Link Mobility Tracking for MANETs

Brian L. Mark and Varnit Suri
Dept. of Electrical and Computer Engineering
George Mason University
Fairfax, Virginia 22030
Email: { bmark, vsuri }@gmu.edu

Shubhranshu Singh and Young Sung Roh
Comm. & Network Lab, Samsung AIT
San 14-1, Giheung-eup
Yongin-si, South Korea.
Email: { shubhranshu,yongsung.roh}@samsung.com

***Abstract*— We propose the concept of link mobility state and present a model and algorithm for link mobility tracking in a mobile ad hoc network (MANET). The link mobility state is a function of the relative distance between any pair of nodes at the two ends of a given link and evolves according to a first-order autoregressive model. The link mobility tracking algorithm is based on a Kalman filter and a parameter estimation algorithm, which provide an estimate of the probability that a given link remains available at a future time. The link availability probability can be taken as a measure of link stability and applied to the route discovery and route maintenance phases of MANET routing protocols to reduce control overhead and packet loss due to route breakages. Our simulation results demonstrate the effectiveness of the proposed link mobility tracking algorithm.**

I. Introduction

Mobile ad hoc networks (MANETs) are self-organizing, rapidly deployable, and require no fixed infrastructure. They are comprised of wireless nodes, which can be deployed anywhere, and must cooperate in order to dynamically establish communications. The flexibility of a highly dynamic MANET introduces many complexities into the tasks of network control and management, including routing, flow control, and power management. For example, traffic routes change over time, subject to the movement of the mobile nodes. The links of the network are not fixed entities, rather their states change over time and are dependent on the relative spatial locations of the nodes, transmitter and receiver characteristics, and the signal propagation properties of the environment.

In this paper, we introduce the concept of link mobility tracking as a means to determine the real-time status of the link between any two mobile nodes and to anticipate the future availability of the link. We define the link mobility state to consist of the relative distance, relative velocity, and relative acceleration between the pair of nodes at the two ends of any given link. We propose a first-order autoregressive (AR-1) model to describe the evolution of the link mobility state in discrete-time. The link mobility model is similar to the AR-1 model used to model *node* mobility in [1]. The key difference is that the link mobility state captures relative distance information, whereas the node mobility state represents absolute location information. The link mobility model provides the basis for a discrete-time algorithm to jointly estimate the link mobility state using a Kalman filter, and the link mobility model parameters using a recursive algorithm.

In the face of a dynamically changing MANET topology, what is needed is a means to transmit packets over the more stable routes available in the network and a trigger mechanism to initiate re-routing and/or route repair in advance of route breakages. By providing this capability, link mobility tracking can be applied to improve the performance of any MANET routing protocol. The computation of link stability metric tends to be more straightforward in proactive routing protocols [2], since such protocols have built-in mechanisms for periodic link state dissemination and neighbor discovery, which can be re-used for link mobility tracking and the dissemination of the link stability metric. Reactive routing [3], [4] protocols can be made mobility-aware by introducing some additional neighbor discovery mechanisms.

The remainder of the paper is organized as follows. In section II, we provide a brief discussion of related work to set the context for the present work. Section III presents the link mobility model and discusses its application to the prediction of future link availability. Section IV discusses how the link state and the parameters of the link mobility model can be jointly tracked in real-time. Section V discusses how the link stability metric can be derived from the link mobility model. Section VI presents numerical results demonstrating the effectiveness of the link mobility estimator. Finally, the paper is concluded in Section VII.

II. Related Work

The idea of using mobility or location prediction to improve routing performance has been proposed in earlier papers. In [6], each node maintains and periodically checks a list of its neighbor nodes. This scheme provides a coarse measure of route stability, but may unfairly penalize relatively stationary nodes that have highly mobile neighbors. By contrast, link mobility tracking provides information on the "mobility" of the *link* between a pair of nodes, rather than the mobility of a single node. Both [7] and [8] represent the mobility behavior of a node in terms of a random walk model. While the random walk model is useful for conceptual modeling, the model does not accurately reflect node movements in realistic scenarios. Moreover, estimation the state and parameters of the random walk model cannot be done efficiently in real-time.

