On Integrated Location and Service Management for Minimizing Network Cost in Personal Communication Systems

Ing-Ray Chen, *Member*, *IEEE Computer Society*, Baoshan Gu, and Sheng-Tzong Cheng

Abstract—We investigate the notion of per-user integrated location and service management in personal communication service (PCS) networks by which a per-user service proxy is created to serve as a gateway between the mobile user and all client-server applications engaged by the mobile user. The service proxy is always *colocated* with the mobile user's location database such that whenever the MU's location database moves during a location handoff, a service handoff also ensues to colocate the service proxy with the location database. This allows the proxy to know the location of the mobile user all the time to reduce the network communication cost for service delivery. We investigate four integrated location and service management schemes. Our results show that the centralized scheme performs the best when the mobile user's SMR (service to mobility ratio) is low and CMR (call to mobility ratio) is high, while the fully distributed scheme performs the best when both SMR and CMR are high. In all other conditions, the dynamic anchor scheme is the best except when the service context transfer cost is high, under which the static anchor scheme performs the best. Through analytical and simulation results, we demonstrate that different users with vastly different mobility and service patterns should adopt different integrated location and service management methods to optimize system performance. Further, the best integrated scheme always performs better than the best decoupled scheme that considers location and service managements separately and management schemes that do not use any service proxy.

Index Terms—Integrated location and service management, service handoff, personal communication services, wireless networks, performance analysis.

1 Introduction

Future personal communications service (PCS) networks will provide a wide range of information services, such as personal banking service, personalized stock market information, location dependent travel information, etc. [8] for which a mobile user (MU) sends requests to a server and the server sends replies to the mobile user. To deliver a reply from the server to an MU, the server needs to know the MU's location information, which may be changed after requests are sent. For this reason, it has been suggested that a per-user service proxy be created for each mobile user to tackle the problem of personal mobility [16]. The service proxy performs tasks such as tracking locations of the MU, maintaining service context information for the service engaged, accepting service requests from the MU, transforming requests into proper formats, and forwarding server replies to the MU. As the personal proxy explicitly tracks the MU location, it eliminates the overhead for the server application to first check with the underlying location management system to know the current MU location before data delivery. However, since all client-server communications must go through the personal proxy, if the personal proxy is static in location, it is likely that inefficient server-proxy-MU triangle routes may be used for data delivery, resulting in high communication costs. On the other hand, if the proxy is mobile to stay closer to the MU, extra network costs will be incurred to inform the server applications of the address change whenever the proxy changes its location. Thus, there is a design trade-off between adopting static versus mobile server proxy in terms of the communication cost incurred.

Location and service managements have often been separately addressed in the literature [3], [6], [14]. For location management, the most popular scheme in PCS networks is the basic Home Location Register/Visitor Location Register (HLR/VLR) [4] scheme where each MU has an HLR. Whenever an MU enters a VLR, the system updates its HLR location database so that, when a call arrives, the HLR location database knows exactly which VLR contains the MU. Variations to the basic HLR/VLR scheme have been proposed in recent years to process location update and search operations more efficiently, e.g., Local Anchor (LA) [7], Forwarding and Resetting [9], Two-Level Pointer Forwarding [12], Hybrid Replication with Forwarding [2], etc. These location management schemes are designed to handle location update and search operations without consideration to service management.

IETF Mobile IP [15] allows an MU to maintain ongoing connections while roaming among IP subnets and requires the MU to inform its Home Agent (HA) of the new Foreign Agent (FA) address whenever it moves from one subnet to another. The function of an HA in Mobile IP is similar to an HLR in PCS networks for location management. Similar to the LA scheme in PCS networks, a variant of Mobile IP, called Mobile IP dynamic regional registration [19], has been proposed to group FAs into a gateway foreign agent (GFA) dynamically to minimize signaling costs in Mobile IP. These

- *I.R. Chen and B. Gu are with the Department of Computer Science, Virginia Tech, Northern Virginia Center, 7054 Haycock Road, Falls Church, VA 22043. E-mail: irchen@cs.vt.edu, bgu@vt.edu.*
- *S.-T. Cheng is with the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan. E-mail: stcheng@mail.ncku.edu.tw.*

Manuscript received 13 Dec. 2002; revised 19 Apr. 2004; accepted 7 Sept. 2004; published online 15 Dec. 2005.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number 4-122002.

1536-1233/06/$20.00 © 2006 IEEE Published by the IEEE CS, CASS, ComSoc, IES, & SPS

solutions, although elegant, solve only location management issues. For service management, a delivery protocol using a service proxy has been proposed [5] to provide reliable delivery of messages to MUs. However, the proxy used to forward messages to an MU must explicitly track the location of the MU, so extra communication costs are incurred to notify the proxy when the MU moves across a location registration area boundary. Jain and Krishnakumar [8], [10] discussed the notion of distributed servers each covering a service area and, thus, a *service handoff* occurs when the MU crosses a service area boundary. Peng and Chen [14] assumed the existence of service handoffs and analyzed cache retrieval schemes to be employed during a transaction execution to improve the cache hit ratio by selecting the best server from which the MU will retrieve cached items upon a service handoff. In another related work, Dunham and Kumar [3] investigated the impact of mobility on mobile transaction management. In particular, they analyzed a service handoff scheme to move the transaction management from one service area to another as the MU crosses a service area in the PCS network. These studies assumed that replicated servers exist in service areas. No integrated location and service management was considered to reduce the overall cost due to location and service management operations.

Jain and Krishnakumar [8], [10] suggested that location and service handoffs be integrated to reduce the overall communication cost, but no follow-up analysis or research has been done to investigate the potential benefit of the integrated approach. Their notion of service handoff is based on the assumption of fully replicated servers in service areas such that, whenever an MU crosses a service area, its ongoing service can be handed off from one server to another by means of service context information transfer to allow the service to be continued at the new server. An example would be a video on-demand application with replicated servers such that the context information could include the video title, minutes played, and the current frames being buffered at the server and played at the MU. While it is possible to have integrated location and service management for these applications, the difficulty of integrating location handoffs (due to movements of the MU crossing VLR boundaries) with service handoffs (due to movement of the MU crossing service area boundaries) in the PCS network lies in the very large scale deployment of a large number of replicated servers in VLRs.

In this paper, we investigate the notion of integrated location and service management for minimizing network cost without making the assumption of fully replicated servers in VLRs in the PCS network. Instead, we target general personalized services in the PCS network including personal banking, stock market, and location-dependent services for which the MU will communicate with a backend server. Our notion of integrated location and service management is based on the concept of using a per-user service proxy as a gateway between the MU and all client-server applications engaged by the MU concurrently. The proxy keeps track of service context information such as the current state of the execution for maintaining service continuity. For example, for banking services, the context information could include the data that have been read or written and the state of the transaction execution. All user requests and server replies would pass through the proxy. If the backend server is replicated, e.g., for multimedia streaming applications [17], [20], the server may change its location for load balance and performance reasons, in which case, the server would inform the proxy of its location change without involving the MU.

A distinguishing feature of our integrated location and service management scheme is that we always colocate the MU's service proxy with the MU's location database that stores the current location of the MU, so that the service proxy knows the current location of the MU all the time so as to eliminate the cost associated with tracking the user location on behalf of the server applications for data delivery. In the PCS network, whenever the MU moves across a registration area boundary, a *location handoff* occurs for the location management system to update the location database. In our integrated location and service management scheme, associated with a location handoff is a *service handoff* [1] to update the service proxy. If a location handoff results in moving the MU's current location database to stay closer to the MU (e.g., as in the LA scheme), then the associated service handoff will also move the service proxy to the same location. Whether the MU's service proxy should move with the MU as the MU crosses VLRs in the PCS network depends on the specific integrated location and service management scheme employed. An integrated scheme that frequently moves the proxy would have the advantage of low-cost service and call management because of the proximity of the service proxy with the MU at the expense of high-cost location management and vice versa.

In this paper, we investigate and analyze four integrated location and service management schemes to explore this cost trade-off with the goal to identify conditions under which a particular scheme should be adopted by an MU based on the MU's own mobility and service characteristics for network cost minimization. These four schemes derive from the basic HLR/VLR and LA schemes for location management and the personal service proxy scheme for service management in the PCS network. We are motivated to investigate and identify the best integrated location and service management scheme that can be applied on an individual user basis to minimize the overall cost incurred to the PCS network per time unit for servicing location and service operations of all users. The amount of cost saving is relative to the speed of the PCS network and is proportional to the number of users, so the benefit is especially pronounced for slow and congested networks with a large number of mobile users. Here, we note that the use of smart terminals capable of reporting their locations may necessitate new location and service management schemes to be used (e.g., paging and letting smart terminals inform ongoing services of their location changes) rather than based on the HLR/VLR structure in the PCS network as is considered in this paper.

The contributions of the paper are: 1) We propose and analyze new integrated location and service management schemes not considered before and show that integrated location and service management is a viable concept applicable to the PCS network on a per-user basis for general server applications; 2) we show that, when given an MU's mobility and service characteristics through a set of parameters identified in the paper, there exists an optimal integrated location and service management scheme that would minimize the overall network communication cost as a result of executing the MU's location and service operations; 3) we show that the best integrated location and service

[1] A service handoff refers to the event that the MU crosses a service area, which, in our case, would coincide with a location registration area. Note that we do not consider fully replicated servers in service areas, which we consider would be difficult to deploy due to their large economical scale. Therefore, service handoffs in our schemes actually involve migrating the MU's service proxy from one service area to another area.

