

End-to-End Versus Hop-by-Hop Soft State Refresh for Multi-hop Signaling Systems

Jianhua He[†], Xiaoming Fu[‡], Zuoyin Tang[§]

[†]School of Engineering, Swansea University, UK. Email: j.he@swansea.ac.uk

[‡]Institute of Computer Science, University of Göttingen, Germany. Email: fu@cs.uni-goettingen.de (corresponding author)

[§]Department of Computing Science and Technology, University of Bedfordshire, UK. Email: zuoyin.tang@beds.ac.uk

Abstract—To ensure state synchronization of signalling operations, many signaling protocol designs choose to establish “soft” state that expires if it is not refreshed. The approaches of refreshing state in multi-hop signaling system can be classified as either end-to-end (E2E) or hop-by-hop (HbH). Although both state refresh approaches have been widely used in practical signaling protocols, the design tradeoffs between state synchronization and signaling cost have not yet been fully investigated. In this paper, we investigate this issue from the perspectives of state refresh and state removal. We propose simple but effective Markov chain models for both approaches and obtain closed-form solutions which depict the state refresh performance in terms of state consistency and refresh message rate, as well as the state removal performance in terms of state removal delay. Simulations verify the analytical models. It is observed that the HbH approach yields much better state synchronization at the cost of higher signaling cost than the E2E approach. While the state refresh performance can be improved by increasing the values of state refresh and timeout timers, the state removal delay increases largely for both E2E and HbH approaches. The analysis here shed lights on the design of signaling protocols and the configuration of the timers to adapt to changing network conditions.

I. INTRODUCTION

Recently, the Internet has evolved from a pure best effort service to offering multiple classes of service. In order to support these services, it is necessary for network nodes to maintain states to distinguish packets of different flows and treat them differently. The increasing use of middle-boxes such as NATs and firewalls also demands a means to install, refresh and release flow-associated states in network nodes.

Signaling protocols are hence designed for managing these flow-associated states. These protocols can be classified as either soft state-based, hard state-based or a mixture of both [10]. Hard state-based approaches use reliable messages to install a state and keep the state unless it is explicitly removed by a teardown message, while a soft state can expire after a certain period of time if not refreshed. As a soft state approach can provide faster adaptation and greater robustness to dynamically changing network conditions, it has been widely used in designing network signaling protocols.

Although the soft state concept was first presented two decades ago [5], the design tradeoffs between the level of state synchronization and the cost of soft state signaling have not yet been fully investigated.

In a so-called single-hop system, a state source and state receiver(s) are connected as a single logic hop, which consists

of one or more physical hops not actively involved in signaling [11]. A state receiver is a node where the state from the state source needs to be installed. There is only one state source and one or multiple state receivers involved in the signaling protocol. An example of the single-hop signaling system is shown in Fig. 1(a) for one state receiver and in Fig. 1(b) for multiple state receivers. The blank squares in the figure represent the routers not actively involved in the signaling protocol. RTCP and SIP are examples of signaling protocols designed for single-hop system.

As more and more applications are demanding multi-hop signaling, such as QoS resource reservation and multicast routing, there is an increasing need to investigate the design implications of so-called multi-hop soft state protocols, with which a soft state is supposed to be maintained by a number of routers along the communication paths. An example of a multi-hop signaling system is shown in Fig. 1(c) with two intermediate routers. There are two categories of state refresh used in the multi-hop soft state protocols, namely end-to-end (E2E) and hop-by-hop (HbH) approaches. The major difference is that in the E2E approach refresh messages are exclusively generated by the source, while in the HbH approach the intermediate routers as well as the source maintain local refresh timers and generate refresh messages.

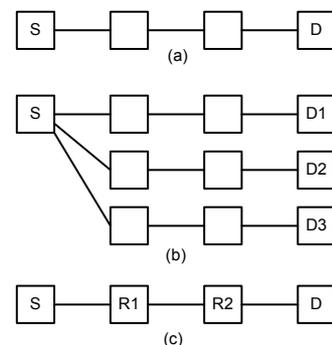


Fig. 1. Network topology for single-hop and multi-hop signaling system.

Intuitively, the E2E approach is simple but probably less efficient in adapting to heterogeneous links and dynamic network conditions. In contrast, the HbH approach is more adaptive as each router can configure its local refresh timer and soft state lifetime to adapt to the local link characteristics,

at the cost of higher signaling overhead (e.g., higher refresh message rates and larger delays in removing stale states). To date most existing research on soft state signaling protocols has been focused on single hop systems or studied multi-hop systems from a local perspective (i.e., looking at the signaling performance of a single hop in the multi-hop system). However, the end-to-end performance of the HbH approach has not been reported in the literature. In addition, little effort has been devoted to addressing the issue of state removal overhead for both approaches. We believe an in-depth analysis of the various choices of refreshes in multi-hop systems will be helpful in designing new soft state protocols.

In this paper we quantitatively evaluate the performance of two soft state refresh approaches from the perspectives of both state refresh and state removal. The outcomes of our research can be applied to the design and configuration of soft state signaling protocols. The contributions of this paper are threefold. Firstly, we introduce a novel model to study the steady state refresh performance metrics of state consistency and refresh message rate for the E2E and HbH approaches. Closed-form solutions are obtained for the performance metrics. Secondly, we find that the delay of removing stale states can cause a low resource utilization (e.g., required memory for maintaining stale states, and unnecessarily reserved bandwidth). The delay is an important metric for the overall design and performance evaluation of the soft state signaling protocols. However, the overhead of delayed state removal has not been studied. We propose an analytic model for the E2E and HbH approaches to study the transient behavior of the state removal process. We obtain closed-form solutions for the state removal delay, and derive a simple upper bound on the HbH state removal delay. To the best of our knowledge, the analytic models for E2E and HbH on state removal delay are the first ones reported in the literature. Thirdly, based on our analysis, we thoroughly investigate how the state refresh and removal performances are affected by the system and protocol parameters, knowledge of which is helpful for adaptively configuring the parameters of the E2E and HbH protocols.

The rest of the paper is structured as follows. Related works are presented in Section II. We introduce the E2E and HbH state refresh approaches and model assumptions in Section III. Analytic model for the steady state refresh performances of E2E and HbH approaches are proposed in Section IV. Section V presents the transient state removal performances of both approaches. Section VI presents numeric results, and Section VII concludes the paper.

II. RELATED WORK

Soft state is an important design aspect for a signaling protocol with an ability to adapt to extreme network situations, such as packet loss, node or link failures. Soft state based signaling has been investigated in the literature and is reflected in protocol designs, such as RSVP [4], [23], YESSIR [16], Boomerang [7], RTCP [19], PIM-SM [6] and more recently, NSIS [8], [9].

Sharma et al. [21] study the issue of scalable timers between a state sender and a state receiver for single-hop soft state signaling protocols and propose an adaptive algorithm for 1) dynamically adjusting the sender's refresh rate according to the number of states and available bandwidth, and 2) estimating the sender's refresh rate and determine state timeout timer values. Their work in [21] is focused on the maintenance of multiple states. Also, their algorithm is evaluated by simulation from a local perspective, while our paper presents an end-to-end evaluation (including state consistency and message rate). For addressing the scalability issue, the authors of [1], [13], [14], [22] and some other proposals study state aggregation techniques.