The concept of link mobility state discussed here is unique to the present paper, but is related to the node mobility state concept in [1], [5]. Our proposed link mobility tracking scheme is based on a first-order autoregressive model, which

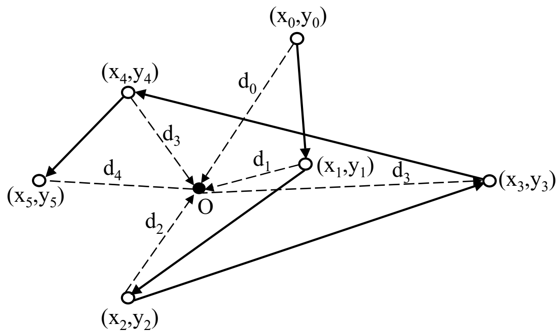

Fig. 1. Illustration of link mobility tracking vs. node mobility tracking.

has been shown (cf. [1]) to accurately model node mobility. More importantly, the model lends itself to efficient and accurate real-time state estimation via the Kalman filter and MMSE (Minimum Mean Squared Error) parameter estimation.

III. Link Mobility Model

Consider a particular mobile node X in an ad hoc network. Another node Y in the network is defined to be a *neighbor* if it lies within the radio transmission/reception range of node X. In this case, a bidirectional wireless link exists between nodes X and Y. This link will continue to exist as long as nodes X and Y remain in radio range of each other, assuming they are also within line-of-sight (LOS). If the nodes move apart, eventually the link will break. The purpose of the link mobility model is to predict, in advance, the availability of the link after a certain time interval. We then use the probability of link availability as a measure of link stability.

A. Link mobility state

We propose a link mobility model based on an autoregressive model similar to the AR-1 model for *node mobility state* presented in [1]. In this case, however, we are not interested in the *absolute* mobility state of each node. Rather, we are interested in the *relative* mobility state between a pair of nodes, i.e., the link mobility state. The link mobility state between two nodes at time t is defined by a (column) vector [1]

$$\boldsymbol{s}(t) = [d(t), \dot{d}(t), \ddot{d}(t)]', \tag{1}$$

where $d(t)$ denotes the relative distance between the nodes, $\dot{d}(t)$ is the first derivative of $d(t)$ or relative velocity, and $\ddot{d}(t)$ is the second derivative of $d(t)$ or relative acceleration.

Note that the relative distance $d(t)$ is always nonnegative, i.e., $d(t) \geq 0$, but the relative velocity $\dot{d}(t)$ and relative acceleration may take on positive or negative values. A value of $\dot{d}(t) < 0$ indicates that node Y is moving *towards* node X at time t, while a value of $\dot{d}(t) > 0$ indicates that node Y is moving *away* from node X. A similar comment applies to the relative acceleration $\ddot{d}(t)$. The link mobility state defined by (5) is applicable whether the nodes are constrained to move in a two-dimensional plane or are free to move in three-dimensional space. The information contained in the link mobility state captures the dynamics of the relative distance between the two nodes of interest, and by inference, the behavior of the associated link.

The evolution of the link mobility state can be considered in discrete-time, assuming a uniform sampling interval τ. The discrete-time link mobility state is defined by

$$\boldsymbol{s}_n \triangleq s(n\tau) = [d_n, \dot{d}_n, \ddot{d}_n]', \quad -\infty < n < \infty, \tag{2}$$

where $d_n \triangleq d(n\tau)$, $\dot{d}_n \triangleq \dot{d}(n\tau)$, and $\ddot{d}_n \triangleq \ddot{d}(n\tau)$. In comparison, the *node* mobility state for two-dimensional space in the discrete-time AR-1 model of [1] is defined by

$$\boldsymbol{u}_n = [x_n, \dot{x}_n, \ddot{x}_n, y_n, \dot{y}_n, \ddot{y}_n]', \tag{3}$$

where (x_n, y_n) denote the (x, y)-position coordinates of the node at the discrete-time instant n. The components $\dot{x}_n$ and $\ddot{x}_n$ represent the velocity and acceleration in the x-direction, while the components $\dot{y}_n$ and $\ddot{y}_n$ represent the velocity and acceleration in the y-direction, at time n.

Figure 1 illustrates the concept of link mobility tracking in comparison to node mobility tracking. The solid circle represents a node O thats track the link and node mobility state of the open circle, which represents a node X that is moving with respect to node O. The node X is initially located at position (x_0, y_0) and moves to the positions (x_1, x_1), $\cdots$, (x_5, y_5), in sequence. The solid lines represent the trajectory taken by node X. The dashed lines represent the sequence of relative distance values d_0, d_1, $\cdots$, d_5. Node mobility tracking involves the estimation of the position sequence $\{(x_n, y_n)\}$, together with the corresponding velocity and acceleration sequences $\{(\dot{x}_n, \dot{y}_n)\}$ and $\{(\ddot{x}_n, \ddot{y}_n)\}$. By contrast, link mobility tracking estimates the sequence of relative distance values $\{d_n\}$, together with the corresponding velocity and acceleration sequences $\{\dot{d}_n\}$ and $\{\ddot{d}_n\}$. In order to extend the node mobility state to three-dimensional space, the state $\boldsymbol{u}_n$ should be expanded to include the z-coordinate and the associated velocity and acceleration components in the z-direction. However, the form of the link mobility state remains the same in the three-dimensional case.

