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. These messages are carried by the transport layer, and the IETF is developing an architecture to clarify the mechanisms that should be used. The proposed architectures employ both unreliable transport using User Datagram Protocol (UDP) and reliable stream transport using Transmission Control Protocol (TCP). In this paper we model the handover latency for the cases where UDP and TCP carry MIH signaling messages, and we discuss some of the design tradeoffs.**

***Index Terms*— IEEE 802.21, Media Independent Handover (MIH), TCP, UDP, protocol design, handover delay, 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). With the proliferation of sundry Layer 2 wireless technologies (e.g. IEEE 802.11 (WiFi) and IEEE 802.16 (WiMAX)), manufacturers of mobile telecommunications devices have recognized the utility of developing wireless mobiles with multiple antennas that are capable of performing so-called vertical handovers. In addition to carrying out horizontal handovers, these devices can switch active connections from a network access point that uses one Layer 2 technology (such as 3G cellular) to another network access point that uses a different Layer 2 technology (such as WiMAX).

Various standards bodies such as the IETF and IEEE 802.21 have been developing new protocols to support fast handovers at different layers. The MIPSHOP and 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, however, do not solve the problem of handover latency, which can cause numerous packets to be dropped while a mobile user is migrating from one access network to another. The latency problem is being addressed by the IEEE 802.21 Media Independent Handover (MIH) working group, which is developing mechanisms to use Layer 2 triggers associated with events such as decreases in received signal strength to provide early warning of impending handovers and thereby decrease handover latency. In addition, the MIH working group is developing an architecture that would define handover managers at the application layer that would direct the processing of Layer 2 triggers and maintain awareness of available access networks.

Because the MIH application resides above the transport layer, it must use a reliable transport protocol to carry signaling messages to MIH applications located at network access points or at a remote mobility manager. These messages must be transported reliably; the MIPSHOP working group is considering a number of possible solutions. One approach uses the User Datagram Protocol (UDP) with reliability enhancements in the form of retransmission timers in the MIH application. Another solution is to use the Transmission Control Protocol (TCP), which offers congestion control in addition to reliability. A hybrid approach combining the previous two solutions would be to use the General Internet Signaling Transport (GIST) protocol.

This paper develops models to characterize MIH signaling exchanges over a variety of transport layer technologies. 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 combine the results of the UDP and TCP analyses to produce a model of MIH over the General Internet Signaling Transport (GIST) protocol. We use simulation results to verify our models in Section V We also use our model and simulations to examine some of the design trade-offs associated with each of the transport layer options. We summarize out results in Section VI.

II. MIH Signaling over UDP

We first consider the case where the MIH application 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 $T_{\mathrm{MIH}}(i)$. 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 a handover that involves the exchange of M MIH messages. We assume that an MIH application that receives an

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).

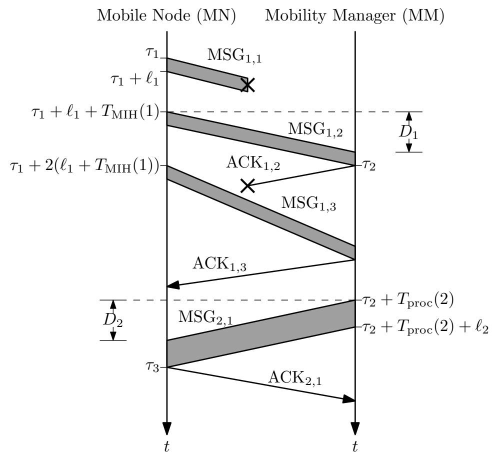

Fig. 1. Sample signal flow between MN and MM, in which packet loss occurs.

MIH message immediately generates an MIH ACK followed by an MIH response message that constitutes the next part of the handover information exchange. 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 $T_{\text{proc}}(i)$. This reflects the amount of time that the receiving node needs to perform whatever tasks are mandated by the arriving MIH message. We define τ_i to be the time when the ith 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 time when the message is generated. The amount of time required to move the message out the transmission buffer is $\ell_1 = L_1/B$ s, where L_1 is the length of the first message in bits and B is the bandwidth of the channel in bits/s. The message traverses the network and, if it is not lost, it arrives at its destination after some random delay that we denote as D_1. In general D_i is the delay experienced by the ith message in the handover sequence.

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

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

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 the absence of packet loss, the characteristic function of the handover delay, $\Phi_H(\omega)$, is $\big(\Phi_D(\omega)\big)^{M+1} \exp\big(j\omega \sum_{i=1}^{M} [\ell_i + T_{\text{proc}}(i)]\big)$.

To model the case where packet losses occur, we 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 therefore successful with probability $P_s = (1-p)^2$. For the handover to be successful, each of the M messages composing the message exchange 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 message transmission attempts does not fail is $1 - p^{R_i+1}$. Thus the probability that a handover fails is

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

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 handover. A sender will also re-transmit a message if the sum of the message's transit time and transit time of its corresponding ACK is greater than the timeout, $T_{\text{MIH}}(i)$. We assume that each resent 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 handover exchange, which was first transmitted at time τ_1. Let $T_{\text{MIH}}(i)$ be the amount of time that must pass before the ith MIH message is re-transmitted by its sender. 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 $\tau_1 + \ell_1 + T_{\text{MIH}}(i)$. 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 application and a retransmission of the original message, up to a limit of $R_i + 1$ 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 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 $k = 0, 1, \ldots, R_i$.

We can perform a similar analysis for each of the M messages in the handover. We assume that the loss performance of each message 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

$$\Phi_{\Delta_i^{\mathrm{D}}}(\omega) = \frac{1-p}{1-p^{R_i+1}} \sum_{k=0}^{R_i} p^k e^{jk\omega\delta_i} = \frac{1-p}{1-p^{R_i+1}} \frac{1-(pe^{j\omega\delta_i})^{R_i+1}}{1-pe^{j\omega\delta_i}}, \quad (3)$$

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

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

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

where $T_P = \sum_{i=1}^{M} T_{\mathrm{proc}}(i)$ 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 application allows the same number of retransmissions for each message, so that $R_i = R$ for all i, then Equation (4) becomes

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

Once we have the characteristic function of the handover time H, we can obtain the probability that the handover time exceeds some time t by using the cumulative distribution function of H to compute $1 - F_H(t)$. We can get the distribution function by inverting the characteristic function using Equation (3.6) from [3], which is

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

Using the trapezoidal rule, we can approximate this integral as follows, which is Equation (4.4) from [3]:

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

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 now consider an example to illustrate our results. In Fig. 2, we consider the case described in Section 10.4 of [1], 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.

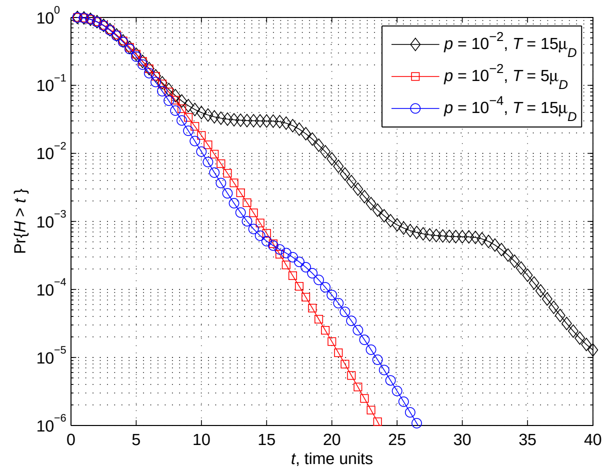

Fig. 2. Probability that the handover time, H, exceeds t seconds for three sets of values of packet loss probability p and MIH timeout, T.