Raman and McCanne [18] propose a formal model for soft state communication and analytically model the state consistency and performance tradeoffs for an open-loop announce/listen protocol in single-hop signaling system. The protocol is modeled based on classic queueing networks by taking into account workload and network loss rate. A protocol incorporating feedback is studied by simulation and is observed to improve system performances. Lui et al. [12] introduce three scenarios, namely denial of service attacks, overload and rapid change of broadcast membership, to examine the robustness property of signaling protocols. A simple analytic model is proposed to study the costs associated with events of initialization, refresh, inconsistency and stale state in a single-hop system. However, the assumption in [12] that soft state information expires if a single refresh message is not received may not hold in practical protocols.

To provide reliability for signaling message delivery, the work in [2], [15] introduces hop-by-hop retransmission to RSVP, while the work in [20] reuses existing reliable transport protocols. Ji et al. [10], [11] systematically classify soft state and hard state signaling protocols into five "soft" level classes. They also model these classes and study their consistency ratio. The impacts of refresh timer and reliability of signaling messages on E2E refresh approach are modeled for the multi-hop system [11]. However, the state removal performance of E2E approach has not been studied. The HbH refresh approach is also not investigated in [11]. It is noted that signaling cost related to state removal has been mentioned in [10] [12] [21], but not quantitatively analyzed.

The first work addressing both E2E and HbH approaches is [16], which gives an empirical study of processing overhead of both approaches and concludes that end-to-end refreshes in YESSIR yield to a lower overhead than hop-by-hop refreshes in RSVP. However, [16] focuses on the comparison between a specific YESSIR implementation and an RSVP implementation, instead of looking at this issue in more generality. In [17] the state refresh performance of E2E and HbH has been investigated. However, the proposed models in [17] are quite complex while state removal performance is not addressed.

III. STATE REFRESH APPROACHES AND ASSUMPTIONS

We consider a scenario of soft-state signaling in a multi-hop system (as shown in Fig. 1c) with sessions from source

to destination. In the path from source to destination, there are N intermediate routers (denoted by router n , $n \in [1, N]$) and $N + 1$ links. The soft states with the sessions are generated by the source, and then recorded/refreshed at the N routers. We further divide the life cycle of a state into three phases, i.e., state installation phase, state refresh phase (SFP) and state removal phase (SRP). The state installation phase starts with the generation of a state at the Source and ends at the installation of the states at all the N routers. The SFP follows the state installation phase and ends at the deletion of the state at the source. The SRP follows the SFP and ends at the instance that the state is removed by all the N routers.

In practical signaling protocols, installation and update of state associated with a session are normally implemented with an end-to-end confirmation. State installation can be source initiated or destination initiated (e.g., RSVP). Here we consider the source initiated installation, which is performed identically in both the E2E and the HbH approaches. The source sends out a state installation message (Inst_Msg) towards the receiver. We assume the underlying signaling message transport mechanism is unreliable (such as IP or UDP used in RSVP, PIM-SM, YESSIR [16], or the datagram mode of GIST [20]). The Inst_Msg message may be lost during transport. An intermediate router which receives an Inst_Msg message forwards the message to its downstream router. If the last hop router receives the Inst_Msg message, it sends a state confirmation message back to the source. The intermediate routers that received the confirmation message install the state, and forwards the confirmation message to the source. If the source does not receive a confirmation message it will retransmit Inst_Msg message after a certain time.

Once the source receives a confirmation that a state has been installed at the N routers, it sends state refresh message (Fresh_Msg) to the routers to keep the state alive. Here an alive state at a router means that a state is installed and not removed at the router. For simplicity, we focus here on the state refresh and state removal for a single soft state rather than multiple states. A stale state is a soft state that has been deleted by the source but is still maintained by an intermediate router. The processing of multiple soft states can generally be considered as multiple instances of the single state processing. Since we focus on the properties of different refresh mechanisms, we do not consider state updates in this paper.

The main task of state refresh is to make the state of the routers synchronized to that generated at the source. For simplicity, we assume that clocks on all routers are perfectly synchronized and tick at the same rate. In the basic E2E refresh, right after a new state generated by the source is installed in a router, a state timeout timer is set in the router to a predefined value of L_e seconds. The soft state of the router expires and is removed if the router does not receive any Fresh_Msg message from the source at the expiration of the timeout timer. The source will set a refresh timer to send periodic Fresh_Msg message along the path with an average period of R_e . Any intermediate router that receives a Fresh_Msg message resets its timeout timer to L_e and

forwards the received Fresh_Msg message to its downstream router if any. If a router's timeout timer expires, the router removes the state even the state is alive at the source.

In the basic HbH approach, the intermediate routers as well as the source maintain their own state refresh timers. Each intermediate router generates periodic Fresh_Msg message for a alive state to its downstream router, rather than forward the Fresh_Msg message received from the source or its upstream router. The values of the refresh period used by the source and the intermediate routers can be configured differently and distributively. For simplicity, we assume the refresh period used to generate a Fresh_Msg message for router n in the HbH approach is fixed as R_n , where $n \in [1, N]$. The refresh period R_n is sent with the Fresh_Msg message to router n , based on which router n can determine the value for its local state timeout timer. We let L_n denote the value of the state timeout timer for router n , $n \in [1, N]$. The soft state in router n expires and is removed if no Fresh_Msg message is received from its upstream router $n - 1$ (or source) at the expiration of router n 's timeout timer.

We assume that each soft state associated with a session has a limited but relatively long lifetime. After a session is finished, the source deletes the state associated with the session and send out a state removal message (Del_Msg) to remove the state. The state removal process is the same for both E2E and HbH approaches. The transport of a Del_Msg message is unreliable. The probability that a Del_Msg message transmitted over link n is lost is denoted by $p_{d,n}$. We assume that a lost Del_Msg message is not retransmitted. Therefore the routers that do not receive the Del_Msg message are not explicitly informed to remove the stale state, but eventually remove the stale state with a delay and associated overheads.

As mentioned in [11] and [21], the cost associated with a soft state signaling protocol can be summarized from the following three aspects:

- State initialization cost, e.g. the performances of state installation delay and bandwidth overhead;
- State refreshment cost, e.g. cost due to state inconsistency and bandwidth consumption due to the transmission of Fresh_Msg message;
- State removal cost, e.g. cost for the state removal messages and for the delay of removing stale state.

Assume that message loss for an Inst_Msg message is random and independent with probability p_n over the n th link, the propagation delay is $T_{P,n}$ over the n th link, and the value of retransmission timer for an Inst_Msg message is T_{rtn} . Then we can simply get a state initialization delay by (1):

$$D_I = \sum_{i=0}^{\infty} \left[\prod_{n=1}^N (1-p_n) \right] \left[1 - \prod_{n=1}^N (1-p_n) \right]^i \left[iT_{rtn} + \sum_{n=1}^N 2T_{P,n} \right]. \quad (1)$$

It is noted that we have ignored the queuing delay in the above delay analysis for simplicity. In the next sections, we analyze the performances related to state refresh and state removal, respectively.

Table I summarizes the notations used in this paper.