B. Link mobility state equation

The AR-1 model for the *node* mobility state $\boldsymbol{u}_n$ is given as follows [1]:

$$\boldsymbol{u}_{n+1} = B\boldsymbol{u}_n + \boldsymbol{v}_n, \tag{4}$$

where B is a 6×6 transformation matrix, the vector $\boldsymbol{v}_n$ is a 6×1 discrete-time zero mean, white Gaussian process with autocorrelation function $R_w(k) = \delta_k V$, where $\delta_0 = 1$ and $\delta_k = 0$ when $k \neq 0$. The matrix V is the covariance matrix of $\boldsymbol{v}_n$. The transformation $B\boldsymbol{s}_n$ in (4) captures the equations of motion for the state variables from time n to time $n+1$, modulo the noise term $\boldsymbol{v}_n$.

A model similar to (4) can be applied to represent the evolution of the link mobility state. However, care must be

[1]The notation $'$ indicates the matrix transpose operator.

taken to ensure that the computed value of d_n is always positive. Thus, we specify the link mobility state equation as follows:

$$\boldsymbol{s}_{n+1} = \operatorname{sgn}\langle A\boldsymbol{s}_n + \boldsymbol{w}_n\rangle_1 \cdot (A\boldsymbol{s}_n + \boldsymbol{w}_n), \tag{5}$$

where $\langle \boldsymbol{s}\rangle_1$ denotes the first component of the vector $\boldsymbol{s}$. The matrix A has the following form:

$$A = \begin{bmatrix} 1 & \tau & \tau^2/2 \\ 0 & 1 & \tau \\ 0 & 0 & \alpha \end{bmatrix}, \tag{6}$$

and

$$\boldsymbol{w}_n = [w_{n,1}, w_{n,2}, w_{n,3}]'$$

is a 3×1 discrete-time zero mean, white Gaussian process with autocorrelation function $R_w(k) = \delta_k Q$, where $\delta_0 = 1$ and $\delta_k = 0$ when $k \neq 0$. The matrix Q is the covariance matrix of $\boldsymbol{v}_n$. The term α in (6) is a parameter of the model that must be specified or estimated.

In the state equation (5), if the first component of $A\boldsymbol{s}_n + \boldsymbol{w}_n$ is negative, $\boldsymbol{s}_{n+1}$ should be set equal to $-(A\boldsymbol{s}_n + \boldsymbol{w}_n)$. In this case, since the distance component has gone negative from time n to $n+1$, the polarities of all three components of the vector $A\boldsymbol{s}_n + \boldsymbol{w}_n$ should be reversed in order to obtain the next state vector s_{n+1}. Similar to the node mobility state equation (3), the transformation $A\boldsymbol{s}_n$ in (5) captures the evolution of the state $\boldsymbol{s}$ from time n to time $n+1$, modulo the noise term $\boldsymbol{w}_n$. Let

$$g_n \triangleq \operatorname{sgn}(d_n + \dot{d}_n\tau + \ddot{d}_n\tau^2/2 + w_{n,1}).$$

Then the state equation (5) is equivalent to the following system of equations:

$$d_{n+1} = g_n(d_n + \dot{d}_n\tau + \ddot{d}_n\tau^2/2 + w_{n,1}), \tag{7}$$
$$\dot{d}_{n+1} = g_n(\dot{d}_n + \ddot{d}_n\tau + w_{n,2}), \tag{8}$$
$$\ddot{d}_{n+1} = g_n(\alpha\ddot{d}_n + w_{n,3}). \tag{9}$$

IV. Link Mobility Tracking

The link mobility model given by (5) allows us to track the link state given a sequence of observation data. Signal measurements typically available in wireless networks such as RSSI (received signal strength indicator) or TOA (Time-of-Arrival) can serve as observation data for tracking link mobility. A single signal measurement of either kind is sufficient to perform link mobility tracking using a Kalman filter. For concreteness, we shall assume that RSSI is used as the source of observation data, although TOA could just as easily be used instead.

