Performance Metrics for IEEE 802.21 Media Independent Handover (MIH) Signaling

David Griffith, Richard Rouil, and Nada Golmie

Abstract— **The IEEE 802.21 Media Independent Handover (MIH) working group is developing a set of mechanisms to facilitate migration of mobile users between access networks that use different link-layer technologies. Among these are mobility managers that create and process signaling messages to facilitate handovers. The MIH signaling architecture being developed in the Internet Engineering Task Force (IETF) allows any transport layer protocol to carry MIH messages. The IETF has considered using the unreliable but lightweight transport available with the User Datagram Protocol (UDP) as well as the reliable stream-oriented transport with congestion control offered by the Transmission Control Protocol (TCP). In this paper we develop mathematical models that result in expressions for the characteristic function of the time required to complete exchanges of an arbitrary number of MIH signaling messages between a mobile node (MN) and a remote mobility manager (MM). Our models also provide expressions for the average amount of overhead associated with MIH message exchanges due to retransmissions either by the MIH signaling entities or by the transport-layer protocol. In addition, we provide simulation results that confirm the results from the mathematical model and illustrate the effect of varying transport parameters such as the TCP maximum retransmission timeout.**

Index Terms— **Media Independent Handover (MIH), TCP, UDP, handover latency, signaling overhead**

I. Introduction

Handovers in mobile wireless networks have been thoroughly studied for the case where the handover is horizontal (i.e. a mobile node's movement causes it to end a connection with one access point and begin another with a different access point that uses the same Layer 2 technology as the previous one)[1],[2]. With the proliferation of Layer 2 wireless technologies (e.g. IEEE 802.11 (WiFi) and IEEE 802.16 (WiMAX)), manufacturers of mobile telecommunications devices have recognized the benefit of developing wireless mobiles with multiple antennas that are capable of performing so-called vertical handovers, in which the mobile switches active connections from a network access point that uses one Layer 2 technology to another network access point that uses a different Layer 2 technology.

Various standards bodies such as the IETF and IEEE 802.21 have been developing new protocols to support fast handovers at different layers. The IETF's Mobility for IP: Performance, Signaling and Handoff Optimization (MIPSHOP) and IP Mobility Optimizations (MOBOPTS) working groups in the IETF have developed enhanced versions of IPv4 and IPv6 that allow a mobile user to receive packets by maintaining information on the user's current location in the network. These enhancements include Mobile IPv6 [3], Fast Mobile IPv6 (FMIPv6) [4], Hierarchical Mobile IPv6 (HMIPv6) [5]. FMIPv6 and HMIPv6 in particular were designed to solve the problem of excessive handover latency at Layer 3. The key to reducing latency lies in beginning the handover process early enough that the mobile node's connection to the target access network is fully prepared before the old connection fails (make before break). This requires giving higher layers access to link-layer information such as triggers related to received signal quality. The IEEE 802.21 Media Independent Handover (MIH) working group is developing mechanisms to use Layer 2 triggers associated with events to provide early warning of impending handovers and thereby decrease handover latency [6].

The IEEE 802.21 architecture consists of MIH Users located above Layer 4 that use MIH Service Access Points (SAPs) to communicate with MIH services at lower layers. MIH Users are the initiation and termination points for MIH signaling sessions. The MIH Function (MIHF), located between Layer 2 and Layer 3, provides handover services (including the Event Service (ES), Information Service (IS), and Command Service (CS)) through SAPs that are defined by the IEEE 802.21 working group. The IETF working group identified low latency and reliable delivery as two key requirements that will be imposed by some MIH Users [7]. In addition, congestion control may be required depending on the amount of data an MIH User generates. Thus some MIH signaling messages should be carried over a reliable transport protocol like TCP, which offers reliable delivery and congestion control at the expense of low latency. Other messages may be better delivered by a less reliable but lightweight transport protocol like UDP. The MIPSHOP working group is considering a number of possible transport solutions, discussed in [8].

The draft [8] allows any transport protocol to carry MIH signaling messages, and identifies several factors that can affect the choice of transport protocol that should be considered when designing an MIH implementation. These include message size, message generation rate, and the effect of retransmission on performance (e.g. latency). The draft notes that short time-sensitive messages such as those associated with CS and ES should travel over UDP but use retransmissions managed by the MIH User itself, while longer messages such as the database updates that are part of the IS need TCP's congestion control. These recommendations do not include a quantitative analysis to support them; it is our goal in this paper to supply a general model that can be used to get performance metrics for MIH messages over either UDP or TCP while varying network parameters such as data rate, MIH

The authors are with the National Institute of Standards and Technology, Gaithersburg, MD 20899 USA.

This work was supported in part by the Office of Law Enforcement Standards (OLES).

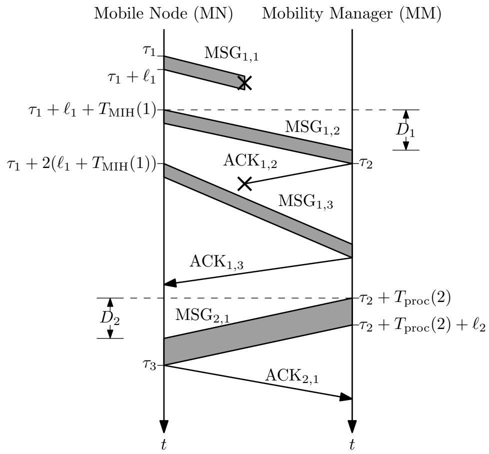

Fig. 1. Sample signal flow between MN and MM, in which packet loss occurs.

message size, and the network transit time delay distribution. In addition, we show results obtained using simulation tools that we developed that allow network planners to consider various of deployment scenarios.

The rest of this paper is organized as follows. In Section II we consider the UDP transport where MIH uses ACKs and retransmission timers. In Section III, we consider TCP; in the course of this analysis, we examine the effect of introducing MIH retransmission timers in addition to TCP's recovery mechanisms. We use simulation results to verify our models in Section IV, and examine some of the design trade-offs associated with the transport layer protocols. We summarize our results in Section V.

II. MIH Signaling over UDP

We first consider the case where the MIH User uses UDP to carry signaling messages. In this case, MIH ensures reliability by requesting that each outgoing message be acknowledged by its recipient. In addition, MIH maintains a retransmission timer for each sent message. We denote the timer duration for the ith MIH message by $T_{\rm MIH}(i)$. If the ACK for the ith message is not received by the time the retransmission timer expires, MIH will send a new copy of the message, up to a limit of R_i retransmissions.

A. MIH over UDP Latency

We can derive the distribution of the time required to complete an exchange of M MIH messages. We assume that an MIH User that receives an MIH message immediately generates an MIH ACK followed by an MIH response message that constitutes the next part of the information exchange, such as a handover signaling sequence. We show an example message exchange between a mobile node (MN) and mobility manager (MM) in Fig. 1. The time between the generation of the ith ACK and the transmission of the subsequent $(i+1)$th MIH message is a deterministic quantity that we denote as $T_{\rm proc}(i)$. This reflects the amount of time used by the receiving node to perform any tasks associated with the arriving MIH message. We define τ_i to be the time when the $(i-1)$th message in the handover message exchange is received. For the case of the first message in the handover sequence, τ_1 is the time when the message is generated. The amount of time required to move the ith message out of the transmission buffer is $\ell_i = L_i/B$ s, where L_i is the length of the first message in bits and B is the bandwidth of the channel in bits/s. The ith message traverses the network and, if it is not lost, it arrives at its destination after some random delay that we denote as D_i.

The node that receives the first MIH message immediately transmits an MIH ACK and, at time $\tau_2 = \tau_1 + \ell_1 + D_1 + T_{\rm proc}(i)$, sends the second MIH message back to the node that initiated the message exchange. In the figure, we show the MIH ACKs as requiring no time to transmit. We use this simplification because the total time to complete the exchange, H, is not affected by the amount of time spent sending MIH ACKs, and because the exchange's state advances only when each of the two endpoints receives MIH messages. Each message in the exchange sequence is generated at time $\tau_i = \tau_{i-1} + \ell_{i-1} + D_{i-1} + T_{\rm proc}(i)$, and the exchange finishes when the MIH ACK for the Mth MIH message is received. The total time required to complete the exchange is

$$\begin{aligned} H &= \tau_M + \ell_M + D_M + D_{M+1} - \tau_1 \\ &= D_{M+1} + \sum_{i=1}^{M} \big(D_i + \ell_i + T_{\rm proc}(i)\big), \end{aligned} \tag{1}$$

where D_{M+1} is the time for the Mth MIH message's ACK to traverse the network.

We assume that the delays $\{D_i\}_{i=1}^{M+1}$ are independent identically distributed random variables with cumulative distribution function $F_D(t)$ and characteristic function $\Phi_D(\omega) = \int_0^\infty f_D(t)\exp(j\omega t)dt$, where $f_D(t) = dF_D(t)/dt$ is the density of D. In practice, the delays will be correlated, since the forward and reverse paths pass through the same network elements; we assume independence to simplify the analysis. In the absence of packet loss, the characteristic function of the handover delay, $\Phi_H(\omega)$, is $\big(\Phi_D(\omega)\big)^{M+1}\exp\big(j\omega\sum_{i=1}^{M}[\ell_i + T_{\rm proc}(i)]\big)$.

To model the case where packet losses occur, let p be the probability that an MIH message or MIH ACK is lost while transiting the network. The exchange of a message and ACK is successful with probability $P_s = (1-p)^2$. For the exchange of M messages to be successful, each message must be received successfully. This requires that at least one of $R_i + 1$ attempts to transmit the ith message be successful, for $i = 1, 2, \ldots, M$. The probability that a series of message transmission attempts does not fail is $1 - p^{R_i+1}$. Thus the probability that a message exchange fails is

$$P_{\rm fail} = 1 - \prod_{i=1}^{M}(1 - p^{R_i+1}). \tag{2}$$

The product is taken over M messages, not $M+1$, because we assume that the loss of the final MIH ACK message will not prevent completion of the message exchange. A sender will also re-transmit a message if the sum of the message's transit time and the transit time of its corresponding ACK is greater than the timeout, $T_{\rm MIH}(i)$. We assume that each retransmitted message arrives after its preceding copy, so that it is not possible for a message retransmission to arrive at the destination before a retransmission that was sent earlier.

Consider the first message in the exchange, which was first transmitted at time τ_1. If the message is lost, as shown in Fig. 1, the sender will not receive an MIH ACK and its MIH event timer will expire at time $\tau_1 + \ell_1 + T_{\text{MIH}}(1)$. At that time, the sender will generate a second copy of the first message. Each additional message that is lost will cause a timeout at the sender's MIH User and a retransmission of the original message, up to a limit of $R_1 + 1$ attempts in total. Fig. 1 shows an MIH ACK being lost during the the first message's second transmission attempt. This does not add to the handover delay because the MM is already responding to the first MIH message. Lost MIH ACKs also trigger retransmissions. The probability that k transmission attempts fail to reach the destination prior to a final, successful attempt is $p^k(1-p)$. Each failure to send the ith message adds an additional

$$\delta_i \triangleq \ell_i + T_{\text{MIH}}(i)$$

units of time to the total time required to complete the signaling for the handover. We condition on the event that the message is successfully delivered during one of its allowed $R_i + 1$ transmission attempts; thus the probability that the ith message transmission is preceded by a delay of $k \cdot T_{\text{MIH}}(i)$ is $p^k(1-p)/(1-p^{R_i+1})$ for $k = 0, 1, \ldots, R_i$.