IV. STATE REFRESH PERFORMANCE

The performance metrics of interest for the refresh performance of both E2E and HbH are state inconsistency ratio and message rate. In the SFP a router's state is called *consistent* if the state is the same as the source's state, otherwise the state is called *inconsistent*. For the multi-hop signaling system, there can be many ways to measure the state consistency performance, for example, state consistency over the whole path or at the individual routers. Here we use *inconsistency ratio* as the measure of end-to-end state consistency, defined as the ratio of the time in the SFP during which not all of the routers' states are inconsistent to the lifetime of the source state. Message rate is defined as the average number of Fresh_Msg messages sent over the whole network in one second in the SFP.

For simplicity, we assume relationships of $L_e=K_eR_e$ for E2E refresh and $L_n=K_nR_n$ in router n for HbH refresh, where K_e and K_n ($n \in [1, N]$) are small integers that are called lifetime coefficient. Each Fresh_Msg message sent over any hop independently experiences a random loss with probability p_n over the n th link, for $n \in [1, N]$. For the modeling purpose, we assume that the state lifetime and refresh period are exponentially distributed random variables for both E2E and HbH approaches. Compared to the refresh period (e.g., 30 seconds by default for RSVP) or the soft state lifetime, the link propagation delay is normally very small and thus assumed to be negligible.

A. E2E Refresh

Let us first study the average message rate for E2E refresh (denoted by γ_e). The source generates a Fresh_Msg message every R_e seconds on average and sends it toward Destination. The Fresh_Msg message can be lost at any link before reach router N . Let $p_{x,n}$ denote the probability that the message is lost exactly at the n th hop, which can be calculated as p_1 for $n=1$ and $p_n \prod_{i=1}^{n-1} (1 - p_i)$ for $n > 1$. It is obvious that γ_e is $\frac{1}{R_e}$

for $N = 1$. We can easily obtain the average message rate γ_e for $N > 1$:

$$\begin{aligned} \gamma_e &= \frac{1}{R_e} \left[\sum_{n=1}^{N-1} np_{x,n} + N \prod_{i=1}^{N-1} (1 - p_i) \right] \\ &= \frac{1}{R_e} \left[p_1 + \sum_{n=2}^{N-1} np_n \prod_{i=1}^{n-1} (1 - p_i) + N \prod_{i=1}^{N-1} (1 - p_i) \right]. \end{aligned} \quad (2)$$

A received Fresh_Msg message can make a router consistent with the source. A state is removed by a router only if the router losses all refresh messages during the router's state lifetime. Obviously, if a router removes its state, all the downstream routers will also remove their states as those routers will not receive any Fresh_Msg message before their timeout timers expire as well. Therefore we only need to study the state refresh at router N to compute the inconsistency ratio for E2E refresh.

Assuming that the SFP is sufficiently long, we model the state refresh process in router N as a continuous Markov process. The Markov chain is shown in Fig. 2 with Markov state spaces $\Omega_e=\{(k)\}$, where $0 \leq k \leq K_e$ is an integer variable, with the meaning of remaining lifetime of the state at router N divided by refresh period R_e . Router N in state (k) means that its timeout timer value (TTV) is kR_e and the state will be removed if router N does not receive any of the next k consecutive Fresh_Msg message from the source.

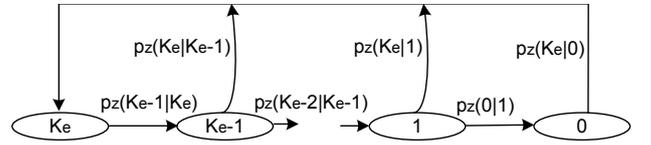


Fig. 2. Markov chain for E2E refresh approach. K_e is shown as K in the figure.

Let $p_z(i_2|i_1)$ denote the transition rate from Markov state (i_1) to (i_2) , where $i_1, i_2 \in \Omega_e$. The transition rates $p_z(i_2|i_1)$ between the Markov states in Ω_e can be expressed by (3):

$$\begin{aligned} p_z(i-1|i) &= \frac{1}{R_e} \left[1 - \prod_{n=1}^N (1 - p_n) \right], i \in [1, K_e] \\ p_z(K_e|i) &= \frac{1}{R_e} \left[\prod_{n=1}^N (1 - p_n) \right], i \in [0, K_e - 1] \\ p_z(i|j) &= 0, \text{ otherwise.} \end{aligned} \quad (3)$$

The first equation in (3) corresponds to the event that a Fresh_Msg message is lost in the transport to router N . The second equation in (3) corresponds to the event of a successful Fresh_Msg message reception by router N .

Let π_k denote the stationary distributions of the Markov state k in Ω_e in the SFP, where $k \in [0, K_e]$. A simple numeric approach can be used to solve the Markov chain from (3) and obtain the stationary distributions. Then, the inconsistency ratio for E2E refresh (denoted by ζ_e) is obtained as π_0 .

TABLE I
NOTATIONS.

T_{rtn}	Retransmission delay for Inst_Msg
T_P	Propagation delay
p_n	Message loss prob. for Inst_Msg, Fresh_Msg over n th link
$p_{d,n}$	Message loss prob. for Del_Msg over n th link
R_e	State refresh period for E2E
L_e	Lifetime of a refreshed state in E2E
K_e	Lifetime coefficient, a constant with $K_e = L_e/R_e$
R_n	State refresh period for HbH over the n th link
L_n	Lifetime of a refreshed state at router n in HbH
K_n	Lifetime coefficient, a constant with $K_n = L_n/R_n$
γ_e	Average message rate of E2E refresh
ζ_e	State inconsistency ratio of E2E approach
ω_e	State removal delay of E2E approach
γ_h	Average message rate of HbH refresh
ζ_h	State inconsistency ratio of HbH approach
ω_h	State removal delay of HbH approach

Additionally, we can achieve a closed-form solutions for the Markov states. Define three temporary variables as follows:

$$\begin{aligned} a &= \frac{1}{R_e} \left[1 - \prod_{n=1}^N (1 - p_n) \right], \\ b &= \frac{1}{R_e} \left[\prod_{n=1}^N (1 - p_n) \right], \quad c = \frac{a+b}{a}. \end{aligned} \quad (4)$$

We have

$$\begin{aligned} a\pi_{K_e} &= b \sum_{k=0}^{K_e} \pi_k, \\ a\pi_k &= (a+b)\pi_{k-1}, \quad k \in [2, K_e - 1], \\ a\pi_1 &= b\pi_0. \end{aligned} \quad (5)$$

After some derivations, we can get

$$\pi_0 + \frac{b}{a}\pi_0 \sum_{k=0}^{K_e-1} c^k = 1. \quad (6)$$

Therefore we obtain a closed-form solution for π_0 :

$$\pi_0 = \frac{1}{1 + \frac{b}{a} \frac{1-c^{K_e}}{1-c}} = c^{-K_e} = \left(\frac{a}{a+b} \right)^{K_e}. \quad (7)$$

and obtain closed-form solutions for π_k ($k \in [1, K_e]$):

$$\begin{aligned} \pi_1 &= \frac{b}{a}\pi_0, \\ \pi_k &= c^{k-1}\pi_1, \quad k \in [2, K_e]. \end{aligned} \quad (8)$$

B. HbH Refresh

In the HbH approach the routers periodically send Fresh_Msg messages in the SFP to their downstream routers provided their states are not removed. A router's state is now directly refreshed by its upstream router (or the source) rather than by the source only. Unlike the E2E approach, it is possible that a router's state expires but the state of its downstream routers does not. To facilitate the modeling, we make an assumption that the state refresh process of a router is independent of those of other routers, except that the process is affected by the probability that the state of its upstream router (or the source) is alive.