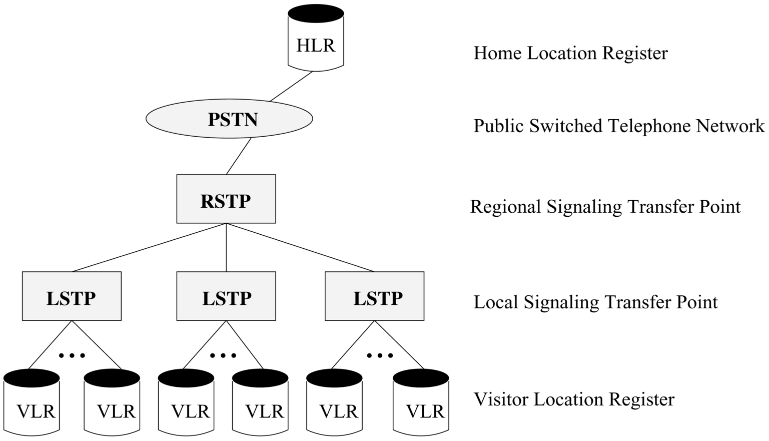
HLR
PSTN
RSTP
LSTP
LSTP
LSTP
VLR
VLR
VLR
VLR
VLR
VLR
Home Location Register
Public Switched Telephone Network
Regional Signaling Transfer Point
Local Signaling Transfer Point
Visitor Location Register

Fig. 1. PCS signaling network architecture.

scheme identified always performs better than the best decoupled scheme that considers location and service managements separately.

The rest of the paper is organized as follows: Section 2 gives a description of the system model and assumptions. Section 3 describes in detail the four integrated schemes to be investigated and analyzed in the paper. Section 4 analyzes the cost incurred under each of the four integrated schemes by means of Petri nets. Section 5 presents analytical results with simulation validation and reveals conditions under which one scheme may perform the best in terms of the overall communication cost incurred with physical interpretations given. It also demonstrates that the best integrated scheme always outperforms the best decoupled scheme as well as schemes that do not use any service proxy. Finally, Section 6 summarizes the paper.

2 System Model

We first describe a PCS system model for location management services. Then, we describe an extended system model for integrated location and service management. We consider a PCS network architecture as shown in Fig. 1, where the PCS service areas are divided into registration areas (RAs). A two-tier HLR/VLR scheme is used for location management. The HLR stores the user profiles of its assigned subscribers and the VLR in an RA stores profiles of users not at home and currently located in that RA. For simplicity, we assume that each VLR corresponds to one RA. Conceptually, the HLR of an MU is at a higher level, while all VLRs are at the lower level. The intermediate switches, such as Local Signaling Transfer Points (LSTPs) and Regional Signaling Transfer Points (RSTPs), are used for connecting VLRs and the HLR.

We assume that a particular MU will stay in a VLR before moving to another. For simplicity, the residence time is assumed to be exponentially distributed with an average rate of σ. Such a parameter can be estimated using the approach described in [11] on a per-user basis. We also assume the interarrival time between two consecutive calls to a particular MU, regardless of the current location of the MU, is exponentially distributed with an average rate of λ. We consider client-server computational models in which an MU sends requests to the server and the server replies with responses via message passing. The interarrival time between two consecutive service requests from an MU is assumed to be exponentially distributed with an average rate of γ. These assumptions allow a model to be developed in the paper based on Stochastic Petri Nets (SPN). The assumption of exponentially distributed times can be relaxed by using SPN evaluation tools that support specifications of general distributions such as SPNP [18] and TimeNET [21].

Under the basic HLR/VLR scheme, a mobile user is permanently registered under a location register HLR. When the mobile user enters a new VLR area, it reports to the new VLR, which, in turn, informs the HLR by means of a location update operation. When a call is placed, the system first searches the MU's current location through the HLR and then the call is delivered. For notational convenience, let the average round trip communication cost between a VLR and HLR be T, representing the cost for a location update operation, as well as for a search operation under the basic HLR/VLR scheme.

To provide integrated location and service management, we explore the notion of integrated location/service areas such that a service handoff occurs when a location handoff occurs as a location boundary area is crossed by the MU. One such boundary registration/service area considered is based on the notion of local anchor proposed by Ho and Akyildiz [7] in the context of location management. The basic idea is that, within an anchor area, we use a local anchor to maintain a location management database to keep track of the location of the MU within the anchor area. Specifically, the VLR which performs the last registration operation with the HLR will become the anchor in an anchor area. Since an anchor area may cover a large geographic area spanning several VLRs, when an MU crosses a VLR boundary, it may still be in the anchor area. In this case, a location update operation within the anchor area is only processed by the anchor without going to the HLR database, thus reducing the communication cost for update operations. Let P_{InA} and $P_{OutA} = 1 - P_{InA}$ be the probabilities of "intraanchor" and "interanchor" moves, respectively, when the MU crosses a VLR boundary. Also, let the average communication cost between the anchor and a VLR within the anchor area be τ_1. Then, a boundary crossing movement will incur a cost of τ_1 with probability P_{InA} and a cost of T with probability P_{OutA}. For a location search operation, we must always go to the HLR database to know the current anchor, which, in turn, points to the current VLR to get the user location information, resulting in a total cost of $T + \tau_1$. When applying the anchor scheme to integrated location and service management, the cost model must

TABLE 1
Parameters

λ	the average rate at which the MU is being called.
σ	the average rate at which the MU moves across VLR boundaries.
γ	the average rate at which the MU requests services.
CMR	call to mobility ratio, e.g., λ/σ.
SMR	service request to mobility ratio, e.g., γ/σ.
T	the average round trip communication cost between a VLR and the HLR (or between a VLR and the server) per message.
τ_1	the average round trip communication cost between the anchor and a VLR in the anchor area per message.
τ_2	the average round trip communication cost between two neighboring anchor areas per message.
τ_3	the average round trip communication cost between two neighboring VLRs per message.
M_{cs}	the number of packets required to transfer the service context.
N_s	the number of server applications concurrently engaged by the MU.
P_{InA}	the probability that a MU moves within the same anchor area when a VLR boundary crossing movement occurs.
P_{OutA}	the probability that a MU moves out of the current anchor area when a VLR boundary crossing movement occurs. $P_{OutA} = 1 - P_{InA}$.

include not only location update/search costs, but also the communication cost between an MU and its servers. Also, to deliver responses from a server to an MU through the proxy, the proxy must know the MU's current location. It is desirable not to query the HLR to obtain the location information because of high communication cost. Thus, in an integrated local anchor scheme to serve both location and service handoffs, whenever the MU moves to a new anchor area, it may be desirable to also migrate the service proxy to the new anchor area to be "colocated" with the new anchor in an anchor area so that the service proxy can query the anchor to know the current location of the MU without going to the HLR. Consequently, both a location handoff and a service handoff would occur when the MU crosses an anchor boundary in the integrated scheme.

A service handoff that migrates the service proxy involves two operations, namely, an address-change operation to inform all application servers of the location change and a service context transfer. The cost of the address change operation per server is T. The service context transfer is unique for the service handoff operation, with the amount of context information being application dependent. The context transferred may include both static context information such as user profile and authentication data as well as dynamic context information such as files opened, objects updated, locks and time-stamps, etc. Let τ_2 be the average communication cost between two neighboring anchor areas (per packet) and M_{cs} be the number of packets required to transfer the service context. Then, a context transfer cost will be $\tau_2 \times M_{cs}$.

We list system parameters considered in the paper in Table 1, including user parameters (such as λ, σ, and γ) and application-specific parameters (such as M_{cs}). Their effects on the performance of integrated location and service management schemes are to be analyzed in the paper.

Note that, for the case in which an MU concurrently interacts with multiple servers, there would still be only one per-user service proxy colocated with the location database under our proposed integrated schemes. In this case, the service rate parameter, γ, would reflect the *aggregate* rate at which the MU makes requests to these multiple services, while the context transfer cost parameter, M_{cs}, would reflect the *aggregate* context transfer cost for moving the service context information of multiple concurrent services from one location to another.

3 Integrated Location and Service Management

In this section, we discuss four possible schemes, i.e., centralized, fully distributed, dynamic anchor, and static anchor for integrated location and service management. We describe the operational procedures used to handle location update, call delivery, and service requests in these four schemes. One should note that the best integrated scheme is selected on a per-user basis for network cost minimization, not to be affected by other users in the system.

3.1 Centralized Scheme

Under the centralized scheme, the location management operations are handled as the basic HLR/VLR scheme and the service proxy is centralized and "colocated" with the HLR to avoid extra costs to locate the MU by the service proxy when forwarding server responses to the MU. A location update operation to the HLR is performed when an MU moves across a VLR boundary. A search operation at the HLR database is performed when a call is placed to the MU. A service request operation involves a high communication cost for the MU to send the request through the service proxy to the server. A service reply operation travels the opposite way.