For each of the M messages in the handover, we assume that the loss performance is independent of that of all the other messages. The characteristic function of the additional delay Δ_i^{D} associated with timeouts caused by losses of the ith MIH message is therefore

$$\begin{aligned} \Phi_{\Delta_i^{\text{D}}}(\omega) &= \frac{1-p}{1-p^{R_i+1}} \sum_{k=0}^{R_i} p^k e^{jk\omega\delta_i} \\ &= \frac{1-p}{1-p^{R_i+1}} \frac{1-(pe^{j\omega\delta_i})^{R_i+1}}{1-pe^{j\omega\delta_i}}, \end{aligned} \tag{3}$$

where the D superscript indicates that MIH is in datagram mode (i.e. using UDP transport).

Since the total exchange time H accounting for dropped messages is $H = \sum_{i=1}^{M+1} D_i + \sum_{i=1}^{M} \left(\ell_i + T_{\text{proc}}(i) + \Delta_i^{\text{D}}\right)$, the characteristic function for H is

$$\Phi_H(\omega) = (\Phi_D(\omega))^{M+1} e^{j\omega(T_P + T_\ell)} \prod_{i=0}^{M} \Phi_{\Delta_i^{\text{D}}}(\omega), \tag{4}$$

where $T_P = \sum_{i=1}^{M} T_{\text{proc}}(i)$ is the total message processing time spent by both the connection endpoints, and $T_\ell = \sum_{i=1}^{M} \ell_i$ is the total time spent transmitting the messages that successfully reach their destinations. If the MIH User allows the same number of retransmissions for each message, so that $R_i = R$ for all i, then Equation (4) becomes

$$\begin{aligned} \Phi_H(\omega) &= \frac{(1-p)^M}{1-P_{\text{fail}}} (\Phi_D(\omega))^{M+1} e^{j\omega(T_P+T_\ell)} \\ &\quad \times \prod_{i=0}^{M} \frac{(1-(pe^{j\omega\delta_i})^{R+1})}{(1-pe^{j\omega\delta_i})}. \end{aligned} \tag{5}$$

Because the characteristic function of H is the Fourier transform of the density $f_H(t)$, we can use it to get any desired statistic for H. For instance, we may want to know the probability that the exchange time exceeds some threshold t, which may be associated with a desired level of Quality of Service (QoS) for the MN. We compute this probability by getting $F_H(t) = \Pr\{H \le t\}$, the cumulative distribution function of H, and then computing $1 - F_H(t)$. We can get the distribution function by inverting the characteristic function using Equation (3.6) from [9], which is

$$F_H(t) = \frac{2}{\pi} \int_0^\infty \text{Re}[\Phi_H(\omega)] \frac{\sin(\omega t)}{\omega} \, d\omega. \tag{6}$$

Using the trapezoidal rule, we can approximate this integral as follows, which is Equation (4.4) from [9]:

$$F_H(t) \approx \frac{ht}{\pi} + \frac{2}{\pi} \sum_{k=1}^{N_h} \text{Re}[\Phi_H(kh)] \frac{\sin(kht)}{k}, \tag{7}$$

where h is the step size for the summation on the ω-axis, and N_h is the number of points taken in the summation. N_h must be large enough to produce an accurate approximation, and h must be small enough to guarantee accuracy while not being so small that it results in rounding errors. For the computations that we performed for this paper we found that $h \approx 5 \times 10^{-3}$ radians/s and $N_h = 2 \times 10^4$ produced good results.

We consider an example to illustrate our results. In Fig. 2, we consider the case described in Section 10.4 of [10], in which a mobile node (MN) signals a remote mobility manager (MM) to perform a handover from an IEEE 802.11 WLAN network to a cellular network. We plot $1 - F_H(t)$, the probability that the handover time exceeds a given time, t, for three cases in which the probability of packet loss and the MIH timeout duration vary. The number of messages required to complete the handover is $M = 3$. The number of retransmissions that are allowed for each message is $R = 2$. The one-way transit delay is assumed to be exponential with an average value of 1 normalized unit of time. The processing time associated with each message is assumed to be zero. The MIH timeout values for all three messages are the same for each case plotted in the figure, and are given as multiples of μ_D, the expected value of D, which is unity.

The effect of message retransmissions is most clearly visible in the black curve with diamonds that is associated with $p = 0.01$ and $T_{\text{MIH}} = 15\mu_D$. The performance curve has a staircase shape with plateaus that correspond to the packet loss probability and whose location on the time axis is determined by the MIH timeout. Decreasing the timeout duration to $T_{\text{MIH}} = 5\mu_D$ produces the behavior shown in the red curve with squares, which is much steeper and which does not exhibit the same plateau effect that can be seen in the curve associated with $p = 0.01$ and $T_{\text{MIH}} = 15\mu_D$. If we decrease the packet loss probability to $p = 10^{-4}$ while keeping the MIH timeout at $T_{\text{MIH}} = 15\mu_D$, we obtain behavior shown by the curve with blue circles. Some plateauing is visible in this curve; the first plateau is located at the same position on the time access as the first plateau the same packet loss probability and the larger MIH timeout. The value of $1 - F_H(t)$ at this plateau is two orders of magnitude below the plateau

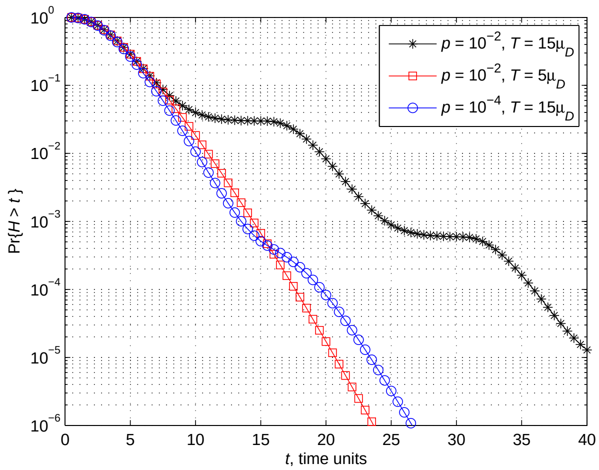

Fig. 2. Probability that the handover time, H, exceeds t seconds for three sets of values of packet loss probability p and MIH timeout, T.

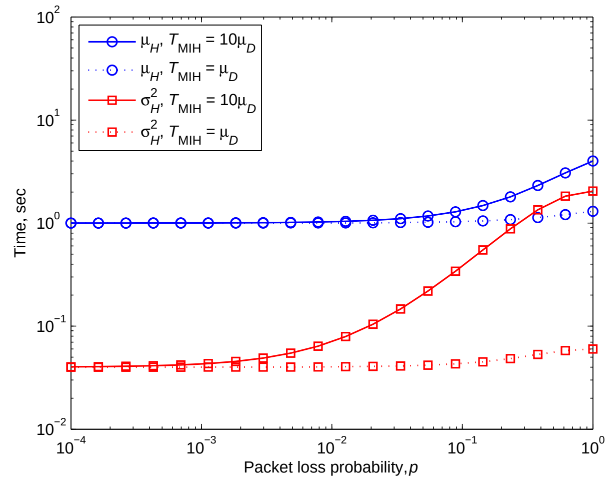

Fig. 3. Mean and variance of the handover time H versus packet loss probability, p, for short and long MIH timeout values ($T_{\text{MIH}} = \mu_D$ and $T_{\text{MIH}} = 10\mu_D$, respectively).

associated with the packet loss probability value of 0.01. The three curves in Fig. 2 demonstrate that the performance of MIH over UDP does not depend on the values of the retransmission timeout if the timeout period is on the order of the round-trip time between the MIH connection endpoints. A small packet loss probability will also eliminate the effect of retransmissions.

We can get the first and second order moments of H from $\Phi_H(\omega)$. The expected value of H is

$$\mu_H = \frac{1}{j}\frac{d\Phi_H(\omega)}{d\omega}\bigg|_{\omega=0}, \tag{8}$$

which we can evaluate in the case of Equation (4) as

$$\mu_H = (M+1)\mu_D + T_P + T_\ell + \sum_{i=1}^{M}\mu_{\Delta_i^{\text{D}}}, \tag{9}$$

where μ_D is the expected one-way transit time, and $\mu_{\Delta_i^{\text{D}}}$ is

$$\begin{aligned}\mu_{\Delta_i^{\text{D}}} &= \frac{1}{j}\frac{d\Phi_{\Delta_i^{\text{D}}}(\omega)}{d\omega}\bigg|_{\omega=0} \\ &= \left[\frac{(1-p)R_i+1}{1-p} - \frac{R_i+1}{1-p^{R_i+1}}\right]\delta_i, \end{aligned}\tag{10}$$

where $\lim_{p\to 1}\mu_{\Delta_i^{\text{D}}} = R_i\delta_i/2$.

Recall that the transit times $\{D_i\}_{i=1}^M$ are mutually independent, and are independent of the additional delays $\{\Delta_i^{\text{D}}\}_{i=1}^M$, which are themselves mutually independent. The processing times and packet transmit times are deterministic. Thus the variance of H is

$$\sigma_H^2 = (M+1)\sigma_D^2 + \sum_{i=1}^{M}\sigma^2_{\Delta_i^{\text{D}}}, \tag{11}$$

Each of the variances of $\{\Delta_i^{\text{D}}\}_{i=1}^M$ are:

$$\begin{aligned}\sigma^2_{\Delta_i^{\text{D}}} &= \mathsf{E}\{(\Delta_i^{\text{D}})^2\} - \mu^2_{\Delta_i^{\text{D}}} \\ &= (-1)\frac{d^2\Phi_{\Delta_i^{\text{D}}}(\omega)}{d\omega^2}\bigg|_{\omega=0} - \mu^2_{\Delta_i^{\text{D}}} \\ &= \left[\frac{p}{(1-p)^2} - \frac{(R_i+1)^2p^{R_i+1}}{(1-p^{R_i+1})^2}\right]\delta_i^2, \end{aligned}\tag{12}$$

where $\lim_{p\to 1}\sigma^2_{\Delta_i^{\text{D}}} = (R_i+2)R_i\delta_i^2/12$.

We can use Eq. (9)-(12) to plot the mean and variance of H for MIH signaling over UDP versus the p, the probability of message loss, in Fig. 3. For the plots in the figure, we used the network parameters given in Table I, but we let $\mu_D = 100$ ms. We let T_{MIH} take values of μ_D and $10\mu_D$ to show the effect of varying the MIH timeout. For this set of network parameters, the value of T_{MIH} does not have an impact on the statistics of H for $p < 10^{-3}$, and the effect on μ_H is not noticeable for $p < 10^{-2}$. Thus a proactive approach to handovers, including initiating the handover before the message loss probability becomes too large, allows consistent handover performance over a wide range of MIH timeout values.

B. MIH over UDP Overhead

As indicated by Fig. 2, setting the timeout interval to be on the order of the network delay will minimize the effect of packet loss on the handover delay. Indeed, as $T_{\text{MIH}}(i) \to 0$, the additional time required to complete a handover vanishes. However, setting the timeout interval to be too small will result in unnecessary generation of duplicate copies of messages, adding to the overhead associated with the handover. We can quantify the overhead penalty associated with a particular set of values for the timeouts $\{T_{\text{MIH}}(i)\}_{i=1}^M$ by computing the expected number of messages generated per attempted exchange.