The effect of message retransmissions is most clearly visible in the black curve with diamonds that is associated with $p = 0.01$ and $T_{\mathrm{MIH}} = 15\mu_D$. 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 $T_{\mathrm{MIH}} = 5\mu_D$ 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 $T_{\mathrm{MIH}} = 15\mu_D$. If we decrease the packet loss probability to $p = 10^{-4}$ while keeping the MIH timeout at $T_{\mathrm{MIH}} = 15\mu_D$, 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 associated with the packet loss probability value of 0.01. The three curves shown in Fig. 2 demonstrate that the performance of MIH over UDP depends very little on the retransmission parameters if the retransmission timeout period is chosen to be on the order of the round-trip time between the MIH connection endpoints. Furthermore, a packet loss probability that is sufficiently small will also eliminate the effect of retransmissions.

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

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

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

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

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

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

We can also compute σ_H^2, the variance of H, using the characteristic function $\Phi_H(\omega)$. We recall that the transit times $\{D_i\}_{i=1}^M$ are mutually independent, and are independent of the additional delays $\{\Delta_i^{\mathrm{D}}\}_{i=1}^M$, which are themselves mutually independent. The processing times and packet transmit times are deterministic and contribute nothing to the variance of H. Thus the variance of H is

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

We get each of the variances of $\{\Delta_i^{\mathrm{D}}\}_{i=1}^M$ as follows:

$$\begin{aligned} \sigma^2_{\Delta_i^{\mathrm{D}}} &= \mathsf{E}\{(\Delta_i^{\mathrm{D}})^2\} - \mu^2_{\Delta_i^{\mathrm{D}}} \\ &= (-1)\frac{d^2\Phi_{\Delta_i^{\mathrm{D}}}(\omega)}{d\omega^2}\bigg|_{\omega=0} - \mu^2_{\Delta_i^{\mathrm{D}}} \\ &= \left[\frac{p}{(1-p)^2} - \frac{(R_i+1)^2 p^{R_i+1}}{(1-p^{R_i+1})^2}\right]\delta_i^2 \end{aligned} \tag{11}$$

B. MIH over UDP Overhead

As indicated by Figure 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 $T_{\mathrm{MIH}}(i) \to 0$, 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 $\{T_{\mathrm{MIH}}(i)\}_{i=1}^M$ by computing the expected number of messages generated per handover attempt.

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 $T_{\mathrm{MIH}}(i)$. Thus the probability that a given message transmission attempt results in another copy of the message being sent is

$$\begin{aligned} P_{rt}(i) &= p + (1-p)[p + (1-p)(1 - F_{\mathrm{RTT}}(T_{\mathrm{MIH}}(i)))] \\ &= (2-p)p + (1-p)^2[1 - F_{\mathrm{RTT}}(T_{\mathrm{MIH}}(i))] \\ &= 1 - (1-p)^2 F_{\mathrm{RTT}}(T_{\mathrm{MIH}}(i)), \end{aligned} \tag{12}$$

where $F_{\mathrm{RTT}}(t)$ is the cumulative distribution of the round-trip time, RTT. 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 $F_{\mathrm{RTT}}(t) = \int_0^\infty f_D(u)F_D(t-u)du$ where $f_D(t)$ is the density of D. Equivalently, the characteristic function of RTT is $\Phi_{\mathrm{RTT}}(\omega) = (\Phi_D(\omega))^2$. For example, if D is exponential with mean μ_D, the cumulative distribution of RTT is

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

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

$$\pi_m = \begin{cases} (1-P_{rt}(i))P_{rt}^{m-1}(i), & m = 1, 2, \ldots, R_i \\ P_{rt}^{R_i}(i), & m = R_i + 1. \end{cases} \tag{13}$$

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

$$\begin{aligned} \overline{n}_i &= \sum_{m=1}^{R_i+1} m\pi_m \\ &= (1-P_{rt}(i))\sum_{m=1}^{R_i} mP_{rt}^{m-1}(i) + (R_i+1)P_{rt}^{R_i}(i) \\ &= \frac{1 - P_{rt}^{R_i+1}(i)}{1 - P_{rt}(i)}. \end{aligned} \tag{14}$$

The total number of messages required for the handover, on average, is thus

$$\overline{N}_{\mathrm{MSG}} = \sum_{i=1}^{M}\overline{n}_i = \sum_{i=1}^{M}\frac{1 - P_{rt}^{R_i+1}(i)}{1 - P_{rt}(i)}. \tag{15}$$

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 application receives an MIH message. For the ith message in a M-message handover sequence, the sending node will produce m copies with probability π_m as defined in Equation (13). 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

$$\begin{aligned} \overline{a}_i &= \sum_{m=1}^{R_i+1}\pi_m\sum_{k=0}^{m}\binom{m}{k}kp^{m-k}(1-p)^k \\ &= (1-p)\sum_{m=1}^{R_i+1} m\pi_m \;=\; (1-p)\overline{n}_i. \end{aligned} \tag{16}$$

To get the expected number of messages of both types sent during the handover, we sum $\overline{a}_i$ over all messages $\{i\}$ and combine the resulting expression with the expected number of MIH messages from Equation (15), giving

$$\overline{N} = (2-p)\sum_{i=1}^{M}\frac{1 - P_{rt}^{R_i+1}(i)}{1 - P_{rt}(i)}. \tag{17}$$

From Equation (14), $\overline{n}_i \to R_i + 1$ as $P_{rt}(i) \to 1$, which happens when $p \to 1$ or when $T_{\mathrm{MIH}}(i) \to 0$. We ignore the first case because the probability of handover failure is 1. In

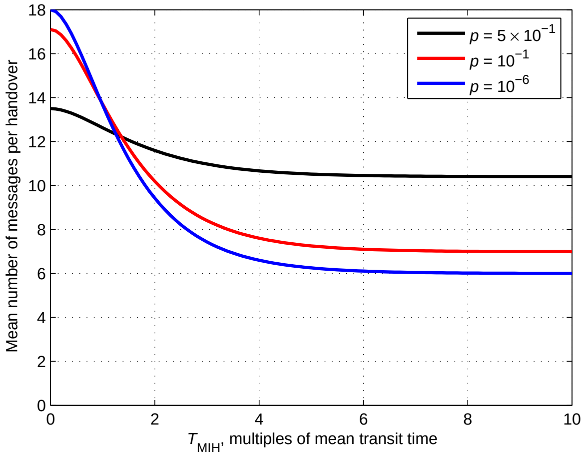

Fig. 3. Mean number of messages generated by MIH signaling endpoints during a handover where $M = 3$ and $R_i = 2$ for $i = 1, 2, 3$.

the second case, $\overline{N} \to 2\sum_{i=1}^{M}(R_i+1)$, 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 $\mu_D = 1$. 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. 3. The resulting curves show that the average number of messages per handover 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 $p \to 1$ we would obtain a horizontal line corresponding to $\overline{N} = M \cdot R = 9$. The curves in Fig. 3 suggest that we can reduce the number of messages generated per handover while minimizing the probability that the handover is excessively long if we let the MIH timeout be two to four times larger than the measured packet transit time.