A Kalman filter can be used to perform accurate link state estimation as long as the mobility characteristics of the link do not change significantly over time. The link mobility characteristics are defined by the parameters A_n and Q_n. From the form of A_n, it can be seen that the only undetermined parameter is the parameter α. If the link mobility characteristics change over time, state estimation will become inaccurate unless the mobility parameters A_n and Q_n are re-estimated. To accommodate changes in the link mobility parameters, we propose an integrated link mobility and parameter estimator similar to the one described in [1] for the AR-1 node mobility model. The link parameter estimator uses the sequence of state estimates to re-estimate the link mobility parameters.

A. Link mobility state estimation

Assuming a lognormal shadowing model, the RSSI at time n, measured in dB, received at a given mobile node X from a neighbor node Y is given follows [9]:

$$o_n = \kappa - 10\gamma \log(d_n) + \psi_n, \tag{10}$$

where κ is a constant determined by the transmitted power, antenna height, wavelength, and gain of the node Y, γ is a slope index (typically γ is between $2-5$), ψ_n is a zero mean, stationary Gaussian process with standard deviation σ_ψ typically in the range from $2 - 10$ dB, and d_n is the relative distance between nodes X and Y, as defined earlier. The propagation model (10) can be written in the form of an observation equation incorporating the link mobility state as follows:

$$o_n = h(\boldsymbol{s}_n) + \psi_n, \tag{11}$$

where

$$h(\boldsymbol{s}_n) = \kappa - 10\gamma \log(d_n). \tag{12}$$

The observation sequence $\{o_n\}$ for link mobility tracking consists of scalar values, whereas for node mobility tracking, the observation sequence consists of vectors containing at least three components (cf. [1]). This is due to the fact that localization of a node in two-dimensional space requires at least three independent observations. By contrast, estimation of the relative distance between two nodes requires only a single signal measurement.

Although the link mobility state equation (5) does not, strictly speaking, represent a linear system, the system is approximately linear in practice because the term $\langle A\boldsymbol{s}_n + \boldsymbol{w}_n\rangle_1$ rarely falls below zero in realistic scenarios. This has been confirmed in our numerical experiments. Therefore, we can apply an extended Kalman filter to estimate the link mobility state from the observation data. To that end, the observation signal o_n is linearized as follows (cf. [10]):

$$o_n = h(\boldsymbol{s}_n^*) + H_n \Delta\boldsymbol{s}_n + \psi_n, \tag{13}$$

where $\boldsymbol{s}_n^*$ is the nominal or reference state vector and

$$\Delta\boldsymbol{s}_n = \boldsymbol{s}_n - \boldsymbol{s}_n^*$$

is the difference between the true and nominal state vectors. In the extended Kalman filter, the nominal vector is obtained from the estimated link mobility state at time $n-1$, i.e., $\boldsymbol{s}_n^* = \hat{\boldsymbol{s}}_{n-1}$. The 1×3 vector H_n is given by

$$H_n = \frac{\partial h}{\partial \boldsymbol{s}}\Big|_{\boldsymbol{s}=\hat{\boldsymbol{s}}_{n-1}} = [-10\gamma/\langle\hat{\boldsymbol{s}}_{n-1}\rangle_1, 0, 0]. \tag{14}$$

The steps in the extended Kalman filter are given below (cf. [10]). In the extended Kalman filter, $\hat{\boldsymbol{s}}_{n|n}$ denotes the state estimate at time n given the observation vector at time n

and $\hat{\boldsymbol{s}}_{n|n-1}$ denotes the state estimate at time n given the observation vector at time $n-1$. In terms of the earlier notation, $\hat{\boldsymbol{s}}_n = \hat{\boldsymbol{s}}_{n|n}$.

Initialization:

1) $\hat{\boldsymbol{s}}_{0|-1} = \mathbf{0}$
2) $M_{0|-1} = I_3$

Recursive estimation ($n = 1, 2, \cdots$):

1) $H_n = [-10\gamma/\langle \hat{\boldsymbol{s}}_{n-1|n-1}\rangle_1, 0, 0]$
2) $K_n = M_{n|n-1} H_n' (H_n M_{n|n-1} H_n' + \sigma_\psi^2 I_3)^{-1}$
   Correction steps:
3) $\hat{\boldsymbol{s}}_{n|n} = \hat{\boldsymbol{s}}_{n|n-1} + K_n(o_n - h(\hat{\boldsymbol{s}}_{n|n-1}))$
4) $M_{n|n} = (I_3 - K_n H_n) M_{n|n-1} (I_3 - K_n H_n)' + \sigma_\psi^2 K_n K_n'$
   Prediction steps:
5) $\hat{\boldsymbol{s}}_{n+1|n} = \text{sgn}\langle A_n \hat{\boldsymbol{s}}_{n|n}\rangle_1 A_n \hat{\boldsymbol{s}}_{n|n}$
6) $M_{n+1|n} = A_n M_{n|n} A_n' + Q$