Assuming that there are retransmission opportunities remaining, the sending node will transmit an additional copy of a message if the previous copy is lost. If the previous copy is not lost, a retransmission will take place if the corresponding MIH ACK is lost. If neither the message nor its ACK are lost, a retransmission will occur if the time from the message transmission to the ACK reception is greater than $T_{\text{MIH}}(i)$. Thus the probability that a given message transmission attempt results in another copy of the message being sent is

$$\begin{aligned} P_{rt}(i) &= p + (1-p)[p + (1-p)(1 - F_{\text{RTT}}(T_{\text{MIH}}(i)))] \\ &= (2-p)p + (1-p)^2[1 - F_{\text{RTT}}(T_{\text{MIH}}(i))] \\ &= 1 - (1-p)^2 F_{\text{RTT}}(T_{\text{MIH}}(i)), \end{aligned} \tag{13}$$

where $F_{\text{RTT}}(t)$ is the cumulative distribution of the message round-trip time, RTT. For ease of computation, we assume that the RTT is the sum of two independent one-way transit times that each have the cumulative distribution $F_D(t)$. Thus $F_{\text{RTT}}(t) = \int_0^\infty f_D(u)F_D(t-u)du$ where $f_D(t)$ is the density of D. Equivalently, the characteristic function of RTT is $\Phi_{\text{RTT}}(\omega) = (\Phi_D(\omega))^2$. For example, if D is exponential with mean μ_D, the cumulative distribution of RTT is

$$F_{\text{RTT}}(t) = \begin{cases} 0, & t < 0 \\ 1 - (1 + t/\mu_D)e^{-t/\mu_D}, & t \geq 0. \end{cases}$$

The probability that the sending node generates m copies of the ith message, given that R_i retransmissions are allowed, is

$$\pi_m = \begin{cases} (1 - P_{rt}(i))P_{rt}^{m-1}(i), & m = 1, 2, \ldots, R_i \\ P_{rt}^{R_i}(i), & m = R_i + 1. \end{cases} \tag{14}$$

From this expression we can get the expected number of messages sent,

$$\begin{aligned} \overline{n}_i &= \sum_{m=1}^{R_i+1} m\pi_m \\ &= (1 - P_{rt}(i)) \sum_{m=1}^{R_i} mP_{rt}^{m-1}(i) + (R_i + 1)P_{rt}^{R_i}(i) \\ &= \frac{1 - P_{rt}^{R_i+1}(i)}{1 - P_{rt}(i)}, \end{aligned} \tag{15}$$

and the average total number of messages required for the exchange is

$$\overline{N}_{\text{MSG}} = \sum_{i=1}^{M} \overline{n}_i = \sum_{i=1}^{M} \frac{1 - P_{rt}^{R_i+1}(i)}{1 - P_{rt}(i)}. \tag{16}$$

This expression does not account for the MIH ACKs that are generated each time a node receives an MIH message. We can count those as well; an ACK is generated by a node each time its MIH User receives an MIH message. For the ith message in a M-message sequence, the sending node will produce m copies with probability π_m as defined in Equation (14). The probability that any one of these m copies is lost is p; the probability that k copies out of m are successfully received by the destination node is ${}_mC_k p^{m-k}(1-p)^k$. Therefore the expected number of MIH ACKs that are generated in connection with the ith message is

$$\begin{aligned} \overline{a}_i &= \sum_{m=1}^{R_i+1} \pi_m \sum_{k=0}^{m} \binom{m}{k} kp^{m-k}(1-p)^k \\ &= (1-p) \sum_{m=1}^{R_i+1} m\pi_m = (1-p)\overline{n}_i. \end{aligned} \tag{17}$$

To get the expected number of messages of both types sent during the exchange, we sum $\overline{a}_i$ over all messages $\{i\}$ and combine the resulting expression with the expected number of MIH messages from Equation (16), giving

$$\overline{N} = (2-p) \sum_{i=1}^{M} \frac{1 - P_{rt}^{R_i+1}(i)}{1 - P_{rt}(i)}. \tag{18}$$

From Equation (15), $\overline{n}_i \rightarrow R_i + 1$ as $P_{rt}(i) \rightarrow 1$, which happens when $p \rightarrow 1$ or when $T_{\text{MIH}}(i) \rightarrow 0$. We ignore the first case because the probability of the exchange failure is 1. In the second case, $\overline{N} \rightarrow 2\sum_{i=1}^{M}(R_i + 1)$, which is the maximum possible number of messages that can be generated; R_i+1 copies of the ith message are generated for all i because the timer immediately expires after each message transmission, and each copy results in an MIH ACK's being sent back.

We consider the example from Section II-A in which $M = 3$ and $R = 2$, and $\mu_D = 1$, so that all times are normalized to the average transit time. We plot $\overline{N}$ for three values of the packet error probability, p: 0.5, 0.1, and 10^{-6} in Fig. 4. The resulting curves show that the average number of messages per exchange is insensitive to changes in p over a wide range of values for the MIH timeout, particularly when the timeout is on the order of the average packet transit time across the network. As the packet loss probability increases to near unity, we see a significant deviation in $\overline{N}$; if we let $p \rightarrow 1$ we would obtain a horizontal line corresponding to $\overline{N} = M \cdot R = 9$. The curves in Fig. 4 suggest that we can prevent the number of messages generated per exchange from being excessively large if we let the MIH timeout be at least two times larger than the measured packet transit time.

So far, our treatment of the message exchange overhead has considered only those cases where the exchange completes successfully. In these cases, at least one of the $R_i + 1$ transmission opportunities succeeds at each of the M stages of the exchange. In general, however, exchanges fail if all of the allowed number of transmission attempts at one of the stages result in the MIH message's not reaching its destination. In order to assess the load that the MIH Users collectively offer to the network, we must consider the effect of transmissions associated with incomplete exchanges. The probability that all copies of the ith MIH message are lost is p^{R_i+1}. Thus the probability that the mth message is the last message in a given exchange is

$$\chi_m = \begin{cases} p^{R_1+1}, & m = 1 \\ p^{R_m+1} \prod_{i=1}^{m-1}(1 - p^{R_i+1}), & 1 < m < M \\ \prod_{i=1}^{M-1}(1 - p^{R_i+1}), & m = M \end{cases}$$

The expected number of MIH messages that are generated during an exchange, accounting for the loss of all R_i+1 copies

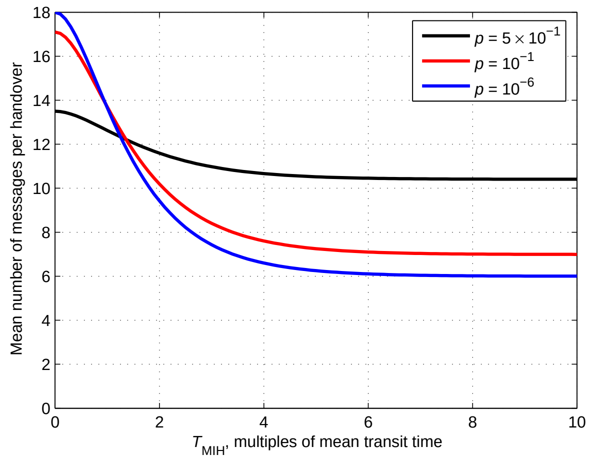

Fig. 4. Mean number of messages generated by MIH signaling endpoints during an exchange where $M = 3$ and $R_i = 2$ for $i = 1, 2, 3$.

of the ith message at any point in the exchange, is

$$\overline{N}_{\text{MSG}} = \sum_{m=1}^{M} \chi_m \sum_{i=1}^{m} \overline{n}_i.$$

We can obtain a simplified version of this expression if we assume that $R_i = R$ and $T_{\text{MIH}}(i) = T_{\text{MIH}}$ for $i \in \{1, 2, \ldots, M\}$. Then, $P_{rt}(i) = P_{rt}$ for $i \in \{1, 2, \ldots, M\}$ as well. Defining $q \triangleq 1 - p^{R+1}$ and $n \triangleq \frac{1-P_{rt}^{R+1}}{1-P_{rt}}$, we get

$$\begin{aligned}\overline{N}_{\text{MSG}} &= (1-q)n \sum_{m=1}^{M-1} mq^{m-1} + Mq^{M-1}n \\ &= \frac{(1-q^M)n}{1-q} \\ &= \frac{(1-P_{rt}^{R+1})(1-(1-p^{R+1})^M)}{(1-P_{rt})p^{R+1}}. \end{aligned} \quad (19)$$

Since $\overline{a}_i = (1-p)\overline{n}_i$, the total number of MIH messages in the exchange is $\overline{N} = (2-p)\overline{N}_{\text{MSG}}$.

III. MIH Signaling over TCP

MIH provides reliable transport over UDP by using internal timers to trigger retransmission of lost or excessively delayed messages. We consider the case of MIH over TCP, which is a connection-based, stream-oriented protocol that provides reliable transport of application data. Here, MIH allows TCP to be solely responsible for guaranteeing message reliability and does not use its own timeouts. We compute the performance metrics associated with a message exchange between MIH Users by using the techniques from Section II.

There has been voluminous work on TCP behavior. In particular, [11] and the work that builds on it, [12], have respectively characterized the loss and delay behavior of TCP during a steady-state bulk data transfer and during the transmission of an arbitrarily small amount of data. A TCP model for short-lived flows, such as those associated with MIH signaling, was developed in [13] but does not incorporate a maximum retransmission timeout (RTO). Our latency analysis considers the degenerate case in the model [12] in which the number of segments to be transferred at any time is one. In addition, we assume that the SYN/SYN-ACK exchange has already taken place during the start up of the MIH User.

For this discussion, we assume for the sake of tractability in the analysis that the size of each MIH message is equal to the size of the data payload of a TCP segment. Thus, each segment contains exactly one MIH message, which will be sent as soon as it is put into TCP's transmit buffer. The time to send each message is therefore ℓ s for all $i \in \{1, 2, \ldots, M\}$. In practice, an MIH message may not fit exactly within a TCP segment. This mismatch will present problems with regard to completing a message exchange in a timely manner. If the MIH message is smaller than a TCP segment, TCP will not send the segment until enough bits have been put into the transmission buffer by the MIH User. This is because TCP is a stream-oriented protocol rather than a packet-oriented one. If MIH timeouts, described in Section II, are in use, the MIH message may not get sent until enough duplicate copies are put into the TCP transmit buffer to fill a segment and trigger its transmission by TCP. If MIH is not using timeouts, the MIH message may not get sent at all. Conversely, if the MIH message is larger than a segment, the first portion of the MIH message, up to an integer multiple of a segment length, will get sent while the remainder of the message will remain in the TCP's buffer until enough additional bits are generated by the MIH User to cause the segment to be transmitted.

A. MIH over TCP Latency

In the absence of MIH timeouts, TCP will automatically retransmit messages if its own retransmission timer expires or if it receives three duplicate acknowledgments from the destination node at the other endpoint of the connection. We assume that there are no unacknowledged segments when the exchange begins. Because the TCP transmission window will be at least one segment long, the node initiating the message exchange will be able to transmit the first message immediately. If there is no packet loss, the destination node will generate an acknowledgment for the source node as soon as its TCP layer receives the source's segment. When the source node receives the acknowledgment, it will advance its own transmission window and possibly expand its congestion window, depending on whether it is in Slow Start mode or Congestion Avoidance mode.

Because we assume that the source node does not have any outstanding unacknowledged segments when it begins the exchange, it will not receive any duplicate acknowledgments from the destination if the first message is lost. Therefore, a source node will not respond to a packet loss by going into Fast Retransmit/Fast Recovery mode; instead, it will wait until its retransmission timer expires. When this happens, the source node will shrink its congestion window to one segment, retransmit the message, and double the RTO value. If subsequent retransmitted messages are also lost, the source node will continue to double its RTO value until it reaches a maximum, $\text{RTO}_{\max} = 2^K \text{RTO}$, where K is the maximum

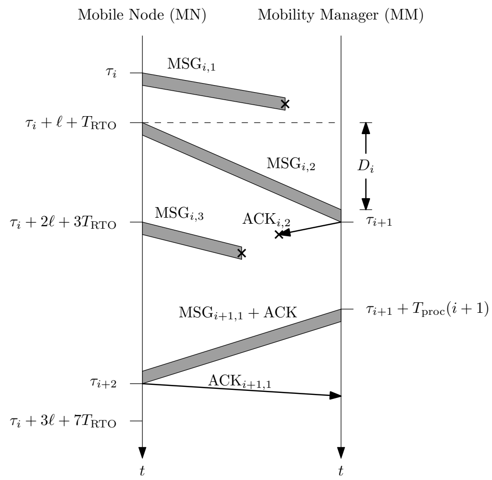

Fig. 5. Sample MIH signal flow between MN and MM when TCP transport is used and when some messages and TCP ACKs are lost in transit.