We illustrate the centralized scheme in Fig. 2. As the MU moves from VLR A to VLR B and, subsequently, to VLR C, the HLR and the service proxy are updated to point to VLR B and then to VLR C sequentially.

3.2 Fully Distributed Scheme

Under the fully distributed scheme, both the location and service handoffs occur whenever the MU moves into a new VLR. The location handoff behaves the same as the basic HLR/VLR scheme. The service handoff migrates the service proxy along with the service context to the new serving VLR that the MU just enters into. Thus, the service proxy is always

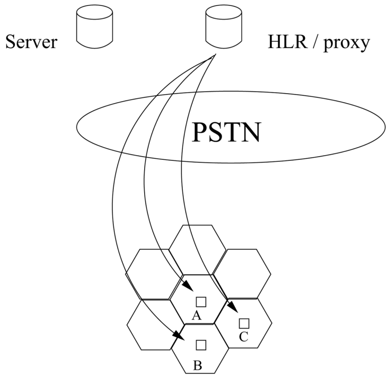

Fig. 2. Centralized scheme.

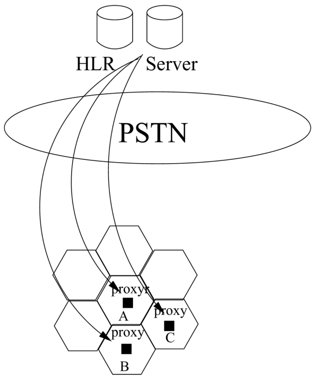

Fig. 3. Fully distributed scheme.

colocated with the location database at the current VLR pointed to by the HLR.

We illustrate the fully distributed scheme in Fig. 3. When the MU moves from VLR A to VLR B, the service proxy migrates from VLR A to VLR B and the HLR and the server are updated to point to VLR B. The subsequent move to C behaves similarly. To service a location search request (not initiated from the current VLR), the HLR database is accessed first to know the current VLR (A, B, or C) and then the MU is found within the current VLR. When the service proxy needs to forward replies to the MU, no extra search cost is required to find the current VLR since the service proxy is located in the current VLR.

3.3 Dynamic Anchor Scheme

Under the dynamic anchor scheme, a location anchor is used for location management such that the anchor changes whenever the MU crosses an anchor boundary. In addition, the anchor may also change its location within an anchor area when a call delivery operation is serviced. The service proxy dynamically moves with the anchor and is always colocated with the anchor. Below, we give the algorithmic description of the dynamic anchor scheme for processing location update, call delivery, and service requests.

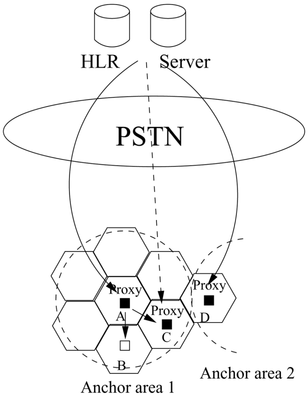

Fig. 4. Dynamic anchor scheme.

Location Update:

```
If (this is an anchor boundary crossing movement)
    A location update message is sent to the HLR through
        the new VLR
    The service context is moved to the new VLR who
        now serves as the new anchor
    A location update message is sent to all application
        servers
Else
    The new VLR sends location update message to the
        anchor
```

Call Delivery:

```
A location request message is sent to the HLR to know
    the anchor of the called user
If (the local anchor is the current serving VLR)
    The anchor sends a response to the HLR that the MU
        is found
Else
    The local anchor forwards the request to the current
        serving VLR
    The current VLR sends a location response to the HLR
    The HLR updates its record such that the current VLR
        becomes the new anchor
    The service context is moved to the current VLR (who
        is the new anchor)
    A location update message is sent to all application
        servers
```

Service Request:

```
A request is sent from the MU to its current VLR
If (the current VLR is the local anchor)
    The request is sent to the server and then a response is
        sent back to the MU
Else
    The current VLR forwards the request to the anchor
    The anchor forwards the service request/response to
        the server/MU
```

In Fig. 4, when an MU moves within anchor area 1 from VLR A to VLR B, only the local anchor in VLR A is updated to point to the current location. Thus, the location update to the HLR and application servers is avoided. Suppose that a call arrives after the MU moves into VLR C. The call will invoke a search operation in the HLR database and a subsequent search operation in the anchor. Once the call is serviced, the HLR database will be updated to point to VLR C, the anchor and the service context are moved from VLR A to VLR C, and the application servers are informed of the address change. Later, if the MU subsequently moves from VLR C to VLR D due to an interanchor movement, the HLR database will subsequently be updated to point to

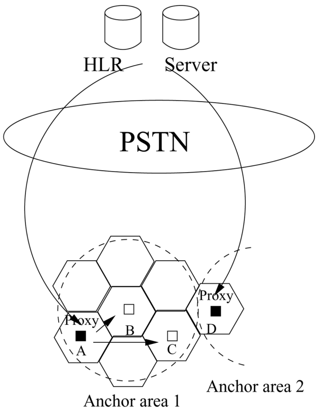

Fig. 5. Static anchor scheme.

VLR D, which will become the new anchor after the service context is transferred to it. Data delivery from the server will pass through the service proxy colocated with the anchor to reach the MU.

3.4 Static Anchor Scheme

Under the static anchor scheme, the service proxy is again colocated with the anchor. However, the anchor will remain at a fixed location as long as the MU stays in the same anchor area. The only condition under which the anchor would move (along with the service context transferred) is when the MU moves across an anchor boundary. The procedures for processing location update, call delivery, and service requests are the same as in the dynamic anchor scheme except that, upon a successful call delivery, the anchor's location remains unchanged. Thus, there is no need to migrate the service proxy to the current serving VLR (if they are not the same) after serving a call delivery operation.

We illustrate static anchor in Fig. 5. When the MU moves within anchor area 1 from VLR A to VLR B and then to VLR C, the local anchor in VLR A is updated to point to the current VLR without updating the HLR. An incoming call will invoke a search operation at the HLR database to first find the anchor and, then, from the anchor to find the current VLR. The location of the anchor (where the service proxy is colocated) remains unchanged after a call is serviced. The anchor moves only when the MU moves out of the current anchor area (from VLR C to VLR D in this case). For each service request issued from MU, it is serviced by the service proxy colocated with the anchor. As in dynamic anchor, there is no extra cost for the service proxy to find the MU, since the service proxy is colocated with the anchor.

4 MODELING

In this section, we develop analytical models for evaluating and comparing various integrated schemes introduced in Section 3. We first define the performance metric as the basis for evaluation. Then, we show how the performance metric can be assessed for various schemes. In particular, we develop stochastic Petri net models for analyzing the static and dynamic anchor schemes.

4.1 Cost Model

Our performance metric used for evaluating various integrated schemes is based on the total communication cost per time unit for handling three basic operations, namely, location update, call delivery, and user service requests. To be more specific, our cost model consists of three cost components: 1) update cost C_{update}—the cost for updating the locations of the MU and service proxy and transferring the service context, if needed, when a user moves across a VLR boundary, 2) search cost C_{search}—the cost for locating the MU to deliver a call, and 3) service request cost $C_{service}$—the cost for the MU to communicate with the server through the proxy. Note that the cost here stands for the "average" cost. Let C_{total} be the average cost of the PCS network in servicing the above three types of basic operations per time unit. Then, our performance metric C_{total}, defined as the total cost incurred to the PCS network per time unit for servicing location and service operations of the MU, is given by:

$$C_{total} = C_{update} \times \sigma + C_{search} \times \lambda + C_{service} \times \gamma, \qquad (1)$$

where σ, λ, and γ are the MU's VLR boundary crossing rate, call arrival rate, and service request rate, respectively, as described in Table 1. Note that the paging cost for locating the location of the MU within the current VLR is not considered in the cost model because the paging cost is the same for all schemes.

4.2 Centralized Scheme

For the centralized scheme, each operation incurs a communication cost between the user's current VLR and the HLR colocated with the centralized service proxy. Thus, we have

$$C_{update} = T,$$
$$C_{search} = T,$$
$$C_{service} = T + T,$$

where, in the last equation, the first T accounts for the round trip cost from the MU to the service proxy while the second T accounts for the cost from the proxy to the server. Thus,

$$C_{total}^{centralized} = T \times \sigma + T \times \lambda + 2T \times \gamma. \qquad (2)$$

4.3 Fully Distributed Scheme

In the fully distributed scheme, each time the MU moves across a VLR boundary, three costs occur, i.e., a cost of T is required to update the HLR database for keeping track of the MU, a cost of $M_{cs} \times \tau_3$ is required to transfer the service context to the new VLR to provide continuous services, where τ_3 stands for the communication cost between two neighboring VLRs and, finally, a cost of N_sT is required to inform N_s application servers of the address change of the service proxy. Each time a call is placed for the mobile user, the HLR consults the current VLR to get the location information with the communication cost T. For each service request, since the service proxy is always colocated with the current VLR of the MU, the only communication cost is from the proxy to the server. Summarizing above,

$$C_{update} = T + M_{cs} \times \tau_3 + N_sT,$$
$$C_{search} = T,$$
$$C_{service} = T.$$

Therefore, based on the cost metric defined in (1),

$$C_{total}^{distributed} = (T + M_{cs} \times \tau_3 + N_sT) \times \sigma + T \times \lambda + T \times \gamma. \qquad (3)$$

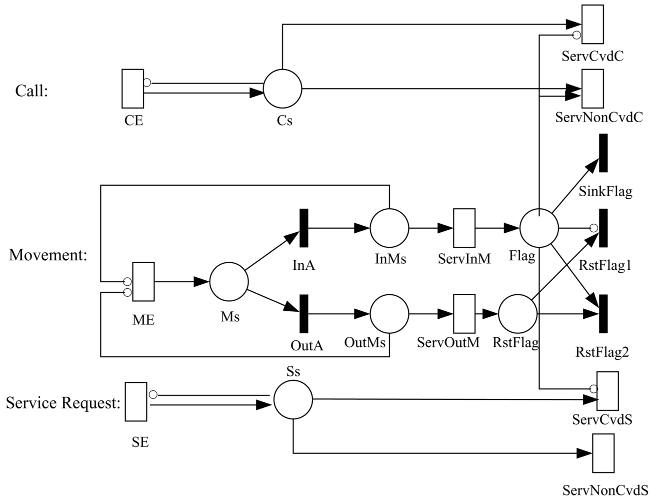

Fig. 6. Petri net model for the dynamic anchor scheme.