Here, $\mathbf{0}$ denotes the 3×1 vector of all zeros, I_3 denotes the 3×3 identity matrix, $M_{i|j} \triangleq \text{Cov}(\hat{\boldsymbol{s}}_{i|j})$, and K_n is the Kalman gain matrix. The most involved computation in the Kalman filter is the inversion of a 3×3 matrix in the calculation of K_n, which should be well within the capabilities of most modern mobile devices.

B. Link mobility parameter estimation

Due to the autoregressive form of the link mobility model, the MMSE (Minimum Mean Squared Error) estimates of the parameters A and Q can be obtained using a form of the Yule-Walker equations [11]. Recall from (6) that the matrix A is determined by the parameter α. Define the residual error at time n by

$$\boldsymbol{e}_n \triangleq \boldsymbol{s}_n - \hat{A}_{n-1}\boldsymbol{s}_{n-1}, \tag{15}$$

where $\hat{A}_n$ denotes the MMSE estimate of A_n at time n. The orthogonality principle requires that

$$E[\boldsymbol{s}_{n-1}' \boldsymbol{e}_n] = 0, \tag{16}$$

From (16), the MMSE estimate of α at time n can be obtained as

$$\hat{\alpha}_n = \begin{cases} r_n(1)/r_n(0), & \text{if } r_n(1) \neq 0, \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

where

$$r_n(0) = E[(\ddot{d}_n)^2] \text{ and } r_n(1) = E[\ddot{d}_n \ddot{d}_{n-1}]. \tag{18}$$

Given $\hat{\alpha}_n$, the estimate $\hat{A}_n$ is completely determined by (6) and the sampling interval T_n. The corresponding estimate for the noise covariance matrix Q_n is calculated as follows:

$$\hat{Q}_n = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{e}_i \boldsymbol{e}_i'. \tag{19}$$

The link mobility state estimates $\hat{\boldsymbol{s}}_n$, which are obtained using the extended Kalman filter as discussed in subsection IV-A, are used to re-estimate the model parameters at time n. The steps involved in estimating the link mobility parameters are listed below.

Initialization:

1) $a_2 = \langle \hat{\boldsymbol{s}}_1 \rangle_3$
2) $\hat{A}_2 = \hat{A}$, $\hat{Q}_2 = \hat{Q}$
3) $r_2(0) = \hat{r}(0)$, $r_2(1) = \hat{r}(1)$

where the initial parameter values $\hat{A}$, $\hat{Q}$, $\hat{r}(0)$, and $\hat{r}(1)$ can be determined from a set of training samples.

Link mobility parameter estimation ($n = 3, 4, \cdots$):

1) $\boldsymbol{e}_n = \boldsymbol{s}_n - \hat{A}_{n-1}\boldsymbol{s}_{n-1}$
2) $\hat{Q}_n = \frac{1}{n-1}[(n-2)\hat{Q}_{n-1} + \boldsymbol{e}_n \boldsymbol{e}_n']$.
3) $a_n = \langle \hat{\boldsymbol{s}}_n \rangle_3$
4) $r_n(0) = \frac{1}{n-1}[(n-2) r_{n-1}(0) + a_n^2]$
5) $r_n(1) = \frac{1}{n-2}[(n-3) r_{n-1}(1) + a_{n-1} a_n]$
6) Obtain $\hat{A}_n$ from $\hat{\alpha}_n = r_n(1)/r_n(0)$.

The link mobility parameter estimates $\hat{A}_n$ and $\hat{Q}_n$ depend only on the estimated state sequence up to $\hat{\boldsymbol{s}}_n$. Therefore, $\hat{A}_n$ and $\hat{Q}_n$ can be applied in the prediction step of the Kalman filter in subsection IV-A in place of A_n and Q_n, respectively. This results in a joint link mobility state and estimation scheme, which combines the Kalman filter-based state estimation, with MMSE parameter estimation as discussed above.

V. Link Stability Metric

Given the link mobility model, which is estimated in real-time as discussed in section IV, a link stability metric can be derived based on the predicted probability of link availability at a future time. In this section, we derive a link stability metric based on the probability of link availability predicted at a future time.