Thus far, our treatment of the handover overhead has considered only those cases where the handover completes successfully. In these cases, at least one of the $R_i + 1$ transmission opportunities succeeds at each of the M stages of the handover. In general, however, handovers 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 applications collectively offer to the network, we must consider the effect of incomplete handover signaling 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 handover 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 a handover, accounting for the loss of all $R_i + 1$ copies 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

$$\begin{aligned} \overline{N}_{\text{MSG}} &= (1-q)n\sum_{m=1}^{M-1} mq^{m-1} + Mq^{M-1}n \\ &= \frac{(1-q^M)n}{1-q} \\ &= \frac{(1-P_{rt}^{R+1})(1-(1-p^{R+1})^M)}{(1-P_{rt})p^{R+1}}. \end{aligned} \qquad (18)$$

Since $\overline{a}_i = (1-p)\overline{n}_i$, the total number of MIH messages in the handover 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 now consider the case of MIH over TCP, which is a connection-based stream-oriented protocol that provides reliable transport of application data. If MIH allows TCP to be solely responsible for guaranteeing message reliability and does not use its own timeouts, we can compute the performance metrics associated with the handover by applying the same techniques that we used in Section II.

There has been voluminous work on TCP behavior. In particular, [5] and the work that builds on it, [6], have respectively characterized the loss and delay behavior of TCP during a steady-state bulk data transfer and a during the transmission of an arbitrarily small amount of data. Our latency analysis considers the degenerate case in the model [6] 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 application.

For this discussion, we assume 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 handover 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 application. 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 in this case, 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 application 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 handover begins. Because the TCP transmission window will be at least one segment long, the node initiating the handover 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 handover, it will not receive any duplicate acknowledgments from the destination if the first handover 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 retransmission timeout (RTO) value, If subsequent retransmitted messages are also lost, the source node will continue to double its RTO value until it reaches a maximum, $\mathrm{RTO}_{\max} = 2^K \mathrm{RTO}$, where K is the maximum number of times that TCP doubles the RTO value. The RTO reaches $\mathrm{RTO}_{\max} = 2^K \mathrm{RTO}$ when the Kth retransmission occurs. 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 in Fig. 4. In the figure, the ith message in the handover 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

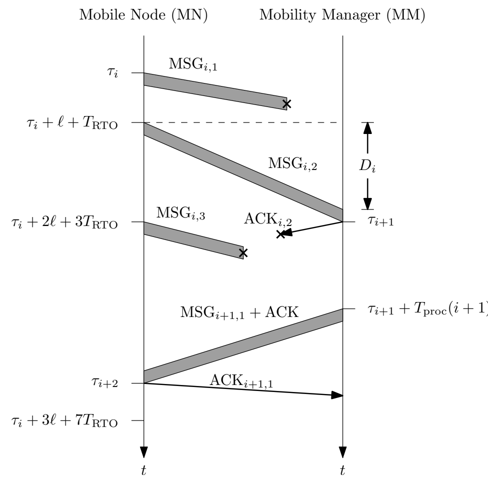

Fig. 4. Sample MIH signal flow between MN and MM when TCP transport is used and when some messages and TCP ACKs are lost in transit.

message at time τ_{i+2}, its TCP layer notes the acknowledgment and stops retransmission of the ith message.

Our analysis of handover latency 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 handover 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

$$
\begin{aligned}
&\Phi_{\Delta_i^{\mathrm{C}}}(\omega) \\
&= (1-p)\sum_{r=0}^{K} p^r e^{j\omega[r\ell + (2^r - 1)T_{\mathrm{RTO}}]} \\
&\quad + (1-p)\sum_{r=K+1}^{\infty} p^r e^{j\omega[r\ell + (r-K+1)\mathrm{RTO}_{\max} - T_{\mathrm{RTO}}]} \\
&= (1-p)\sum_{r=0}^{K} p^r e^{j\omega[r\ell + (2^r - 1)T_{\mathrm{RTO}}]} \\
&\quad + \frac{(1-p)p^{K+1} e^{j\omega[(K+1)\ell + 2\mathrm{RTO}_{\max} - T_{\mathrm{RTO}}]}}{1 - p e^{j\omega(\ell + \mathrm{RTO}_{\max})}}. \qquad (19)
\end{aligned}
$$

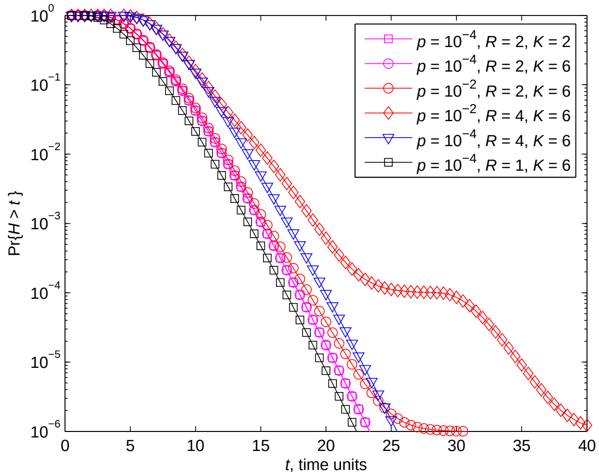

Fig. 5. Probability that the handover time, H, exceeds t seconds for six sets of values of packet loss probability p, RTO, and number of RTO stages, K.

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 $\Pr\{H > t\}$ for various values of t. We plot performance curves for the case where $M = 3$ in Fig. 5.

We can use the characteristic function to obtain the expected value and the variance of the handover time. The expression for μ_H is analogous to Equation (9); the only additional computation that we require is the expected value of Δ_i^{C}, which we obtain from Equation (19) as follows:

$$\mu_{\Delta_i^{\mathrm{C}}} = \frac{p}{1-p}\ell + \frac{(1-p-2^K p^{K+1})p}{(1-p)(1-2p)} T_{\mathrm{RTO}}. \quad (20)$$

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 (20) is closely related to Equation (18) in [6], which itself comes from [5]. We can manipulate our Equation (20) 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 [6], $G(p) = 1 + \sum_{k=1}^{K} 2^{k-1} p^k$, 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 (20) 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 use the characteristic function to get the variance of Δ_i^{C} in a fashion similar to that which gave us Equation (11). Doing this produces

$$\begin{aligned}\sigma^2_{\Delta_i^{\mathrm{C}}} = {} & \tfrac{2p((1-p)^2 - p(2p)^K(2+K-3(K+1)p+2Kp^2))}{(1-p)^2(1-2p)^2} \ell T_{\mathrm{RTO}} \\ & + \Big[\tfrac{4^K p^{4+2K}}{(1-p)^2(1-2p)^2} + \tfrac{p^2(7-p)(4p)^K}{(1-p)^2(1-4p)} \\ & - \tfrac{(1-p)p + (1-4p)p(2p)^{K+1}}{(1-p)^2(1-4p)} \Big] T^2_{\mathrm{RTO}} + \tfrac{p}{(1-p)^2} \ell^2. \quad (21)\end{aligned}$$

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

$$\begin{aligned}\lim_{p\to 1/4} \sigma^2_{\Delta_i^{\mathrm{C}}} = {} & \frac{4}{9}\ell^2 + \frac{2(-5+9\cdot 2^K) - 3K}{9\cdot 2^K} \ell T_{\mathrm{RTO}} \\ & + \frac{4^K(27K-1) + 9\cdot 2^{K+1} - 1}{9\cdot 4^{K+1}} T^2_{\mathrm{RTO}} \quad (22)\end{aligned}$$

and

$$\begin{aligned}\lim_{p\to 1/2} \sigma^2_{\Delta_i^{\mathrm{C}}} = {} & 2\ell^2 + \frac{8+5K+K^2}{2} \ell T_{\mathrm{RTO}} \\ & + \frac{2(-9+13\cdot 2^K) - 8K - K^2}{4} T^2_{\mathrm{RTO}}. \quad (23)\end{aligned}$$

