Optimal Accounting Policies for AAA Systems in Mobile Telecommunications Networks

Said Zaghloul, *Student Member*, *IEEE*, and Admela Jukan, *Senior Member*, *IEEE*

Abstract—Authentication, Authorization, and Accounting (AAA) deployments are expected to grow significantly in emerging mobile systems as they offer a plethora of services and mobile applications. In current systems, network access servers (NAS) periodically report the service usage of mobile users located within their coverage areas. The periodic reports are used by the billing systems to minimize the incurred capital losses if the serving NAS fails. While shorter reporting intervals are desired for lower losses, they can potentially result in undesirably high signaling load. Because it is prohibitively difficult to obtain optimal reporting intervals in mobile systems due to multitudes of services with different mobility profiles, current accounting standards offer no quantitative measures for selecting a proper reporting interval and AAA systems are typically designed via over provisioning. To address this issue, we propose an adaptive optimization mechanism in multiservice AAA systems which limits the potential loss without excessively generating unnecessary usage reports. Our optimization mechanism embraces the current AAA IETF standards RADIUS and its successor Diameter and does not require any modifications to the AAA protocols nor to the network access servers' implementation, and its implementation scope is limited to the AAA systems. The results demonstrate that our mechanism is robust under various operational conditions, easy to implement, and offers considerable potential for loss control compared to the current static approaches.

Index Terms—AAA, accounting, RADIUS, Diameter, accounting interim interval.

1 Introduction

The success of next generation IP-based mobile systems in terms of the operator's revenue growth largely depends on the abilities to implement smart charging and accounting strategies for the supported Quality of Service (QoS). Toward this goal, the next generation wireless mobile systems are adopting the Authentication, Authorization, and Accounting (AAA) systems and their dedicated protocols RADIUS and Diameter [1], [2], [3], [4], [5], [6], [7]. In every AAA-based system, an IP gateway element, referred to as the Network Access Server (NAS), meters the service usage and reports it to the AAA system via a sequence of messages; an accounting *start* record is issued when a service session starts and an accounting *stop* is issued when the session terminates. During the service, the accounting *interim* records are issued periodically as a way of protection against server or network failures or loss of accounting stop messages where unreported usage can lead to a significant loss of revenue [2], [8], [9]. For instance, for a typical size equipment [10], the failure of a network access server (NAS) serving 24,000 active users from 800 base stations with average session duration of 10 minutes and a charge of 10 cents a minute, results in a loss of 12,000 USD when the reporting interval equals 10 minutes. A reduction of the potential loss by half via reducing the reporting intervals, would result in requirements to handle about 30 percent more signaling load; a further loss reduction to 1,000 USD would require the signaling server capacity to go up to 314 percent. Clearly, there is a trade-off between the potential loss and the signaling load; the shorter the reporting interval the smaller the potential loss, but also the larger the signaling load, and hence, the required size of the AAA system [8]. As the current AAA standards [1], [2] leave the determination of the reporting periods open to the operators, the question arises of how to minimize the potential losses while avoiding excessive server overprovisioning, especially as the number of mobile services is expected to grow and energy and data center sizes are becoming a concern [11].

Finding an optimal trade-off between the potential loss and the signaling load is particularly complex in mobile and multiservice network systems, as the multiservice and mobile scenario results in a multicommodity trade-off due to the potential loss from each service, its session statistics which vary with mobility, and the corresponding signaling load from all services. The impact of mobility is nontrivial. For mobile services, the optimality for the reporting periods can only be achieved by adapting the reporting intervals to the expected service session arrival rates, service durations, and their costs. These expected values vary and often do not exhibit long term stationarity. For some mobile users, only a portion of the session is observed by the serving NAS. Depending on the users' concentration in the border areas of the cellular coverage area under consideration, the service sessions arrival rates and their effective service time within the NAS area may also largely fluctuate. Hence, even though operators can choose to determine the reporting intervals empirically and based on past observation, future services can be better served by a formal characterization of the accounting intervals which can optimally relate signaling load to the potential loss. This is especially true for the emerging IP

• *The authors are with the Institute of Computer and Network Engineering, Technische Universitaet Carolo-Wilhelmina zu Braunschweig, Hans-Sommer-Str. 66, D-38106, Brunswick, Germany.*
E-mail: {zaghloul, jukan}@ida.ing.tu-bs.de.

Manuscript received 23 Feb. 2009; revised 10 Oct. 2009; accepted 25 Oct. 2009; published online 11 Jan. 2010.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-2009-02-0069.
Digital Object Identifier no. 10.1109/TMC.2010.19.

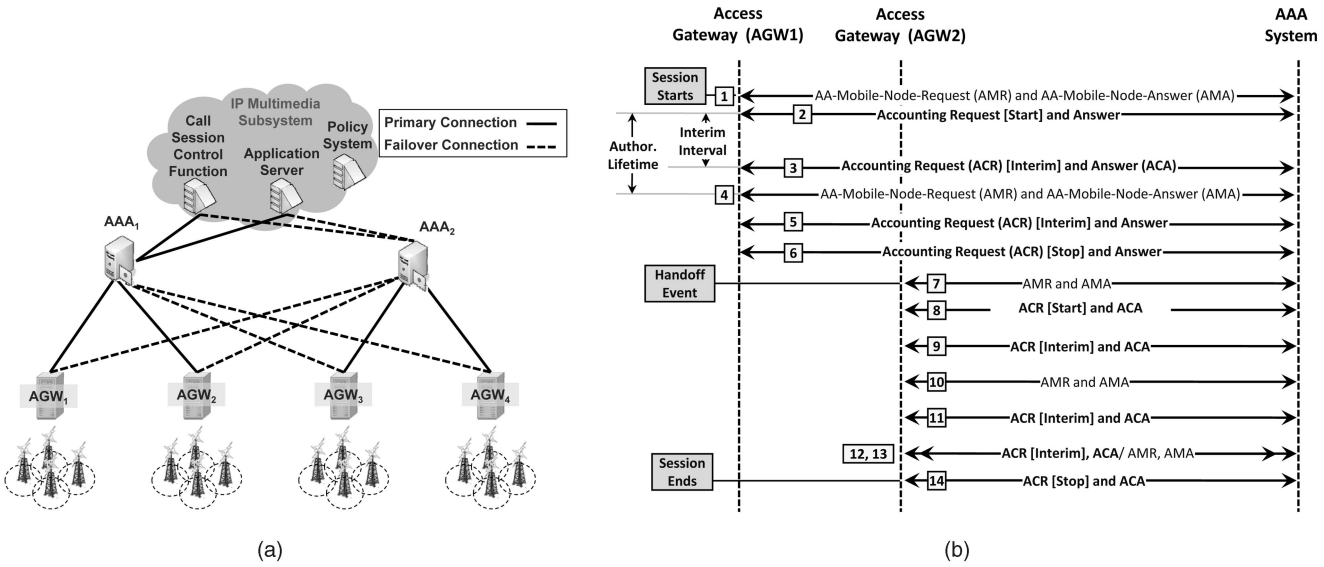

Fig. 1. (a) AAA system architecture and signaling flows. (b) The *Reauthentications* (AMR) are omitted for clarity. (Acronyms: AGW: Access Gateway, AAA: Authentication, Authorization, and Accounting). Diameter signaling traffic model.

Multimedia Subsystem (IMS) [12], which is a standardized multiservice framework for network convergence where multiple innovative and third-party services are created to be economically viable with shorter launching times.

To address these issues, in this paper we propose the first formal framework that quantifies the trade-off between the potential loss and the signaling load in multiservice mobile networks. We furthermore propose two optimization policies, which can adaptively and optimally trade off the potential loss and the AAA signaling load. In our framework, we utilize stochastic and renewal theoretic concepts to obtain simple estimates of the signaling load and the potential loss to be used by the optimization policies. To account for statistical variability due to mobility, our method uses standard protocol attributes [3], [4] to categorize mobile service sessions into four distinct types relevant to their initiation and termination locations. The statistics of the four components are then used to estimate the load and loss by extending concepts of holding time, based on our past work [13], [14], [15]. Our optimization mechanism embraces the current AAA IETF standards, RADIUS, and its successor Diameter [1], [2] and does not require any modifications to the AAA protocols nor to the network access servers, and its implementation scope is limited to the AAA systems. As such the method is easy to implement and scalable with the number of services. We show with numerical results that our adaptive mechanism is lightweight as it was observed that the optimization converges rapidly and shows fast execution time even on a low-end desktop machine.

We emphasize that the AAA accounting process analyzed here is agnostic to whether accounting systems are post- or prepaid. However, this work considers postpaid systems only, due to the following three aspects in prepaid systems. First, a credit control server (a.k.a. prepaid server) is used to interact with the Business Support System (BSS) to ensure that the user has sufficient quota for the consumed service. Hence, credit control signaling needs to be considered. Second, the granularity of the credit resources in prepaid systems is typically finer than in postpaid systems (e.g., every 2 minutes versus 10 minutes), which implies lower potential loss than in postpaid systems. Finally, the signaling load estimate in prepaid systems requires a rather small, but important modification of the performance analysis to account for the case that users can be dropped due to insufficient credits. For these reasons, we leave the discussion related to the prepaid systems for future studies.

This paper is organized as follows: Section 2 presents the necessary background on AAA protocols and past research relevant to our mechanism. In Section 3, we describe the details of our mechanism including signaling load and loss estimation. In Section 4, we formulate the optimization policies. In Section 5, we validate our mechanism and show relevant numerical results. In Section 6, we conclude the paper and give directions for future work.

2 AAA Background

Currently, the Remote Authentication Dial In User Service (RADIUS) and its successor Diameter [1], [2] are the commonly adopted AAA protocols in the wireless network standards and in deployments. Since both protocols incorporate the same AAA signaling message types and procedures, we use the message names from the newer protocol, Diameter [2], [16]. Fig. 1a shows a simplified all-IP wireless network architecture which consists of four access gateways serving four cellular regions. In this regard, an access gateway (AGW) is a generic term that refers to the first IP network element which interacts with the terminal and usually implements the network access server (NAS) functionality. Examples of AGWs are Access Serving Node Gateway (ASN-GW) in WiMAX, Packet Data Serving Node (PDSN) in 3GPP2 networks, or Serving Gateways in 3GPP R6+ systems. The four AGWs connect to two AAA systems in a redundant pair configuration.

AGWs identify service flows using charging rules supplied by policy systems residing in IMS [12] and

metered accordingly. Other IMS components such as call session control functions (CSCFs) or application servers may report accounting information to the AAA system. Hence, we use the NAS as a general term to refer to the AGW and CSCFs. When IP service flows are identified and metered by the AGW, the accounting process is usually referred to as flow-based accounting. In this paper, we use the term *service session* to refer to the duration of the chargeable service flows rather than the mere connectivity time at the IP level.

Fig. 1b illustrates the corresponding signaling flow. When a user establishes a data session (step 1), Diameter authentication exchanges (i.e., AA-Mobile-Node-Request (AMR)) are conducted with the AAA server to authenticate and/or authorize the incoming session.[1] The authentication response (i.e., AA-Mobile-Node-Answer) returns the user's profile and any necessary network settings to the requesting AGW. Two important settings are the so-called *accounting interim interval* and *the authorization lifetime*. Whereas the interim interval determines the reporting frequency of the usage, the authorization lifetime is used to indicate the time by which the mobile node must reauthenticate. Upon successful authentication, an Accounting Request, ACR type Start message, is sent by the AGW to the AAA server (step 2). The AAA acknowledges the receipt of the ACR message by sending an accounting answer message (ACA). Note that accounting messages contain Attribute Value Pairs (AVPs) which usually convey session state information such as the user's identity, IP address, byte usage, usage time, and various other parameters. The accounting ACR Start message is typically followed by periodic ACR type Interim message reporting the latest subscriber's usage every accounting interim interval throughout the user's session (steps 5). When an AGW handoff occurs (i.e., when the user moves from area 1 to area 2), the accounting session at the source AGW is terminated with an ACR type Stop message (step 6). Simultaneously, a new accounting session is created at the target AGW after optionally authenticating the user (i.e., by sending an AMR message) (step 7). Similar steps to (1-6) take place at the new AGW. Once the session is terminated (step 14), an ACR type (Stop) message is sent reporting the final subscriber's usage to the AAA system.