4.4 Dynamic Anchor

For the dynamic anchor scheme, a Stochastic Petri net model as shown in Fig. 6 is developed to analyze its behavior.[2] Table 2 gives the meanings of places and transitions defined in the Petri net model. Here, $mark(\texttt{p})$ returns the number of tokens in place `p`. The Petri net model is constructed as follows:

- When a call arrives, a token is placed in place `Cs`. The system serves the call based on the current status stored in place `Flag`:
  - If place `Flag` contains a token, or $mark(\texttt{Flag}) > 0$, then transition `ServNonCvdC` is enabled, which means that the current VLR is not the same as the anchor VLR. To deliver the call, the HLR is first queried to locate the anchor which, in turn, queries the current serving VLR to return the MU's current location. Note that, after the search operation is performed, the anchor is moved to the current VLR as modeled by resetting $mark(\texttt{Flag}) = 0$.
  - If $mark(\texttt{Flag}) = 0$, transition `ServCvdC` is enabled. It means that the user resides in the same VLR with the anchor, so the search request is sent to HLR, which, in turn, forwards the request to the local anchor. The local anchor returns the MU's location immediately.
- When an MU moves across a VLR boundary, a token is placed in place `Ms`.
  - If this is an intraanchor movement with probability P_{InA}, transition `InA` will consume the token immediately, after which a token will be placed in `InMs`, which subsequently disables transition `ME` and enables transition `ServInM`, representing that a local anchor update operation is being performed. After that, a token is placed in `Flag` to indicate that the current VLR is not the anchor VLR. If multiple tokens exist in `Flag`, `SinkFlag` is enabled and only one token remains in `Flag`.
  - If it is an interanchor movement with probability P_{OutA}, transition `OutA` will consume the token immediately, after which a token will be generated in `OutM`, which subsequently disables transition `ME` and enables transition `ServOutM`. To serve the interanchor movement, the HLR is updated to point to the current VLR (e.g., the new anchor), the service context is transferred from the old anchor to the new anchor and the application servers are updated with the new address of the proxy, after which, a token is placed in `RstFlag` to reset the token of `Flag` to 0 using immediate transitions `RstFlag1` and `RstFlag2`. This models the fact that the current VLR is the anchor VLR.
- When the MU sends a service request, a token is placed in `Ss`. The request is serviced by the service proxy colocated with the local anchor:
  - If $mark(\texttt{Flag}) > 0$, then transition `ServNonCvdS` is enabled, which means that the current VLR is not the same as anchor VLR. The request is sent to the service proxy colocated with the local anchor to forward to the server and a server response is sent back through the proxy to the MU. The time to execute these operations is modeled by the time to execute transition `ServNonCvdS`. Note that the service proxy is colocated with the anchor, so there is no extra cost to obtain the MU's current location.
  - If $mark(\texttt{Flag}) = 0$, transition `ServCvdS` is enabled. The service request is sent to the service proxy colocated with the anchor resided in the current VLR. This is modeled by transition `ServCvdS`.

To calculate C_{total} of the dynamic anchor scheme, we introduce additional cost parameters in Table 3 for ease of presentation.

These cost parameters can be calculated as follows:

$$\begin{aligned}
C_{\mathrm{ServInM}} &= \tau_1, \\
C_{\mathrm{ServOutM}} &= T + M_{cs} \times \tau_2 + N_s T, \\
C_{\mathrm{ServCvdC}} &= T, \\
C_{\mathrm{ServNonCvdC}} &= T + \tau_1 + M_{cs} \times \tau_1 + N_s T, \\
C_{\mathrm{ServCvdS}} &= T, \\
C_{\mathrm{ServNonCvdS}} &= \tau_1 + T.
\end{aligned}$$

Suppose N states exist in the underlying Markov model[3] of the Petri net. Let P_i be the steady state probability that the system is found in state i. The average cost to serve location update, call delivery, and service requests can be obtained by assigning "cost" values to these N system states. Specifically, let $C^{da}_{i,call}$ be the search cost assigned to state i given that a search operation is being serviced in state i under the

[2] We could have directly used a finite-state continuous-time Markov chain for performance analysis except that the number of states would be large and the state diagram would be unwieldy. Instead of using a Markov model, we have used a Stochastic Petri Net (SPN) model to provide a concise definition of the corresponding finite-state continuous Markov chain. Tools such as SPNP [18] allow us to automatically generate the underlying finite-state Markov chain corresponding to an SPN model defined for stochastic analysis.

[3] The underlying model of a stochastic Petri net model is a Markov model when all times are exponentially distributed. We can relax the exponential distribution assumption, e.g., using other general distributions for the residence time of the MU staying in a VLR, in which case, the underlying model would be a semi-Markov model. Tools such as SPNP version 6 allow this functionality. For simplicity, we assume all times are exponentially distributed and, thus, the underlying model is only a Markov chain.

TABLE 2
Places and Transitions for the Petri Net Model Shown in Fig. 6

`Cs`	$mark(\texttt{Cs}) = 1$ indicates that a call has just arrived.
`Ms`	$mark(\texttt{Ms}) = 1$ indicates that the MU has just moved across a VLR boundary.
`Ss`	$mark(\texttt{Ss}) = 1$ indicates that a service request has just arrived.
`InMs`	$mark(\texttt{InMs}) = 1$ indicates that an intraanchor movement has just been made.
`OutMs`	$mark(\texttt{OutMs}) = 1$ indicates that an interanchor movement has just been made.
`Flag`	$mark(\texttt{Flag}) > 0$ indicates that the current VLR is different from the anchor, i.e., the MU is located in a VLR that is not the same as the anchor VLR.
`RstFlag`	$mark(\texttt{RstFlag}) = 1$ indicates that an interanchor movement has just been serviced. The `Flag` should be reset.
`CE`	call arrival transition with a rate of λ.
`ME`	VLR boundary crossing transition with a rate of σ.
`SE`	service request transition with a rate of γ.
`ServCvdC`	transition to service a call when the current VLR is the same as the anchor.
`ServNonCvdC`	transition to service a call when the current VLR is different from the anchor.
`InA`	immediate transition with probability P_{InA} that the movement is an intraanchor movement.
`OutA`	immediate transition with probability P_{OutA} that the movement is an interanchor movement.
`ServInM`	transition to service an intraanchor movement.
`ServOutM`	transition to service an interanchor movement.
`SinkFlag`	immediate transition to consume one token from `Flag` if multiple tokens exist in `Flag`. It is used to ensure that at most one token exists in `Flag`. The enabling function is $mark(\texttt{Flag})>1$.
`RstFlag1`	immediate transition to consume the token generated after an interanchor movement is serviced.
`RstFlag2`	immediate transition to reset the `Flag` after an interanchor movement is serviced.
`ServCvdS`	transition to serve a service request when the current VLR is the same as the anchor VLR.
`ServNonCvdS`	transition to service a service request when the current VLR is different from the anchor VLR. The enabling function is $mark(\texttt{Flag})>0$.