B. MIH over TCP overhead

We also can compute the expected number of messages generated during a handover 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 (12), the probability that the kth transmission attempt fails and requires another attempt is

$$\phi_k = \begin{cases} 1-(1-p)^2 F_{\mathrm{RTT}}(2^{k-1}T_{\mathrm{RTO}}), & k \le K \\ 1-(1-p)^2 F_{\mathrm{RTT}}(\mathrm{RTO}_{\max}), & k > K. \end{cases} \quad (24)$$

where we assume that T_{RTO} is the same for all M messages in the handover 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

$$\pi_m = \begin{cases} 1-\phi_1, & m = 1 \\ (1-\phi_m)\prod_{k=1}^{m-1}\phi_k, & 1 < m \le K \\ (1-\psi)(\psi)^{m-K-1}\prod_{k=1}^{K}\phi_k, & m > K \end{cases} \quad (25)$$

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:

$$\begin{aligned}\overline{n}_i &= \sum_{m=1}^{\infty} m\pi_m \\ &= 1-\phi_1+\sum_{m=2}^{K} m(1-\phi_m)\prod_{k=1}^{m-1}\phi_k \\ &\quad + \sum_{m=K+1}^{\infty} m(1-\psi)(\psi)^{m-K-1}\prod_{k=1}^{K}\phi_k \\ &= 1-\phi_1+\sum_{m=2}^{K} m(1-\phi_m)\prod_{k=1}^{m-1}\phi_k \\ &\quad + \frac{1+K(1-\psi)}{1-\psi}\prod_{k=1}^{K}\phi_k. \end{aligned} \tag{26}$$

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 handover 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 a handover is $\overline{N} = (2-p)M\overline{n}$, where $\overline{n}$ is given in Equation (26).

In Fig. 6, we plot the average number of messages in a handover exchange 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 application. The figure bears a strong resemblance to Fig. 3, which plots $\overline{N}$ versus the MIH timeout for the MIH/UDP case. An important difference is the asymptote at $\mathrm{RTO} = 0$, 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 signaling application.

C. The Effect of Combining MIH Timeouts with TCP

In Section III we developed an analytical model of the behavior of the MIH signaling application during a handover, in which we assumed that MIH timeouts were not in effect. In other words, TCP's retransmission features would be the sole means of guaranteeing reliable message delivery. In our simulations, which we will describe in Section V, we examined the effect of implementing timeouts and retransmissions in the MIH application on top of the timeout and retransmission features of TCP. The simulation results show that there is a significant negative impact on performance if one combines the two reliability mechanisms in MIH and TCP.

In this subsection, we present an analytical model that quantifies some of the negative effects that result from combining

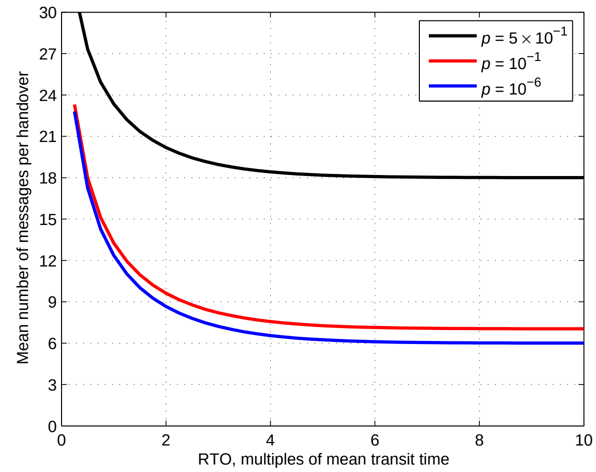

Fig. 6. Mean number of messages generated by MIH signaling endpoints using TCP transport during a handover where $M = 3$ and $K = 6$, for various RTO values.

the TCP layer and MIH layer retransmission mechanisms. Since we are using MIH's reliability features, the MIH layer will generate 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 layers. In addition to the MIH messages, the connection endpoints generate TCP ACKs every time they receive a TCP segment.

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 ($T_{\mathrm{MIH}} \geq T_{\mathrm{RTO}}$). 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 application'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 application generates depends on the amount of time that TCP spends in Timeout retransmitting the original copy of the message. We start our analysis by examining the Request message from the MN to the MM. The MIH layer will produce m additional copies of the 1st message, $0 \leq m \leq R_1$, if the time from the first message transmission attempt until the TCP layer receives an ACK, U_1, lies in the half-open interval $\big[mT_{\mathrm{MIH}}(1), (m+1)T_{\mathrm{MIH}}(1)\big)$. The probability of this event is

$$\begin{aligned}&\Pr\{U_1 \in \big[mT_{\mathrm{MIH}}(1), (m+1)T_{\mathrm{MIH}}(1)\big)\} \\ &\quad = \int_{mT_{\mathrm{MIH}}(1)}^{(m+1)T_{\mathrm{MIH}}(1)} f_{U_1}(u)\,du \\ &\quad = F_{U_1}((m+1)T_{\mathrm{MIH}}(1)) - F_{U_1}(mT_{\mathrm{MIH}}(1)). \end{aligned} \tag{27}$$

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 $R_1 T_{\text{MIH}}(1)$, which is $1 - F_{T_M}(R_1 T_{\text{MIH}}(1))$.

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 $\Phi_D(\omega)\Phi_{\Delta_1^{\text{C}}}(\omega)$. 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 application 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 $\Delta_{1,\text{ACK}}^{\text{C}}$, the additional time required to send the MIH ACK for MSG_1 from the MM to the MN. There are two possibilities that we consider, depending on whether $T_{\text{proc}}(2)$ is larger than T_{RTO}. If $T_{\text{proc}}(2) \geq T_{\text{RTO}}$ 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, $\text{RTO}_{\max} = 2^K T_{\text{RTO}}$.

In Fig. 7, 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 $T_{\text{proc}}(2) > T_{\text{RTO}}$. 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 $D_{1,\text{ACK}}$, 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 $\Delta_{1,\text{ACK}}^{\text{C}}$. 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 $D'_{1,\text{ACK}}$ time units, where $D'_{1,\text{ACK}}$ also has characteristic function $\Phi_D(\omega)$. Like [6], we assume that TCP ACKs are never lost in transit. From Fig. (7),

$$\Phi_{U_1}(\omega) = \left(\Phi_D(\omega)\right)^2 \Phi_{\Delta_1^{\text{C}}}(\omega)\Phi_{\Delta_{1,\text{ACK}}^{\text{C}}}(\omega)e^{j2\omega\ell}. \quad (28)$$

where $\Phi_{\Delta_{1,\text{ACK}}^{\text{C}}}(\omega)$ is given by Equation (19).

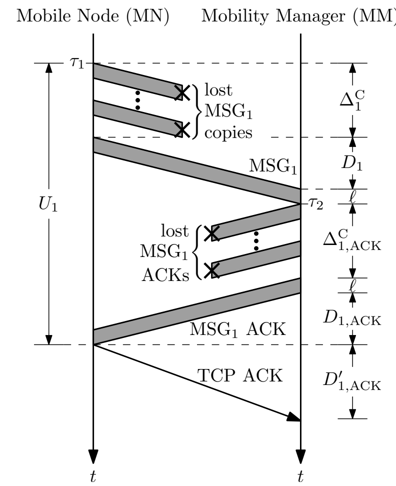

Fig. 7. Signal flow for an exchange of an MIH message and MIH ACK message between MN and MM over TCP, when $T_{\text{proc}}(2) > T_{\text{RTO}}$.