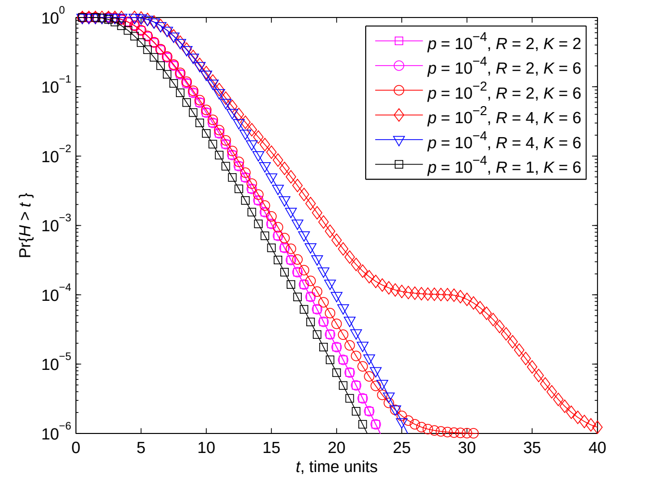

Fig. 6. Probability that the MIH message exchange time, H, exceeds t seconds for six sets of values of packet loss probability p, RTO, and number of RTO stages, K.

number of times that TCP doubles the RTO value. We assume that TCP will continue to send copies of the lost segment until it receives an acknowledgment from the destination endpoint.

We show an example of an MIH message transmission over TCP in Fig. 5. In the figure, the ith message in the message sequence, which is transmitted at time τ_i, is lost in transit. The lack of a TCP ACK causes the sending node's TCP to retransmit the segment and double the RTO. The first retransmission attempt succeeds in the sense that the MM receives the message and begins preparing a response message. However the TCP ACK is lost on the way back to the MN, so the MN's TCP sends a second copy of the segment at time $\tau_i + 2\ell + 3T_{\mathrm{RTO}}$ and doubles the RTO interval again. This segment is also lost, and the MN's TCP will send another copy if no ACK arrives by the time $\tau_i + 3\ell + 7T_{\mathrm{RTO}}$. Because the destination node receives the first transmission of the MIH message at time $\tau_{i+1} = \tau_i + \ell + D_i$, it generates a response MIH message at time $\tau_{i+1} + T_{\mathrm{proc}}(i+1)$. This message is contained within a TCP segment whose header's acknowledgment field contains the byte number that follows the last byte of the ith message. When the MN receives this message at time τ_{i+2}, its TCP layer notes the acknowledgment and stops retransmission of the ith message.

Our latency analysis is similar to our treatment of MIH over UDP from Section II. The principal differences lie in the exponential expansion of TCP's timeout window, up to a maximum amount, with each successive timeout and the lack of a limit on the number of retransmission attempts. Note that the use of TCP does not affect the time penalty associated with each successful transit, nor does it affect the processing times for each message at the MIH layer. Thus the only change is a new expression for the additional time penalty imposed on the ith message in the message sequence. Because the RTO doubles with each retransmission, the time when the rth retransmission occurs (given the first transmission attempt occurred at time τ_i) is $\tau_i + r\ell + T_{\mathrm{RTO}} + 2T_{\mathrm{RTO}} + \cdots + 2^{r-1}T_{\mathrm{RTO}} = \tau_i + r\ell + (2^r - 1)T_{\mathrm{RTO}}$ for $r = 1, 2, \ldots, K$. Immediately after the rth retransmission attempt, TCP expands the RTO to $2^r T_{\mathrm{RTO}}$. At the Kth retransmission attempt, the retransmission occurs at time $\tau_i + K\ell + 2^K T_{\mathrm{RTO}} - T_{\mathrm{RTO}} = \tau_i + K\ell + \mathrm{RTO}_{\max} - T_{\mathrm{RTO}}$, after which the RTO expands to $\mathrm{RTO}_{\max}$. For subsequent retransmissions ($r > K$), the RTO remains at its maximum value and the rth retransmission occurs at time $\tau_i + r\ell + (r - K + 1)\mathrm{RTO}_{\max} - T_{\mathrm{RTO}}$. The characteristic function of the extra delay Δ_i^{C} caused by packet loss when MIH is using TCP transport (i.e. in connection mode) is thus

$$\begin{aligned}
&\Phi_{\Delta_i^{\mathrm{C}}}(\omega) \\
&= (1-p)\sum_{r=0}^{K} p^r e^{j\omega[r\ell + (2^r - 1)T_{\mathrm{RTO}}]} \\
&\quad + (1-p)\sum_{r=K+1}^{\infty} p^r e^{j\omega[r\ell + (r-K+1)\mathrm{RTO}_{\max} - T_{\mathrm{RTO}}]} \\
&= (1-p)\sum_{r=0}^{K} p^r e^{j\omega[r\ell + (2^r - 1)T_{\mathrm{RTO}}]} \\
&\quad + \frac{(1-p)p^{K+1} e^{j\omega[(K+1)\ell + 2\mathrm{RTO}_{\max} - T_{\mathrm{RTO}}]}}{1 - pe^{j\omega(\ell + \mathrm{RTO}_{\max})}}. \qquad (20)
\end{aligned}$$

Inserting this expression into Equation (4) in place of $\Phi_{\Delta_i^{\mathrm{D}}}(\omega)$ gives us the characteristic function for H. By inverting $\Phi_H(\omega)$ to obtain $F_H(t)$ using Equation (6), we can again get $\Pr\{H > t\} = 1 - F_H(t)$ for various values of t. We plot performance curves for the case where $M = 3$ in Fig. 6, which shows that, similar to the UDP case, latency is minimized by using small timeouts and fewer backoff stages. In addition, we observe the same plateau effects that we saw in the UDP case; they are most noticeable when the packet loss rate is high.

We can use the characteristic function to obtain the expected value and the variance of the exchange time, H. The expression for μ_H is nearly identical to Equation (9); the only change is replacing $\mu_{\Delta_i^{\mathrm{D}}}$ with the expected value of Δ_i^{C}, which we obtain from Equation (20) as follows:

$$\mu_{\Delta_i^{\mathrm{C}}} = \frac{p}{1-p}\ell + \frac{(1 - p - 2^K p^{K+1})p}{(1-p)(1-2p)} T_{\mathrm{RTO}}. \qquad (21)$$

The second fraction assumes the indeterminate form $0/0$ when $p = 1/2$. We can use L'Hôpital's Rule to get

$$\lim_{p \to 1/2} \mu_{\Delta_i^{\mathrm{C}}} = \ell + \frac{K+2}{2} T_{\mathrm{RTO}}.$$

Equation (21) is closely related to Equation (18) in [12], which itself comes from [11]. We can manipulate our Equation (21) to produce Equation (18), with an additional factor of p, by letting $\ell \to 0$ and observing that the general form of Equation (19) from [12], $G(p) = 1 + \sum_{k=1}^{K} 2^{k-1} p^k$, can be written as

$$G(p) = \frac{1 - p - 2^K p^{K+1}}{1 - 2p}.$$

The cause of the additional factor of p is that Equation (18) gives the average duration of Z^{TO}, the TCP timeout period, given that a timeout occurred; thus Z^{TO} must be at least T_{RTO} time units long. Our Equation (21) gives the average time to transmit a segment in addition to the network transit delay; the occurrence of a timeout is not a given condition, and so Δ_i^{C} has a minimum value of zero. We can also relate Equation (21) to the expression for C_1^1 (the mean time to send one segment with an initial congestion window size of one segment) in [13] by letting $K \to \infty$, since their model assumes no upper bound on the RTO value. The expression in [13] has an additional factor of p as well, because we are computing the additional delay due to message losses.

The variance of H when MIH messages use TCP is given by Eq. (11) with $\sigma^2_{\Delta_i^{\mathrm{C}}}$ replacing $\sigma^2_{\Delta_i^{\mathrm{D}}}$. We get the variance of Δ_i^{C} in the same way that we got Equation (12), which gives us a more complex but still closed-form expression:

$$\begin{aligned}\sigma^2_{\Delta_i^{\mathrm{C}}} &= 2\tfrac{(1-p)^2-(2p)^K p(2+K-3(K+1)p+2Kp^2)}{(1-p)^2(1-2p)^2} p\ell T_{\mathrm{RTO}} \\ &+ \Big[\tfrac{1-p}{(1-2p)^2(1-4p)} + \tfrac{(2p)^{K+1}(1-6p-(2^{K+4}-9)p^2-4p^3)}{(1-p)^2(1-2p)^2(1-4p)} \\ &- \tfrac{(4p)^K p(7-29p-4p^3+p^{K+2}-4p^{K+3})}{(1-p)^2(1-2p)^2(1-4p)}\Big] p T^2_{\mathrm{RTO}} + \tfrac{p\ell^2}{(1-p)^2}.\end{aligned} \tag{22}$$

This quantity becomes undefined when $p = 1/2$ or $p = 1/4$. As in the case of $\mu_{\Delta_i^{\mathrm{C}}}$, we can take limits and get

$$\begin{aligned}\lim_{p \to 1/4} \sigma^2_{\Delta_i^{\mathrm{C}}} &= \frac{4}{9}\ell^2 + \frac{2(-5 + 9 \cdot 2^K) - 3K}{9 \cdot 2^K} \ell T_{\mathrm{RTO}} \\ &+ \frac{4^K(27K-1) + 9 \cdot 2^{K+1} - 1}{9 \cdot 4^{K+1}} T^2_{\mathrm{RTO}}\end{aligned} \tag{23}$$

and

$$\begin{aligned}\lim_{p \to 1/2} \sigma^2_{\Delta_i^{\mathrm{C}}} &= 2\ell^2 + \frac{8 + 5K + K^2}{2} \ell T_{\mathrm{RTO}} \\ &+ \frac{2(-9 + 13 \cdot 2^K) - 8K - K^2}{4} T^2_{\mathrm{RTO}}.\end{aligned} \tag{24}$$

We show plots of μ_H and σ^2_H in Fig. 7, using the same parameters that we used to generate Fig. 3. Here we consider two values of the TCP RTO value, μ_D and $10\mu_D$. Note that the performance of UDP and TCP is identical for small values of p, and that the value of p for which there is a noticeable increase in the value of μ_H and σ^2_H is around the same for both the UDP and TCP transport cases. When TCP is used,

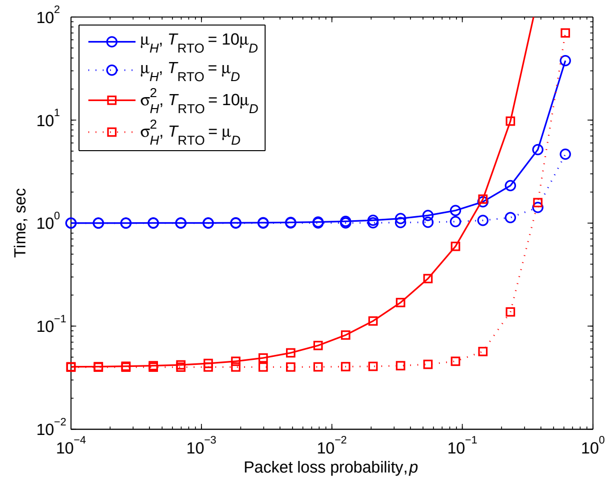

Fig. 7. Mean and variance of the handover time H versus packet loss probability, p, for short and long TCP RTO values ($T_{\mathrm{RTO}} = \mu_D$ and $T_{\mathrm{RTO}} = 10\mu_D$, respectively).

the mean and variance of H increase without bound as $p \to 1$, because TCP attempts to deliver an unacknowledged segment until it receives an ACK. The gain from doing this is the eventual success of the transmission as long as $p \neq 1$, but this benefit is less meaningful in the case of signaling for handovers, because the amount of time available to complete message delivery is short. Again, this figure demonstrates the importance of beginning critical handover signaling exchanges early, so that degrading link conditions do not expand the amount of time required to successfully deliver signaling messages.