TABLE 3
Additional Parameters for Dynamic Anchor

$C_{ServInM}$	the average cost of performing an intraanchor location update operation when the MU changes its VLR within the same anchor area.
$C_{ServOutM}$	the average cost of performing an interanchor location update operation when the MU moves out of the current anchor area.
$C_{ServCvdC}$	the cost to handle a call delivery operation when the current VLR is the same as the anchor VLR, i.e., the cost from the HLR to the anchor, namely, T.
$C_{ServNonCvdC}$	the cost for handling a call delivery operation when the current VLR is different from the anchor VLR, including a cost from the HLR to the anchor (i.e, T), a communication cost (τ_1) from the anchor to the current VLR, a service context transfer cost (i.e. $M_{cs} \times \tau_1$) to migrate the anchor to the current VLR, and a cost (N_sT) to inform all N_s application servers of the address change of the proxy.
$C_{ServCvdS}$	the cost to handle a service request when the anchor resides in the current serving VLR; it is T under the dynamic scheme.
$C_{ServNonCvdS}$	the cost to handle a service request when the anchor is different from the current serving VLR; it is $\tau_1 + T$ under the dynamic scheme.

dynamic anchor scheme. Then, the average search cost under dynamic anchor, C^{da}_{search}, can be calculated as the expected value of $C^{da}_{i,call}$ weighted by the state probability, i.e.,

$$C^{da}_{search} = \sum_{i=1}^{N} P_i \times C^{da}_{i,call},$$

where

$$C^{da}_{i,call} = \begin{cases} C_{\text{ServNonCvdC}} & \text{if } mark(\texttt{Flag}) > 0 \\ C_{\text{ServCvdC}} & \text{Otherwise.} \end{cases}$$

Here, $C^{da}_{i,call}$ is $C_{\text{ServNonCvdC}}$ if, in state i, the current VLR is different from the anchor, i.e., $mark(\texttt{Flag}) > 0$. Otherwise, $C^{da}_{i,call}$ is assigned the value of C_{ServCvdC} to account for the fact that the current VLR is the same as the anchor in state i.

Similarly, let $C^{da}_{i,update}$ and $C^{da}_{i,service}$ be the costs for serving location update and service requests in state i, respectively. We have:

$$C^{da}_{update} = \sum_{i=1}^{N} P_i \times C^{da}_{i,update},$$

$$C^{da}_{service} = \sum_{i=1}^{N} P_i \times C^{da}_{i,service},$$

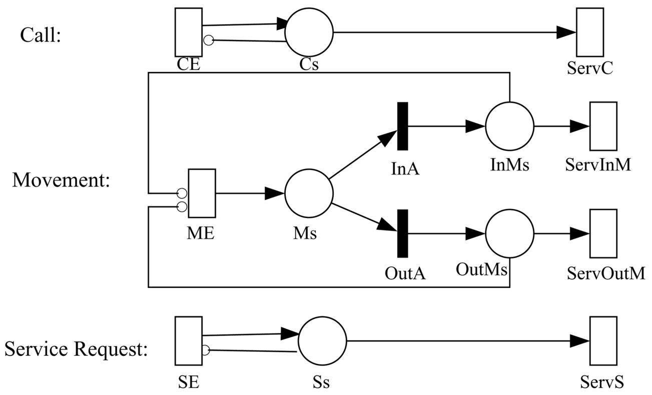

Fig. 7. Petri net model for the static anchor scheme.

where

$$C^{da}_{i,update} = \begin{cases} C_{\text{ServInM}} & \text{if } enabled(\texttt{ServInM}) \\ C_{\text{ServOutM}} & \text{if } enabled(\texttt{ServOutM}) \\ P_{InA} \times C_{\text{ServInM}} + & \text{Otherwise,} \\ P_{OutA} \times C_{\text{ServOutM}} & \end{cases}$$

$$C^{da}_{i,service} = \begin{cases} C_{\text{ServNonCvdS}} & \text{if } mark(\texttt{Flag}) > 0 \\ C_{\text{ServCvdS}} & \text{Otherwise.} \end{cases}$$

Here, enabled(T) means that transition T is enabled. In the first equation above, $C^{da}_{i,update}$ is assigned a value that reflects if the movement is intraanchor or interanchor. If, in state i, the MU has just made an intraanchor movement, transition ServInM would be enabled. Thus, the location update cost in state i would be C_{ServInM}. If the MU has just made an interanchor movement, transition ServOutM would be enabled instead. Thus, the location update cost would be C_{ServOutM}. If, in state i, the MU has not yet made a move, then the location update cost in state i is the average cost weighted on the probability of whether the user's next move is inter or intraanchor, i.e., $P_{InA} \times C_{\text{ServInM}} + P_{OutA} \times C_{\text{ServOutM}}$. In the second equation above, $C^{da}_{i,service}$'s value depends on if the current VLR is different from the anchor VLR in state i. If yes (modeled by $mark(\texttt{Flag}) > 0$ in the Petri net), then the service request cost in state i is $C_{\text{ServNonCvdS}}$. Otherwise, the cost is C_{ServCvdS}.

The total cost per time unit incurred to PCS network under dynamic anchor, C^{da}_{total}, can be calculated by (1), i.e.,

$$C^{da}_{total} = C^{da}_{update} \times \sigma + C^{da}_{search} \times \lambda + C^{da}_{service} \times \gamma. \tag{4}$$

4.5 Static Anchor

In the static anchor scheme, the local anchor and the service proxy remain static in one VLR as long as the MU resides in an anchor area. Its behavior is modeled by an SPN model as shown in Fig. 7. Table 4 lists the meanings of transitions and places in the petri net. Table 5 lists the cost parameters for the static anchor scheme. The major difference between the static anchor model and the dynamic anchor model is that there is no Flag to indicate whether the anchor VLR is located in the current serving VLR because, unlike in the dynamic anchor scheme, the anchor is at a fixed location upon entry to a new anchor area and remains there until the MU departs the anchor area. Therefore, we only need to consider the average cost of accessing the anchor from *any* VLR in the anchor area without having to track if the current VLR is the same as the anchor VLR. Let τ_1 be this average communication cost between the anchor and a VLR in the anchor area as described in Table 1. (Note: We will show how to parameterize this parameter in Section 5.) Then, the cost parameters listed in Table 5 can be calculated as:

$$\begin{aligned} C_{\text{ServInM}} &= \tau_1, \\ C_{\text{ServOutM}} &= T + M_{cs} \times \tau_2 + N_s T, \\ C_{\text{ServC}} &= T + \tau_1, \\ C_{\text{ServS}} &= \tau_1 + T. \end{aligned}$$

By following a similar approach performed for the dynamic anchor scheme, the costs incurred to the PCS system per time unit under the static anchor scheme for serving location update, call delivery, and service requests can be calculated, respectively, as:

$$\begin{aligned} C^{sa}_{search} &= \textstyle\sum_{i=1}^{N} P_i \times C^{sa}_{i,call} = \sum_{i=1}^{N} P_i \times C_{\text{ServC}}, \\ C^{sa}_{update} &= \textstyle\sum_{i=1}^{N} P_i \times C^{sa}_{i,update}, \\ C^{sa}_{service} &= \textstyle\sum_{i=1}^{N} P_i \times C^{sa}_{i,service} = \sum_{i=1}^{N} P_i \times C_{\text{ServS}}, \end{aligned}$$

TABLE 4
Places and Transitions for the Petri Net Model Shown in Fig. 7

Cs	$mark(\texttt{Cs}) = 1$ indicates that a call has just arrived.
Ms	$mark(\texttt{Ms}) = 1$ indicates that the MU has just moved across a VLR boundary.
Ss	$mark(\texttt{Ss}) = 1$ indicates that a service request has just arrived.
InMs	$mark(\texttt{InMs}) = 1$ indicates that an intraanchor movement has just been made.
OutMs	$mark(\texttt{OutMs}) = 1$ indicates that an interanchor movement has just been made.
CE	call arrival transition with a rate of λ.
ME	VLR boundary crossing transition with a rate of σ.
SE	service request transition with a rate of γ.
ServC	transition to service an incoming call.
InA	immediate transition with probability P_{InA} that the movement is an intraanchor movement.
OutA	immediate transition with probability P_{OutA} that the movement is an interanchor movement.
ServInM	transition to service an intraanchor movement.
ServOutM	transition to service an interanchor movement.
ServS	transition to service a service request.

TABLE 5
Additional Parameters for Static Anchor

$C_{ServInM}$	the average cost of performing an intraanchor location update operation when the MU changes its VLR within the same anchor area.
$C_{ServOutM}$	the average cost of performing an interanchor location update operation when the MU moves out of the current anchor area.
C_{ServC}	the cost to handle a call delivery.
C_{ServS}	the cost to handle a service request.

where

$$C^{sa}_{i,update} = \begin{cases} C_{\mathrm{ServInM}} & \text{if } enabled(\texttt{ServInM}) \\ C_{\mathrm{ServOutM}} & \text{if } enabled(\texttt{ServOutM}) \\ P_{InA} \times C_{\mathrm{ServInM}} + \\ P_{OutA} \times C_{\mathrm{ServOutM}} & \text{Otherwise.} \end{cases}$$

Therefore, based on (1), the total cost per time unit incurred to PCS network under static anchor, C^{sa}_{total}, is calculated as:

$$C^{sa}_{total} = C^{sa}_{upsate} \times \sigma + C^{sa}_{search} \times \lambda + C^{sa}_{service} \times \gamma. \tag{5}$$

5 Evaluation

In this section, we first parameterize the performance models developed by means of a hexagonal network coverage model for describing a PCS network to evaluate the performance of the four integrated location and service management schemes proposed so as to identify conditions under which one scheme could perform the best when given a set of parameters characterizing an MU's mobility and service behaviors. Then, we present analytical results with physical interpretation given. We compare integrated versus decoupled location and service management and show that the best integrated scheme outperforms the best decoupled scheme, as well as management schemes that do not use any service proxy, under all conditions. Last, a simulation study is conducted to validate the analytical results.