It is noteworthy to state that currently for postpaid schemes RADIUS and Diameter only support time-based interim reporting. The proposals in [17], [18] have suggested the triggering for interim records based on consumed data volumes for data-based services, for instance after 500 KB of data are consumed by a terminal. When volume-based interim reporting is possible, our method can be directly applied by merely using volume rather than time units as the distribution of the packet volumes that transmitted in a service session can be mapped to a specific service session holding time distribution [19]. For the rest of the article, we will focus on time-based metering. For more sophisticated charging plans where, for instance, users from two providers are able to negotiate their payment shares of the session [20], our mechanism can handle such cases by using service costs for the loss estimates that reflect the negotiated payment proportions between the users. Furthermore, our mechanism can be combined with specialized pricing tools such as the HP DNA tool in [17] which helps to analyze pricing plans and service prices. This allows automatically configuring the proposed mechanism with the calculated service prices on the fly, and hence, maximizes the return on investment of service providers and enhances the customers experience. With respect to accounting, some efforts addressed service metering configuration and management [21], enhancements to accounting schemes in high mobility networks [22], and challenges for fraud detection as in [23].

1. Without loss of generality, we adopt the CHAP based authentication mechanism used in 3GPP2 systems.

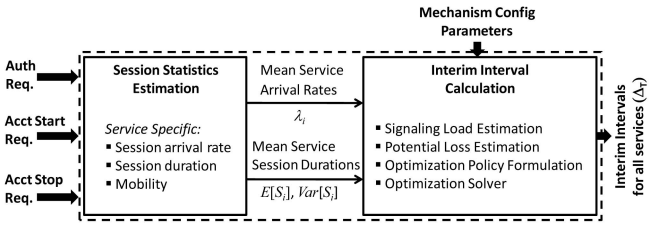

Fig. 2. Optimization logic.

3 The Optimization Mechanism

Fig. 2 shows a high level diagram of the proposed optimization mechanism. Our scheme can be viewed as an AAA module which receives the authentication, accounting start, and accounting stop requests and use them to update the accounting interim intervals from all services that will be used by currently arriving and future service sessions. Our mechanism consists of two major blocks: one responsible for estimating service load and session duration statistics and another that uses such estimates to resolve the trade-off between the load and the potential loss to produce optimal interim intervals for all services based on the current state of the system. The mechanism can be completely implemented in software modules (e.g., [24]) in the AAA servers, or split into components where statistics collection is implemented on the AAA system while the rest of the optimization logic is implemented in a separate server. We emphasize that our scheme is not an overload handling mechanism but rather targets resolving the trade-off between the loss and the load, and leaves the overload handling mechanism intact. Since according to the RADIUS and Diameter standards [1], [2], it is generally not possible to change the interim intervals for the admitted sessions, the optimized interim settings only affect future sessions.

In a nutshell, the statistics estimation block tracks the current service session arrival rates, duration, and mobility statistics from all NASes. When a sufficient change in the service session arrival rate or duration statistics or a change in the system's parameters is detected, interim recalculation is invoked. In this regard, the estimates of the potential loss and the signaling load are updated based on the estimated statistics, which are then used along with configuration parameters by the optimization policies. The optimization policies are then solved by the optimization solver and the interim intervals are updated based on the latest state of the

system. The typical triggers of a new statistic estimation can be tariff switching, NAS failover, NAS addition or removal, but triggers can also be periodic, e.g., for administrative reasons. The interim interval calculation also considers the configuration parameters. For our mechanism, each service's configuration includes the administrative range for the interim intervals [8] denoted in vector form as $\mathbf{\Delta}_{\mathbf{T}}^{min}$ and $\mathbf{\Delta}_{\mathbf{T}}^{max}$ and service costs. The configuration parameters also include the capacity of the AAA system P, and whether optimization is allowed. The last parameter is useful in cases where the interim interval for some services is required to be fixed such as in some roaming agreements. In the following sections, we provide details on each functional block shown in Fig. 2.

3.1 The Session Statistics Estimation Block

The major functions of the statistics collection block is to keep track of the services session arrival rate and duration statistics (e.g., mean and variance) including mobility effects, and then trigger an interim interval recalculation when the service statistics change by an amount greater than a preset threshold.

3.1.1 The Service-Specific Session Statistics

In our system, each service is identified by unique properties such as NAS IP address, service type (e.g., VoIP, video, gaming, etc.), cost, etc. We use moving average windows to maintain the most recent statistics for the arrival rate and the session duration of each service served by the AAA system. The moving windows are used to adapt to changes in service statistics during the day. The collected statistics for each service include the access request rate, the rejected authentications rates (e.g., misconfigured devices), and session durations. In practice, this is possible as many of the available AAA solutions today already implement traffic counting abilities and offer them for network operations and management systems [25], [26]. The mean session arrival rate is estimated by the interarrival time between accounting-start requests and the session duration is directly read from the `Session-Time` attribute in the accounting stop messages [1], [2]. To account for mobility effects, other attributes are used as we describe in the next section. The estimated mean arrival and session durations for each service are used to trigger a recalculation of the interim interval when a change in the mean arrival rate or session durations exceeds a preset threshold (e.g., 5 percent since the last interim optimization). To ensure resilience against transient perturbations in service statistics, we also wait for a minimum grace period to pass since the last optimization operation.

3.1.2 Impact of Mobility on Session Statistics

When users move between NAS regions, the accounting sessions are closed on the source NAS (i.e., access gateway) and new accounting sessions are started at the target NAS. Consequently, this has an impact on the session statistics observed at the AAA system from a particular NAS. To capture this important aspect, let us define the service session duration as the sequence of all durations a session spends in a given NAS region before it terminates or moves to another NAS area. We refer to the duration spent in each

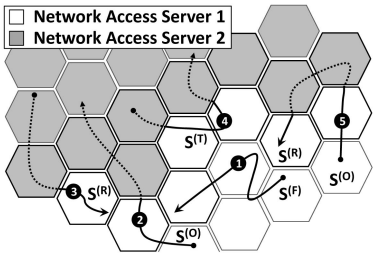

Fig. 3. Session holding times (solid lines) from the perspective of NAS 1 for various mobility patterns (session types: (1) full, (2) originating, (3) terminating, (4) transit, (5) mixed (originating and terminating)).

NAS region as the session holding time akin to the channel holding time in cellular call performance theory [27]. This definition leads to four basic service session holding times categories, as illustrated in Fig. 3, i.e.,

1. *Full Sessions*, $S^{(F)}$: Sessions that originate and terminate in the NAS area under consideration.
2. *Originated Sessions*, $S^{(O)}$: Sessions that originate in the NAS area under consideration and last long enough to handoff to other NAS serving areas.
3. *Terminating Sessions*, $S^{(R)}$: Sessions that originate in another NAS area and terminate in the NAS area under consideration.
4. *Transit Sessions*, $S^{(T)}$: Sessions that pass through the NAS area under consideration (i.e., start and terminate in other NAS areas).

Notice that for NAS 1 in Fig. 3, mixed mobility cases such as case 5 can be decomposed into cases 2 and 3, and hence, do not need to be addressed separately. In fact, this characterization is sufficiently general to handle both fixed and mobile systems. Thus, for each service i, the output of the statistics estimation block is given as four components for the session arrival rates, $\lambda_i^{(x)}$, the four components for the session holding time, $S_i^{(x)}$, where $x \in \{F, O, R, T\}$, and service authentication success rates. For instance, depending on the size of the NAS area and its surrounding NASes, different behaviors can be observed. For instance, for networks with large NAS areas, handoffs are unlikely, and hence, $\lambda_i^{(F)}$ is high, and hence, the relative proportions of $S_i^{(F)}$ dominate. If the NAS under consideration was large and surrounded by small NAS areas then $\lambda_i^{(F)}$ and $\lambda_i^{(R)}$ will be large, and hence, $S_i^{(F)}$ and $S_i^{(R)}$ will dominate. Similar arguments can be made when significant user concentrations are located in its border cells of the NAS coverage area.

The protocol attributes necessary to obtain the four session holding times based on RADIUS and Diameter [1], [2] are shown in Table 1. The attributes in the table are used in several wireless systems such as WiMAX and 1xEVDO systems [3], [4]. The `Beginning-of-Session` attribute is used to mark the first accounting period in a session and appears only in accounting start messages. The `Session-Continue` appears only in accounting stop messages and is used to indicate whether there are any subsequent accounting periods. The session holding times can be read directly from the accounting stop records from the standard `Acct-Session-Time` [1] attribute

TABLE 1
Session Types Categorization Using RADIUS/Diameter AVPs

Session Type	BOS AVP [Acct. Start]	SC AVP [Acct. Stop]
Full, $S^{(F)}$	true	false
Originated, $S^{(O)}$	true	true
Terminating, $S^{(R)}$	false or N/A	true
Transit, $S^{(T)}$	false or N/A	false

Acronyms, AVP: Attribute Value Pair, BOS: Beginning-Of-Session, SC: Session-Continue.

which reports the service time for the session by a particular NAS element.

3.2 The Load and Loss Estimation

In the interim interval calculation block, services are grouped into NAS sets, denoted as $\mathbb{N}$, which identify all service sessions coming from the same NAS node. This is needed for the loss estimate because failures usually impact one NAS and not all NASes simultaneously. The global service set, $\mathbb{A}$, which is used to estimate the signaling load, is the union of all NAS sets and is given as $\mathbb{A} = \mathbb{N}_1 \cup \mathbb{N}_2 \cdots \cup \mathbb{N}_k$ where $\mathbb{N}_k$ is the kth NAS set.

3.2.1 Estimating the AAA Signaling Load

Let us assume the generic case that the AAA signaling traffic consists of both authentication and accounting messages, otherwise the authentication terms are simply ignored. For clarity, we first explain the estimation of the signaling load in the absence of mobility and then show how to incorporate mobility effects. Since without mobility, the session holding time and the session duration are synonymous as only full session categories are observed, we drop the (x) superscript from $\lambda_i^{(x)}$ and $S_i^{(x)}$. Let us denote the mean AAA signaling rate as ζ. Let ζ_A, ζ_R, ζ_{Start}, ζ_{Int}, and ζ_{Stop} denote the mean authentication, reauthentications, accounting start, interim, and stop rates, respectively. Let p_a denote the estimated AAA authentication success rate probability (i.e., the estimated proportion of the accepted access requests). Let us also assume that the service session arrival process is Poissonian. The resulting signaling rate is then the sum of all the rates from all services including authentications, accounting starts, interims, and stops and is given as,

$$\zeta = \sum_{i \in \mathbb{A}} [\zeta_{A,i} + (\zeta_{R,i} + \zeta_{Start,i} + \zeta_{Int,i} + \zeta_{Stop,i}) p_{a,i}]. \quad (1)$$

In (1), we make the assumption that reauthentications are always successful for already authenticated users. This is a practical assumption for operational networks. Following a similar approach as in [13], [14], the authentication rate[2] for service i denoted as λ_i is related to the rates of accounting start and stop messages as

$$\zeta_{A,i} = p_{a,i}^{-1} \zeta_{Start,i} = p_{a,i}^{-1} \zeta_{Stop,i} = \lambda_i. \quad (2)$$

2. For brevity, we assume that authentications consist of one exchange, as in 3GPP2 systems. Otherwise the rates can be multiplied by a constant reflecting the number of messages and processing costs.