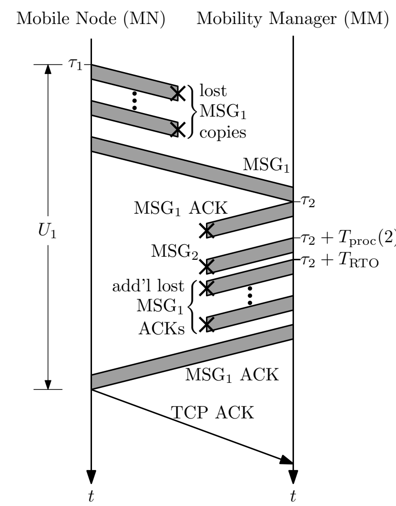

Fig. 8. Signal flow for an exchange of an MIH message and MIH ACK message between MN and MM over TCP, when $T_{\text{proc}}(2) \leq T_{\text{RTO}}$.

If $T_{\text{proc}}(2) < T_{\text{RTO}}$, 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. 8. 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 $t = \tau_2 + T_{\text{RTO}}$, 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 $\Delta_{1,\text{ACK}}^{\text{C}}$ changes slightly from its form in Equation (19), as we now show.

In this case, $\Delta_{1,\text{ACK}}^{\text{C}}$ takes the values 0 and $T_{\text{proc}}(2)$ with probabilities $(1-p)$ and $(1-p)p$, respectively. The first retransmission of the MIH ACK happens at time $t = \tau_2 + \ell + T_{\text{RTO}}$ if no ACK is received for either the MIH ACK or the 2nd MIH message, so $\Delta_{1,\text{ACK}}^{\text{C}} = \ell + T_{\text{RTO}}$ with probability

$(1-p)p^3$. Continuing this pattern and using the results of our analysis in Section III, we see that $\Delta^{\mathrm{C}}_{1,\mathrm{ACK}} = (2^r - 1)T_{\mathrm{RTO}}$ 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 $\Delta^{\mathrm{C}}_{1,\mathrm{ACK}} = (r - K + 1)\mathrm{RTO}_{\max} - T_{\mathrm{RTO}}$ with probability $(1-p)p^{r+1}$ for $r \geq K$. Thus we obtain the following expression for the characteristic function when $T_{\mathrm{proc}}(2) < T_{\mathrm{RTO}}$:

$$\begin{aligned}
&\Phi_{\Delta^{\mathrm{C}}_{1,\mathrm{ACK}}}(\omega) \\
&= (1-p) + (1-p)pe^{j\omega T_{\mathrm{proc}}(2)} \\
&\quad + (1-p)\sum_{r=1}^{K} p^{r+1} e^{j\omega[r\ell + (2^r-1)T_{\mathrm{RTO}}]} \\
&\quad + (1-p)\sum_{r=K+1}^{\infty} p^{r+1} e^{j\omega[r\ell + (r-K+1)\mathrm{RTO}_{\max} - T_{\mathrm{RTO}}]} \\
&= (1-p)[1 + pe^{j\omega T_{\mathrm{proc}}(2)}] \\
&\quad + (1-p)\sum_{r=1}^{K} p^{r+1} e^{j\omega[r\ell + (2^r-1)T_{\mathrm{RTO}}]} \\
&\quad + \frac{(1-p)p^{K+2} e^{j\omega[(K+1)\ell + 2\mathrm{RTO}_{\max} - T_{\mathrm{RTO}}]}}{1 - pe^{j\omega(\ell + \mathrm{RTO}_{\max})}}. \qquad (29)
\end{aligned}$$

From Fig. (8), we can work out the characteristic function of U_1 by conditioning on the whether MSG_2 is lost in transit, as follows:

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

where $\Phi_{\Delta^{\mathrm{C}}_{1,\mathrm{ACK}}}(\omega)$ is given by Equation (29).

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 (27) and the inversion formula in Equation (6), we get $\beta(m) \triangleq \Pr\{m \text{ messages in backlog}\}$:

$$\begin{aligned}
\beta(m) &= \Pr\{mT_{\mathrm{MIH}}(1) \leq U_1 < (m+1)T_{\mathrm{MIH}}(1)\} \\
&= F_{U_1}\big((m+1)T_{\mathrm{MIH}}(1)\big) - F_{U_1}\big(mT_{\mathrm{MIH}}(1)\big) \\
&= \frac{2}{\pi}\int_0^{\infty} \mathrm{Re}[\Phi_{U_1}(\omega)]\Big[\sin\big((m+1)\omega T_{\mathrm{MIH}}(1)\big) \\
&\quad - \sin\big(m\omega T_{\mathrm{MIH}}(1)\big)\Big]\,\frac{d\omega}{\omega} \qquad (30)
\end{aligned}$$

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

$$\begin{aligned}
\beta(R_1) &= \Pr\{U_1 > R_1 T_{\mathrm{MIH}}(1)\} \\
&= 1 - F_{U_1}\big(R_1 T_{\mathrm{MIH}}(1)\big) \\
&= 1 - \frac{2}{\pi}\int_0^{\infty} \mathrm{Re}[\Phi_{U_1}(\omega)]\sin\big(\omega R_1 T_{\mathrm{MIH}}(1)\big)\,\frac{d\omega}{\omega}. \qquad (31)
\end{aligned}$$

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 application needs to send is $1 + \overline{B}$.

IV. MIH over General Internet Signaling Transport (GIST)

In this section we consider the case where MIH uses the General Internet Signaling Transport (GIST) protocol, which was developed as part of the Next Steps in Signaling (NSIS) architecture by the IETF [9]. The NSIS architecture consists of two major layers. The lower layer is the NSIS Transport Layer Protocol (NTLP), which contains GIST itself and the transport layer protocols that GIST uses to carry signaling messages. The upper layer contains the NSIS Signaling Layer Protocols (NSLPs) that are responsible for formatting and processing NSIS signaling messages. GIST has a range of Layer 4 protocols that it can use, depending on the signaling application requirements. In Datagram Mode (D-Mode), GIST uses UDP transport, while TCP is used for Connection Mode (C-Mode).

A. *MIH over GIST Handover Delay*

In the most general scenario, the set of M messages in the handover is divided into the following two subsets: $\mathcal{D}$ and $\mathcal{C}$, which are the sets of messages sent using D-mode and C-mode respectively. Since each message is an element of only one of these two sets, i.e. $\mathcal{C} \cap \mathcal{D} = \varnothing$, $|\mathcal{C}| + |\mathcal{D}| = M$. We can get the characteristic function of H by examining a given message i whose first transmission attempt begins at time τ_i. The additional delay due to packet loss, MIH ACK loss (if in D-mode), or TCP ACK loss (if in C-mode) has characteristic function $\Phi_{\Delta^{\mathrm{D}}_i}(\omega)$ if $i \in \mathcal{D}$ and has characteristic function $\Phi_{\Delta^{\mathrm{C}}_i}(\omega)$ if $i \in \mathcal{C}$. Thus the characteristic function of H becomes

$$\begin{aligned}
\Phi_H(\omega) &= \big(\Phi_D(\omega)\big)^{M+1} e^{j\omega(T_P + T_\ell)} \\
&\quad \times \left(\prod_{i\in\mathcal{C}} \Phi_{\Delta^{\mathrm{D}}_i}(\omega)\right)\left(\prod_{j\in\mathcal{D}} \Phi_{\Delta^{\mathrm{C}}_j}(\omega)\right), \qquad (32)
\end{aligned}$$

where $T_\ell = |\mathcal{C}|\ell + \sum_{i\in\mathcal{D}} \ell_i$. Again, we can invert this function using Equation (6) to get $F_H(t)$, which we can use to get the probability that the handover time H is greater than some time of interest, t.