Suppose the link mobility state between two nodes X and Y is tracked using the joint state and parameter estimation scheme discussed in section IV. From the prediction steps (5 and 6) of the extended Kalman filter specified in subsection IV-A, the one-step predicted link mobility state estimate at time $n+1$ given the set of all observations up to time n is given by

$$\hat{\boldsymbol{s}}_{n+1|n} = \text{sgn}\langle A_n \hat{\boldsymbol{s}}_{n|n}\rangle_1 A_n \hat{\boldsymbol{s}}_{n|n} \tag{20}$$

and the corresponding covariance matrix is given by

$$M_{n+1|n} = A_n M_{n|n} A_n' + Q_n. \tag{21}$$

The predicted relative distance estimate at time $n+1$ given the set of observations up to time n is given by

$$\hat{d}_{n+1|n} = \langle \hat{\boldsymbol{s}}_{n+1|n}\rangle_1. \tag{22}$$

The corresponding variance of $\hat{d}_{n+1|n}$ is given by

$$\sigma_{n+1|n}^2 = \langle M_{n+1|n}\rangle_{(1,1)}. \tag{23}$$

Then $d_{n+1|n}$ is Gaussian-distributed with approximate mean $\hat{d}_{n+1|n}$ and variance $\sigma_{n+1|n}^2$.

Let d_{th} be the transmission/reception range of nodes in the network. We shall assume for simplicity that all nodes have the same transmission/reception range. Then the probability that the link between nodes X and Y will remain active at time $n+1$, given the set of observations up to time n can be expressed as follows:

$$p_{n+1|n} \triangleq P[d_{n+1|n} \leq d_{th}] \tag{24}$$

$$= \frac{1}{\sqrt{2\pi}\sigma_{n+1|n}} \int_0^{d_{th}} e^{\frac{-(x-\hat{d}_{n+1|n})}{2\sigma^2_{n+1|n}}} dx \tag{25}$$

$$= Q\left(\frac{d_{th} - \hat{d}_{n+1|n}}{\sigma_{n+1|n}}\right) - Q\left(\frac{-\hat{d}_{n+1|n}}{\sigma_{n+1|n}}\right), \tag{26}$$

where $Q(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-y^2/2}\, dy$ denotes the standard Q-function. In principle, the one-step predicted link availability probability could be extended to an m-step predicted value by repeating the Kalman filter prediction step to derive approximations for $\hat{d}_{n+m|n}$ and $\sigma_{n+m|n}$. In this paper, however, we shall take the probability $p_{n+1|n}$ as a measure of the stability of the link between nodes X and Y at time n, which we shall denote by $p_n(X, Y)$.

VI. Numerical Results

To evaluate link mobility tracking, the link mobility behavior between a pair of mobile nodes was simulated using Matlab. The serving area is assumed to be a 800 m $\times$ 800 m square bounded by the points $(-400, -400)$ and $(400, 400)$. Since the link mobility state captures the relative distance, velocity, and acceleration between the two nodes, one of the nodes, labelled O, can be fixed at the origin $(0, 0)$ system without loss of generality. The other node, labelled X, moves in the serving area according to the random waypoint model [3], [12].

In the representative scenario of Fig. 2, node O tracks the link mobility state between itself and node X. The mobility of node X is governed by a random waypoint model with velocity uniformly distributed in the range $[5, 15]$ m/s and a pause time of zero, which represents the highest degree of mobility. The transmission/reception range is set to $d_{th} = 250$ m. The link stability threshold is set to $p_{th} = 0.5$. The signal propagation parameters in (10) are set to $\kappa = 0$ and $\gamma = 3$. The variance of the noise process ψ_n is set to $\sigma_\psi = 4$ dB.

Figure 2 shows a sample path of the actual relative distance over a time interval of 200 s. The vertical axis on the left represents the relative distance in units of meters, while the right-hand vertical axis represents the link stability metric, taking values in the range $[0, 1]$. The estimated relative distance trajectory is obtained from the link mobility tracking procedure described in section IV. The sampling interval is assumed to be a constant $\tau = 2$ s. One can see that the estimated distance tracks the actual distance closely over the observation interval. When the slope of the actual distance curve changes sign, the estimated distance deviates from the actual distance but then recovers within several seconds. The graph also shows the trajectory of the link stability metric computed by node O. Observe that the link stability metric trajectory always crosses the threshold p_{th} in the vicinity of a crossing, in the opposite sense, of the estimated distance trajectory with respect to the threshold d_{th}. That is, an upcrossing of p_{th} always corresponds to a downcrossing of d_{th} and vice versa.