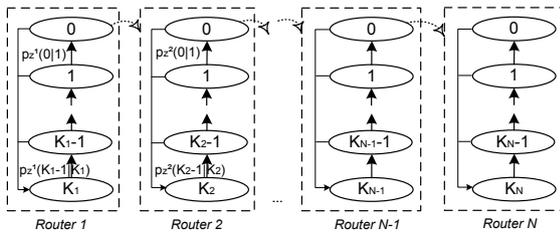


Fig. 3. Markov chains for HbH refresh approach.

In the SFP the source state is always alive. We use a Markov chain identical to that shown in Fig. 2 to model the HbH refresh process for each of the N routers. An illustration of the N Markov chains for the N routers is presented in Fig. 3.

The Markov state spaces for router n is denoted by $\Omega_n = \{(k)\}$, where $0 \leq k \leq K_n$ has the same meaning as that of (k) defined in Subsection IV-A. Let $p_z^n(i_2|i_1)$ denote the transition rate from the Markov state (i_1) to (i_2) for router n , where $(i_1), (i_2) \in \Omega_n$ and $1 \leq n \leq N$. Let $\mu_{n,0}$ denote the stationary probability that the state of router n 's upstream neighbor is not maintained. The transition rates for the Markov states of router n can be expressed by:

$$\begin{aligned} p_z^n(i-1|i) &= \frac{\mu_{n,0} + (1 - \mu_{n,0})p_n}{R_n}, \quad i \in [1, K_n] \\ p_z^n(K_n|i) &= \frac{(1 - \mu_{n,0})(1 - p_n)}{R_n}, \quad i \in [0, K_n - 1] \\ p_z^n(i|j) &= 0, \quad \text{otherwise.} \end{aligned} \quad (9)$$

Let $\pi_{n,k}$ denote the stationary distributions of the Markov states (k) of router n , where $(k) \in \Omega_n$ and $0 \leq n \leq N$. As we have $\mu_{1,0}=0$ in the SFP, we can solve the Markov chain for router 1 numerically by (9) and obtain $\pi_{1,k}$, where $0 \leq k \leq K_1$. Similarly, replacing $\mu_{2,0}$ by $\pi_{1,0}$, we can solve the Markov chain for router 2 and obtain $\pi_{2,k}$, where $0 \leq k \leq K_2$. The stationary distributions of the Markov states for the remaining routers can be computed sequentially.

Similar to the E2E approach, we can obtain closed-form solutions for the stationary distributions of the Markov states for each intermediate router in the HbH approach. Define three sets of temporary variables $\{\alpha_n\}, \{\beta_n\}, \{\theta_n\}$ with $1 \leq n \leq N$ as follows:

$$\begin{aligned} \alpha_n &= \frac{\mu_{n,0} + (1 - \mu_{n,0})p_n}{R_n}, \\ \beta_n &= \frac{(1 - \mu_{n,0})(1 - p_n)}{R_n}, \\ \theta_n &= 1 + \frac{\beta_n}{\alpha_n}. \end{aligned} \quad (10)$$

We can obtain the stationary state balance equations for the distributions of router n 's Markov states $\pi_{n,k}$ ($k \in [1, K_n]$):

$$\begin{aligned} \pi_{n,0} &= \left(\frac{\alpha_n}{\alpha_n + \beta_n} \right)^{K_n}, \\ \pi_{n,1} &= \frac{\beta_n}{\alpha_n} \pi_{n,0}, \\ \pi_{n,k} &= \theta_n^{k-1} \pi_{n,1}, \quad k \in [2, K_n]. \end{aligned} \quad (11)$$

Once the stationary distributions $\pi_{n,k}$ for all routers are determined, the inconsistency ratio for HbH refresh (denoted by ζ_h) can be calculated by:

$$\zeta_h = 1 - \prod_{n=1}^N (1 - \pi_{n,0}). \quad (12)$$

The average message rate γ_h for HbH refresh is obtained by:

$$\gamma_h = \sum_{n=1}^N \frac{1 - \mu_{n,0}}{R_n}. \quad (13)$$

V. STATE REMOVE PERFORMANCES

At the beginning of the state removal phase, the source deletes its state and sends out a Del_Msg message towards the destination to remove state installed along the path. If an intermediate router receives the Del_Msg message, it removes its state if the state is alive and forward the Del_Msg message. Otherwise, the stale state at the router is removed with a delay after the router's timeout timer expires. The performance measures of the SRP are the number of transmitted signaling messages (only Del_Msg message for the E2E approach, both Fresh_Msg and Del_Msg messages for the HbH approach) and how long elapsed time until the routers remove the stale state in the SRP.

A. E2E Approach

Let $N_{srp,e}$ denote the number of signaling messages transmitted in the SRP for the E2E refresh approach. Only Del_Msg messages are transmitted during the SRP for the E2E approach. It is obvious that $N_{srp,e} = 1$ for $N = 1$. We can obtain $N_{srp,e}$ for $N > 1$ by:

$$N_{srp,e} = p_{d,1} + \sum_{n=2}^{N-1} n p_{d,n} \prod_{i=1}^{n-1} (1 - p_{d,i}) + N \prod_{i=1}^{N-1} (1 - p_{d,i}). \quad (14)$$

With regard to the delay in removing the stale state, it is observed that if a router receives the Del_Msg message, its state is immediately deleted with a zero delay. Otherwise, if the router fails to receive the Del_Msg message, the delay is equal to the router's TTV for the state. For simplicity, we assume that at the end of the SFP, the TTV of router n is an integer multiple of the refresh period (R_e and R_n for E2E and HbH approaches, respectively) with the same distribution as that of the stationary Markov states in the SFP. Next we extend the Markov chain proposed for router N in Subsection IV-A to router n , $1 \leq n \leq N$.

The Markov chains for all N routers are identical to the one shown in Fig. 2. Note that the Markov chains of the routers are independent. For a general router n , the Markov state space is $\Omega_e = \{(k)\}$, where k ($0 \leq k \leq K_e$) is the TTV of n divided by refresh period R_e . Define three sets of temporary variables for router n as follows:

$$\begin{aligned} a_n &= \frac{1}{R_e} \left[1 - \prod_{i=1}^n (1 - p_i) \right], \\ b_n &= \frac{1}{R_e} \left[\prod_{i=1}^n (1 - p_i) \right], \quad c_n = \frac{a_n + b_n}{a_n}. \end{aligned} \quad (15)$$

Let $\pi_{n,k}$ denote the stationary distribution of Markov state (k) at router n , $k \in [2, K_e]$. We can get the state transition probabilities for router n as the following:

$$\begin{aligned} a_n \pi_{n,K_e} &= b_n \sum_{k=0}^{k=K_e} \pi_{n,k}, \\ a_n \pi_{n,k} &= (a_n + b_n) \pi_{n,k-1}, \quad k \in [2, K_e - 1], \\ a_n \pi_{n,1} &= b_n \pi_{n,0}. \end{aligned} \quad (16)$$

After some derivations, we can get closed-form expressions for router n 's stationary Markov state distributions:

$$\begin{aligned} \pi_{n,0} &= c_n^{-K_e} = \left(\frac{a_n}{a_n + b_n} \right)^{K_e}, \\ \pi_{n,1} &= \frac{b_n}{a_n} \pi_0, \\ \pi_{n,k} &= c_n^{k-1} \pi_{n,1}, \quad k \in [2, K_e]. \end{aligned} \quad (17)$$

Let ω_e denote the average delay in removing stale state over all N routers for the E2E approach. Then we can calculate the average delay in removing stale state at each router and sum them up to get ω_e by:

$$\omega_e = \sum_{n=1}^N R_e \left[1 - \prod_{m=1}^n (1 - p_{d,m}) \right] \sum_{i=1}^{K_e} i \pi_{n,i}. \quad (18)$$

Note that the average delay in removing stale state for the E2E approach is a linear function of R_e and a nonlinear function of K_e .