B. *MIH over GIST Overhead*

In a similar fashion we can compute the expected number of messages that are generated in an exchange when GIST is used at the transport layer. The total number of MIH messages is just

$$\overline{N}_{\mathrm{MSG}} = \sum_{i\in\mathcal{D}} \overline{n}^{\mathrm{D}}_i + \sum_{j\in\mathcal{C}} \overline{n}^{\mathrm{C}}_j, \qquad (33)$$

where $\overline{n}^{\mathrm{D}}_i$, is given by Equation (14) and $\overline{n}^{\mathrm{C}}_j$ is given by Equation (26). The total number of messages including MIH ACKs is

$$\overline{N}_{\mathrm{MSG}} = (2-p)\sum_{i\in\mathcal{D}} \overline{n}^{\mathrm{D}}_i + \sum_{j\in\mathcal{C}} \overline{n}^{\mathrm{C}}_j. \qquad (34)$$

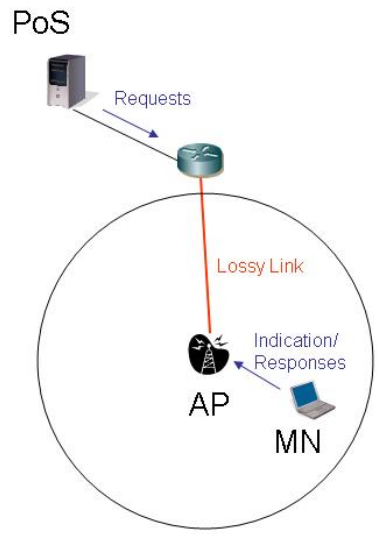

Fig. 9. Network topology used in simulations of Indication and Request/Response MIH message exchanges.

V. Simulation Results

To further quantify the performance of the MIH application over different transport layers, we performed a set of simulations using the ns-2 tool. To generate our results, we used a simple network topology shown in Fig. 9 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 4000 seconds of simulation time, between the 5 s and 4005 s marks.

The parameters that we used 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.

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
$T_{\mathrm{proc}}(2)$ (s)	0.2
Simulation configuration	
Duration (s)	6005
loss probability, p	variable [0, 50%]
$\mathrm{RTO}_{\max}$ (s)	0.2, 0.3, 0.5, 0.75, 1
Indications/s	2
Requests/s	2
MIH Packet size (bytes)	200

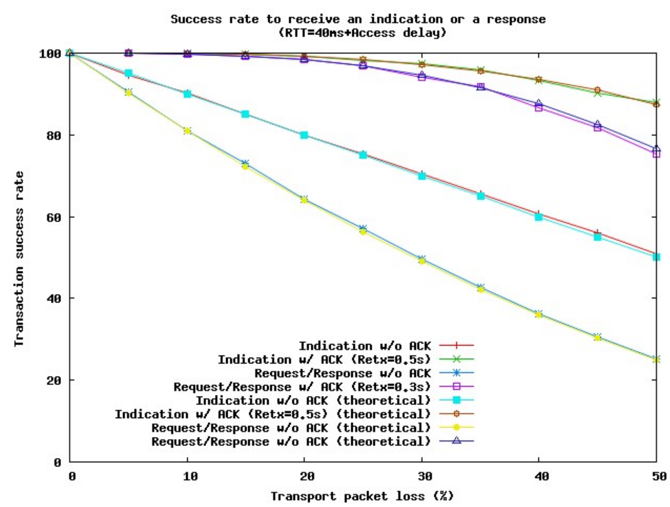

Fig. 10. Theoretical and simulated values for MIH transaction completion probability.

We plot values for the probability that MIH transaction completes successfully in Fig.10. We compare the results from ns-2 with theoretical results that we obtained by computing $1 - P_{\mathrm{fail}}$ 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.

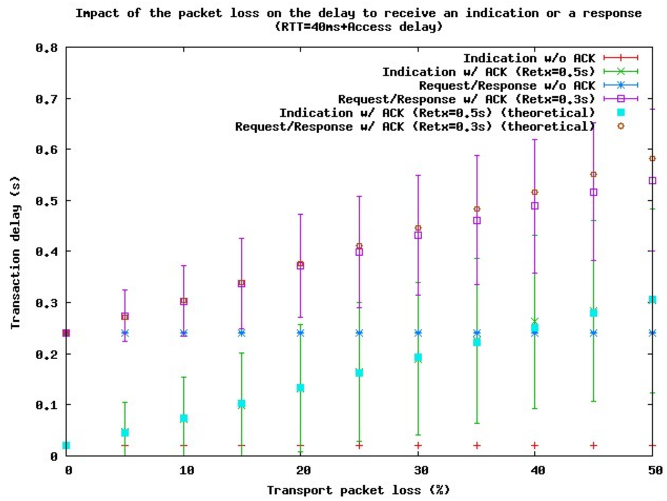

Fig. 11. Theoretical and simulation values of mean MIH handover time for MIH handovers where $M = 1$ (Indication) and $M = 2$ (Request/Response) over UDP.

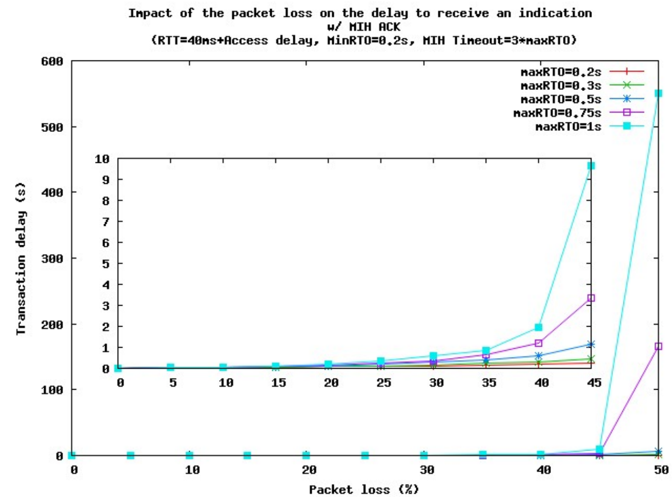

Fig. 12. Mean MIH Indication transaction completion time over TCP transport with MIH timeouts and acknowledgments.

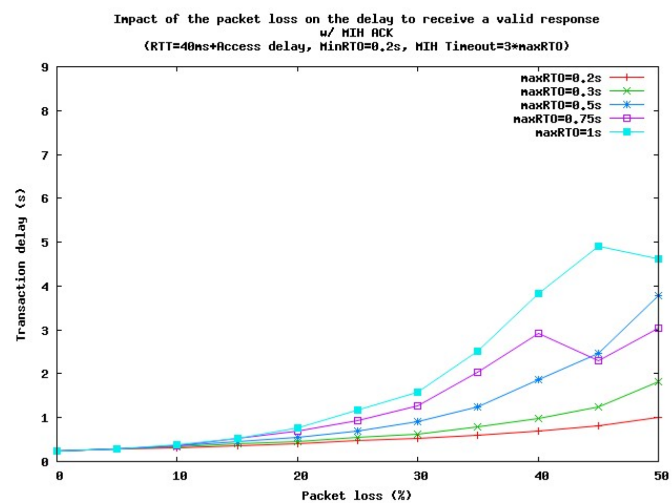

Fig. 13. Mean MIH Request/Response transaction completion time over TCP transport with MIH timeouts and acknowledgments.

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.