The mean interims rate is the product of the number of interim messages during each service session and the session arrival rate. Let the session time duration follow a generic distribution $F_S(s)$ with a mean of E_s and a coefficient of variation of

$$c_s = \frac{\sqrt{Var[S]}}{E_s}.$$

Let us denote the interim interval as Δ_T and the authorization lifetime as Δ_M. Then, for service i, the number of interims can be obtained by taking the expectation of the floor of the ratio of the duration of the service session and the interim interval (i.e., the reporting interval of metering information) Δ_{T_i} as $E[\lfloor \frac{S_i}{\Delta_{T_i}} \rfloor]$. It can be shown (see Appendix A) that the interim rate (i.e., the rate of ACR interim messages) from all services is

$$\zeta_{Int} = \sum_{i \in \mathbb{A}} \lambda_i E\left[\left\lfloor \frac{S_i}{\Delta_{T_i}} \right\rfloor\right] = \sum_{i \in \mathbb{A}} \lambda_i \sum_{j=1}^{\infty} \bar{F}_{S_i}(j\Delta_{T_i}). \quad (3)$$

The mean number of reauthentications can be evaluated similarly to the mean number of interims ζ_R by substituting Δ_{M_i} instead of Δ_{T_i} in (3). Substituting (2)-(3) in (1), the mean signaling rate is given as,

$$\zeta = \sum_{i \in \mathbb{A}} \lambda_i \left[1 + p_{a_i}\left(2 + \sum_{j=1}^{\infty} \bar{F}_{S_i}(j\Delta_{T_i}) + \sum_{j=1}^{\infty} \bar{F}_{S_i}(j\Delta_{M_i})\right)\right]. \quad (4)$$

To get an insight to the general formula in (4), let us consider an exemplary case of a single service with an exponentially distributed session duration. It directly follows that (4) simplifies to,

$$\zeta = \lambda \left[1 + p_a\left(2 + \frac{1}{e^{\frac{\Delta_T}{E_s}} - 1} + \frac{1}{e^{\frac{\Delta_M}{E_s}} - 1}\right)\right] \quad (5)$$

which matches the result in [13], [14]. From (5), it is clear that there is a nonlinear relationship between the interim setting and the mean signaling load. Notice that when $\Delta_T > E_s$, the signaling load barely changes. This is because the mean number of interims per session falls significantly below one (i.e., $\frac{1}{e-1} = 0.58$ interim/session). It should be noted that (5) is convex because it is the sum of convex functions (i.e., $\bar{F}_S(s)$) and has a diagonal hessian matrix with positive elements.

Let us now extend our results to incorporate mobility. In this case, the total signaling rate due to each service is the weighted sum of the signaling load due to its four mobility components denoted as $\zeta_i^{(x)}$ and is given as,

$$\zeta = \sum_{i \in \mathbb{A}} \sum_{x \in \{F,O,R,T\}} \zeta_i^{(x)}, \quad (6)$$

where $\zeta_i^{(x)}$ is obtained using (4). To obtain an estimate for $\zeta_i^{(x)}$ to use in our mechanism and without loss of generality, we assume that the four components of the session holding time, $S_i^{(x)}$, follow the LogNormal distribution as it is widely observed in measurement studies for VoIP and data sessions [28], [29], [30], [31]. Since the complementary distribution for the LogNormal is

$$\bar{F}_S(s) = \frac{1}{2}\ \mathrm{erfc}\left(\frac{\ln(ks) - \mu_i^{(x)}}{\sqrt{2\left(\sigma_i^{(x)}\right)^2}}\right),$$

then, using (4), it follows that

$$\zeta_i^{(x)} = \lambda_i^{(x)}\left[1 + p_{a_i}^{(x)}\left(2 + \frac{1}{2}\sum_{k=1}^{\infty}\mathrm{erfc}\left(\frac{\ln(k\Delta_{T_i}) - \mu_i^{(x)}}{\sqrt{2\left(\sigma_i^{(x)}\right)^2}}\right) + \frac{1}{2}\sum_{k=1}^{\infty}\mathrm{erfc}\left(\frac{\ln(k\Delta_M) - \mu_i^{(x)}}{\sqrt{2\left(\sigma_i^{(x)}\right)^2}}\right)\right)\right], \quad (7)$$

where the parameters $\mu_i^{(x)}$ and $\sigma_i^{(x)}$ are given in terms of the mean session holding time and its coefficient of variation as

$$\mu_i^{(x)} = \ln\left(E_{s_i}^{(x)}\right) - \frac{\left(\sigma_i^{(x)}\right)^2}{2}, \quad \left(\sigma_i^{(x)}\right)^2 = \ln\left(\left(\mathrm{c}_{s_i}^{(x)}\right)^2 + 1\right).$$

3.3 The Potential Loss

The potential loss L is defined as the unreported usage from impacted services when their serving NAS fails. The potential loss due to a given service i is given as the service consumption since the last interim report or since the service starting instant if no interims were generated yet. For clarity, we first study the potential loss in the absence of mobility and incorporate mobility afterwards. Assuming that the simultaneous failure of multiple gateways is unlikely, the loss due to the failure of a single gateway (NAS), L_j, is the sum of the unreported usage from all services belonging to the service set, $\mathbb{N}_j$. Let us denote the cost of a unit time for service i which belongs to $\mathbb{N}_j$ as C_i and the session duration until the failure moment as $\tilde{S}_i$, where it denotes the age (a.k.a the residual) lifetime of the session duration with a distribution of $\frac{\bar{F}_S(s)}{E_s}$. Using renewal theoretic concepts of renewals and random variable residuals [32], it can be shown (see Appendix B) that the potential loss due to the impacted NAS j is,

$$L_j = \sum_{i\in\mathbb{N}_j}\lambda_i E_{s_i} C_i U_i, \quad (8)$$

where U_i denotes the unreported usage of service i and is given as

$$\left[E\{\tilde{S}_i\} - \Delta_{T_i} E\left\{\left\lfloor\frac{\tilde{S}_i}{\Delta_{T_i}}\right\rfloor\right\}\right].$$

Let us now briefly discuss the physical interpretation of the potential loss in (8). Notice that the loss is given as the sum of the products of the losses from all impacted user sessions from all services belonging to the NAS service set (i.e., the $\lambda_i E_{s_i}$ term), the cost of the service per unit time C_i, and the mean unreported usage U_i. The mean unreported usage is intuitively the difference between the mean *age* of the session time at the moment of failure (i.e., $E\{\tilde{S}_i\}$) and the last interim report of the usage given by (i.e., $\Delta_{T_i} E\{\lfloor\frac{\tilde{S}_i}{\Delta_{T_i}}\rfloor\}$). For exponentially distributed sessions, similar to (5) and due to the memoryless property (i.e., $\tilde{S}_i = S_i$), the mean unreported usage for service i, U_i, is,

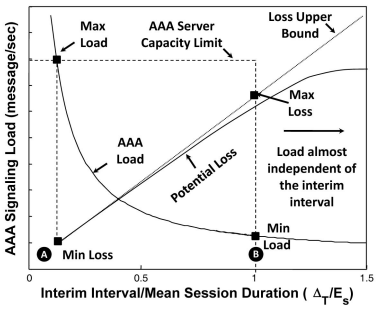

Fig. 4. Signaling load and potential loss trade-off.

$$U_i \;=\; E_{s_i} - \Delta_{T_i} E\left[\left\lfloor\frac{S_i}{\Delta_{T_i}}\right\rfloor\right] = E_{s_i} - \frac{\Delta_{T_i}}{e^{\frac{\Delta_{T_i}}{E_{s_i}}} - 1}. \quad (9)$$

Observing the limiting behavior of (9) as a function of Δ_{T_i}, we notice that as $\Delta_{T_i} \to 0$ then the unreported usage U_i approaches 0 which matches our intuition that continuous interim updates result in no loss at the event of failure. Similarly U_i approaches E_{s_i} as $\Delta_{T_i} \to \infty$. When Δ_{T_i} equals the mean session duration $\Delta_{T_i} = E_{s_i}$, then $U_i \to 0.418\Delta_{T_i} \leq 0.5\Delta_{T_i}$. Hence, in the worst case when the reporting interval equals the mean session duration, the upper bound in (8) is only an overestimate by approximately $0.418/0.5 = 16\%$. It was shown in Appendix B that U_i is upper bounded by $0.5\Delta_{T_i}$ (i.e., $\sum_{i\in\mathbb{N}_j}\lambda_i E_{s_i} C_i U_i \leq \sum_{i\in\mathbb{N}_j}\lambda_i E_{s_i} C_i \frac{\Delta_{T_i}}{2}$). Thus, in our optimization formulation, we can use the upper bound estimate of the loss which linearizes the loss as a function of the interim interval for $\Delta_{T_i} \leq E_{S_i}$ since very little control over the signaling rate is attained for higher interim values. Finally, the potential loss estimate in the presence of mobility is simply obtained by modifying (8) by summing the loss due to each mobility component as,

$$L_j = \sum_{i\in\mathbb{N}_j}\sum_{x\in\{F,O,R,T\}}\lambda_i^{(x)} E_{s_i}^{(x)} C_i \frac{\Delta_{T_i}}{2}. \quad (10)$$

3.4 The Trade-Off between the Load and the Loss

It is clear from (4), (10) that there is a trade-off between the potential loss and the signaling load ζ. This is because if the interim intervals Δ_{T_i} are decreased to reduce the potential loss in the event of the NAS failure, the corresponding signaling rate ζ increases. To illustrate this behavior, let us for simplicity assume a single service. As shown in Fig. 4, the load and the loss are given as functions of the interim interval normalized to the mean session duration. The loss is a linearly increasing function of the interim interval while the load is a nonlinear deceasing function. Notice that when the interim setting is increased beyond the mean session duration, the AAA signaling load changes very slowly. This is due to the fact that in this case, the session would most likely terminate before any interim messages are sent. On the other hand, significantly reducing the interim values may result in an excessive AAA system overloading resulting in undesired network instabilities (i.e., failovers or redirections). Hence, the desirable optimization region for the interim intervals should be selected such that they neither violate the server capacity, P, nor exceed the mean session duration. In the next section, we design policies to resolve this trade-off.

4 The Optimization Policies

In this section, we propose two optimization policies, i.e., the Constrained Loss Policy (CLP) and the Adaptive Policy with Weight Control (APWC).

4.1 Constrained Loss Policy (CLP)

The target of this policy is to maintain the loss from each NAS j below given loss limits by using the minimum signaling load. This policy is formulated as a constrained nonlinear minimization problem. The objective is to minimize the signaling load ζ from all NASes subject to two classes of linear constraints: one set limiting the range of the interim intervals for all services within their administrative limits (i.e., $\mathbf{\Delta_T^{(j)}} \in [\mathbf{\Delta_T^{(j)min}}, \mathbf{\Delta_T^{(j)max}}]$) and another limiting the potential loss from each NAS L_j to an upper bound L_j^{max}. When a new NAS is configured or detected (e.g., failover or network expansion), new loss and interim range constraints are simply added accordingly. If the minimum signaling load exceeds the AAA capacity P, either overload handling mechanisms (e.g., standard request redirection [2]) are invoked or the the maximum losses for all NASes are relaxed by a percentage, ϵ. As shown in Policy1, before performing the optimization operation, we check the preconditions in order to avoid solving infeasible problems. In this regard, we first check whether the estimate of the load ζ at the most relaxed settings (i.e., $\mathbf{\Delta_T} = \mathbf{\Delta_T^{max}}$ in (7)) is below the AAA system capacity P, otherwise $\mathbf{\Delta_T}$ is set to $\mathbf{\Delta_T^{max}}$ and standard overload handling mechanisms are triggered. We then check for each NAS j that the maximum allowed loss, L_j^{max}, can be satisfied using the corresponding minimum interim intervals (i.e., $\mathbf{\Delta_T} = \mathbf{\Delta_T^{(j)min}}$), otherwise we attempt to relax the loss constraints `MaxNumberOfInreases` times before reporting infeasibility. If all the preconditions are met, we calculate the optimal interim settings by minimizing the signaling load. If the signaling load at the optimal interim settings exceeds the capacity limit, we relax the maximum losses L_j^{max} from all NASes and try again.