5.1 Parameterization

We use a hexagonal network coverage model to describe a PCS network where cells are assumed to be hexagonally shaped, with each cell having six neighbors. At the lowest level of Fig. 1, an n-layer VLR covers $3n^2 - 3n + 1$ cells where n is equal to either two or three [11]. Going into the second lowest level of Fig. 1, we can again view each hexagonally-shaped cell as corresponding to a VLR and, therefore, an n-level LSTP will contain $3n^2 - 3n + 1$ VLRs. This view continues as we recursively go up to the higher levels of the PCS network until the RSTP level is reached. For the dynamic and static anchor schemes, we consider an anchor area corresponding to one LSTP area.

For a PCS system described by the hexagonal network coverage model as such, it can be shown that [1] with random movements, the probability that an MU moves within the same anchor area (e.g., same LSTP area), that is, the probability of an intraanchor movement, as the MU moves across a VLR boundary, is given by:

$$P_{InA} = \frac{3n^2 - 5n + 2}{3n^2 - 3n + 1}. \tag{6}$$

Thus, the probability of an interanchor movement, when the MU moves across a VLR boundary, is given by:

$$P_{OutA} = 1 - \frac{3n^2 - 5n + 2}{3n^2 - 3n + 1} = \frac{2n - 1}{3n^2 - 3n + 1}. \tag{7}$$

Without loss of generality, consider $n = 2$ for n-layer VLRs, LSTPs, and RSTPs composing the PCS. Then, the probability P_{InR} that an MU moves within the same RSTP, that is, the probability of an intra-RSTP movement, when the MU moves across a VLR boundary, is given by:

$$P_{InR} = \frac{21n^2 - 27n + 10}{7(3n^2 - 3n + 1)}. \tag{8}$$

Let C_{vl} be the cost of transmitting a message (round trip) between a VLR and its LSTP. Let C_{lr} be the cost of transmitting a message (round trip) between an LSTP and its RSTP. Let C_{pstn} be the communication cost (round trip) to pass through a PSTN. The communication between a VLR and the HLR will traverse through a VLR-LSTP-RSTP-PSTN path sequence. Therefore,

$$T = C_{vl} + C_{lr} + C_{pstn}.$$

For the centralized scheme, there are no additional parameters to parameterize. For the fully distributed scheme, we need to parameterize τ_3 standing for the average communication cost between two neighboring VLRs. With reference to the PCS network shown in Fig. 1, the communication cost between two VLRs within the same LSTP (with probability P_{InA}) is $2C_{vl}$, the communication cost between two VLRs out of the same LSTP but within the same RSTP (with probability $P_{InR} - P_{InA}$) is $2(C_{vl} + C_{lr})$ and the communication cost between two VLRs out of the same RSTP (with probability $1 - P_{InR}$) is $2C_{vl} + 2C_{lr} + C_{pstn}$. Therefore, τ_3 can be parameterized as:

$$\begin{aligned} \tau_3 = {} & 2C_{vl} \times P_{InA} + 2(C_{vl} + C_{lr}) \times (P_{InR} - P_{InA}) \\ & + (2C_{vl} + 2C_{lr} + C_{pstn}) \times (1 - P_{InR}). \end{aligned}$$

For the dynamic anchor scheme, we need to parameterize τ_1 for the average communication cost between the anchor VLR and another VLR (other than the anchor VLR itself) in an anchor area, as well as τ_2 for the average signaling communication cost between two neighboring LSTP areas. τ_1 is equal to the communication cost between two VLRs within the same LSTP. To calculate τ_2, two scenarios are considered: the communication between two VLRs within the same RSTP with cost $2(C_{vl} + C_{lr})$ and the communication between two VLRs out of the same RSTP with cost $2C_{vl} + 2C_{lr} + C_{pstn}$. Thus,

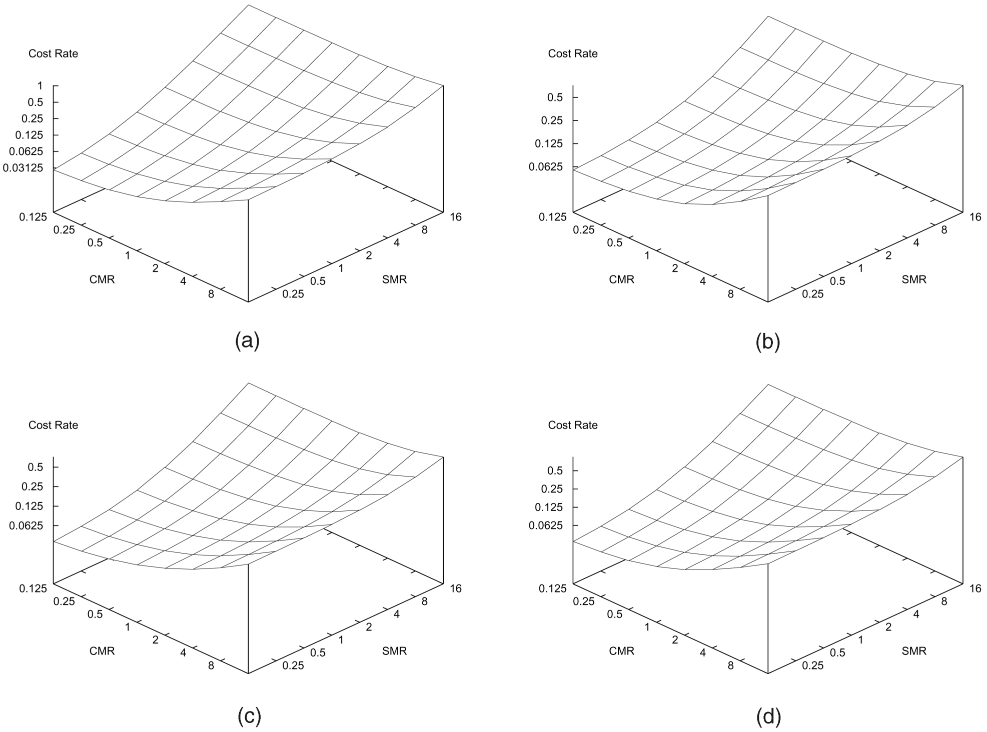

Fig. 8. Cost rate under different CMR and SMR values.

$$
\begin{aligned}
\tau_1 &= 2C_{vl}, \\
\tau_2 &= 2(C_{vl} + C_{lr}) \times \frac{P_{InR} - P_{InA}}{1 - P_{InA}} \\
&\quad + (2C_{vl} + 2C_{lr} + C_{pstn}) \times \frac{1 - P_{InR}}{1 - P_{InA}}.
\end{aligned}
$$

For the static anchor scheme, we need to parameterize τ_1 for the average communication cost between the anchor VLR and *any* VLR (including possibly the static anchor VLR itself) in an anchor area, as well as τ_2 for the average signaling communication cost between two neighboring LSTP areas. Since the static anchor scheme does not track the location of the MU within an anchor area, the MU can reside in each VLR with equal probability. Thus, for a PCS network with $n = 2$, where each LSTP has seven VLRs,[4] we have:

$$
\begin{aligned}
\tau_1 &= 2C_{vl} \times \frac{6}{7} + 0 \times \frac{1}{7} = C_{vl} \times \frac{12}{7}, \\
\tau_2 &= 2(C_{vl} + C_{lr}) \times \frac{P_{InR} - P_{InA}}{1 - P_{InA}} \\
&\quad + (2C_{vl} + 2C_{lr} + C_{pstn}) \times \frac{1 - P_{InR}}{1 - P_{InA}}.
\end{aligned}
$$

5.2 Results

In the section, we present numerical data obtained based on our analysis for a PCS network consisting of two-layer VLRs, LSTPs, RSTPs, and HLR as shown in Fig. 1 modeled by the hexagonal network coverage model as discussed earlier. Fig. 8 summarizes the performances of the centralized, fully distributed, dynamic anchor, and static anchor schemes in the PCS network in terms of the communication cost incurred to the network per time unit as a function of CMR and SMR under identical network signaling-cost conditions in which all costs are normalized with respect to the cost of transmitting a message between a VLR and its LSTP, i.e., $C_{vl} = 1$, such that $C_{lr} = 0.5$ and $C_{pstn} = 6$. The data presented in Fig. 8 were obtained based on (2), (3), (4), and (5) for the case in which there is a single server $N_s = 1$. For the dynamic anchor and static anchor schemes, we used SPNP [18] as a tool to evaluate their respective Petri net models, defined in Figs. 6 and 7, to obtain the data.

Fig. 9 shows the cost incurred to the PCS network per second as a function of the MU's CMR for the four integrated schemes. The X coordinate represents the CMR value in the range of [0.1, 16] with the mobility rate σ fixed at 10/hour while changing the call arrival rate λ. To isolate the effect of CMR, we let SMR=1 and M_{cs}=1 such that the service request rate γ is the same as the mobility rate σ and the average number of packets to transfer the service context is 1. The Y coordinate is the cost rate, i.e., the total cost incurred per second (normalized with respect to the cost of transmitting a message between a VLR and its LSTP) to the network. The cost difference among different integrated schemes presented in Fig. 9 is expressed in terms of cost rate, or "cost per second," so even a cost difference of 0.01 per second (normalized with respect to the cost of transmitting a message between a VLR and its LSTP) is considered significant as the effect of cost difference is cumulative over time.