We first plot the average time required to complete the simple Indication transmission as well as the Request/Response message exchange. The error bars in the figure indicate the standard deviation of the completion time, i.e. the upper terminus of an error 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 correspond 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. 12 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 3RTO_{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. 13. We observe that the exchange completion time decreases when the packet loss reaches 45% for larger values of 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 reliability features 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. 14. If no MIH ACK is used, the MIH application sends a Request and waits for a Response. In our scenario, the Responses received are always considered valid although that might not be the

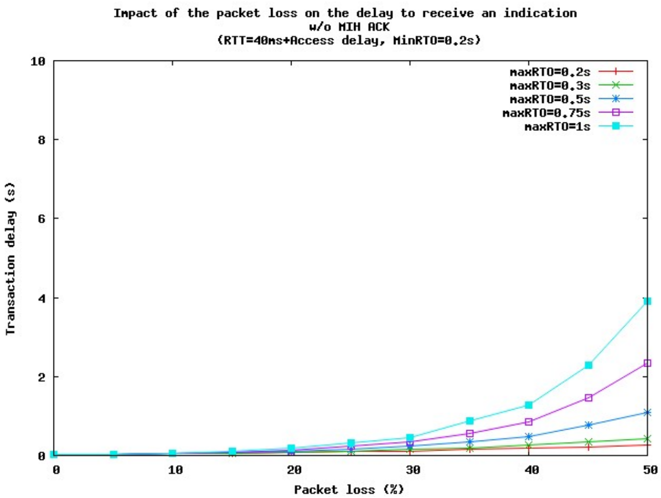

Fig. 14. Mean MIH Indication transaction completion time over TCP transport without MIH timeouts and acknowledgments.

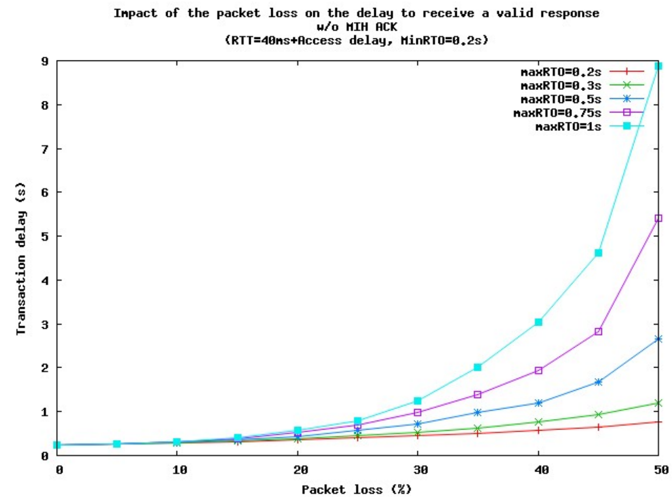

Fig. 15. Mean MIH Request/Response transaction completion time over TCP transport without MIH timeouts and acknowledgments.

case in real implementations. The figure does not exhibit the type of severe performance degradation that we observed in Fig. 12, 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. 15 we show the average Request/Response completion time when MIH reliability is 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 shown in Fig. 14. 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. First wish a 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 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 (18), 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.

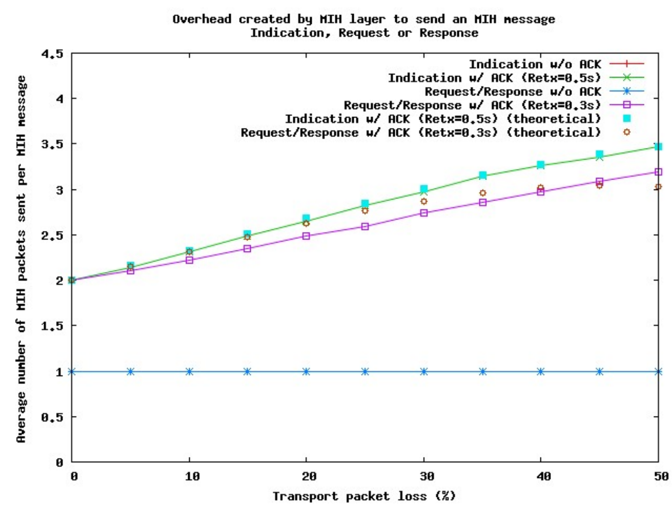

Fig. 16. Theoretical and simulation values of MIH overhead for Indication and Request/Response exchanges over UDP.

Next we examine overhead associated with using TCP transport. In Fig. 17 and Fig. 18, we plot the expected number of MIH packets sent to the transport layer per MIH message generated by the MIH application 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 application 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. 17. In the case of Request/Response exchanges, if the TCP delay

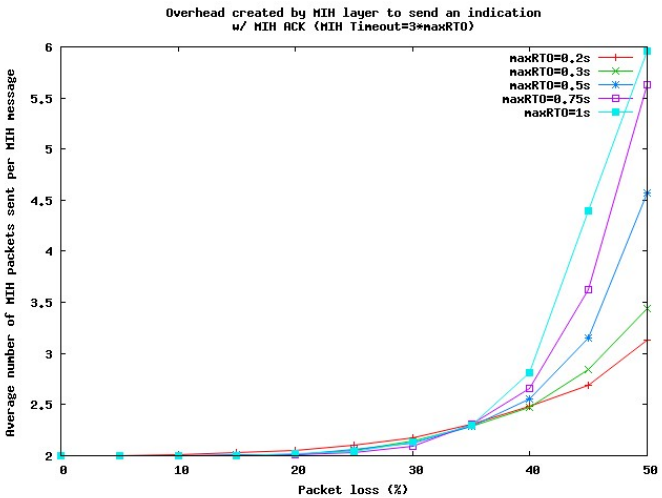

Fig. 17. Simulation values of MIH overhead for Indication messages over TCP with MIH ACKs.

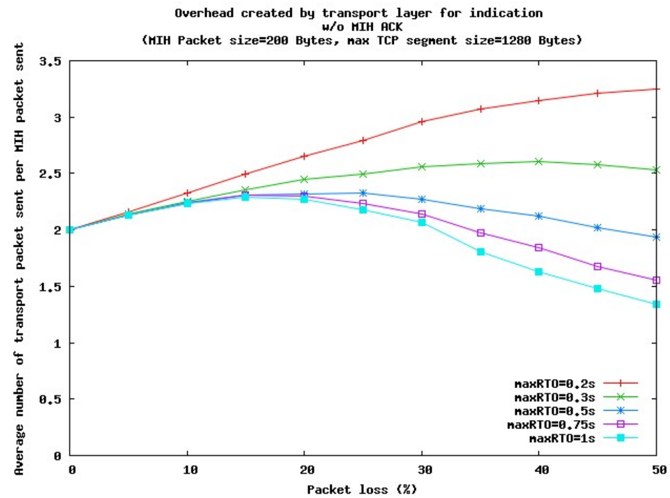

Fig. 19. Simulation values of TCP overhead for Indication messages over TCP without MIH ACKs.

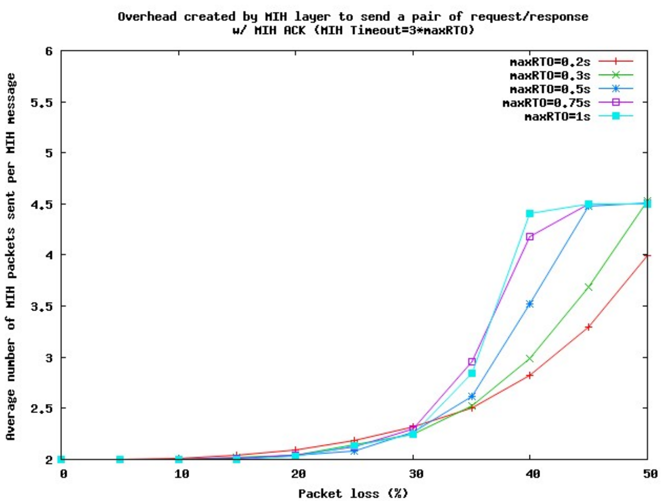

Fig. 18. Simulation values of MIH overhead for Request/Response exchanges over TCP with MIH ACKs.

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 at 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.