B. HbH Approach

Let $N_{srp,h}$ denote the number of signaling messages transmitted in the SRP for the HbH refresh approach. Del_Msg message and possibly Fresh_Msg message are transmitted during the SRP for the HbH approach. The average number of Del_Msg messages transmitted in the SRP for the HbH approach is the same as $N_{srp,e}$. The number of Fresh_Msg messages transmitted in the SRP will be calculated after we get state removal delay.

For the E2E approach, the processes on stale state in the SRP are independent for the N routers. This is different for the HbH approach. Suppose that the Del_Msg message from the source is lost exactly over the n th link. The states of router n 's upstream routers are removed and their delay in removing stale state is zero. However, the TTV of router n now affects the delay of removing stale state at router n , as well as the delay at the downstream routers if $n < N$.

We let E_n denote the event that the Del_Msg message is lost exactly at the n th hop ($1 \leq n \leq N$) and let p_{E_n} denote the probability of event E_n . p_{E_n} is $p_{d,1}$ for $n = 1$ and can be calculated for $2 < n \leq N$ by:

$$p_{E_n} = p_{d,n} \prod_{m=1}^{n-1} (1 - p_{d,m}) \quad (19)$$

The delay of removing stale state at a router l due to event E_n is 0, if $1 \leq l < n \leq N$. For the event E_n , we consider a worst case for the delay of removing stale state which occurs in the following scenario:

- At the end of the SFP, the timeout timer value of router n is $K_n R_e$; and
- the Fresh_Msg messages sent by router n and its downstream routers (if any) are all successfully received by the target receivers.

The worst case delay of removing stale state at router l with event E_n can be calculated for $n < l \leq N$ by:

$$R_l + \sum_{m=n}^l R_m(K_m - 1). \quad (20)$$

Let $\omega_{h,u}$ denote the average delay of removing the stale state over the N routers in the worse case for the HbH approach, which is an upper bound for the average delay of removing stale state. By counting all the events E_n ($1 \leq n \leq N$), we obtain the average delay in worst case $\omega_{h,u}$ by:

$$\omega_{h,u} = \sum_{n=1}^N p_{E_n} \sum_{l=n}^N \left[R_l + \sum_{i=n}^l R_i(K_i - 1) \right]. \quad (21)$$

It is obvious that the overall wasted resource for the HbH approach is also a linear function of R_n .

Next we give an analysis on the delay of removing stale state for the HbH approach. Define a variable (denoted by $\vartheta_{n,l,t}$) as the probability that the state of router l is alive between the t th and $t+1$ th refresh points for router l since the end of SFP, under the condition of event E_n , where $1 \leq n \leq l \leq N$ and $t \geq 1$. Define an intermediate variable $\kappa_{n,l}$ for router l under the condition of event E_n , with the following expression:

$$\begin{aligned} \kappa_{n,l} &= 0, n > l, \\ \kappa_{n,n} &= K_n, \\ \kappa_{n,l} &= \kappa_{n,l-1} - 1 + K_l, n < l \leq N. \end{aligned} \quad (22)$$

For $t > \kappa_{n,l}$, we always have $\vartheta_{n,l,t} = 0$.

With event E_n , the state at router n is not refreshed and is removed after router n 's timeout timer expires. The distribution of router n 's TTV is obtained from the stationary Markov state distribution of router n in the SFP, which is $\pi_{n,k}$ ($0 \leq k \leq K_n$). We can easily obtain $\vartheta_{n,n,t}$ by:

$$\begin{aligned} \vartheta_{n,n,t} &= \sum_{k=t}^{K_n} \pi_{n,k}, 1 \leq t \leq K_n, \\ \vartheta_{n,n,t} &= 0, t \geq K_n. \end{aligned} \quad (23)$$

Under the event E_n , it is more difficult to calculate the delay of removing stale state at a downstream router l with $l > n$. We compute $\vartheta_{n,l-1,t}$ first, which is the probability that router $l-1$ send a refresh at the t th refresh period since the end of SFP, where $1 \leq n < l \leq N$ and $t \geq 1$. To take into account the impact of the probabilistic Fresh_Msg message from the upstream router ($l-1$) on a tagged router l , we define a transient Markov chain for router l . The transient Markov chain has the same structure as that shown in Fig. 2, but the state transition probabilities are different and change with time according to $\vartheta_{n,l-1,t}$ due to the Fresh_Msg messages from the upstream router (router $l-1$). We denote the transient Markov state at the t th refresh instance of the SRP by $\langle k \rangle_{n,l,t}$, where $0 \leq k \leq K_l$, and let $v_{n,l,t,k}$ denote the corresponding distribution of the transient Markov states $\langle k \rangle_{n,l,t}$ right after the t th refresh point. Further, let $V_{n,l,t}$ be a $(K_l + 1)$ by 1 vector, which denotes the distribution of the transient Markov

states at the t th refresh period in the SRP, under the condition of event E_n , with $V_{n,l,t}(k) = v_{n,l,t,k}$ for $k \in [0, K_l]$. If the upstream router $l-1$ does not generate a Fresh_Msg message at the first refresh instance of the SRP (with probability $1 - \vartheta_{n,l-1,1}$), the distribution of the transient Markov states $\langle k \rangle_{n,l,1}$ at the end of the SRP equals to the distribution of the stationary Markov states in SFP $\pi_{l,k}$. However, if the upstream router $l-1$ sends a Fresh_Msg message to the tagged router l (with probability $\vartheta_{n,l-1,1}$), the distributions of $\langle k \rangle_{n,l,1}$ are obtained from the distribution of the stationary Markov states $\pi_{l,k}$ with one-step state transition, the probability of which will be calculated next.