Policy 1 Constrained Loss Policy (CLP)

Input: P, $\mathbf{\Delta_T^{min}}$,$\mathbf{\Delta_T^{max}}$, ϵ, `MaxNumberOfIncreases`, L_j^{max}
Output: $\mathbf{\Delta_T}$
if ζ *at* $\mathbf{\Delta_T^{max}} < P$ **then**
  repeat
    `IncreaseLmax = false`
    if *at* $\mathbf{\Delta_T^{(j)\,min}}$, $\forall_j L_j \le L_j^{max}$ **then**
      Minimize ζ subject to
        $0 < L_1 \le L_1^{max}$, $\mathbf{\Delta_T^{(1)}} \in [\mathbf{\Delta_T^{(1)\,min}}, \mathbf{\Delta_T^{(1)\,max}}]$
        $0 < L_2 \le L_2^{max}$, $\mathbf{\Delta_T^{(2)}} \in [\mathbf{\Delta_T^{(2)\,min}}, \mathbf{\Delta_T^{(2)\,max}}]$
        $\vdots$
        $0 < L_j \le L_j^{max}$, $\mathbf{\Delta_T^{(j)}} \in [\mathbf{\Delta_T^{(j)\,min}}, \mathbf{\Delta_T^{(j)\,max}}]$
      if *at optimal* $\mathbf{\Delta_T}$, $\zeta > P$ **then**
        $\forall_j L_j^{max} = (1+\epsilon) L_j^{max}$, `IncreaseLmax = true`
      end
    else
      $\forall_j L_j^{max} = (1+\epsilon) L_j^{max}$, `IncreaseLmax = true`
    end
  until *NumberOfIncreases > MaxNumberOfIncreases OR IncreaseLmax = false* ;
else
  `Trigger overload handling mechanisms`
end

4.2 The Simplified Constrained Loss Policy (SCLP)

A simplified version of the CLP policy can be formulated by solving the linear constraint equations for each NAS when the loss bound $L_{max}^{(i)}$ is binding. Clearly, SCLP is suboptimal to CLP as it does not guarantee that the solution results in minimal system load. To do so, for each NAS j, we simply start from the minimal loss at $\mathbf{\Delta_T^{(j)min}}$ and obtain $\mathbf{\Delta_T^{(j)}}$ at the NAS loss boundary in one step by moving in the gradient descent direction[3] (i.e., $-\nabla \mathbf{L_j}$) as $\mathbf{\Delta_T^{(j)}} = \mathbf{\Delta_T^{(j)min}} - \alpha \nabla \mathbf{L_j}$, where the constant α and the loss gradient $\nabla \mathbf{L_j}$ are derived in Appendix C. We then range limit $\mathbf{\Delta_T^{(j)}}$ to $\mathbf{\Delta_T^{(j)max}}$. Once optimal settings are obtained for all NASes, we check if the load ζ is below the capacity P, otherwise we relax the loss limits L_j^{max} by moving a small amount ϵ in the gradient *ascent* direction as $\mathbf{\Delta_T^{(j)}} = \mathbf{\Delta_T^{(j)min}} + \epsilon \nabla \mathbf{L_j}$ until the capacity limit is satisfied. The SCLP logic is summarized in Policy 2.

Policy 2 Simplified Constrained Loss Policy (SCLP)

Input: P, $\mathbf{\Delta_T^{min}}$,$\mathbf{\Delta_T^{max}}$, ϵ, `MaxNumberOfIncreases`, L_j^{max}
Output: $\mathbf{\Delta_T}$
if ζ *at* $\mathbf{\Delta_T^{max}} < P$ **then**
  repeat
    `IncreaseLmax = false`
    if *at* $\mathbf{\Delta_T^{(j)\,min}}$, $\forall_j L_j \le L_j^{max}$ **then**
      for each *NAS* j **do**
        $\mathbf{\Delta_T^{(j)}} = \mathbf{\Delta_T^{(j)\,min}} - \alpha \nabla \mathbf{L_j}$, range_limit $\left(\mathbf{\Delta_T^{(j)}}\right)$
      end
      while $\zeta > P$ **do**
        $\forall \mathrm{NAS}_j, \mathbf{\Delta_T^{(j)}} = \mathbf{\Delta_T^{(j)}} + \epsilon \nabla \mathbf{L_j}$, range_limit $(\mathbf{\Delta_T})$
      end
    else
      $\forall_j L_j^{max} = (1+\epsilon) L_j^{max}$, `IncreaseLmax = true`
    end
  until *NumberOfIncreases > MaxNumberOfIncreases OR IncreaseLmax = false* ;
else
  `Trigger overload handling mechanisms`
end

4.3 Adaptive Policy with Weight Control (APWC)

The CLP method requires the setting of loss bounds for NASes, (i.e., L_j^{max}), which may not be always desirable from operations and management perspective. As an alternative, we propose the APWC policy which does not require the definition of loss bounds on NASes by attempting to optimally minimize the losses using the available capacity and without excessively using up the system's resources. The APWC policy is formulated as a nonlinear minimization problem with an objective defined as the sum of a weighted average of the loss from all NASes,[4] L_A, and a weight (or penalty) function of the signaling load $W(\zeta)$ as $L_A + W(\zeta)$. L_A is defined as $L_A = \beta_1 L_1 + \beta_2 L_2 + \cdots + \beta_j L_j$ with NASes with lower potential losses assigned lower weights. The weights are given as $\beta_j = L_j / \sum_j L_j$. Notice that β_js are constants as L_j is calculated at unity interim intervals for the weights, and hence, L_A is linear. The weight function $W(\zeta)$ can be any suitable convex function of the signaling load ζ given that it becomes very low when

3. When the gradient for service j is zero, the maximum interim setting for service j is used instead.

4. Consider the case of two NASes; one posing a potential loss of \$2,000 while the other posing a risk of losing \$20,000 in the event of failure. The arithmetic mean of \$11,000 underestimates the real loss of \$20,000 if the second NAS fails.

the system utilization ($\rho = \frac{\zeta}{P}$) is low and starts to grow after crossing a given utilization, ρ_0 (e.g., 60 percent). In this paper, we use an exponential weight function as,

$$W(\zeta) = ae^{\frac{b}{P}\zeta}, \tag{11}$$

where $a = Ke^{-b}$, $b = \frac{\ln(K)}{1-\rho_0}$, and K is a suitable constant (e.g., $K = 10L_{max}$, where L_{max} is given by L_A at $\mathbf{\Delta_T} = \mathbf{\Delta_T^{max}}$). Note that ρ_0 acts as a "knob" parameter for the policy and determines the load at which we start to consider the system utilization. Thus, when the system load is high (i.e., $\rho > \rho_0$), the system utilization is considered, otherwise the loss is minimized. The constraints include a convex constraint that the signaling load ζ does not exceed the capacity P and that the interim intervals $\mathbf{\Delta_T}$ fall in their respective administrative limits. Policy 3 summarizes the APWC logic.

Policy 3 Adaptive Policy with Weight Control (APWC)

Input: P, $\mathbf{\Delta_T^{min}}$, $\mathbf{\Delta_T^{max}}$, $W(\zeta)$
Output: $\mathbf{\Delta_T}$
if ζ *at* $\mathbf{\Delta_T^{max}} < P$ **then**
  Minimize $L_A + W(\zeta)$ such that
    $\zeta \leq P$, $\mathbf{\Delta_T} \in [\mathbf{\Delta_T^{min}}, \mathbf{\Delta_T^{max}}]$
else
  `Trigger overload handling mechanisms`
end

5 Validation and Numerical Results

In this section, we examine the operation of our optimization mechanism under wide range of operational conditions. In this regard, we study realistic scenarios of fixed and mobile networks under conditions of variable loads, tariff switching, failovers, and roaming scenarios. We then examine the execution delay and the rate of invocation of the proposed mechanism. An important objective is to show that the mechanism is lightweight and easy to implement.

5.1 Simulation Environment and System Settings

We implement the proposed mechanism in a JAVA-based event driven simulator and link it to MATLAB's Sequential Quadratic Programming method to solve constrained nonlinear optimization problems. Our simulation environment consists of several modules for multiservice session generation, network topology and user mobility, and Diameter protocol. The AAA messages are generated according to the AAA standards [1], [2] and according to the accounting model in [4] for mobile networks. Authentications are considered successful by tossing a random variable and comparing to p_a.

The service session arrivals are Poissonian, and their session durations are generated following Lognormal distributions to match experimental findings for VoIP and wireless data traffic [29], [30], [31]. For the mobile network layout, we assume without loss of generality that the analyzed network is an area composed of 3×3 NASes. NAS coverage areas are different and the movement between their areas is assumed to be random. As users move between NAS regions, they can randomly trigger any of the four possible session scenarios (see Table 1). The optimization logic is only invoked when the statistics change by 5 percent and when at least a grace period of 75 seconds since the last optimization elapses. The CLP, SCLP, and APWC optimization policies are simulated based on the session statistics using the estimates in (6) and (10) and are solved using MATLAB Optimization Toolbox. For the APWC policy, we set the knob parameter, ρ_0, in (11) to 60 percent. Finally, since both the AAA signaling load in (7) and the potential loss in (10) are proportional to the session arrival rates λ_i, all of the results are normalized and given in terms of load (i.e., authentication plus accounting divided by the AAA server capacity) as well as the normalized loss to the target potential loss. Hence, our results apply to arbitrary session loads and AAA system capacities.

5.2 Optimizer Operation

In order to assess the benefits of our adaptive scheme, we compare it with three policies with static interim interval settings, to mimic current systems, i.e., Static_Min, Static_Med, and Static_Max. The interim settings for Static_Min are set to 1 minute for all services. For Static_Med and Static_Max policies, the interim settings are fixed to half and full mean session durations, respectively. For example, for two services of 5 and 15 minutes, the corresponding interim settings are [1, 1], [2.5, 7.5], and [5, 15] for the Static_Min, Static_Med, and Static_Max policies, respectively.

5.2.1 Basic Operation

Let us start by investigating the mean potential loss and AAA system load (i.e., authentication and accounting requests) in a scenario with two services served by one NAS in a network environment with no mobility (fixed). Services 1 and 2 have mean durations of 5 minutes and 15 minutes, respectively, and have the same session arrival rates to facilitate comparison. For both services, the mean load varies during the day following a sinusoid with a period of 24 hours and a peak to average ratio of 1.4. The costs for services 1 and 2 are set to 0.1 and 0.4 price units, respectively. The tariff for service 2 is halved between 11 pm and 6 am. For illustration, let us assume that a reduction in the tariff is assumed to result in doubling the mean session duration from 15 to 30 minutes. We now observe the results obtained.

- *The session holding time (Fig. 5a).* The estimated session holding times are equal to the mean session durations for both services due to the absence of mobility. The duration doubles for service 2 in the tariff switching period. The estimate for the arrival rate (not shown) also matches our sinusoidal setting.
- *The system load (Fig. 5b).* As expected, the minimum and maximum loads are achieved by the Static_Max and Static_Min policies, respectively, and hence, the loads of all other policies fall in between. This confirms that the administrative bounds for the interim intervals are respected by our proposed policies. We also observe that for all static policies the load and loss performance clearly follow the sinusoidal session arrival rate which leaves the system load and the potential loss open to the variations in the session statistics (see Fig. 5c).
- *The potential loss (Fig. 5c).* For comparison purposes, let us normalize the potential losses from all policies to the target potential loss for the CLP and SCLP mechanisms (i.e., L_1^{max}). As expected, the Static_Min and Static_Max policies set the loss bounds and all policies result in losses that fall in between. For the Static_Med policy, we observe that while halving the