B. MIH over TCP overhead

We also can compute the expected number of messages generated during a message exchange as a function of the RTO value T_{RTO} and the packet loss probability, p. Similar to the UDP case, retransmission of an MIH message occurs if the message is lost, its TCP ACK is lost or the RTO timer expires before the sender receives the ACK. However, we have to account for the exponential dilation of the timeout interval until it reaches its final value, $\mathrm{RTO}_{\max}$. Thus, similar to Equation (13), the probability that the kth transmission attempt fails and requires another attempt is

$$\phi_k = \begin{cases} 1 - (1-p)^2 F_{\mathrm{RTT}}(2^{k-1} T_{\mathrm{RTO}}), & k \leq K \\ 1 - (1-p)^2 F_{\mathrm{RTT}}(\mathrm{RTO}_{\max}), & k > K. \end{cases} \tag{25}$$

where we assume that T_{RTO} is the same for all M messages in the message sequence. We assume that there is no limit on the number of times TCP attempts to retransmit a segment. The probability that m copies of the ith message gets sent is the probability that the TCP ACK for the mth transmitted copy of the MIH message arrives within the time limit after $m-1$ failures. Defining $psi \triangleq 1 - (1-p)^2 F_{\mathrm{RTT}}(\mathrm{RTO}_{\max})$, (i.e. $\phi_k = \psi$ for $k > K$) we have

$$\pi_m = \begin{cases} 1-\phi_1, & m=1 \\ (1-\phi_m)\prod_{k=1}^{m-1}\phi_k, & 1 < m \le K \\ (1-\psi)(\psi)^{m-K-1}\prod_{k=1}^{K}\phi_k, & m > K \end{cases} \tag{26}$$

for $m \in \mathbb{Z}^+$, the set of positive integers. Using this expression we can get the mean number of copies sent for a given message:

$$\begin{aligned} \overline{n}_i &= \sum_{m=1}^{\infty} m\pi_m \\ &= 1-\phi_1 + \sum_{m=2}^{K} m(1-\phi_m)\prod_{k=1}^{m-1}\phi_k \\ &\quad + \sum_{m=K+1}^{\infty} m(1-\psi)(\psi)^{m-K-1}\prod_{k=1}^{K}\phi_k \\ &= 1-\phi_1 + \sum_{m=2}^{K} m(1-\phi_m)\prod_{k=1}^{m-1}\phi_k \\ &\quad + \frac{1+K(1-\psi)}{1-\psi}\prod_{k=1}^{K}\phi_k. \end{aligned} \tag{27}$$

As was the case with MIH transport over UDP, k out of m segments generated by a source will arrive at their destination (and cause ACKs to be sent back) with probability ${}_mC_k p^{m-k}(1-p)^k$. Thus the expected number of ACKs is $(1-p)\overline{n}_i$, and the total number of messages created in connection with the ith phase of the exchange is $(2-p)\overline{n}_i$. Because the retransmission parameters do not change from message to message, $\overline{n}_i$ is a constant, and the mean total number of messages sent during an exchange is $\overline{N} = (2-p)M\overline{n}$, where $\overline{n}$ is given in Equation (27).

In Fig. 8, we plot the average number of messages in a handover versus RTO, where the RTO is expressed as multiples of the average one-way network transit time, μ_D. The handover exchange is composed of $M = 3$ MIH messages where TCP is used for transport and there are no timeouts and retransmissions at the MIH User. The figure bears a strong resemblance to Fig. 4, which plots $\overline{N}$ versus the MIH timeout for the MIH/UDP case. An important difference is the asymptote at RTO $= 0$, which is due to our assumption that TCP never stops trying to send an un-ACKd segment. In contrast, MIH is limited to a finite number of retransmission attempts when it uses UDP. The MIH/TCP case is similar to the MIH/UDP case in that reducing the timeout to be on the order of the network round-trip time significantly increases $\overline{N}$ in both cases. Thus a conservative RTO estimate reduces overhead, but this is a TCP function that is not controllable by the MIH User.

C. The Effect of Combining MIH Timeouts with TCP

In this subsection, we present an analytical model that quantifies some of the negative effects that result from combining the TCP layer and MIH User retransmission mechanisms. The MIH User generates an MIH ACK message for each MIH message that it receives. Our model examines the case of an MIH Indication message transmission from a MN to the MM. This exchange involves one MIH message and one MIH ACK, for a total of two messages generated jointly by both nodes' MIH Users. In addition to the MIH messages, the connection endpoints generate TCP ACKs every time they receive a TCP segment.

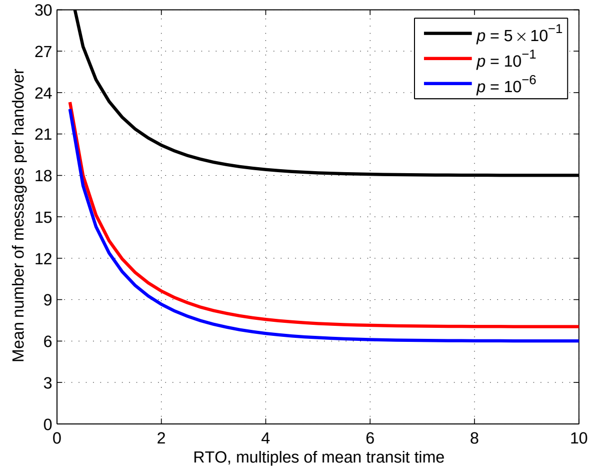

Fig. 8. Mean number of messages generated by MIH signaling endpoints using TCP transport during a handover where $M = 3$ and $K = 6$, for various RTO values.

There are two cases involving the relative values of the MIH timeout T_{MIH} and the TCP timeout, T_{RTO}. We consider only the case where the MIH timeout occurs later than the initial value of the TCP RTO ($T_{\text{MIH}} \ge T_{\text{RTO}}$). Letting the MIH timeout occur before the RTO would introduce additional copies of the message into the TCP queue before TCP has a chance to time out and retransmit, resulting in unnecessary extra traffic. In this situation, the MIH User's timeout occurs after the TCP RTO, so the MIH layer will not begin generating duplicate messages until TCP has been in Timeout mode for some period of time. The number of messages that the MIH User generates depends on the amount of time that TCP spends in Timeout retransmitting the original copy of the message. We first examine the Request message from the MN to the MM. The MIH layer will produce m additional copies of the 1st message, $0 \le m \le R_1$, if the time from the first message transmission attempt until the TCP layer receives an ACK, U_1, lies in the half-open interval $\big[mT_{\text{MIH}}(1), (m+1)T_{\text{MIH}}(1)\big)$. The probability of this event is

$$\begin{aligned} &\Pr\{U_1 \in \big[mT_{\text{MIH}}(1), (m+1)T_{\text{MIH}}(1)\big)\} \\ &\quad = \int_{mT_{\text{MIH}}(1)}^{(m+1)T_{\text{MIH}}(1)} f_{U_1}(u)\,du \\ &\quad = F_{U_1}((m+1)T_{\text{MIH}}(1)) - F_{U_1}(mT_{\text{MIH}}(1)). \end{aligned} \tag{28}$$

Because MIH is limited to a finite number of retransmissions, R_1, it follows that the probability of exactly R_1 retransmissions is the probability that U_1 is greater than $R_1T_{\text{MIH}}(1)$, which is $1 - F_{T_M}(R_1T_{\text{MIH}}(1))$.

From our analysis in Section III, we know that $\tau_2 - \tau_1 = D_1 + \Delta_1^{\text{C}}$, so we have already characterized the random time from the first transmission attempt to the segment's successful reception at the MM; its characteristic function is $\Phi_D(\omega)\Phi_{\Delta_1^{\mathrm{C}}}(\omega)$. We next get the characteristic function of U_1, the time from the 1st message's reception at the MM to the arrival of a TCP ACK at the MN. There are two ways the MN can get a TCP ACK from the MM. The MN can receive a TCP ACK in the header of the MIH ACK that the MM's MIH User generates when it receives the MN's MIH message; this assumes that the MM's TCP uses delayed ACKs. The MN can also receive a TCP ACK in the header of the next MIH message in the handover sequence, as long as the MM's transmission window is large enough to send both the MIH ACK and the second MIH message. When any one of these packets first arrives at the MN, the local TCP will note that the transmitted segment was correctly received. If the ACK is received before T_{RTO} units of time have expired, TCP will continue normal operations. If the MN receives the ACK after TCP has gone into timeout, TCP will end the timeout and begin Slow Start.

We compute the characteristic function for $\Delta_{1,\mathrm{ACK}}^{\mathrm{C}}$, the additional time required to send the MIH ACK for MSG_1 from the MM to the MN. There are two possibilities, depending on whether $T_{\mathrm{proc}}(2)$ is larger than T_{RTO}. If $T_{\mathrm{proc}}(2) \geq T_{\mathrm{RTO}}$ and the MIH ACK gets lost in transit, the MM's TCP will time out before the 2nd MIH message is ready to be sent; the message will go into the transmission queue until the MM's TCP receives an ACK. The MM's TCP will behave exactly like the MN's TCP, retransmitting the MIH ACK's segment and expanding the RTO at each attempt until it reaches its maximum value, $\mathrm{RTO}_{\max} = 2^K T_{\mathrm{RTO}}$.

In Fig. 9, we show the exchange of messages between the MN and MM when MIH reliability mechanisms (ACKs and retransmissions) are used in addition to TCP, and when $T_{\mathrm{proc}}(2) > T_{\mathrm{RTO}}$. The 1st MIH message travels from the MN to the MM and is delayed by D_1 units of time. There is an additional time penalty of Δ_1^{C} units associated with failed attempts by the MN's TCP to transmit MSG_1. Immediately upon receiving MSG_1, the MM's MIH layer sends an MIH ACK to the MN whose TCP header contains the sequence number of the last octet in MSG_1. We assume that all MIH messages and MIH ACKs have the same transmission duration of ℓ units of time. The delay experienced by the MIH ACK is $D_{1,\mathrm{ACK}}$, which has the same characteristic function, $\Phi_D(\omega)$, as D_1. Failed transmission attempts by the MM's TCP cause an additional delay of $\Delta_{1,\mathrm{ACK}}^{\mathrm{C}}$. When the MIH ACK finally arrives at the MN, the MN's TCP immediately responds with a 40-byte TCP ACK, which reaches the MM after a delay of $D'_{1,\mathrm{ACK}}$ time units, where $D'_{1,\mathrm{ACK}}$ also has characteristic function $\Phi_D(\omega)$. Like [12], we assume that TCP ACKs are never lost in transit. From Fig. (9),

$$\Phi_{U_1}(\omega) = \left(\Phi_D(\omega)\right)^2 \Phi_{\Delta_1^{\mathrm{C}}}(\omega)\Phi_{\Delta_{1,\mathrm{ACK}}^{\mathrm{C}}}(\omega)e^{j2\omega\ell}. \qquad (29)$$

where $\Phi_{\Delta_{1,\mathrm{ACK}}^{\mathrm{C}}}(\omega)$ is given by Equation (20).