Define two metrics $\Gamma_{l,0}$ and $\Gamma_{l,1}$ for the tagged router l , where $\Gamma_{l,1}(i, j)$ and $\Gamma_{l,0}(i, j)$ denote the one-step transition rate from a transient Markov state $\langle i \rangle_{n,l,t}$ to $\langle j \rangle_{n,l,t+1}$ under the conditions that the upstream router $l-1$ sends and does not send a Fresh_Msg message to router l at the t th refresh instance of the SRP, respectively, where $i, j \in [0, K_l]$ and $1 \leq t \leq \kappa_{n,l}$. It is easy to obtain $\Gamma_{n,1}$ by:

$$\begin{aligned} \Gamma_{l,1}(i, j) &= p_l, 1 \leq i \leq K_l \text{ and } j = i - 1, \\ \Gamma_{l,1}(0, 0) &= p_l, \\ \Gamma_{l,1}(i, K_l) &= 1 - p_l, 0 \leq i \leq K_l, \\ \Gamma_{l,1}(i, j) &= 0, \text{ otherwise.} \end{aligned} \quad (24)$$

and obtain $\Gamma_{n,0}$ by:

$$\begin{aligned} \Gamma_{l,0}(0, 0) &= 1, \\ \Gamma_{l,0}(i, j) &= 1, 1 \leq i \leq K_l \text{ and } j = i - 1, \\ \Gamma_{l,0}(i, j) &= 0, \text{ otherwise.} \end{aligned} \quad (25)$$

Then we can get the vector $V_{n,l,1}$, the distribution of the initial transient Markov states by:

$$V_{n,l,1} = (1 - \vartheta_{n,l-1,1})(\pi_{l,0}, \pi_{l,1}, \dots, \pi_{l,K_l}) + \vartheta_{n,l-1,1}(\pi_{l,0}, \pi_{l,1}, \dots, \pi_{l,K_l})\Gamma_{l,1}. \quad (26)$$

After obtaining the vector $V_{n,l,1}$, we can sequentially obtain $V_{n,l,t}$ for $2 \leq t \leq \kappa_{n,l-1}$ by:

$$V_{n,l,t} = (1 - \vartheta_{n,l-1,t})V_{n,l,t-1}\Gamma_{l,0} + \vartheta_{n,l-1,t}V_{n,l,t-1}\Gamma_{l,1}. \quad (27)$$

Then $\vartheta_{n,l,t}$ can be calculated by:

$$\begin{aligned} \vartheta_{n,l,t} &= 1 - v_{n,l,t,0}, 1 \leq t \leq \kappa_{n,l-1} - 1, \\ \vartheta_{n,l,t} &= \sum_{i=t-\kappa_{n-1}+1}^{K_n} v_{n,l,\kappa_{n-1},i}, \kappa_{n,l-1} \leq t \leq \kappa_l \\ \vartheta_{n,l,t} &= 0, t > \kappa_{n,l}. \end{aligned} \quad (28)$$

In summary, for any $1 \leq n \leq N$, we can use (23) to calculate $\vartheta_{n,n,t}$ ($1 \leq t \leq K_n$). After obtaining $\vartheta_{n,n,t}$, we can use (28) to calculate $\vartheta_{n,l,t}$ ($n < l \leq N$, $1 \leq t \leq \kappa_{n,l}$). We are now ready to calculate the average number of Fresh_Msg messages transmitted in the SRP for the HbH approach. Under the event E_n , the average number of Fresh_Msg transmissions can be computed by $\sum_{l=n}^N \sum_{t=1}^{\kappa_{n,l}} \vartheta_{n,l,t}$. Therefore we can

compute $N_{srp,h}$ (i.e., the average number of transmissions of signaling messages in the SRP for the HbH approach) by:

$$N_{srp,h} = N_{srp,e} + \sum_{l=n}^N R_l \sum_{t=1}^{\kappa_{n,l}} p_{E_n} \vartheta_{n,l,t}. \quad (29)$$

Let $\omega_{h,n}$ denote the average delay of removing stale state over the N routers under the condition of event E_n for the HbH approach, $1 \leq n \leq N$. Further, let ω_h denote the average delay of removing stale state at the N routers for the HbH approach. Once we obtain $\vartheta_{n,l,t}$, where $1 \leq n \leq N$, $n < l \leq N$ and $1 \leq t \leq \kappa(n, l)$, we can calculate $\omega_{h,n}$ by:

$$\omega_{h,n} = \sum_{l=n}^N R_l \sum_{t=1}^{\kappa_{n,l}} \vartheta_{n,l,t}, \quad (30)$$

and calculate ω_h by:

$$\omega_h = \sum_{n=1}^N p_{E_n} \omega_{h,n}. \quad (31)$$

VI. NUMERICAL RESULTS

We have implemented a discrete event-driven simulator to evaluate the state refresh and removal performances of the E2E and HbH approaches for multi-hop soft state signaling protocols. We performed 20 simulation runs and obtain an average value over all runs for each point in the graphs. Each simulation runs for 10^7 seconds. Unless otherwise specified, we will use the following default parameters: $N = 10$, $R_e = 30$ s, $K_e = 3$, $R_n = 30$ s and $K_n = 3$ (for $1 \leq n \leq N$), $p_n = 0.05$ and $p_{d,n} = 0.05$ (for $1 \leq n \leq N$).

A. State Refresh

1) *Impact of Message Loss Rate*: Fig. 4 compares the inconsistency ratio of E2E and HbH approaches for various message loss rates with 5 and 10 hops. Both simulation and analytical results are presented, denoted in the figure by “sim” and “mod”, respectively. For comparison, the results obtained by Ji’s analytical model for the E2E approach [11] are also presented, which is denoted by “Ji”. In Ji’s Markov chain analytical model, two so-called paths are defined, with a slow path corresponding to a Fresh_Msg message loss in the transport to router N and a fast path corresponding to a successful Fresh_Msg message reception by router N . Compared to Ji’s analytical model, our proposed model is simpler and has higher accuracy, for example, the inconsistency ratio of the E2E approach can be calculated by only one simple formula given in equation (7). Additionally we observe from Fig. 4 that our analytical model has high accuracy for both refresh approaches. As the analytical methods proposed in this paper and in [17] have similar, we did not include the results from [17] for readability.

Fig. 5 plots the message rate of the E2E and HbH approaches. Fig. 4 and Fig. 5 show that the HbH approach is much better in maintaining consistent state in the routers than the E2E approach due to the local refresh and resilience to the message loss of the HbH approach. The inconsistency ratio of

the HbH approach is mostly one order of magnitude lower than that of the E2E approach. However the better consistency of the HbH approach is achieved at the cost of higher message overhead. For both 5 and 10 hops signaling systems, the message rate of the HbH approach does not change much when message loss rate is smaller than 0.4 and then drops significantly when the message loss rate exceeds 0.6.

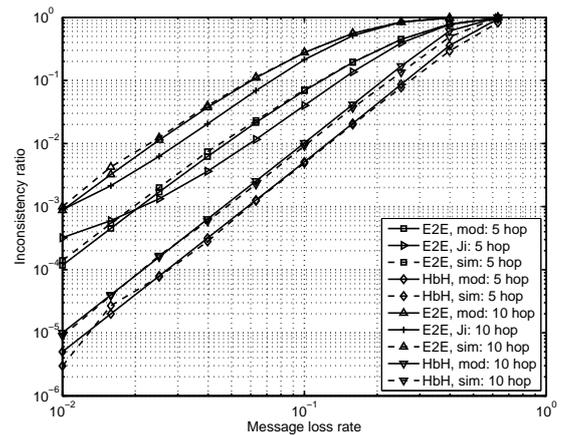


Fig. 4. State inconsistency rate versus link message loss rate.