When the CMR value is low, both the centralized and fully distributed schemes perform worse than the dynamic and static anchor schemes. This is attributed to the fact that the total cost rate is dominated by mobility-related cost factors at low CMR at which the mobility rate is much higher than the call arrival rate. Specifically, the centralized scheme performs badly in this condition because of the high cost of servicing location update operations as these operations need to access the HLR in the centralized scheme. The fully distributed scheme performs badly at low CMR because, with a high mobility rate, the location update cost and the context transfer

[4] If $n = 3$, then each LSTP contains 19 VLRs and τ_2 will need to be calculated accordingly.

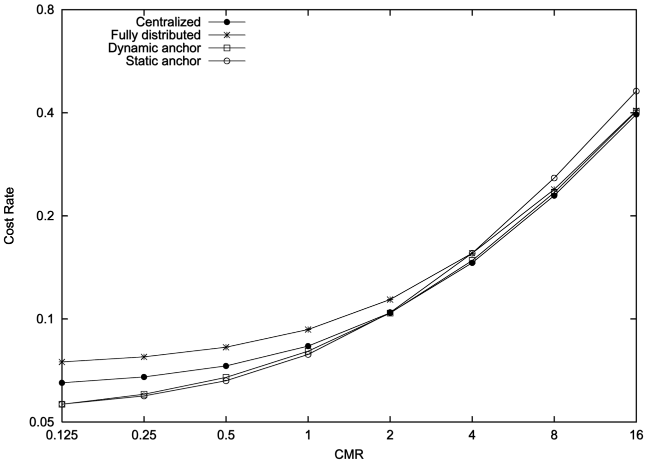

Fig. 9. Cost rate under different Call to Mobility Ratio (CMR) values.

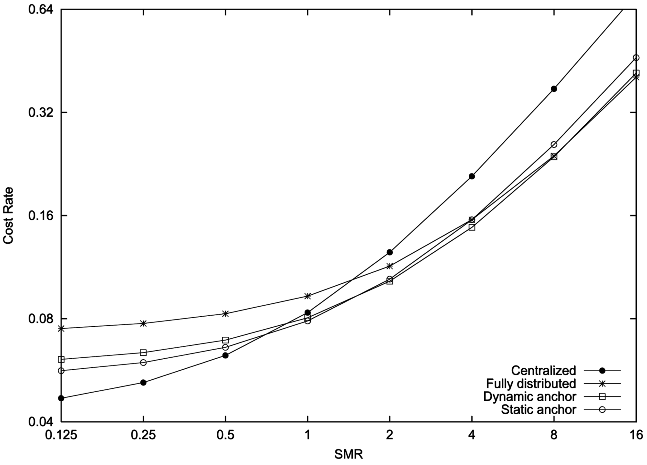

Fig. 10. Cost rate under different Service to Mobility Ratio (SMR) values.

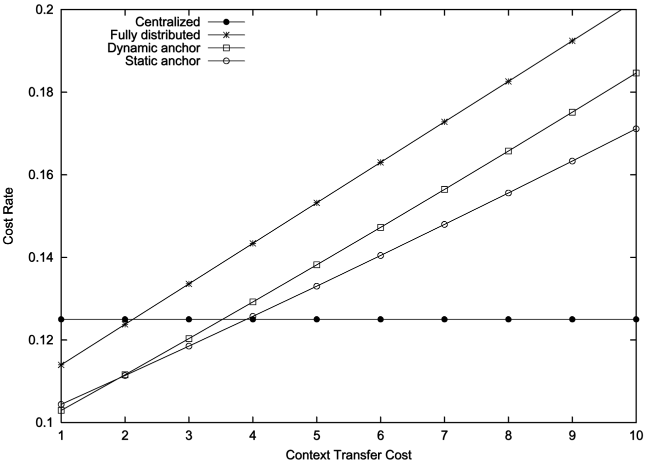

Fig. 11. Cost rate under different context transfer cost values.

cost are high in the fully distributed scheme. On the other hand, the dynamic and static anchor schemes employ an anchor to reduce the location update cost and the context transfer cost, even when the user's mobility rate is high.

As the CMR value increases, the performance of both centralized and fully distributed improves. At very high CMR, the centralized scheme performs the best followed by dynamic anchor over fully distributed and, in the last place, the static anchor scheme. Dynamic anchor performs better than static anchor in this extreme case because, in the dynamic anchor scheme, the anchor colocated with the service proxy is close to the MU. Thus, the cost for service requests and location updates due to movements within an anchor area is low. Another reason is that, when a call arrives and the anchor VLR is not the current serving VLR, the dynamic anchor scheme will update the HLR after the call is serviced and move the anchor to the current VLR. This keeps the HLR database up-to-date and keeps the anchor close to the MU. As a result, it reduces the call delivery cost since the system is able to find the MU quickly on subsequent calls, the effect of which is especially pronounced when CMR is high.

Fig. 10 shows the cost rate as a function of the service request to mobility ratio (SMR) to analyze the effect of the service request rate. Again, we isolate out the effect of SMR by fixing CMR = 1 and M_{cs}=1. Here, by setting CMR=1, we set the calling rate to be the same as the mobility rate fixed at 10/hour. Fig. 10 shows that, as SMR increases, the cost rate under all four schemes increases because, when the mobility rate σ is fixed, increasing SMR increases the service request rate, which, in turn, incurs more service-related costs for all four schemes. At very high SMR, however, the fully distributed scheme performs the best among all, followed by dynamic anchor over static anchor and centralized because, in the fully distributed scheme, the MU's service requests can be serviced quickly by the local service proxy located in the current VLR database, although each service request still unavoidably incurs a communication cost from the service proxy to the server. As the service rate increases while keeping other rates constant, we see that the service request cost dominates other costs, thus making the fully distributed scheme the best scheme at high SMR.

Fig. 11 summarizes the effect of the service context transfer cost on the cost rate. As expected, as the context transfer cost increases, the cost rate under the fully distributed, dynamic anchor, or static anchor scheme all increase, while that for the centralized scheme remains unchanged because there is no service context transfer cost in the centralized scheme. The fully distributed scheme is most sensitive to the increase of the context transfer cost in terms of the increase of the cost rate, followed by dynamic anchor and static anchor. This order corresponds to the context transfer frequency under various schemes. At one end of the spectrum, the fully distributed scheme must transfer the service context with the migrated service proxy whenever the MU moves across a VLR boundary. The dynamic anchor scheme transfers the service context when the MU moves across an anchor boundary, or after a call delivery operation is serviced if the anchor VLR is not the same as the current VLR. In the static anchor scheme, the service context is transferred only when the user moves across an anchor boundary. At the other end of the spectrum, the centralized scheme is entirely insensitive to the increase of the service context transfer cost because the service proxy is colocated with the HLR, which requires no service context transfer.

5.3 Integrated versus Decoupled Location and Service Management

To demonstrate the viability of the integrated location and management scheme, we have conducted a performance study to compare integrated against decoupled location and service management for which location management is decoupled from service management. By decoupling, the MU's service proxy is not colocated with the MU's location database and the MU's location registration areas are decoupled from the MU's service areas. Three location

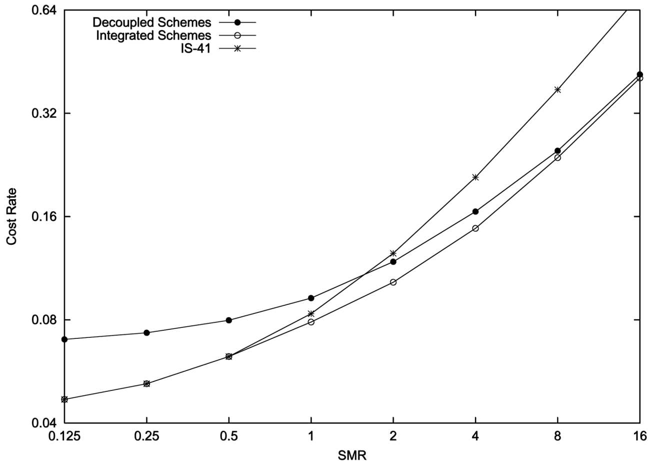

Fig. 12. Integrated versus decoupled location and service management: best cost rate under different SMR values.