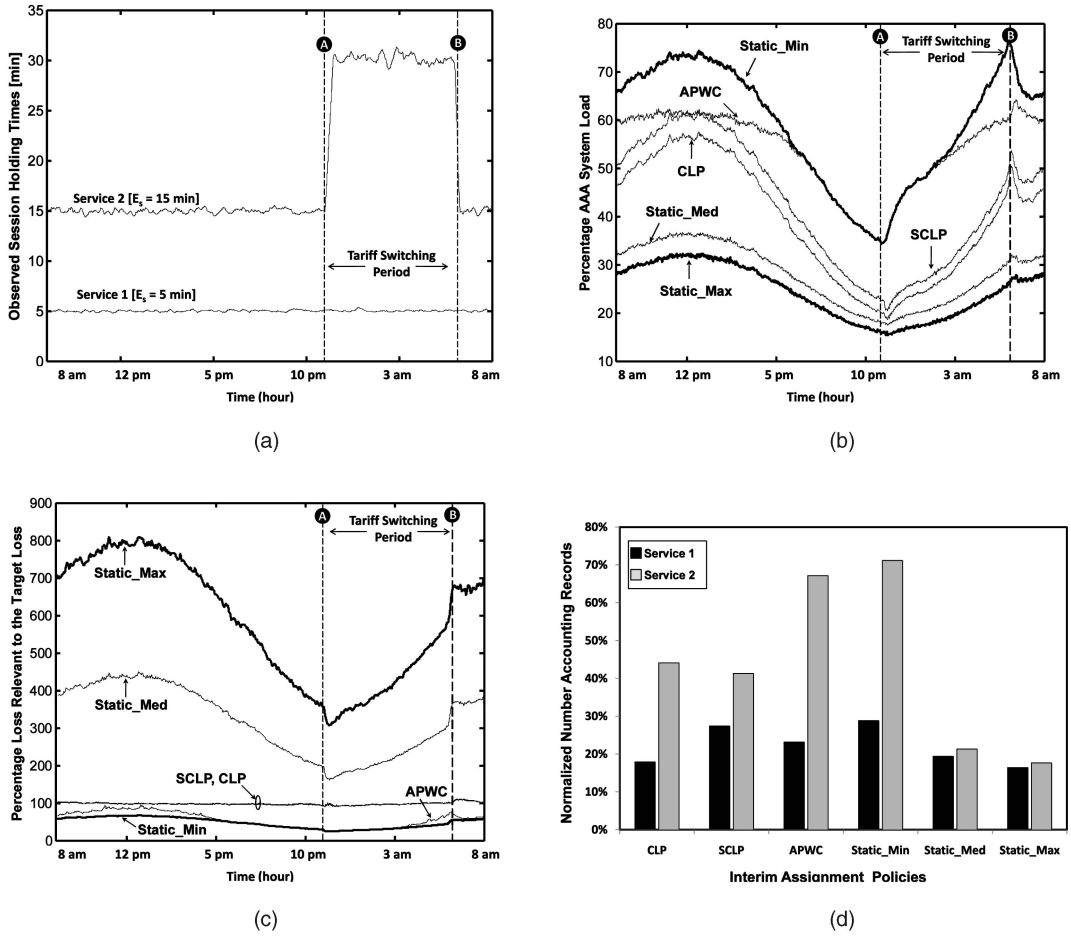

Fig. 5. System's performance in a fixed network environment, (tariff switching occurs between 7 pm and 7 am, $\lambda_i = 1 + 0.4\sin(\frac{2\pi}{24hr}t)/\mathrm{s}$, S/CLP target loss $(L_1^{max}) = 400$ units, AAA capacity $P = 40$ req/s, average window $\mathrm{sizes} = 100$, 30 independent simulation runs, 4 hour warm up period, 95 percent confidence (change within 3 percent variation)). (a) Mean session holding time as observed by the NAS. (b) AAA system load. (c) Potential loss. (d) Number of accounting records per day.

interim reporting period for both services only adds 10 percent extra system load, it potentially results in halving the potential loss. For the APWC policy, we observe that the load curves match the Static_Min policy as long as the load is below our knob setting of 60 percent. When the load exceeds this setting the loss is increased in favor of lower system load which matches our objective (observe the duration from 8 am to 5 pm in Fig. 5b and Fig. 5c). We also observe in Fig. 5c that both the SCLP and the CLP mechanisms maintain the potential loss target irrespective of the system load with minor "blips" due to tariff switching. Moreover, the load due to the CLP scheme is lower than that of the SCLP scheme which confirms the optimality of the CLP scheme.

- *The number of accounting records (Fig. 5d).* For comparison, since Static_Min generates the largest number of interim records, we use the *total* number of accounting records for both services generated by the Static_Min as reference to normalize the number of accounting records from all services generated by all policies. As shown in Fig. 5d, for the Static_Max and Static_Med policies, the number of accounting records is almost equal for both services. The slight difference is due to tariff switching which increases the accounting records for Service 2. The accounting records produced by Static_Med slightly exceed those generated by Static_Max due to the lower interim setting of the Static_Med. The accounting records produced by the Static_Min policy primarily reflect the difference in the session durations of both services irrespective of their costs or the AAA system load. The APWC produces less interim records than the Static_Min because it tries to avoid overloading the AAA system by increasing the interim intervals for both services, and hence, spreading-out the losses. We also observe the similarity of the CLP and the SCLP in terms of the produced accounting records with the CLP resulting in less total number of interim records. The suboptimality of the load performance of the SCLP is clear when observing the number of interims produced by service 1 in Fig. 5d. Common to all proposed policies (i.e., S/CLP and APWC), service 2 results in more accounting records as it contributes more to the potential loss than service 1 (i.e., $.4 > .1$ price units).

5.2.2 Impact of Mobility

Let us now investigate the benefits of our policies and stability in maintaining the expected behavior in mobile environments and under more complex scenarios with more services characterized by different session durations

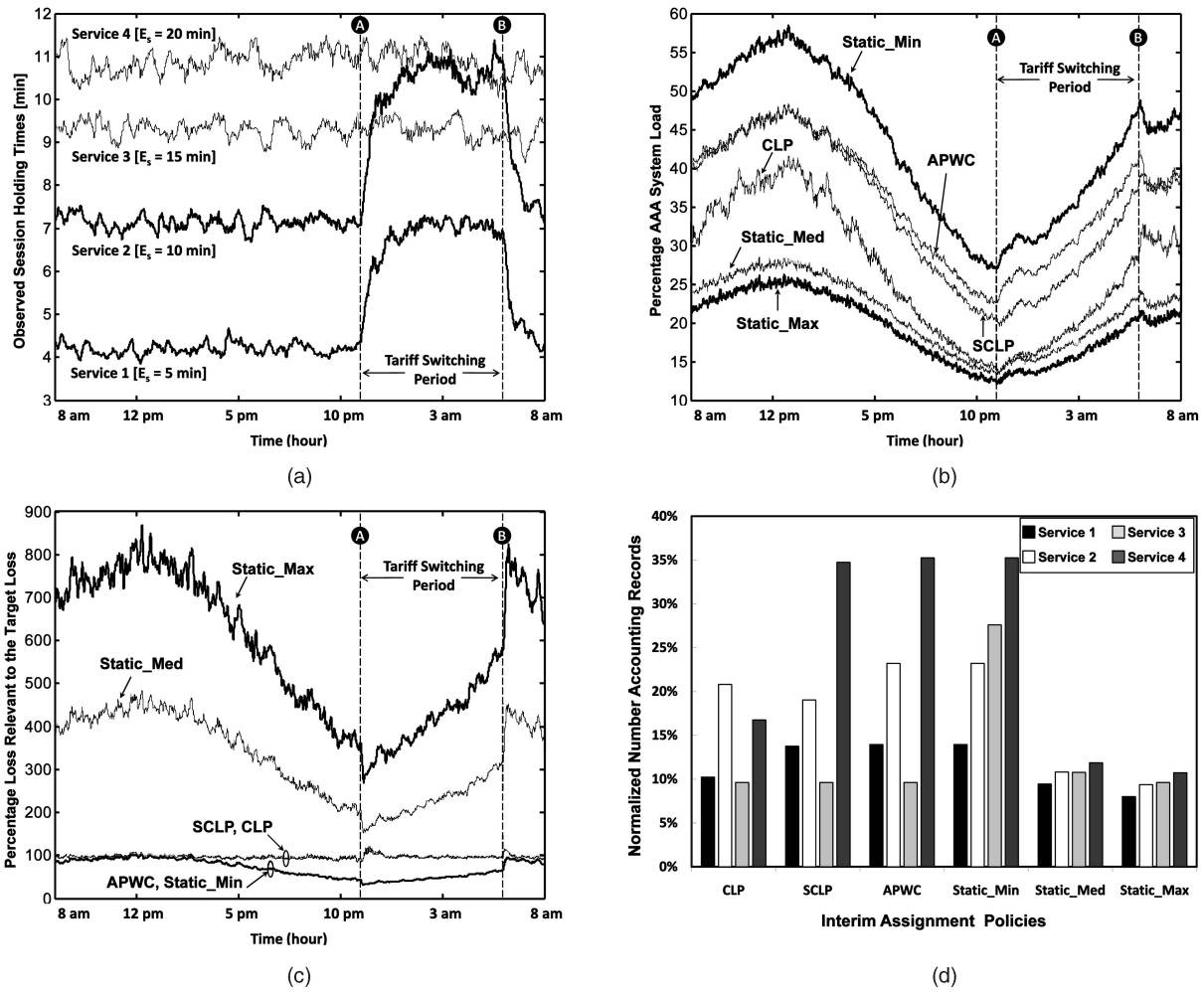

Fig. 6. System's performance in a mobile network environment, (tariff switching occurs between from 7 pm to 7 am, $\lambda_i = 1 + 0.4\sin(\frac{2\pi}{24hr}t)/\mathrm{s}$, S/CLP target loss $(L_1^{max}) = 500$ units, AAA capacity $P = 150$ req/s, average window $\mathrm{sizes} = 100$, mean AGW residence times are {10, 22, 23; 43, 25, 10; 17, 10, 11.6} minutes, 30 independent simulation runs, 4 hour warm up period, 95 percent confidence (change within 3 percent variation)). (a) Mean session holding time as observed by the NAS. (b) AAA system load. (c) Potential loss. (d) Number of accounting records per day.

and tariffs. In this regard, mobility is characterized by simulating an area served by 3×3 NASes with different sizes. Only the central NAS reports usage to the AAA system under consideration while other NASes report to other AAA systems. The central NAS has a mean residence time of 25 minutes. All NASes serve four services with mean durations of 5, 10, 15, and 20 minutes and service rates of 0.1, 1, 0, and 0.02 price units/minute. The zero cost is used to indicate that service 3 is a flat rate service and to investigate the effect of service tariffs on the behavior of our schemes. Further investigation of more complex pricing plans as in [17] is out of the scope of this work and is part of our future work. For comparison purposes we assume that the arrival rate of all services is the same. Tariff switching is applied to services 1 and 2 between 11 pm and 6 am (see Fig. 6a, instants A and B). During this time, the service costs are halved and the session durations double from 5 and 10 to 10 and 20 minutes, respectively. For all services, the mean load varies during the day following a sinusoid as in the previous case study. The AAA capacity here is larger than the fixed network case to accommodate the increased load due to the additional services and due to mobility.

- *The session holding times (Fig. 6a).* In our method, the mean session holding time is estimated as the weighted average of the mean holding time from all mobility components as

$$E_{s_i} = \frac{\sum_x \lambda_i^{(x)} E_{s_i}^{(x)}}{\sum_x \lambda_i^{(x)}},\ x \in \{F, O, R, T\}.$$

To validate the correctness of this method, we compare our estimated session holding time from the four components to the theoretical mean channel holding time from [27] which assumes preknowledge of the session duration S_i and the NAS residence time, R, (i.e., the time a mobile device spends in the NAS coverage area). In short, [27] models the mean session holding time by the minimum of the whole session duration S_i and the residence time in the NAS region, R. Hence, under exponential distribution assumptions, the mean session holding time is given as

$$\frac{E[R]E[S_i]}{E[S_i] + E[R]}.$$

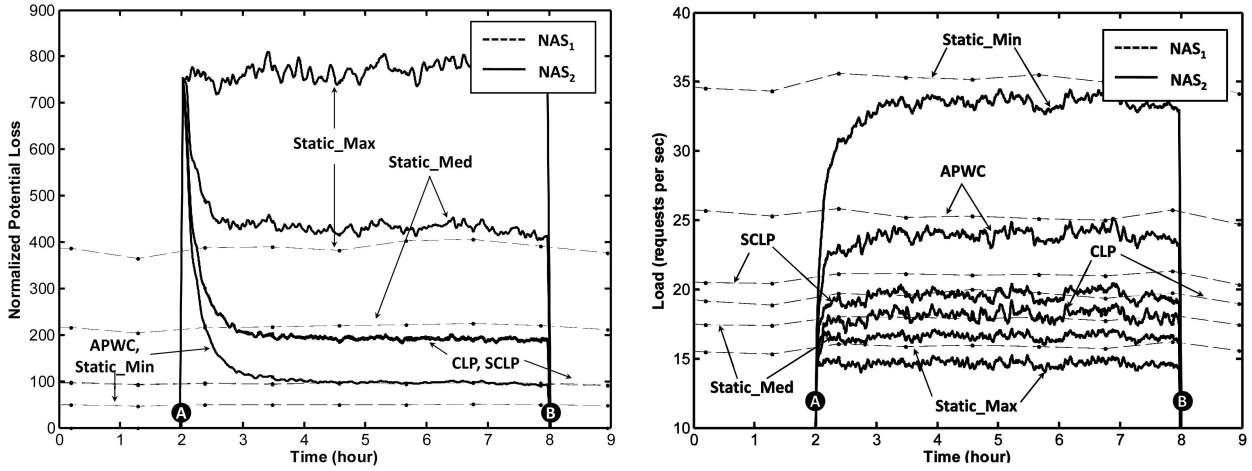

Fig. 7. Failover effect, $\lambda_i = 1/\mathrm{s}$, S/CLP target $\mathrm{loss} = 700$ and 1,400 for NAS_1 and NAS_2 units, AGW residence times are $\{40, 60\}$ minutes for NAS_1 and NAS_2 respectively, AAA capacity $P = 80$ req/s, 30 independent simulation runs, 4 hour warm up period, 95 percent confidence (change within 3 percent variation) (dashed lines are used to represent slightly fluctuating curves in (a)-(b) for clarity). (a) Normalized potential loss. (b) AAA system load from NASes 1 and 2.