If $T_{\mathrm{proc}}(2) \leq T_{\mathrm{RTO}}$, and the MM's TCP transmission window is at least two segments long, the MM will send the 2nd MIH message before the RTO expires, as shown in Fig. 10. In the figure, we show the case where both the MM's MIH ACK and MSG_2 transmissions fail. If no ACK comes back from the MN by the time $t = \tau_2 + T_{\mathrm{RTO}}$, the MM's TCP will enter timeout, shrink its congestion window to one segment, and begin retransmitting the MIH ACK while the 2nd MIH message waits in the transmission queue. So the MM will have one attempt to send the 2nd MIH message before its TCP goes into timeout. The MN has two chances to receive a segment containing an ACK for the 1st MIH message and send an ACK back to the MM. Because of this, the characteristic function for $\Delta_{1,\mathrm{ACK}}^{\mathrm{C}}$ changes slightly from its form in Equation (20), as we now show.

In this case, $\Delta_{1,\mathrm{ACK}}^{\mathrm{C}}$ takes the values 0 and $T_{\mathrm{proc}}(2)$ with probabilities $(1-p)$ and $(1-p)p$, respectively. The first retransmission of the MIH ACK happens at time $t = \tau_2 + \ell + T_{\mathrm{RTO}}$ if no ACK is received for either the MIH ACK or the 2nd MIH message, so $\Delta_{1,\mathrm{ACK}}^{\mathrm{C}} = \ell + T_{\mathrm{RTO}}$ with probability $(1-p)p^2$. Continuing this pattern and using the results of our analysis in Section III, we see that $\Delta_{1,\mathrm{ACK}}^{\mathrm{C}} = r\ell + (2^r - 1)T_{\mathrm{RTO}}$ with probability $(1-p)p^{r+1}$, for $r = 1, 2, \ldots, K$. Recalling that the maximum RTO is $\mathrm{RTO}_{\max} = 2^K T_{\mathrm{RTO}}$, we see that $\Delta_{1,\mathrm{ACK}}^{\mathrm{C}} =$

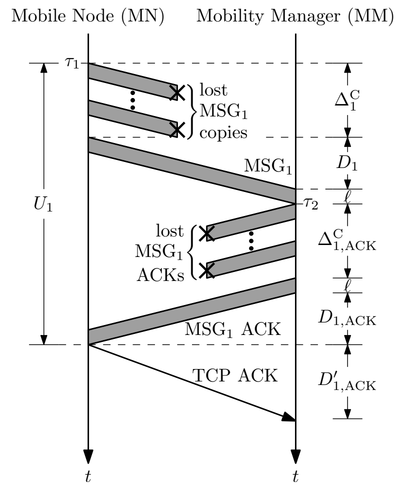

Fig. 9. Signal flow for an exchange of an MIH message and MIH ACK message between MN and MM over TCP, when $T_{\mathrm{proc}}(2) > T_{\mathrm{RTO}}$.

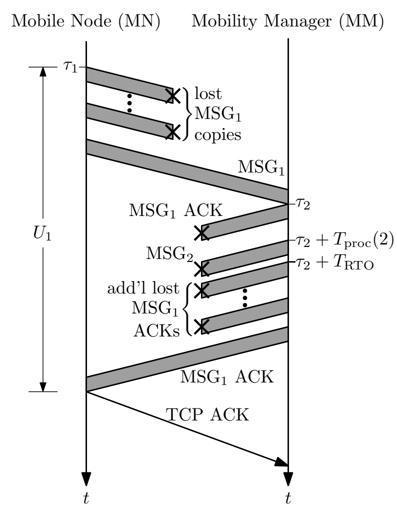

Fig. 10. Signal flow for an exchange of an MIH message and MIH ACK message between MN and MM over TCP, when $T_{\mathrm{proc}}(2) \leq T_{\mathrm{RTO}}$.

$r\ell + (r-K+1)\text{RTO}_{\max} - T_{\text{RTO}}$ with probability $(1-p)p^{r+1}$ for $r \geq K$. Thus we obtain the following expression for the characteristic function when $T_{\text{proc}}(2) < T_{\text{RTO}}$:

$$\begin{aligned}
&\Phi_{\Delta^{\text{C}}_{1,\text{ACK}}}(\omega) \\
&= (1-p) + (1-p)pe^{j\omega T_{\text{proc}}(2)} \\
&\quad + (1-p)\sum_{r=1}^{K} p^{r+1} e^{j\omega[r\ell + (2^r-1)T_{\text{RTO}}]} \\
&\quad + (1-p)\sum_{r=K+1}^{\infty} p^{r+1} e^{j\omega[r\ell + (r-K+1)\text{RTO}_{\max} - T_{\text{RTO}}]} \\
&= (1-p)[1 + pe^{j\omega T_{\text{proc}}(2)}] \\
&\quad + (1-p)\sum_{r=1}^{K} p^{r+1} e^{j\omega[r\ell + (2^r-1)T_{\text{RTO}}]} \\
&\quad + \frac{(1-p)p^{K+2} e^{j\omega[(K+1)\ell + 2\text{RTO}_{\max} - T_{\text{RTO}}]}}{1 - pe^{j\omega(\ell + \text{RTO}_{\max})}}. \qquad (30)
\end{aligned}$$

From Fig. (10), we can work out the characteristic function of U_1 by conditioning on the whether MSG_2 is lost in transit, as follows:

$$\begin{aligned}
\Phi_{U_1}(\omega) &= (1-p)\big(\Phi_D(\omega)\big)^2 \Phi_{\Delta^{\text{C}}_1}(\omega)\Phi_{\Delta^{\text{C}}_{1,\text{ACK}}}(\omega)e^{j2\omega\ell} \\
&\quad + p\big(\Phi_D(\omega)\big)^2 \Phi_{\Delta^{\text{C}}_1}(\omega)e^{j\omega(2\ell + T_{\text{proc}}(2))} \\
&= \Big[(1-p)\Phi_{\Delta^{\text{C}}_{1,\text{ACK}}}(\omega) + pe^{j\omega T_{\text{proc}}(2)}\Big] \\
&\quad \times \big(\Phi_D(\omega)\big)^2 \Phi_{\Delta^{\text{C}}_1}(\omega)e^{j2\omega\ell},
\end{aligned}$$

where $\Phi_{\Delta^{\text{C}}_{1,\text{ACK}}}(\omega)$ is given by Equation (30).

With the characteristic function $\Phi_{U_1}(\omega)$ in hand, we can get an estimate of the MIH overhead by computing the size of the backlog of copies of the MIH Indication message at the MN at the time the MN receives a TCP ACK for the MIH Request message. Using Equation (28) and the inversion formula in Equation (6), we get $\beta(m) \triangleq \text{Pr}\{m \text{ messages in backlog}\}$:

$$\begin{aligned}
\beta(m) &= \text{Pr}\{mT_{\text{MIH}}(1) \leq U_1 < (m+1)T_{\text{MIH}}(1)\} \\
&= F_{U_1}\big((m+1)T_{\text{MIH}}(1)\big) - F_{U_1}\big(mT_{\text{MIH}}(1)\big) \\
&= \frac{2}{\pi}\int_0^{\infty} \text{Re}[\Phi_{U_1}(\omega)]\Big[\sin\big((m+1)\omega T_{\text{MIH}}(1)\big) \\
&\qquad - \sin\big(m\omega T_{\text{MIH}}(1)\big)\Big]\,\frac{d\omega}{\omega} \qquad (31)
\end{aligned}$$

for $m = 0, 1, 2, \ldots, R_1 - 1$ and

$$\begin{aligned}
\beta(R_1) &= \text{Pr}\{U_1 > R_1 T_{\text{MIH}}(1)\} \\
&= 1 - F_{U_1}\big(R_1 T_{\text{MIH}}(1)\big) \\
&= 1 - \frac{2}{\pi}\int_0^{\infty} \text{Re}[\Phi_{U_1}(\omega)]\sin\big(\omega R_1 T_{\text{MIH}}(1)\big)\frac{d\omega}{\omega}. \qquad (32)
\end{aligned}$$

The expected number of messages in the backlog is $\overline{B} = \sum_{m=0}^{R_1} m\beta(m)$. Thus the average number of MIH messages put into the TCP transmission queue for each MIH message that the MIH User needs to send is the original message plus the ones in the backlog, or $1 + \overline{B}$.

In Fig. 11, we we plot the average backlog size versus p, the probability of MIH message loss for three different values of the MIH timeout, T_{MIH}: 150 ms, 200 ms, and 400 ms. We used the set of parameters shown in Table I, including $T_{\text{RTO}} = 200$ ms; we also used $T_{\text{RTO}} = 2$ ms to show the effect of increasing the TCP retransmission rate. For all three values of T_{MIH} and for $T_{\text{RTO}} = 200$ ms, $\overline{B}$ is insensitive to p for $p < 0.01$. The figure also shows that increasing the MIH timeout reduces the size of the backlog. Computing $\overline{B}$ using different values for T_{proc} does not result in a significant change in the curves. Note that using a smaller value of T_{RTO} produce curves that have the same value for small values of p as the curves where $T_{\text{RTO}} = 200$ ms; the small T_{RTO} curves are flatter and begin increasing toward R at a larger value of p. This is because decreasing $T_{\text{RTO}} = 200$ increases the TCP retransmission rate and it becomes more likely that a TCP ACK will arrive at the MN before the MIH timeout expires, thus reducing the backlog, although it will increase the TCP segment transmission rate.

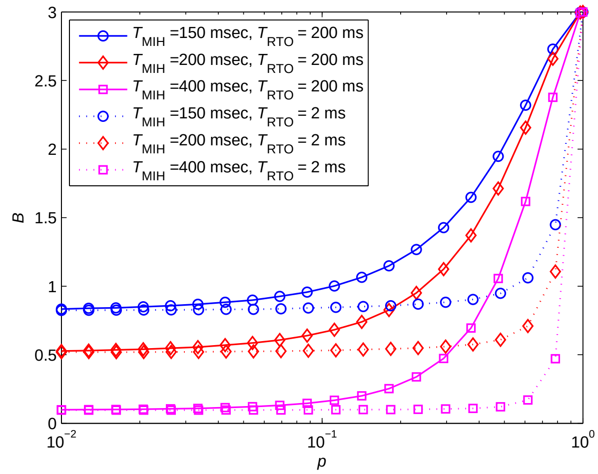

Fig. 11. Plot of $\overline{B}$, the average size of the message backlog generated by MIH retransmissions due to expiration of the MIH timer, versus p, the probability of message loss for different values of the MIH timeout, T_{MIH}.

IV. Simulation Results

To further quantify the performance of the MIH User over different transport layers, we performed a set of simulations using the ns-2 tool, which we enhanced by extending the tool's mobility framework [14]. To generate our results, we used a simple network topology shown in Fig. 12 that consists of a mobility manager located at a remote Point of Service (PoS) connected to an access point via a backbone network that we have abstracted as a single router and a lossy link. Within the coverage area of the access point is a single mobile node that communicates with the access point over a IEEE 802.11 wireless LAN link.