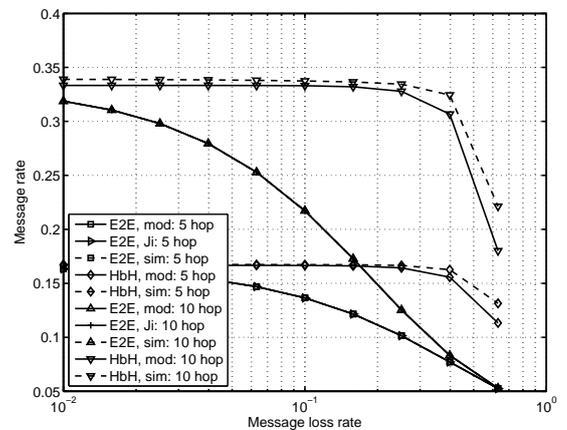


Fig. 5. Message rate versus link message loss rate.

2) *Impact of Soft State Timeout Timer and Number of Hops*: Next we explore the impact of the timeout timer on the E2E and HbH approaches with 5 and 9 hops. Fig. 6 and Fig. 7 show the analytical results of inconsistency ratio and message rate for the E2E and HbH approaches, respectively. The message loss rate is set to 0.056. It is observed that inconsistency ratio of the HbH approach decreases steeply with the lifetime coefficient K_n and the message rate of the HbH approach increases when K_n changes from 1 to 3, but remains unchanged when K_n is equal to or larger than 3. The inconsistency ratio of the E2E approach also decreases linearly with K_e , but at a much slower speed. As predicted by equation (2), the message rate of the E2E approach is not affected by the state timeout timer.

Fig. 8 and Fig. 9 show the analytical results of inconsistency ratio and message rate for the E2E and HbH approaches against the number of hops, respectively. K_e is set to 3 for the E2E approach, and K_n ($1 \leq n \leq N$) are set to 3 for the HbH approach. Message loss rate is set to 0.032 and 0.18 for all the links. It is observed that the inconsistency rate of the E2E approach increases considerably with the number of hops, while that of the HbH approach does not increase much. The message rate of both approaches increases linearly with the number of the hops. When the message loss rate is low (e.g., 0.032), both approaches have comparable message rates, and the impact of message loss rate on the message rate of the HbH approach is almost negligible.

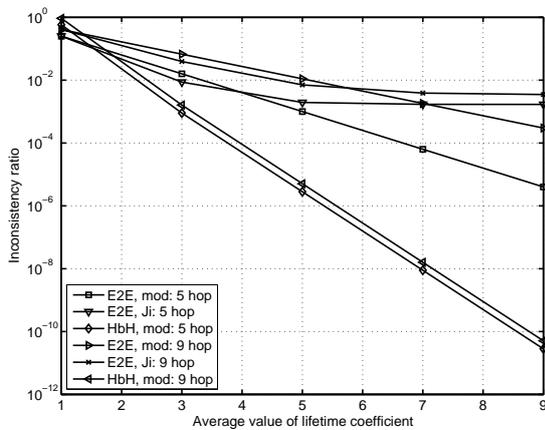


Fig. 6. Average inconsistency ratio against state timeout timer.

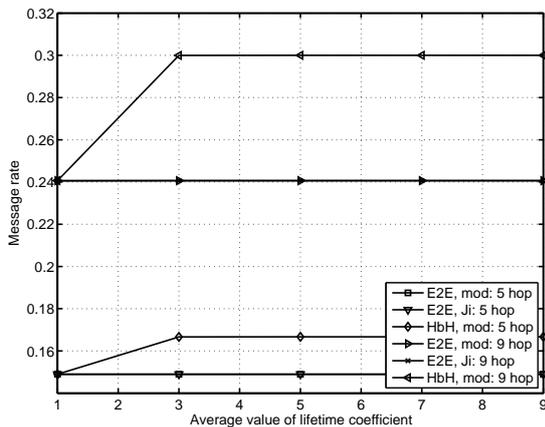


Fig. 7. Average message rate against state timeout timer.

B. State Remove Delay

In this subsection, we explore the state removal delay performance of the E2E and HbH approaches in a multi-hop signaling system. The impact of the message loss rate, state lifetime coefficient and the number of hops are investigated. Fig. 10 presents the average state removal delay of the E2E and HbH approaches against message loss rate. The number of hops is set to 5 and 9, and the lifetime coefficient is

set to 3. As E2E analytical results match perfectly with the simulation results, simulation results for the E2E approach are not presented. Both analytic and simulation results for the HbH approach are presented. Additionally, the upper bound of the state removal delay for the HbH approach is shown in the figure (denoted by “HbH bound, Mod”). Observe that the analytical results match the simulation results well and the upper bound of the state removal delay is very close.

The state removal delay of the HbH approach is several times of that of the E2E approach. Moreover, the state removal delay of the HbH approach increases considerably with the message loss rate, while that of the E2E approach does not increase much with the message loss rate.

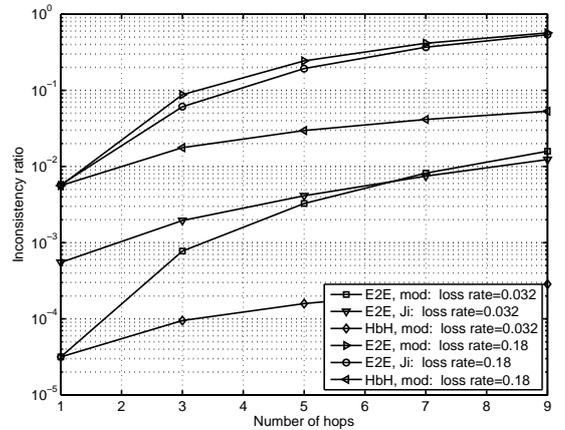


Fig. 8. Average inconsistency ratio against number of hops.

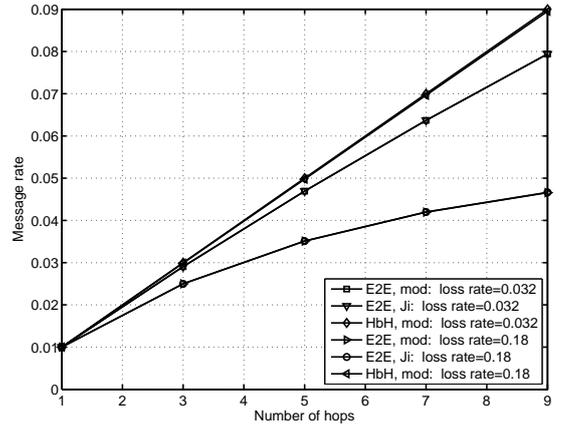


Fig. 9. Average message rate against number of hops.

Fig. 11 presents the average state removal delay of the E2E and HbH approaches against the average value of state lifetime coefficient. The number of hops is again set to 5 and 9, and the message loss rate is set to 0.056. Both analytic and simulation results (for the HbH approach) are presented. It is observed again that the analytical model has high accuracy in predicting the average state removal delay of the HbH approach and the upper bound is very close. The removal delay of both E2E and HbH approaches increases considerably with the lifetime

coefficient. Therefore, from the state removal perspective, the lifetime coefficient should be configured as small as possible.