Notice that we can not directly use such model because estimating R is hard in realtime, and the knowledge of S_i requires communications between all AAA systems which serve the mobile sessions. As shown in Fig. 6a, due to mobility, we observe that the estimated session holding times for all services are less than their mean values. Our holding time estimate matches the theoretical estimates in [27] (e.g., $\frac{5*25}{5+25} = 4.2$ and $\frac{20*25}{20+25} = 11.1$ for services 1 and 4, respectively). This confirms that our estimation method using four mobility components works properly.

- *The AAA system load and potential loss behavior (Fig. 6b and 6c).* In this case, we see similar trends for the static and optimization policies as in the fixed network case in Fig. 5b and 5c. This verifies the proper and consistent operation of our schemes in mobile environments. In this regard, the CLP offers more optimal load performance than the SCLP while both maintain the same loss target. The loss from the APWC mechanism is identical to the Static_Min policy while they differ in the AAA system load. This is due to the fact that Service 3 has zero cost which is considered by the APWC scheme and ignored by the Static_Min policy. Moreover, since the system load is below the knob value of 60 percent for the APWC policy, no load limiting is observed as in the fixed network case (see Fig. 5b).
- *The mean number of interims (Fig. 6d).* We also observe similar trends as in the previous case study in Fig. 5d. However, the effect of the service tariff is reflected on the mean number of interims generated by the proposed policies. Common to all of our optimization mechanisms the number of interims for service 2 is relatively large and that for service 3 is low which reflects their relative tariffs. We also see that the APWC produces the same number of interims as the Static_Min policy for all services except for service 3 due to its cost, and thus, explains the load difference between the APWC and the Static_Min policy in Fig. 6b.

From Fig. 5 and Fig. 6, we confirmed that our policies allow much better control of the potential loss relative to the static policies and are more resilient to changes in session statistics as they manage to either minimize the loss (i.e., in the APWC case) or maintain a constant loss target (i.e., in the CLP and SCLP cases) in fixed and mobile networks.

5.2.3 Impact of NAS Failovers

Let us now investigate the mechanism's behavior when another NAS fails over to the AAA system under consideration. In this regard, we study a mobile network configuration where the AAA system normally serves one NAS, which we refer to as NAS1, and a new NAS (i.e., NAS2) fails over to the AAA system under consideration after its serving AAA fails. The NAS sizes are assumed to be different with NAS2 covering a larger area. In order to clearly see the transient behavior of the policies, we study the system under constant load and we assume that the original AAA server for NAS2 stopped responding due to overload. We assume that NAS2 was always instructed to have $\mathbf{\Delta}_{\mathbf{T}}^{max}$ prior to fail over, and hence, resulting in the largest possible potential loss at the fail over event. Each NAS serves three services with equal arrival rates but with different tariffs. The service tariffs for services 1 to 3 from NAS1 are 0.2, 1, 0 price units and for services 4 to 6 from NAS2 are 0.4, 2, 0 price units. For comparison purposes, we set the loss targets for the CLP and the SCLP policies such that the potential loss of NAS2 is double that of NAS1. The simulation results for the potential loss and the mean AAA system load are shown in Fig. 7a and Fig. 7b.

When NAS2 fails over (i.e., at instant A), then depending on the interim interval settings returned for new sessions coming from NAS2 by the AAA under consideration, a transient behavior of the loss may occur (see Fig. 7a). Since there is no change in the interim setting for the Static_Max policy, no transient behavior is observed for the loss or for the load. Due to the change of the interim interval, all other policies incur a transient behavior. The transient effects in

TABLE 2
Percentage Load and Losses for Two NASes (i.e., NAS1 and NAS2) and a Proxy (30 Runs, 95 Percent Confidence with Error in Loss and Load Below 3 Percent Variation, Load from NAS1 and NAS2 Services is 1/s while that from $\mathrm{NAS}_{visited}$ is 0.1/s, $P = 300$ req/s)

	CLP		SCLP		APWC		Static_Max	
	Loss	*Load*	*Loss*	*Load*	*Loss*	*Load*	*Loss*	*Load*
$\mathbf{NAS}_1$	93.5	16.9	94.8	18.5	59.0	19.3	325	14.5
$\mathbf{NAS}_2$	95.9	19.8	97.0	21.4	86.7	20.6	650	14.3
$\mathbf{NAS}_{visited}$	96.6	1.8	98.7	1.8	124	1.7	434	1.2

the load curves in Fig. 7b are not as significant as in the case of the potential loss. This in fact shows that changing the interim interval for an operational system does not impact the load drastically while it can majorally change the loss behavior depending on the service costs.

We also observe that the CLP and the SCLP methods maintain the loss targets for NAS1 and NAS2 at the 100 percent and 200 percent levels as shown in Fig. 7a. The load for the CLP and the SCLP policies from both NASes in Fig. 7b is very similar due to the fact that the tariff targets are proportional to the total service costs (200 percent/100 percent is $(2 + 0.4)/(1 + 0.2)$). For the APWC policy, the same loss behavior is observed as in the Static_Min policy while the load behavior is observed to be different as the APWC sets the interim settings for services 3 and 6 at $\mathbf{\Delta}_{\mathbf{T}}^{max}$. For all policies, both NASes are jointly optimized and interims are generated to either minimize the loss from both NASes (i.e., the APWC) or to control the loss at the given targets as in the CLP scheme. The slight difference between the loads of NAS1 and NAS2 in Fig. 7b is due to the difference between the NAS sizes where NAS2 poses lower load on the AAA system.

5.2.4 Impact of Roaming Users (Proxy Chains)

In some cases such as in roaming, the AAA system connected to the NAS may forward requests to the destination AAA system through few intermediate AAA proxies [13]. This configuration is referred to as the AAA proxy chain. As a result, the optimization carried by one AAA system might be in conflict with the other AAA systems in the AAA proxy chain. For instance, consider the case for roaming users where NAS_v reports the usage to AAA_v of the visited network which proxies the accounting reports to the home network's AAA_h system. System overload may occur if the optimization is carried out by either of the AAA systems without considering the other. To address this case, when the first request for roaming users is received by the system, a preconfigured capacity Q is requested for the reserved stream from all servers in the chain by AAA_v using an access request message. If the requested capacity is approved by all systems in the proxy chain, then the request is accepted otherwise a reject message is generated. Only one AAA system in the chain (e.g., AAA_v) optimizes the reporting intervals within the prescribed reserved capacity Q while the other AAA systems (i.e., AAA_b and AAA_h) treat these services as nonoptimizable. In this case, our policies are left intact with the simple modification to include constraints that limit the load due to the proxied signaling messages below the preconfigured/negotiated limit Q (i.e., $\zeta(\mathbf{\Delta}_{\mathbf{T}px}) < Q$, where $\mathbf{\Delta}_{\mathbf{T}px}$ denotes the interim intervals of the proxied services from network x). The available capacity for local requests is reduced as $(P - Q)$. This simple prereservation scheme is suitable for proxy chain configurations as roaming traffic is expected to be low compared to home users' traffic.

TABLE 3
Mechanism Execution Delay (ms) (All Results are within 50 ms for APWC and CLP and within 5 ms for the SCLP Scheme with 95 Percent Confidence Using the Mean Batch Method, 30 Batches, Constant Unit Load from all Services)

	No. Services	2	4	8	12	15
Case A	APWC	763	791	857	1618	1919
	CLP	818	895	864	1009	1265
	SCLP	3	5	9	12	16
Case B	APWC	760	773	858	994	1698
	CLP	824	899	869	997	1194
	SCLP	2	4	8	11	14

Let us now consider a typical configuration which supports roaming users. In this regard, the NAS in the roaming partner's network (which we call here as $\mathrm{NAS}_{visited}$) is connected to an AAA system in the visited network. The visited AAA system forwards the accounting traffic to the home AAA system which also supports requests from home NASes (NAS 1 and NAS 2). The visited NAS supports two services each with 1 unit cost to reflect roaming charges while NAS1 serves three services with 0.2, 1, and 0 price units/minute and NAS2 serves another set of services with price units of 0.2, 2, and 0. Before the exchange, both systems negotiate the allocated capacity ($Q = 20$ req/sec to roaming traffic) for the forwarded (proxy) traffic, and thus, both AAA systems dedicate a maximum load. The visited AAA optimizes the interim values while the home AAA system treats the traffic as nonoptimizable. Same results are observed when this is reversed. As shown in Table 2, the mean loss is around the target loss limit for the three NASes when using the CLP and the SCLP policies. The load of the $\mathrm{NAS}_{visited}$ is below the limit. We also observe that all policies offer significantly lower loss for all NASes without significant load requirements compared to the static policy $\mathbf{\Delta}_{\mathbf{T}}^{max}$.

5.3 Computational Performance

In this section, we investigate our mechanism's performance in terms of the required execution time for the optimization operation and the number of mechanism's invocations as a function of the trigger setting. The trigger setting is defined as the amount of change in the load and session statistics for services which triggers updating the current interim settings. This in fact determines the mean duty cycle of the mechanism invocation (i.e., the interim intervals update rate) and should always be larger than the mechanism execution delay. In our study cases, we used a standard desktop machine (Intel Core 2 CPU E6700, 2 GB of memory, Windows XP OS). In order to observe the effect of the APWC knob parameter, ρ_0, in (11) (set at 60 percent) as we did in Fig. 5b, we study the execution time using two AAA capacities (Case A: 210 req/sec and Case B: 300 req/sec) to reflect two different system utilizations. As shown in Table 3, we observe very low execution times for the SCLP method compared to the CLP and the APWC methods. We

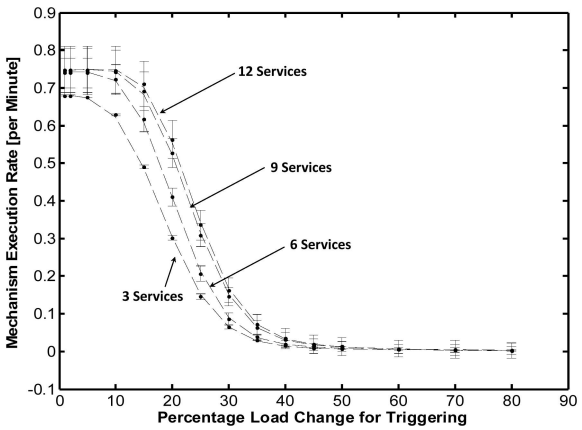

Fig. 8. The effect of the mechanism triggering threshold.

also observe that system utilization (compare Case A and Case B) barely affects the performance of the CLP and the slight difference is rather within the confidence limits of the test. On the other hand, the performance of the APWC scheme is affected with the system load (e.g., compare Cases A and B for 12 and 15 services). This is due to the fact that the system load exceeds the APWC knob setting, and hence, the nonlinear weight function W in (11) starts to have significant values in the objective function, and hence, impacts the optimization time. We conclude that due to the superior performance of the SCLP scheme it might be directly implemented into the AAA servers as a simple module as in [24], [33], [34].

Let us now investigate the effect of the mechanism triggering threshold on the execution rate. As shown in Fig. 8, we observe that increasing the optimization triggering threshold drastically reduces the mechanism invocation rate from approx 0.8 invocations/minute to below 0.1 invocations when the mechanism triggering threshold is set over 30 percent. The shape of the curve is due to the fact that when the triggering threshold is large, the mechanism is barely invoked while when the threshold is too small, the execution rate is upper limited by the grace period setting of 75 seconds (i.e., $1/75 = 0.8/\text{min}$). We also observe that the number of services does not largely impact the mechanism triggering rate.