In Fig. 2, the downcrossing of the link stability curve with respect to p_{th} just prior to $t = 236$ s occurs approximately 2 s before the upcrossing of the relative distance curve with respect to d_{th}. This implies that the link stability metric *anticipates* the breakage of the link approximately $\tau = 2$ s in advance of the link breakage event. Recall from section V that the link stability metric is defined in terms of the estimated probability of link availability τ time units into the future.

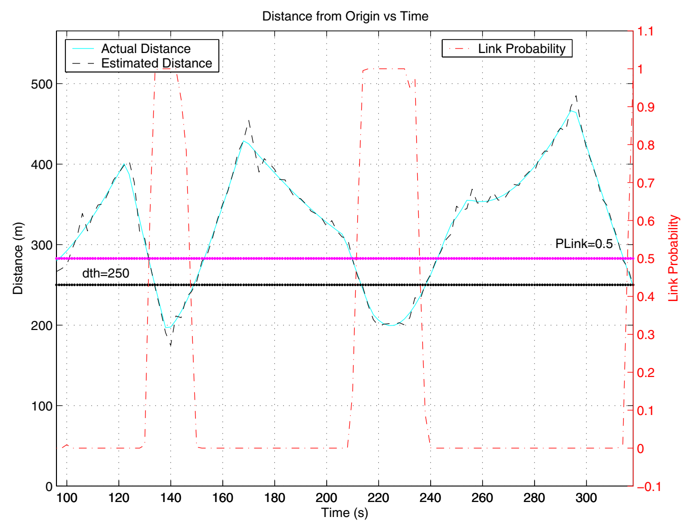

Fig. 2. Tracking of link relative distance and link stability.

Figures 3-5 show the tracking performance of the link mobility estimator for various parameter settings. The average speed of the mobile node was varied between 0 and 40 m/s (i.e., 0 and 144 mph), which spans most real-life speeds. Each simulation run was 2500 s long and the results were averaged over a total of 40 simulation runs. The tracking error is reported as the average of the absolute errors during the simulation. The absolute error is the absolute value of the difference between the actual relative distance at time t and the value predicted by the link mobility estimator, τ seconds in advance. The results indicate that the tracking error increases with increasing node velocity, as should be expected. For a wide range of real-life speeds, the link mobility tracking performance is quite accurate.

Figure 3 provides insight into how the estimation algorithm performs under varying noise levels for time-step $\tau = 1$ s. The noise level is varied by changing the parameter σ_ψ. As expected, the error increases for increasing noise levels and larger time steps. Figures 4 and 5 provide another perspective on the performance of the link mobility estimator when $\sigma_\psi = 1$ dB and 2 dB, respectively, and for different values of the

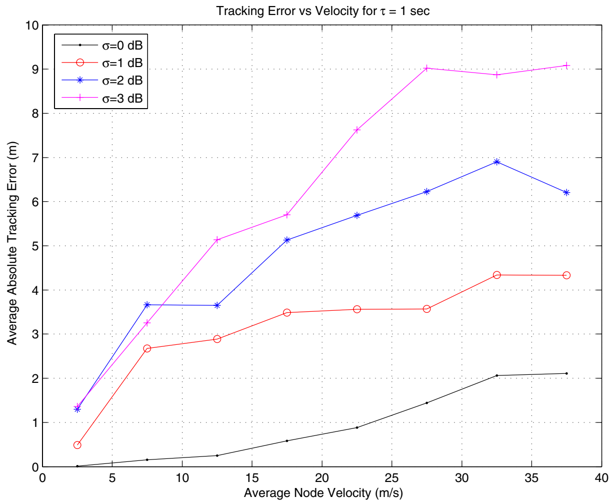

Fig. 3. Tracking error for time-step $\tau = 1$ s.

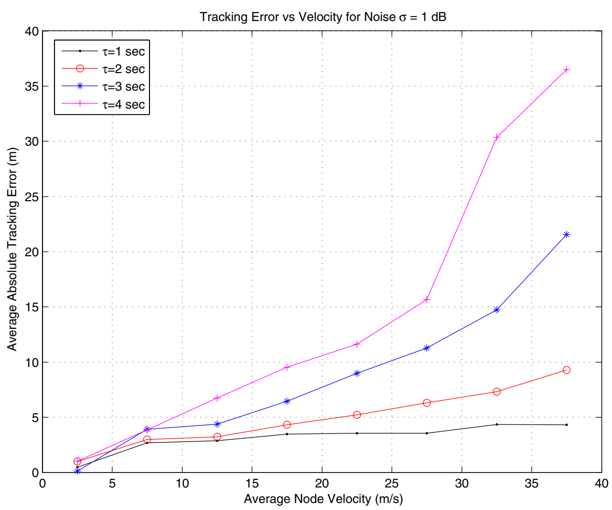

Fig. 4. Tracking error for $\sigma_\psi = 1$ dB.

time-step τ. As expected, the error increases for increasing values of τ and larger values of σ_ψ.