In each scenario that we considered, the MN first connects to the AP and registers with the PoS. Then, depending on whether we are simulating Indications or Request/Response exchanges, the MN generates Indications every 0.5 second or the PoS generates requests every 0.5 second, respectively. We take performance measurements by taking traces of the relevant output parameters and averaging those traces over

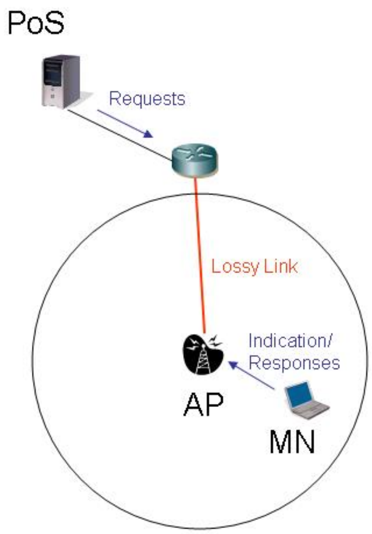

Fig. 12. Network topology used in simulations of Indication and Request/Response MIH message exchanges.

TABLE I
SIMULATION PARAMETERS

IEEE 802.11	
Data rate (Mb/s)	11Mb/s
Coverage area radius (m)	50
Links	
Speed (Mb/s)	100
Delay (s)	0.01
UDP	
Max packet size (byte)	1000
Header size (bytes)	8
TCP	
Max Segment Size (bytes)	1280
Min RTO (s)	0.2
Max retransmission	Unlimited
Queue size	Unlimited
Header size (bytes)	20
IP header	
IPv6 header (bytes)	40
MIH Function	
Transaction timeout (s)	none
R_i	2
$T_{\text{proc}}(2)$ (s)	0.2
Simulation configuration	
Duration (s)	6005
loss probability, p	variable [0, 50%]
$\text{RTO}_{\max}$ (s)	0.2, 0.3, 0.5, 0.75, 1
Indications/s	2
Requests/s	2
MIH Packet size (bytes)	200

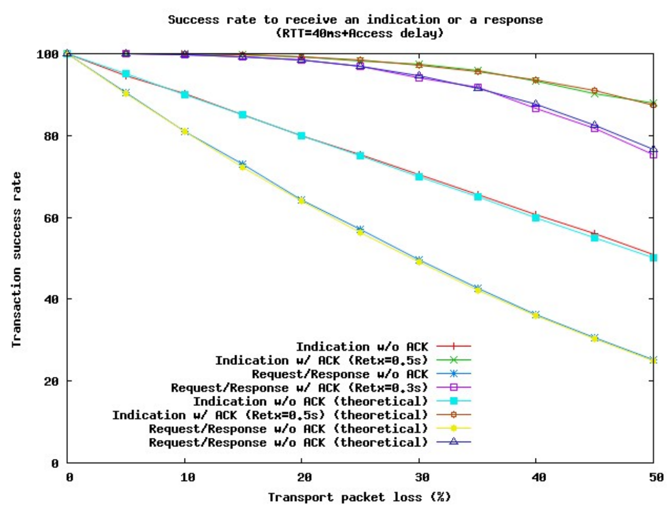

Fig. 13. Theoretical and simulated values for MIH transaction completion probability.

4000 seconds of simulation time, between the 5 s and 4005 s marks. The typical standard deviation of the data in each of the graphs in this section is on the order of 1% of the mean data value plotted; thus error bars would not be visible and so are not plotted.

The parameters that we used in our simulations are shown in Table I. The simulations examined two types of MIH message exchanges. The first type consists of a Indication message sent by the mobile node to the mobility manager. The second type of exchange involves a Request message generated by the mobility manager which produces a Response message from the mobile node. Both types of MIH message exchanges occur according to a Poisson process with an average exchange arrival rate of two events per second. Each simulation run covered a time interval of 6005 seconds. The packet loss rate on the link connecting the router to the access point was allowed to vary between 0 and 0.5. Two different transport layers were used in the simulations: UDP and TCP. The parameters for both transport layers are given in Table I. We used a variety of values for the TCP maximum RTO, as shown in the table, and varied the link loss rate for each maximum RTO value that we used.

We plot values for the probability that MIH transaction completes successfully in Fig. 13. We compare the results from ns-2 with theoretical results that we obtained by computing $1 - P_{\text{fail}}$ from Equation (2). In all the scenarios that we considered, we obtained excellent agreement between the values predicted by the model and the results that we obtained from the simulations. The graph further shows that the probability that a message exchange is unable to complete successfully decreases as the number of the packets in the exchange increases, as we would expect. In addition, using MIH acknowledgments significantly increased the success rate.

A. MIH Delay

In this subsection, we examine the delay performance of MIH over both UDP and TCP transport layers. In all of the figures in this subsection, the sets of parameters that we used are indicated in the legend appearing in the figure.

In Fig. 14 we plot the average time required to complete the simple Indication transmission as well as the Request/Response message exchange. The bars in the figure do not show the variance of the simulation data; rather they show the value of the standard deviation of the completion time, i.e. the upper terminus of a bar is located at a value one standard deviation greater than the corresponding mean. We examine the delay with MIH acknowledgment messages and retransmissions (for which $T_{\text{MIH}} = 0.5$ s for Indications and $T_{\text{MIH}} = 0.3$ s for Request/Response) and without. In the figure, there are two sets of horizontal marks that corre-

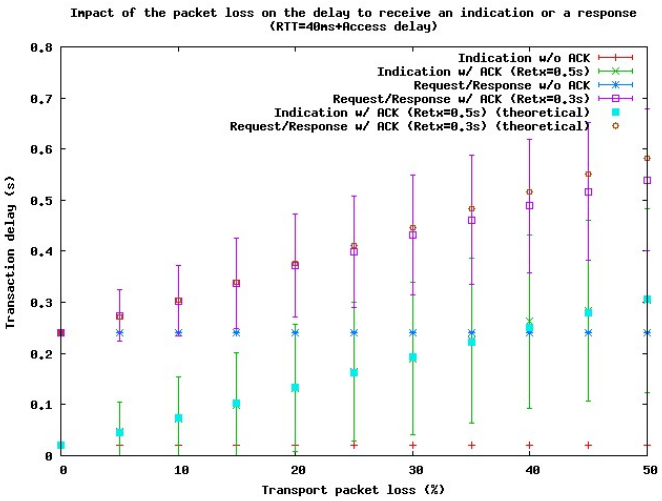

Fig. 14. Theoretical and simulation values of mean MIH handover time for MIH handovers where $M = 1$ (Indication) and $M = 2$ (Request/Response) over UDP.

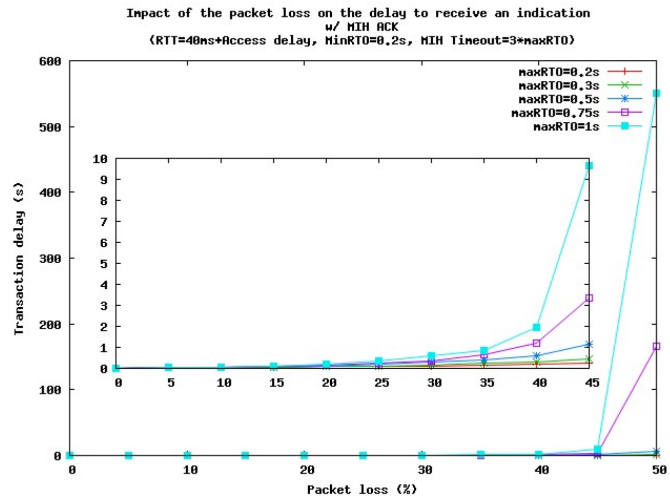

Fig. 15. Mean MIH Indication transaction completion time over TCP transport with MIH timeouts and acknowledgments.

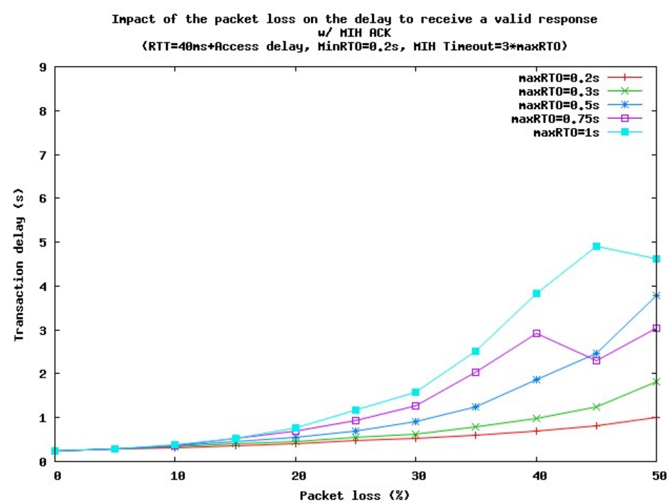

Fig. 16. Mean MIH Request/Response transaction completion time over TCP transport with MIH timeouts and acknowledgments.

spond to the scenarios in which MIH Indications and MIH Request/Response exchanges take place without the use of MIH ACKs. In both cases, the message exchange will fail unless the initial attempt to transmit each MIH message is successful. Thus, the only variance in the exchange completion time comes from random delays in the network which, in this case, are very small. We have an average delay of 20 ms and 240 ms for the Indication and Request/Response exchanges, respectively. The delay curves also show strong agreement between the simulation results and the mathematical model's prediction from Equation (9) and Equation (10), especially for small values of p.

In Fig. 15 we show simulation results showing the mean message exchange time for Indications when TCP is used in addition to MIH ACKs and timeouts. The MIH timeout T_{MIH} is set to $3\text{RTO}_{\max}$. The figure shows much greater sensitivity to the packet loss probability as the transaction delay increases exponentially with respect to the value of T_{RTO}. In addition, using MIH ACKs over TCP further increases the average delay. When MIH ACKs are used and the average TCP round-trip delay is greater than the MIH retransmission timeout interval, MIH will create duplicate packets that go into the TCP queue. These duplicate packets cause additional delays because all of them must be transmitted and acknowledged before the next MIH Indication message can be sent.

We also show simulation results showing the mean transaction completion time for Request/Response exchanges when TCP is used in addition to MIH ACKs and timeouts in Fig. 16. We observe that the exchange completion time decreases when the packet loss reaches 45% for larger values of $\text{RTO}_{\max}$. This is because the delays at this level of packet loss are so large and most Request/Response transactions do not complete within the ACK timeout interval. Thus the delay shown in the figure is associated only with the minority of successful exchanges.

To further show the negative effect of combining MIH retransmissions with TCP's reliable delivery, we plot the average Indication completion time for the case where we use TCP transport without MIH reliability features in Fig. 17. If no MIH ACK is used, the MIH User sends a Request and waits for a Response. In our scenario, the Responses received are always considered valid although that might not be the case in real implementations. The figure does not exhibit the type of severe performance degradation that we observed in Fig. 15, largely because the TCP transmission queue is no longer being filled with extra copies of messages that were lost during transmission attempts. If we compare the average delay to what we observed in connection with UDP, we see that the average completion time is much higher in the case of TCP. However, using TCP results in a much lower probability that a given exchange will fail.

In Fig. 18 we show the average Request/Response completion time when MIH retransmission are not being used over TCP. As we expect, the transaction delay for a given maximum RTO value is approximately 2.25 times greater than it is in the case when only a Indication message is being transmitted, as

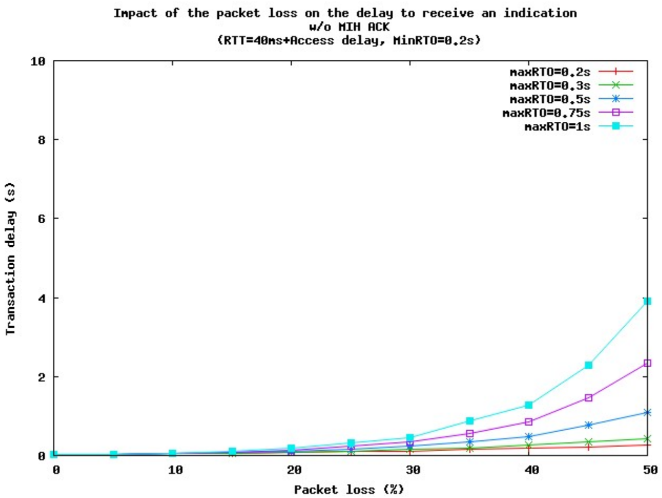

Fig. 17. Mean MIH Indication transaction completion time over TCP transport without MIH timeouts and acknowledgments.