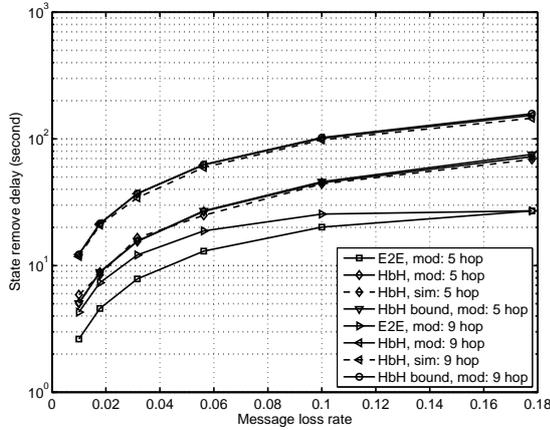


Fig. 10. Average state removal delay against message loss rate.

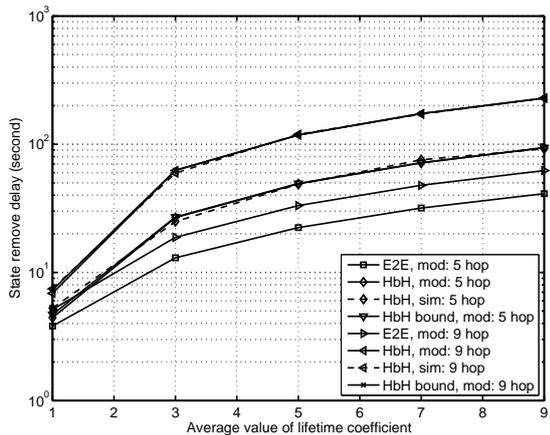


Fig. 11. Average state removal delay against state lifetime coefficient.

VII. CONCLUSION

In this article, we proposed analytical models to study the end-to-end (E2E) and hop-by-hop (HbH) refresh approaches for multi-hop soft state signaling protocols. We obtain closed-form expressions for both state refresh and state removal performances of the E2E and HbH approaches, in terms of state inconsistency ratio, message rate and state removal delay. The analytical models are simple and have been validated by simulations. They are general and can be applied to a wide range of existing soft-state signaling protocols. It is shown that the HbH approach can achieve a much lower state inconsistency performance than the E2E approach with relatively higher message rate in the multi-hop signaling system. However, the state removal delay of the HbH approach is much larger than that of the E2E. The impact of the refresh timer and the state lifetime on application requirements (e.g., state consistency) and network resources (e.g., message overheads due to refresh message and delay in removing stale states)

were investigated. The refresh timer and state lifetime need to be carefully configured to make a tradeoff of application requirement and resource overheads.

Acknowledgement

The authors would like to thank Hsiao-Hua Chen and anonymous reviewers for their insightful comments.

REFERENCES

- [1] F. Baker, C. Iturralde, F. L. Faucheur, and B. Davie, "Aggregation of RSVP for IPv4 and IPv6 Reservations," Internet Engineering Task Force, RFC 3175, Sept. 2001.
- [2] L. Berger, D. Gan, G. Swallow, P. Pan, F. Tommasi, and S. Molendini, "RSVP Refresh Overhead Reduction Extensions," Internet Engineering Task Force, RFC 2961, Apr. 2001.
- [3] R. Braden, D. Estrin, S. Berson, S. Herzog, and D. Zappala, "The design of the RSVP protocol," ISI Tech. Rep., July 1996.
- [4] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification," Internet Engineering Task Force, RFC 2205, Sept. 1997.
- [5] D. Clark, "The Design Philosophy of the DARPA Internet Protocols," in *Proc of ACM SIGCOMM*, 1988.
- [6] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. Wei, "Protocol Independent Multicast-Sparse Mode (PIM-SRP): Protocol Specification," Internet Engineering Task Force, RFC 2362, June 1998.
- [7] G. Feher, K. Nemeth, M. Maliosz, I. Cselenyi, J. Bergkvist, D. Ahlrad, and T. Engborg, "Boomerang – A Simple Protocol for Resource Reservation in IP Networks," in *Proc of IEEE RTAS*, 1999.
- [8] X. Fu, H. Schulzrinne, A. Bader, D. Hogrefe, C. Kappler, G. Karagiannis, H. Tschofenig, and S. Van den Bosch, "NSIS: A New Extensible IP Signaling Protocol Suite," *IEEE Communications Magazine*, vol. 43, no. 10, pp. 131–141, Oct. 2005.
- [9] R. Hancock, G. Karagiannis, J. Loughney, and S. Van den Bosch, "Next Steps in Signaling (NSIS): Framework," Internet Engineering Task Force, RFC 4080, June 2005.
- [10] P. Ji, Z. Ge, J. Kurose, and D. Towsley, "A comparison of hard-state and soft-state signaling protocols," in *Proc of ACM SIGCOMM*, 2003.
- [11] P. Ji, Z. Ge, J. Kurose, and D. Towsley, "A comparison of hard-state and soft-state signaling protocols," *IEEE/ACM Transaction on Networking*, vol. 15, no. 2, pp. 281–294, Apr. 2007.
- [12] J. Lui, V. Misra, and D. Rubenstein, "On the Robustness of Soft State Protocols," *Proc of IEEE ICNP*, 2004.
- [13] J. Manner, G. Karagiannis, and A. McDonald, "NSLP for Quality-of-Service Signaling," Internet Engineering Task Force, Internet-Draft draft-ietf-nsis-qos-nslp-16, Feb. 2008, work in progress.
- [14] P. Pan, E. Hahne, and H. Schulzrinne, "BGRP: A tree-based aggregation protocol for inter-domain reservations," *Journal of Communications and Networks*, vol. 2, no. 2, pp. 157–167, June 2000.
- [15] P. Pan and H. Schulzrinne, "Staged Refresh Timers For RSVP," in *Proc of Global Internet*, 1997.
- [16] P. Pan and H. Schulzrinne, "YESSIR: A Simple Reservation Mechanism for the Internet," in *Proc of NOSSDAV*, 1998.
- [17] J. He, X. Fu, Z. Tang, and H. Chen, "End-to-end versus hop-by-hop state refresh in soft state signaling protocols," *IEEE Communications Letters*, vol. 13, no. 4, pp. 268–270, Apr. 2009.
- [18] S. Raman and S. McCanne, "A model, analysis, and protocol framework for soft state-based communication," in *Proc of ACM SIGCOMM*, 1999.
- [19] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," Internet Engineering Task Force, RFC 3550, July 2003.
- [20] H. Schulzrinne and R. Hancock, "GIST: General Internet Signaling Transport," Internet Engineering Task Force, Internet-Draft draft-ietf-nsis-ntlp-20, June 2009, work in progress.
- [21] P. Sharma, D. Estrin, S. Floyd, and V. Jacobson, "Scalable timers for soft state protocols," in *Proc of IEEE INFOCOM*, 1997.
- [22] L. Wang, A. Terzis, and L. Zhang, "A new proposal for RSVP refreshes," in *Proc of IEEE ICNP*, 1999.
- [23] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "RSVP: A New Resource ReSerVation Protocol," *IEEE Network*, vol. 7, no. 5, pp. 8–18, Sept. 1993.