VII. Conclusion

This paper introduced the concept of link mobility tracking, a simple, but powerful idea that can be applied to improve the performance of MANETs. We presented a link mobility state model based on an autoregressive model and a link tracking algorithm based on combining the Kalman filter and a recursive parameter estimator. A link stability metric was defined as the estimated probability of link availability at the next discrete-time index. The link stability metric can be used to proactively invoke re-routing or route repair to avoid route breakage events in advance. Our numerical results illustrate the predictive power and accuracy of the link mobility tracking algorithm.

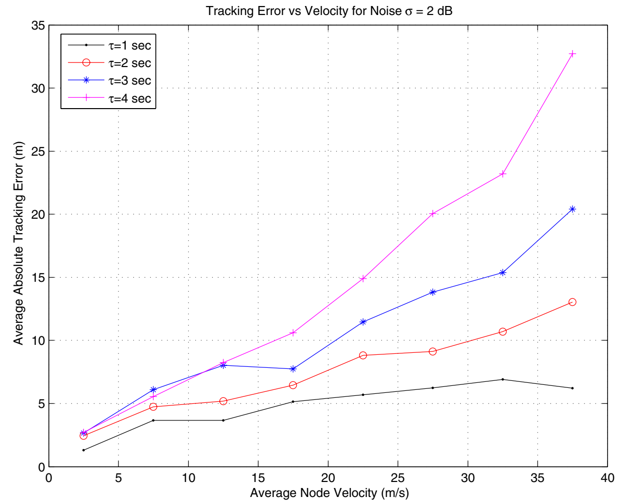

Fig. 5. Tracking error for $\sigma_\psi = 2$ dB.

Acknowledgment

The authors would like to acknowledge Dr. Zainab Zaidi for useful input in an early stage of this work. This work was supported in part by the U.S. National Science Foundation under Grant CCR-0209049 and Grant ECS-0426925.

References

[1] Z. R. Zaidi and B. L. Mark, "Mobility estimation for wireless networks based on an autoregressive model," in *Proc. IEEE Globecom*, (Dallas, TX), pp. 3405–3409, Nov. 2004.

[2] T. H. Clausen and P. Jacquet, "Optimized link state routing," *IETF RFC 3626*, Oct. 2003.

[3] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing* (T. Imielinski and H. Korth, eds.), ch. 5, pp. 153–181, Kluwer Academic Publishers, 1996.

[4] C. Perkins and E. Royer, "Ad hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99)*, pp. 90–100, 1999.

[5] T. Liu, P. Bahl, and I. Chlamtac, "Mobility modeling, location tracking, and trajectory prediction in wireless ATM networks," *IEEE J. Selected Areas in Comm.*, vol. 16, pp. 922–936, August 1998.

[6] S. Abhyankar and D. Agrawal, "Distributed mobility-aware route selection for wireless ad hoc networks," in *Proc. IEEE IPCCC '02*, pp. 241–247, April 2002.

[7] R. Chellappa, A. Jennings, and N. Shenoy, "A comparative study of mobility predicition in fixed wireless and mobile ad hoc networks," in *Proc. IEEE ICC '03*, vol. 2, pp. 891–895, May 2003.

[8] A. B. McDonald and T. Znati, "A mobility based framework for adaptive clustering in wireless ad-hoc networks," *IEEE J. Sel. Areas in Comm.*, vol. 17, pp. 1466–1487, August 1999.

[9] G. L. Stüber, *Principles of Mobile Communication*. Massachusetts: Kluwer Academic Publishers, 2nd ed., 2001.

[10] R. G. Brown and P. Y. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*. New York: John Wiley & Sons, 3rd ed., 1997.

[11] J. S. Lim and A. V. Oppenheim, *Advanced Topics in Signal Processing*. Englewood Cliffs, NJ 07632: Prentice Hall, 1987.

[12] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proc. ACM Mobicom*, pp. 85–97, Oct. 1998.