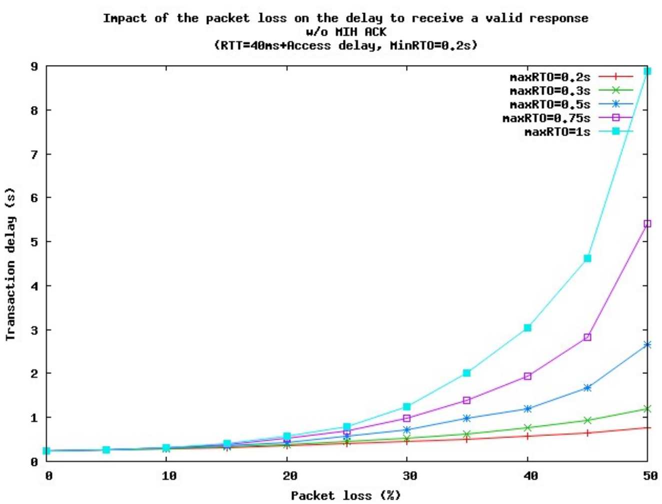

Fig. 18. Mean MIH Request/Response transaction completion time over TCP transport without MIH timeouts and acknowledgments.

shown in Fig. 17. Note that in all four figures we do not begin to see serious increases in the delay for either type of message exchange until the packet loss probability is on the order of 0.1. Also, the large average delay values associated with packet loss rates of near 0.5 are an indication that a greater number of transactions are able to successfully complete within the time limit than we observed when we used MIH reliability features in conjunction with TCP.

B. MIH Overhead

In this subsection, we examine the amount of overhead associated with the various transport layer options. In Fig. 19 we plot theoretical and simulation results for transport over UDP for both the Indication and Request/Response exchanges. We observed very close agreement between the theoretical and simulation results for the Indication exchange, and a maximum error of 7% for the Request/Response exchange when $p = 0.25$. If MIH reliability is not in use, there is no

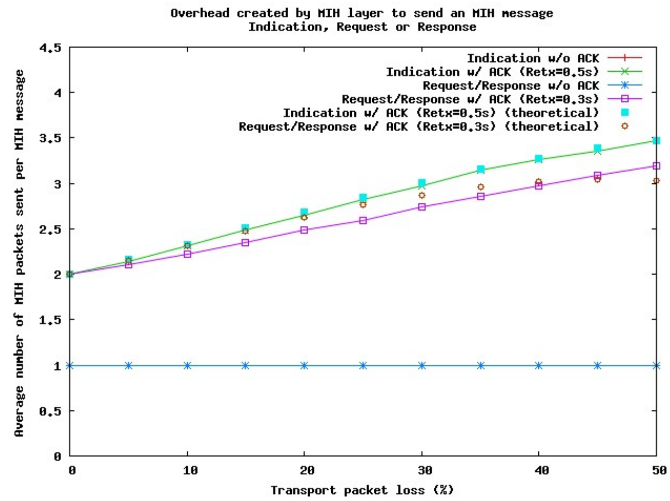

Fig. 19. Theoretical and simulation values of MIH overhead for Indication and Request/Response exchanges over UDP.

overhead penalty although this will result in a lower success rate for both types of exchanges. With MIH acknowledgments and retransmissions, we observed a slightly higher overhead penalty in connection with Indication exchanges relative to the Request/Response case. This follows from Equation (19), which shows that increasing M, the number of messages in an exchange, decreases $\overline{N}_{\mathrm{MSG}}$, with $\overline{N}_{\mathrm{MSG}} \to n/(1-q)$ as M becomes large.

Next we examine overhead associated with using TCP transport. In Fig. 20 and Fig. 21, we plot the expected number of MIH packets sent to the transport layer per MIH message generated by the MIH User for Indications and Request/Response exchanges, respectively. If retransmissions by the MIH layer are turned off, this number is 1. When there is no packet loss ($p = 0$), an MIH ACK message gets sent for each indication/request/response that a node receives. Therefore the minimum number of MIH packets sent per message is two if MIH ACKs are being used. As the packet loss rate increases, the MIH User will retransmit messages up to two times. The maximum number of packets for each MIH Indication is therefore 6 (3 copies of the message and 3 corresponding MIH ACKs), as seen in Fig. 20. In the case of Request/Response exchanges, if the TCP delay is too large, Requests will arrive late at the MN, whose Response will be ignored by the PoS, which will in turn create another Request. The maximum overhead is 4.5 since there are at most 3 Requests, 3 ACKs, and 3 Responses for each Request/Response pair. If we examine both graphs, we observe that the amount of overhead increases as we decrease the maximum RTO, which is what we would expect. Furthermore, significant increases in overhead to not occur until the packet loss probability exceeds 0.1.

V. CONCLUSIONS

In this paper, we developed characteristic functions that let us obtain the distribution and moments of the time to exchange an arbitrary number of MIH messages using either UDP with MIH retransmissions and acknowledgments or using

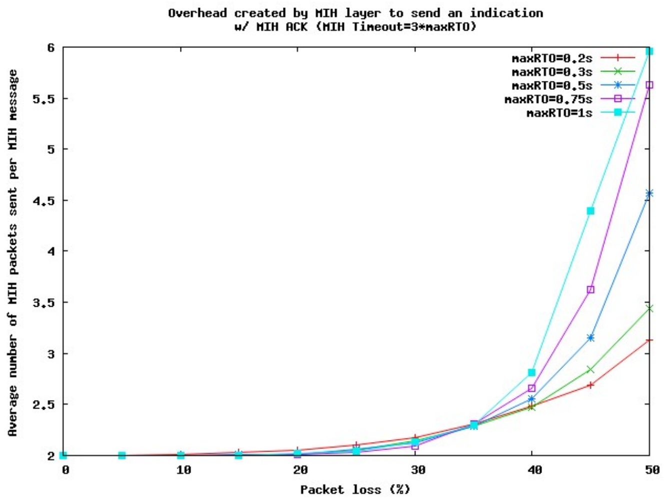

Fig. 20. Simulation values of MIH overhead for Indication messages over TCP with MIH ACKs.

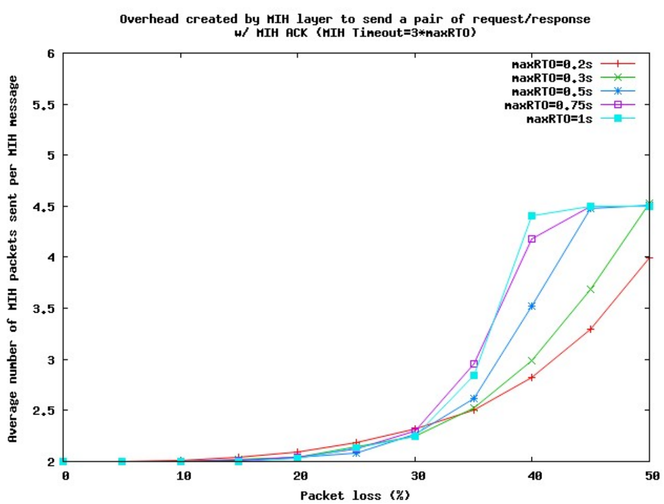

Fig. 21. Simulation values of MIH overhead for Request/Response exchanges over TCP with MIH ACKs.

TCP. We also developed expressions for the average overhead expressed as extra message transmissions due to packet losses. Our results verify that there is a tradeoff between latency and overhead. Adjusting timeout values to reduce latency, whether in the MIH User if UDP is being used or at the transport layer if TCP is being used, invariably results in an increase in the expected number of messages that are transmitted during an exchange. The delay and overhead penalties are of course smaller if the packet loss probability is small. This illustrates the importance of proactive handover signaling that begins well before the signal quality on the wireless link degrades to the point where the packet loss probability is on the order of 0.1 or greater. We also note that TCP is best suited for longer MIH message exchanges since it will hold data until there is enough to fill a segment, which is undesirable for messages that require low latency.

Our simulation results agreed with our mathematical model. We observed low delays and less overhead in the case where we used UDP, but we also experienced a higher rate of failure for a MIH message exchange. TCP displayed a lower exchange failure rate than UDP with retransmissions, largely because the number of retransmissions over UDP was limited while the number of TCP retransmissions was not. TCP's shortcoming was that it resulted in greater overhead than UDP and performed inefficiently when the MIH message generation rate was low. Our results show that using MIH retransmission timers with TCP produces conflicts that severely degrade performance.

Acknowledgment

The authors would like to thank the members of the IETF MIPSHOP MIH design team for many helpful discussions, especially Juan Carlos Zúñiga and Telemaco Melia.

References

[1] D. Hong, Daehyoung and S.S. Rappaport, "Traffic Model and Performance Analysis for Cellular Mobile Radio Telephone Systems with Prioritized and Nonprioritized Handoff Procedures," *IEEE Transactions on Vehicular Technology*, Vol. 35, no. 3, pp. 77-92, Aug 1986.

[2] M. Sidi and D. Starobinski, "New Call Blocking versus Handoff Blocking in Cellular Networks," *IEEE INFOCOM 96*, pp. 35-42, Mar 1996.

[3] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," IETF RFC 3775, Jun 2004.

[4] "Fast Handovers for Mobile IPv6," R. Koodli, Ed., IETF RFC 4068, Jul 2005.

[5] H. Soliman, C. Castelluccia, K. El Malki, and L. Bellier, "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)," IETF RFC 4140, Aug 2005.

[6] "Draft IEEE Standard for Local and Metropolitan Area Networks: Media Independent Handover Services," IEEE LAN/MAN Draft IEEE P802.21/D06.00, Jun 2007.

[7] "Mobility Services Transport: Problem Statement," T. Melia, Ed., IETF RFC 5164, Mar 2008.

[8] "Mobility Services Framework Design," T. Melia, Ed., draft-ietf-mipshop-mstp-solution-01, Feb 2008. IETF Internet Draft, work in progress.

[9] Abate, J. and Whitt, W., "The Fourier-series method for inverting transforms of probability distributions," *Queuing Systems*, Vol. 10, pp. 5–88, 1992.

[10] A. Rahman, U. Olvera-Hernandez, J.C. Zúñiga, M. Watfa, and H.W. Kim, "Transport of Media Independent Handover Messages over IP," draft-rahman-mipshop-mih-transport-03, Jul 2007. IETF Internet Draft, work in progress.

[11] J. Padhye, V. Firoiu, D. Towsley, and J. Krusoe, "Modeling TCP Throughput: A Simple Model and its Empirical Validation," *IEEE/ACM Transactions on Networking*, Vol. 8, No. 2, pp. 133–145, Apr 2000.

[12] N. Cardwell, S. Savage, T. Anderson, "Modeling TCP latency," *INFOCOM 2000: Proceedings of the Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies* , Vol.3, pp. 1742–1751, 26-30 Mar 2000.

[13] M. Mellia, I. Stoica, and H. Zhang, "TCP Model for Short Lived Flows," *IEEE Communications Letters*, Vol. 6, No. 2, pp. 85–87, Feb 2002.

[14] National Institute of Standards and Technology. NS-2 Mobility Package. http://www.antd.nist.gov/seamlessandsecure.shtml, Jul 2007.