Finally, we investigate the effect of the triggering threshold setting on the loss and the AAA system load. We use the same configuration as in Fig. 6 with variable load and use the most granular threshold with 1 percent change in the load or session statistics as a reference. To compare to other threshold settings, we use the root mean square error (RMSE) for the load and the potential loss between the reference case (i.e., 1 percent threshold) and the threshold under consideration. The larger the RMSE, the worse the performance. As shown in Table 4, we observe that in our test case, the potential loss performance is affected significantly more than the system load by the triggering threshold. We also observe that the SCLP is the most sensitive scheme to the threshold setting while the APWC is the least sensitive. This is because the solution of the SCLP is not optimal and is more likely to fluctuate if not optimized frequently enough. On the other hand, the APWC tends to minimize the loss when the system is not overloaded, and hence, will not likely change the interim settings from the last optimal value. Table 5 provides a short comparison between the proposed accounting policies.

TABLE 4
Root Mean Square Error for System Load and Norm

	Policy	Threshold Setting				
		5%	15%	25%	35%	45%
Loss	SCLP	1.2	1.8	6.2	9.1	13.2
	CLP	0.6	0.7	1.9	5.5	9.7
	APWC	0.4	0.5	1.0	1.6	1.6
Load	SCLP	0.2	0.3	0.6	1.0	1.2
	CLP	0.5	0.5	1.2	2.4	2.9
	APWC	0.5	0.5	0.9	1.6	1.6

Potential loss with reference to the 1 percent mechanism triggering threshold setting.

6 Conclusion

In this paper, we proposed an adaptive optimization mechanism for postpaid accounting in multiservice AAA systems. Our mechanism limits the potential loss without excessively generating unnecessary usage reports. The proposed mechanism is based on IETF AAA standards RADIUS and Diameter and does not require changes to the network access servers in the network nor to the standards. Changes are only limited to the AAA systems in the network. Using various simulations, we showed that the mechanism maintains optimal service reporting intervals in dynamic environments which involve mobility, variation of the service load, tariff switching, and failovers. The results showed that our mechanism is light weight and does not pose processing overhead on the system. Future work

TABLE 5
Summary and Comparison between the Accounting Optimization Policies

		CLP	SCLP	APWC	Static
1.	**Loss guarantees**	Supported	Supported	Best Effort	Not Supported
2.	**Overload avoidance**	Medium High	Medium Low	Best	None
3.	**System requirements**	Changes to AAA + Optimization Solver	Changes to AAA	Changes to AAA + Optimization Solver	Already Supported
4.	**Execution time**	Medium [0.8-1.2]s	Very low [< 20] ms	Medium High [0.9-1.9]s	Lowest [< 10]ms
5.	**Threshold sensitivity**	Medium Sensitive	Most Sensitive	Low Sensitivity	Not applicable
6.	**Complexity**	Polynomial Time [SQP]	One iteration	Polynomial Time [SQP]	No computation
7.	**Robustness and adaptability to changes**	Full	Full	Full	None

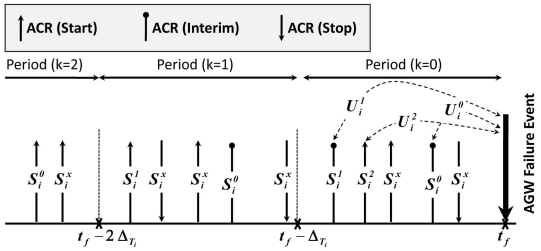

Fig. 9. The unreported usage at the event of NAS failure.

includes the implementation of the mechanism using open source AAA packages such as `Free RADIUS` or `Open Diameter` and validating its performance with real captures of accounting traffic. We also plan to extend our work to cover unified billing architectures combining both prepaid and postpaid mechanisms, as well as integration with dynamic pricing tools.

Appendix A

The Mean Interims Rate

Let J be the number of interims in the ith service flow as $J = \lfloor \frac{S_i}{\Delta_{T_i}} \rfloor$, then the pdf of J is [15],

$$\begin{aligned} f_J(j) &= \int_{j\Delta_{T_i}}^{(j+1)\Delta_{T_i}} f_{S_i}(x)dx = F_{S_i}((j+1)\Delta_{T_i}) - F_{S_i}(j\Delta_{T_i}) \\ &= \bar{F}_{S_i}(j\Delta_{T_i}) - \bar{F}_{S_i}((j+1)\Delta_{T_i}). \end{aligned} \tag{12}$$

Using (12), the mean rate of interims from all services S_i is given as the sum of products of their arrival rate λ_i and the mean number of interims produced during their lifetime (i.e., $E[\lfloor \frac{S_i}{\Delta_{T_i}} \rfloor]$) as

$$\begin{aligned} \zeta_{Int} &= \sum_{i\in\mathbb{A}} \lambda_i E\left[\left\lfloor \frac{S_i}{\Delta_{T_i}} \right\rfloor\right] = \sum_{i\in\mathbb{A}} \lambda_i \sum_{j=0}^{\infty} j f_J(j) \\ &= \sum_{i\in\mathbb{A}} \lambda_i \sum_{j=1}^{\infty} \bar{F}_{S_i}(j\Delta_{T_i}). \end{aligned} \tag{13}$$

Appendix B

The Mean Potential Loss

Consider a NAS failure event which occurs at a time instant denoted as t_f (see Fig. 9). Let us denote the mean number of the users at the system who consume service i as N_i. When a NAS fails, the loss for service i, L_i is given by the product of the number of users, the service cost, and the mean unreported service usage, U_i, as,

$$L_i = N_i C_i U_i. \tag{14}$$

To estimate N_i and U_i at t_f, we start by dividing the time access into Δ_{T_i} steps and move backwards from the loss event (see Fig. 9). By dividing the time axis this way, we can categorize sessions according to the number of interims they incurred (i.e., sessions with zero interims, with only one interim, etc.). For instance in Fig. 9, the lifetime of session S_i^0 is less than Δ_{T_i}, and hence, produced no interims at the moment of failure. The unreported usage in this case is U_i^0 which equals the session lifetime. The age of the session S_i^1 at t_f lies in the interval $[\Delta_{T_i}, 2\Delta_{T_i}]$, and hence, contains one interim message. The Unreported usage in this case is U_i^1. Finally, the age of session S_2 at t_f lies in the interval $[2\Delta_T, 3\Delta_T]$ and results in unreported usage of U_i^2. Any other sessions that finished before the failure event do not contribute to the loss and are marked as S_i^x in Fig. 9. In all of our exemplary cases S_i^0, S_i^1, and S_i^2, we observe that the loss event always falls randomly in the interval $[k\Delta_{T_i}, (k+1)\Delta_{T_i}]$, where $k \in \{0, 1, 2, \ldots\}$. Let us start by considering the sessions initiating in the first interim period such as S_2 (i.e., Period $\mathrm{k} = 0$) in Fig. 9. Assuming Poissonian arrivals, then the number of the corresponding sessions in the system denoted as N_i is given by the sum of the likelihood that a failure happens at instant t, a session arrival occurs (λdt), and that the session survives until the failure event ($\bar{F}_{s_i}(t)$) as

$$N_i^0 = \int_{t=0}^{\Delta_{T_i}} \Pr\{arrival\}\Pr\{failure\}\Pr\{Session\ survives\ until\ t\}$$

or

$$N_i^0 = \frac{1}{\Delta_{T_i}} \int_{t=0}^{\Delta_{T_i}} \lambda_i \bar{F}_{s_i}(t)dt. \tag{15}$$

The corresponding mean unreported usage U_i^0 *per session* is given by the weighted sum of the unreported usage due to each session divided by the number of the impacted sessions N_i^0. This is given as,

$$U_i^0 = \frac{1}{N_i^0}\frac{1}{\Delta_{T_i}} \int_{t=0}^{\Delta_{T_i}} \lambda_i t \bar{F}_{s_i}(t)dt = \frac{\int_{t=0}^{\Delta_{T_i}} t\bar{F}_{s_i}(t)dt}{\int_{t=0}^{\Delta_{T_i}} \bar{F}_{s_i}(t)dt}. \tag{16}$$

Observing that the mean age (or residual lifetime) $E[\tilde{S}]$ for the service session until failure is given as,

$$E[\tilde{S}] = \int_{t=0}^{\infty} t\tilde{f}_S(t)dt = \int_{t=0}^{\infty} t\left(\frac{\bar{F}_S(t)}{E_s}\right)dt = \frac{\int_{t=0}^{\infty} t\bar{F}_S(t)dt}{\int_{t=0}^{\infty} \bar{F}_S(t)dt}. \tag{17}$$

Comparing (16) and (17), it is clear that (16) can be viewed as the average age of the flows that have lifetime in the period of $[0, \Delta_{T_i}]$. We now extend this result to the periods $(\mathrm{k} = 1, 2, 3, \ldots)$. For the period $(\mathrm{k} = 1)$, the number of arrivals is given as

$$N_i^1 = \frac{1}{\Delta_{T_i}} \int_{x=\Delta_{T_i}}^{2\Delta_{T_i}} \lambda_i \bar{F}_{s_i}(x)dt.$$

Similarly, the number of surviving arrivals in the kth period is given as

$$N_i^k = \frac{1}{\Delta_{T_i}} \int_{x=k\Delta_{T_i}}^{(k+1)\Delta_{T_i}} \lambda_i \bar{F}_{s_i}(x)dt.$$

Hence, the total number of surviving arrivals from service i until the loss event is given as

$$N_i = \frac{\lambda_i}{\Delta_{T_i}} \sum_{k=0}^{\infty} \int_{x=k\Delta_{T_i}}^{(k+1)\Delta_{T_i}} \bar{F}_{s_i}(x)dx = \lambda_i E_{s_i}, \tag{18}$$

where

$$\sum_{k=0}^{\infty}\int_{x=k\Delta_{T_i}}^{(k+1)\Delta_{T_i}}\bar{F}_{s_i}(x)dx=\int_{x=0}^{\infty}\bar{F}_{s_i}(x)dx.$$

This result also matches the steady state mean number of users in M/G/∞ systems which is reasonable as in practice a NAS serves thousands of concurrent sessions [35]. Similar to (18), the mean unreported usage, U_i, is given by the sum of the usage from all sessions starting in the periods $[k\Delta_{T_i},(k+1)\Delta_{T_i}]$ (e.g., U_i^0,U_i^1,U_i^2 in Fig. 9).

$$\begin{aligned}U_i&=\frac{1}{N_i}\sum_{k=0}^{\infty}\int_{t=0}^{\Delta_{T_i}}\frac{\lambda}{\Delta_{T_i}}t\bar{F}_{s_i}(k\Delta_{T_i}+t)dt\\&=\frac{1}{E_{s_i}}\sum_{k=0}^{\infty}\int_{t=0}^{\Delta_{T_i}}t\bar{F}_{s_i}(k\Delta_{T_i}+t)dt.\end{aligned}\tag{19}$$

Let us substitute $y=k\Delta_{T_i}+t$ in (19). Then, we have

$$U_i=\frac{1}{E_{s_i}}\sum_{k=0}^{\infty}\int_{y=k\Delta_{T_i}}^{(k+1)\Delta_{T_i}}(y-k\Delta_{T_i})\bar{F}_{s_i}(y)dy.\tag{20}$$

Observing that $\frac{\bar{F}_{s_i}(y)}{E_{s_i}}$ is simply the probability density function of the age of the service session $\tilde{S}_i$ at any random moment (see (17)), then

$$\frac{1}{E_{s_i}}\sum_{k=0}^{\infty}\int_{y=k\Delta_{T_i}}^{(k+1)\Delta_{T_i}}y\bar{F}_{s_i}(y)dy=\int_{y=0}^{\infty}y\frac{\bar{F}_{s_i}(y)}{E_{s_i}}dy=E\{\tilde{S}_i\}.$$

For the other part of (20),

$$\left(\text{i.e.,}-\sum_{k=0}^{\infty}\int_{y=k\Delta_{T_i}}^{(k+1)\Delta_{T_i}}\frac{k\Delta_{T_i}}{E_{s_i}}\bar{F}_{s_i}(y)dy\right),$$

we observe that $\frac{\bar{F}_{s_i}(y)}{E_{s_i}}$ represents the probability density of the age of the session at t_f as $f_{\tilde{S}_i}(y)$. Using (12)-(13), we have

$$-\Delta_{T_i}\sum_{k=0}^{\infty}k\int_{y=k\Delta_{T_i}}^{(k+1)\Delta_{T_i}}f_{\tilde{S}_i}(y)dy.$$

This simplifies to

$$\begin{aligned}&-\Delta_{T_i}\sum_{k=0}^{\infty}k\big(\bar{F}_{\tilde{S}_i}\big(k\Delta_{T_i}\big)-\bar{F}_{\tilde{S}_i}\big((k+1)\Delta_{T_i}\big)\big)\\&=-\Delta_{T_i}\sum_{k=1}^{\infty}\bar{F}_{\tilde{S}_i}(k\Delta_{T_i}).\end{aligned}$$

The infinite sum is the mean number of interims during the lifetime of the session, and hence, we have

$$-\Delta_{T_i}\sum_{k=1}^{\infty}\bar{F}_{\tilde{S}_i}\big(k\Delta_{T_i}\big)=-\Delta_{T_i}E\left\{\frac{\tilde{S}_i}{\Delta_{T_i}}\right\}.$$

Thus, the mean unreported usage per session in (20) is

$$U_i=E\{\tilde{S}_i\}-\Delta_{T_i}E\left\{\left\lfloor\frac{\tilde{S}_i}{\Delta_{T_i}}\right\rfloor\right\}=\epsilon_i\Delta_{T_i},\ \Delta_{T_i}\le E_{s_i}.\tag{21}$$

The upper bound on U_i can be simply obtained by the observation that if all sessions at the failure instant, t_f, have incurred at least one interim then the failure event will fall uniformly in the interval $[0,\Delta_{T_i}]$, and hence, the mean unreported usage per session is $\Delta_{T_i}/2$.