We consider the amount of overhead generated by the TCP itself. We first look at the case where a single Indication message is being generated. We plot the expected overhead for the cases where MIH reliability is being used and where it is not in Fig. 19 and Fig. 20, respectively. We observe in both figures that as the packet loss increases, the average number of TCP segments sent increases with respect to p and then decreases. The value of p where the maximum occurs is larger for smaller values of the maximum RTO, and this phenomenon can be observed for all values of $\mathrm{RTO}_{\max}$. This phenomenon can be explained as follows. When there is no packet loss, only one MIH packet will be carried in one TCP segment due to the long time between consecutive message generation events at the MIH application. Once the MIH message is received, the the receiving node will send a TCP ACK that is carried in another TCP segment, so there will be 2 TCP segments per MIH message. If $p > 0$, packet losses cause TCP to retransmit segments. If the packet loss rate is low, a retransmitted a TCP segment will typically contain only a single copy of the lost MIH message. However, as the packet loss rate increases, additional MIH Indication messages will begin to arrive in the TCP transmission queue. Because an MIH message is smaller than the maximum segment size, multiple MIH messages will then be bundled into a single segment, thereby reducing the average number of TCP segments per MIH packet. The reduction effect is more pronounced in the case where MIH performs retransmissions, as shown in Fig 20. The MIH retransmission timeout is three times the TCP RTO; the retransmissions from the MIH application generate an additional source of data that flows into the TCP's transmission queue. This causes TCP to send out segments that are the maximum segment size at a lower packet loss rate.

In Fig. 21 and Fig. 22, we plot average overhead at the TCP layer versus packet loss rate for the Request/Response exchange when MIH reliability features are not used and used, respectively. Our observations are similar to the ones that we obtained for the Indication message, but the value of the packet loss rate at which the overhead reaches a maximum is lower for a given maximum RTO value in the case of the Request/Response exchange. The explanation for this is similar to that for the Indication case. Because each MIH exchange event involves twice as many messages, there will be more data flowing into each node's TCP transmission buffer for a given packet loss rate than there would be if only Indication messages were being sent. And additional effect of this greater amount of data is that the maximum overhead associated with a given maximum RTO is slightly lower than in the case of the Indication message exchange.

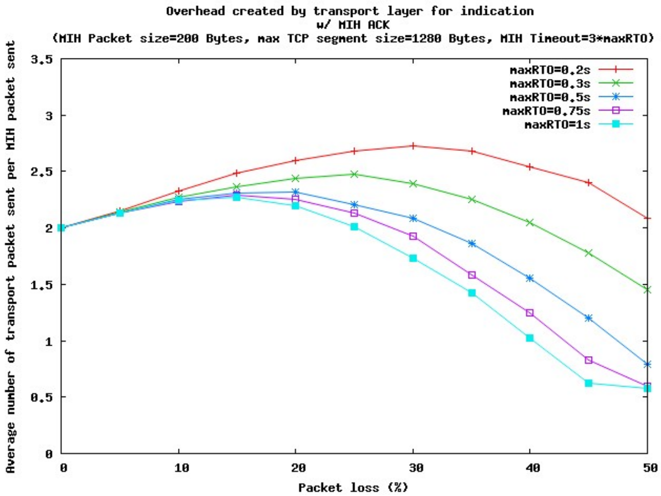

Fig. 20. Simulation values of TCP overhead for Indication messages over TCP with MIH ACKs.

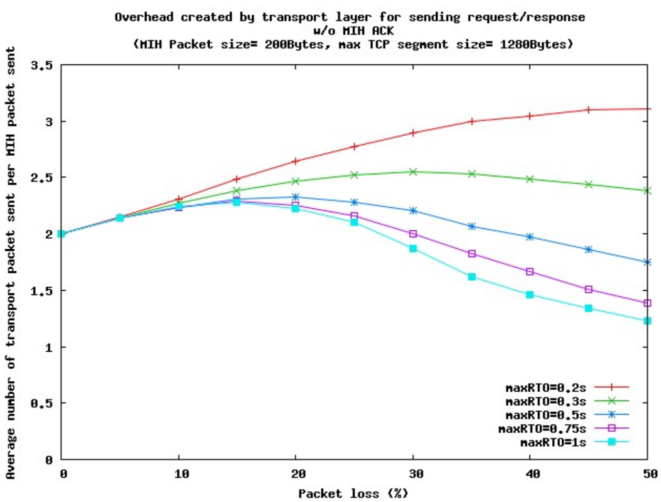

Fig. 21. Simulation values of TCP overhead for Request/Response exchanges over TCP without MIH ACKs.

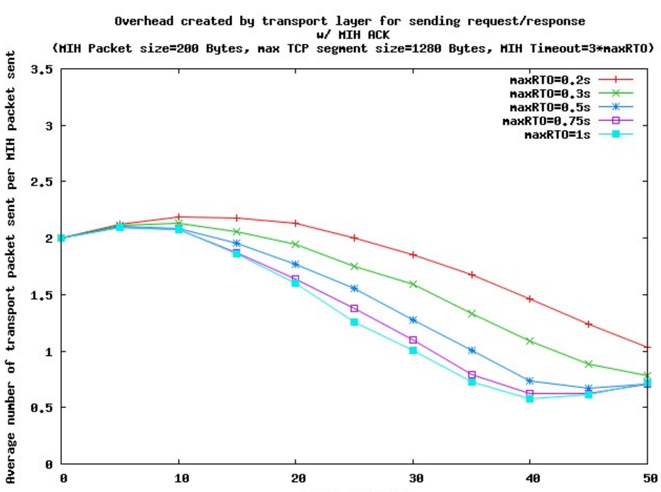

Fig. 22. Simulation values of TCP overhead for Request/Response exchanges over TCP with MIH ACKs.

VI. Summary

We derived the distribution of the time to perform an exchange consisting of an arbitrary number of MIH messages over UDP with MIH retransmissions and acknowledgments, TCP, or GIST. We also computed the expected number of messages in an MIH exchange over UDP, TCP, and GIST. We presented simulation results that evaluate the performance of MIH over different transport layers. We determined that there are important trade-offs between the exchange success rate, delay, and overhead. We had low delays and less overhead in connection with UDP but also had a higher exchange failure rate. TCP performed better with respect to reliability but introduced greater overhead and performed very inefficiently when the MIH message generation rate was small. Our simulation results show that combining MIH reliability mechanisms with the reliability built into 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] A. Rahman, U. Olvera-Hernandez, J.C. Zúñiga, M. Watfa, and H.W. Kim, “Transport of Media Independent Handover Messages over IP,” IETF MIPSHOP Working Group draft, July 5, 2007. Work in progress.

[2] Ross, S., *A First Course in Probability*, Third Edition, Macmillan Publishing Company, New York, 1988.

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

[4] Comer, D. E., *Internetworking with TCP/IP, Volume I*, Second Ed., Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1991.

[5] 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, April 2000.

[6] 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.

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

[8] “Draft IEEE Standard for Local and Metropolitan Area Networks: Media Independent Handover Services,” IEEE LAN/MAN Draft IEEE P802.21/D06.00, June 2007.

[9] Hancock, R., Karagiannis, G., Loughney, J., and Van den Bosch, S., “Next Steps in Signaling (NSIS): Framework,” IETF RFC 4080, June 2005.

[10] Fu, X., Schulzrinne, H., Bader, A., Hogrefe, D., Kappler, C., Karagiannis, G., Tschofenig, H., and Van den Bosch, S., “NSIS: a new extensible IP signaling protocol suite,” *IEEE Communications Magazine*, Vol. 43, No. 10, pp. 133–141, Oct. 2005.