APPENDIX C
THE SCLP DERIVATION

In this policy, we find Δ_{T_i} for each service by solving for the case when the loss constraint is bounding (i.e., $L=L_{max}^{(j)}$) for each NAS j. To simplify the notation we drop the NAS index for the loss and the interim intervals. The interim intervals can be found by solving a linear vector equation of the steepest gradient decent direction towards the loss constraint for each NAS.

$$\boldsymbol{\Delta}_{\mathbf{T}}=\boldsymbol{\Delta}_{\mathbf{T}}^{min}-\alpha\nabla\mathbf{L}.\tag{22}$$

The gradient function $\nabla\mathbf{L}$ for NAS j, is given by the partial derivative of the loss relative to all interim intervals served by that NAS (i.e., $\boldsymbol{\Delta}_{\mathbf{T}}$) as

$$\nabla\mathbf{L}=\left(\frac{dL}{d\boldsymbol{\Delta}_{\mathbf{T}0}}\quad\frac{dL}{d\boldsymbol{\Delta}_{\mathbf{T}1}}\quad\cdots\quad\frac{dL}{d\boldsymbol{\Delta}_{\mathbf{T}i}}\right),\tag{23}$$

where

$$\frac{dL}{d\Delta_{T_i}}=0.5\lambda_iC_iE_{s_i}.$$

Since at the loss boundary we have $L=L_{max}^{(j)}$, the scalar constant α is obtained substituting (22) into (8) and solve for α as

$$\begin{aligned}L_{max}^{(j)}&=\sum_{i\in\mathbb{N}_j}\frac{\lambda_iC_i}{2}E_{s_i}\left(\left\|\boldsymbol{\Delta}_{\mathbf{T}}^{min}\right\|_i-\alpha\left\|\nabla L\right\|_i\right),\\\alpha&=\frac{\sum_{i\in\mathbb{N}_j}\lambda_iC_iE_{s_i}\|\boldsymbol{\Delta}_{\mathbf{T}}^{min}\|_i-2L_{max}^{(i)}}{\sum_{i\in\mathbb{N}_j}\lambda_iC_iE_{s_i}\|\nabla L\|_i}.\end{aligned}\tag{24}$$

ACKNOWLEDGMENTS

The authors would like to thank Ankit Singla for discussions and software development for early versions of this paper. They would also like to thank the anonymous reviewers for their valuable comments and suggestions.

REFERENCES

[1] C. Rigney et al., "Remote Authentication Dial in User Service (RADIUS)," IETF RFC 2865, June 2000.
[2] P. Calhoun et al., "Diameter Base Protocol," IETF RFC 3588, Sept. 2003.
[3] "WiMAX Forum Network Architecture—Stage 2 Part 2—Release 1.1.0," http://www.wimaxforum.org/technology/documents, 2010.
[4] 3GPP2 X.S0011-005-C, "CDMA2000 Wireless IP Network Standard: Accounting Services and 3GPP2 RADIUS VSAs," V1.0, Aug. 2003.
[5] 3GPP TS 32.299, Diameter Charging Applications, 09, 2008.
[6] 3GPP2 X.S0013-000-B, "All IP Core Network Multimedia Domain," V1.0, Dec. 2007.
[7] 3GPP TS 22.258, "Service Requirements for the All IP Network (AIPN)," V8.0.0, Mar. 2006.
[8] D. Nelson et al., "Common Remote Authentication Dial in User Service (RADIUS) Implementation Issues and Suggested Fixes," IETF RFC 5080, Dec. 2007.
[9] P. Calhoun, "Interim Accounting Record Extension Draft," http://www.freeradius.org/rfc/draft-ietf-radius-acct-interim-01.txt, Jan. 1998.
[10] "Motorola's UMTS 'Radio Network Controller' Solution," Datasheet, 2007.
[11] J. Koomey, "Estimating Total Power Consumption by Servers in The U.S. and the World," final report, LBNL, 2007.

[12] G. Camarillo and M. Garcia-Martin, *The 3G IP Multimedia Subsystem (IMS).* John Wiley & Sons, Aug. 2004.
[13] S. Zaghloul and A. Jukan, "On the Performance of the AAA Systems in 3G Cellular Networks," *Proc. IEEE Comm. Conf. (ICC),* 2007.
[14] S. Zaghloul and A. Jukan, "Relating the AAA and the Radio Access Rates in 3G Cellular Networks," *IEEE Comm. Letters,* vol. 11, no. 4, pp. 363-365, Apr. 2007.
[15] S. Zaghloul and A. Jukan, "Signaling Rate and Performance for Authentication, Authorization, and Accounting (AAA) Systems in all IP Cellular Networks," *IEEE Trans. Wireless Comm.,* vol. 8, no. 6, pp. 2960-2971, June 2009.
[16] P. Calhoun, T. Johansson, C. Perkins, T. Hiller, and P. McCann, "DiameterMobile IPv4 Application," IETF RFC 4004, Aug. 2005.
[17] J. Altmann and L. Rhodes, "Dynamic Netvalue Analyzer—A Pricing Plan Modeling Tool for ISPs Using Actual Network Usage Data," *Proc. IEEE Int'l Workshop Advance Issues of e-Commerce and Web-Based Information Systems (WECWIS '02),* June 2002.
[18] *System and Method of Monitoring and Reporting Accounting Data Based on Volume,* US Patent 6999449, Washington, D.C.: Patent and Trademark Office, 2006.
[19] S. Sou et al., "Modeling Credit Reservation Procedure for UMTS Online Charging System," *IEEE Trans. Wireless Comm.,* vol. 6, no. 11, pp. 4129-4135, Nov. 2007.
[20] J. Hwang, J. Altmann, I. Okumus, and P. Aravamudham, "Transaction Management for Sender/Receiver-Payment Schemes in Charging and Accounting Systems for Interconnected Networks," *Proc. IEEE/IFIP Network Operations and Management Symp. (NOMS '04),* Apr. 2004.
[21] F. Eyermann et al., "Diameter-Based Accounting Management for Wireless Services," *Proc. IEEE Wireless Comm. and Networking Conf. (WCNC '06),* Apr. 2006.
[22] J. Na, Y. Chung, M. Yun, and Y. Kim, "An Efficient Diameter-Based Accounting Scheme for Wireless Metropolitan Area Network (WMAN)," *Proc. IEEE Vehicular Technology Conf. (VTC),* 2004.
[23] M. Bella et al., "Using the IPDR Standard for NGN Billing and Fraud Detection," *Proc. Fifth Ann. Information Security South Africa Conf. (ISSA '05),* June 2005.
[24] Modules from FreeRADIUS, http://wiki.freeradius.org/Modules, 2010.
[25] "AAA Service Controller," Datasheet, Bridgewater Systems, http://www.bridgewatersystems.com/products/aaa_service_controller.html, 2007.
[26] Cisco, "User Guide for Cisco Access Registrar, 4.2," http://www.cisco.com/en/US/docs/net_mgmt/access_registrar/4.2/user/guide/CAR4.2_usersguide.pdf, 2008.
[27] K. Yeo and C. Jun, "Modeling and Analysis of Hierarchical Cellular Networks with General Distributions of Call and Cell Residence Times," *IEEE Trans. Vehicular Technology,* vol. 51, no. 6, pp. 1361-1374, Nov. 2002.
[28] F. Barcelo and J. Jordan, "Channel Holding time Distribution in Public Telephony Systems," *IEEE Trans. Vehicular Technology,* vol. 49, no. 5, pp. 1615-1625, Sept. 2000.
[29] E. Casilari et al., "Modelling of Voice Traffic over IP Networks," *Proc. Third Int'l Symp. Comm. Systems, Networks, and Digital Signal Processing (CSNDPS '02),* 2002.
[30] E. Yavuz and V. Leung, "Modeling Channel Occupancy Times for Voice Traffic in Cellular Networks," *Proc. IEEE Comm. Conf. (ICC '07),* June 2007.
[31] L. Peng et al., "Experimental Study on Traffic Model of Wireless Internet Services in CDMA Network," *Proc. IEEE Proc. 61st Vehicular Technology Conf. (VTC '05),* Apr. 2005.
[32] R. Nelson, *Probability, Stochastic Processes, and Queuing Theory.* Springer, 1995.
[33] Y. Ohba, "Diameter NASREQ Application API," http://www.opendiameter.org, 2004.
[34] JDiameter Project, https://jdiameter.dev.java.net, 2010.
[35] "Starent ST16 PDSN&HA Datasheet," http://www.starentnetworks.com/pdf/StarentNetworks_PDSN_Datasheet_0904.pdf, 2010.

Said Zaghloul received the first IEE award for his BSc senior projects in 2002 in Jordan, for his work dedicated to the development of a UMTS cellular planning tool. In 2003, he was granted a Fulbright Scholarship to pursue the MSc degree at the University of Kansas. In 2005, he received the MSc degree with honors in computer engineering for his work in the area of inversely multiplexed satellite connections. He is currently working toward the PhD degree at the Technische Universitaet Carolo-Wilhelmina zu Braunschweig, Germany. Prior to his PhD studies, he was with Sprint-Nextel as a telecommunication design engineer, where he was a major contributor to the design and testing of Sprint's wireless data network architectures in several areas including MVNO and roaming solutions, dual GPRS/CDMA solutions, and hybrid WiFi/CDMA phone products. His current research interests include next generation all-IP wireless architectures, signaling plane performance, mobility, and wireless communications. He is a student member of the IEEE.

Admela Jukan received the MSc degree in information technologies and computer science from the Polytechnic of Milan, Italy, and the PhD degree (cum laude) in electrical and computer engineering from the Vienna University of Technology (TU Wien), Austria. She is a W3 professor of electrical and computer engineering at the Technische Universitaet Carolo-Wilhelmina zu Braunschweig, Germany. Prior to coming to Brunswick, she was a research faculty member at the Institut National de la Recherche Scientifique (INRS), University of Illinois at Urbana Champaign (UIUC), and the Georgia Institute of Technology (GaTech). From 2002 to 2004, she served as a program director in Computer and Networks System Research at the US National Science Foundation (NSF) in Arlington, Virginia. She is the author of numerous papers in the field of networking, and she has authored and edited several books. She serves as a member of the External Advisory Board of the EU Network of Excellence BONE. She has chaired and cochaired several international conferences, including IFIP ONDM, IEEE ICC, and IEEE GLOBECOM. She serves as associate technical editor for *IEEE Communications Surveys*, *IEEE Communications Magazine*, and *IEEE Network*. She is a senior member of the IEEE.