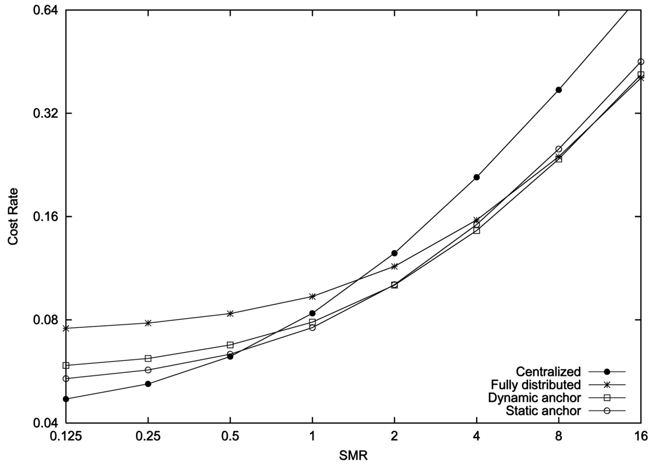

Fig. 13. Simulation results: cost rate under different SMR values.

management schemes are feasible, namely, fully distributed (corresponding to basic HLR/VLR), dynamic anchor, and static anchor. The centralized scheme is not feasible because it is meaningless to put a regional location database colocated with the HLR also pointing to the current VLR. For service management, again fully distributed, static anchor and dynamic scheme are feasible. The centralized scheme is not feasible because it would place the service proxy at a fixed location (not at the HLR), so the communication cost for servicing user requests would be excessively high. The dynamic anchor scheme for service management here refers to the feature that the anchor can change its location to the current serving VLR within the anchor area after serving a user request. Therefore, there are nine possible combinations (i.e., three for location management and three for service management) through which decoupled location and service management can be applied. For fair comparison, we only compare the best cost rate achievable by both schemes, i.e., for the decoupled scheme, the best combination out of the nine selections is used, and, for the integrated scheme, the best out of four is used when given an MU's mobility and service characteristics. Fig. 12 (corresponding to Fig. 10) compares integrated versus decoupled at various SMR values. As a baseline, Fig. 12 also shows a cost curve for the basic HLR/VLR scheme that does not use a service proxy as now in IS-41 and GSM, for which the major cost is in updating and querying the MU's location at the HLR. Fig. 12 demonstrates the superiority of integrated over decoupled schemes and the basic HLR/VLR scheme. We attribute the superiority of the integrated scheme to the fact that the service proxy *knows* the MU's location at all times through integration of location and service management. The superiority of the integrated scheme over the decoupled scheme is especially pronounced when SMR is low, at which the service proxy in the decoupled scheme has to explicitly *track* the MU's location which incurs extra costs. On the other hand, the integrated scheme outperforms the basic scheme significantly, especially at high SMR at which the server in the basic scheme has to query the HLR to know the MU's location and then communicate with the mobile user for data delivery.

5.4 Simulation Validation

We have conducted a simulation study using a discrete event simulation language called SMPL (Simulation Model Programming Language) [13] to validate the analytical results calculated from (2), (3), (4), and (5). The simulation environment consists of a large two-layer RSTP area covering seven LSTPs, each corresponding to an anchor area (for the dynamic and static anchor schemes) that, in turn, covers seven hexagonally-shaped VLRs. The center of the RSTP is at $(0, 0)$. Each hexagonal VLR area is represented by its center location (x, y). An MU is characterized by its own mobility and service behaviors, with the mobility rate of σ and service rate of γ. The MU can move from the current VLR to one of the six neighbor VLRs randomly. If the MU moves out of the simulated RSTP area, its location will be circled to the other side of the simulated area, i.e., its location will be changed from (x, y) to $(-x, -y)$, thus allowing the simulated RSTP area to be reused. At all times, the location of the MU is known. The service proxy moves according to the specific integrated scheme considered. As the simulation program knows the locations of the MU and its service proxy all the time, whenever a location or service management event occurs, such as a call, a move to another VLR, or a service request, it knows exactly the cost incurred in response to the event. These per-event costs are then accumulated to the overall cost during the course of the simulation. At the end of each simulation batch run, the average "cost rate" is computed by dividing the cumulative cost over the simulation period.

To ensure statistical significance of simulation results, a *batch mean analysis* technique has been adopted by which the simulation period is divided into batch runs with each batch consisting of 2,000 "cost rate" observations for computing an average value. A minimum of 10 batches was run to compute a grand mean of the cost rate value. Additional batches were added if necessary until the mean cost rate value is within 95 percent confidence level and 10 percent accuracy from the true mean.

The simulation results show very good correlations with analytical results. For example, Fig. 13 shows the simulation results for the cost rate as a function of the MU's SMR, corresponding to Fig. 10 for analytical results. We see that Figs. 13 and 10 are virtually identical despite the fact that simulation results are obtained based on the cumulative cost over the simulation period in response to mobility, call, and service events divided by the simulation period, while analytical results are obtained based on the average cost rate as calculated by (2), (3), (4), and (5) following the parameterization process. We conclude that the analytical results are valid and there exists an optimal integrated scheme for integrated location and service management on a per-user basis.

6 Conclusion

In this paper, we investigated the concept of integrated location and service management with the objective to reduce the overall communication cost for servicing mobility-related and service-related operations by the integrated PCS network environment. The central concept lies in the design of "colocating" the MU's service proxy with the MU's location databases such that the application server can take full advantage of the location services provided by the "colocated" location database. We investigated and analyzed several possible integrated location and service management schemes by means of Petri net models and identified conditions under which one scheme may perform better than others. The analysis results are useful for identifying the best scheme to be adopted to provide personalized services to individual users based on their user profiles. Our analysis result shows that the dynamic anchor scheme performs the best in most conditions except when the context transfer cost is high (when the server is heavy). The centralized scheme performs the best at low SMR and high CMR. The fully distributed scheme performs the best at high SMR and high CMR. The static anchor scheme is a relatively stable scheme, performing reasonably well under a wide range of parameter values examined in the paper. These results suggest that different users with vastly different mobility patterns should adopt different integrated location and service management methods to optimize system performance.

Acknowledgments

This work was supported in part by US National Science Foundation grant #9987586, a Microsoft Research grant, and an Intel grant.

References

[1] I.R. Chen, T.M. Chen, and C. Lee, "Performance Evaluation of Forwarding Strategies for Location Management in Mobile Networks," *The Computer J.*, vol. 41, no. 4, pp. 243-253, 1998.

[2] I.R. Chen and B. Gu, "Quantitative Analysis of a Hybrid Replication with Forwarding Strategy for Efficient and Uniform Location Management in Mobile Wireless Networks," *IEEE Trans. Mobile Computing*, vol. 2, no. 1, pp. 3-15, 2003.

[3] M.H. Dunham and V. Kumar, "Impact of Mobility on Transaction Management," *Proc. Int'l Workshop Data Eng. for Wireless and Mobile Access (MobiDE '99)*, pp. 14-21, 1999.

[4] EIA/TIA, *Cellular Radio Telecommunication Inter System Operations*, Technical Report IS-41 (Revision B), EIA/TIA, July 1991.

[5] M. Endler, D.M. Silva, and K. Okuda, "RDP: A Result Delivery Protocol for Mobile Computing," *Proc. Int'l Workshop Wireless Networks and Mobile Computing (WNMC) at the 20th Int'l Conf. Distributed Computing Systems (ICDCS)*, 2000.

[6] Y. Fan, "General Modeling and Performance Analysis for Location Management in Wireless Mobile Networks," *IEEE Trans. Computers*, vol. 51, no. 10, pp. 1169-1181, Oct. 2002.

[7] J.S.M. Ho and I.F. Akyildiz, "Local Anchor Scheme for Reducing Signaling Costs in Personal Communications Networks," *IEEE/ACM Trans. Networking*, vol. 4, no. 5, pp. 709-725, Oct. 1996.

[8] R. Jain and N. Krishnakumar, "Network Support for Personal Information Services to PCS Users," *Proc. IEEE Conf. Networks for Personal Comm.*, pp. 1-7, Mar. 1994.

[9] R. Jain, Y.B. Lin, C. Lo, and S. Mohan, "A Forwarding Strategy to Reduce Network Impacts of PCS," *Proc. 14th Ann. Joint Conf. IEEE Computer and Comm. Socs.*, vol. 2, pp. 481-489, 1995.

[10] N. Krishnakumar and R. Jain, "Escrow Techniques for Mobile Sales and Inventory Applications," *ACM Wireless Networks*, vol. 3, no. 3, pp. 235-246, 1997.

[11] W.R. Lai and Y.B. Lin, "Mobility Database Planning for PCS," *Proc. 1996 Workshop Distributed System Technologies and Applications*, pp. 263-269, 1996.

[12] W. Ma and Y. Fang, "Two-Level Pointer Forwarding Strategy for Location Management in PCS Networks," *IEEE Trans. Mobile Computing*, vol. 1, no. 1, pp. 32-45, Jan.-Mar. 2002.

[13] M.H. MacDougall, *Simulating Computer Systems*. MIT Press, 1987.

[14] W.C. Peng and M.S. Chen, "A Dynamic and Adaptive Cache Retrieval Scheme for Mobile Computing Systems," *Proc. IFCIS Int'l Conf. Cooperative Information Systems*, pp. 251-258, Aug. 1998.

[15] C.E. Perkins, "Mobile IP," *IEEE Comm. Magazine*, pp. 84-99, 1997.

[16] M. Roussopoulos et al., "Personal-Level Routing in the Mobile People Architecture," *Proc. USENIX Symp. Internet Technologies and Systems*, pp. 165-176, Oct. 1999.

[17] M.M. Bin-Tariq and A. Takeshita, "Management of Cacheable Streaming Multimedia Content in Networks with Mobile Hosts," *Proc. IEEE Global Telecomm. Conf.*, vol. 3, pp. 2245-2249, Nov. 2002.

[18] K.S. Trivedi, G. Ciardo, and J. Muppala, *SPNP Version 6 User Manual*, Dept. of Electrical Eng., Duke Univ., Durham, N.C., 1999.

[19] J. Xie and I.F. Akyildiz, "A Novel Distributed Dynamic Location Management Scheme for Minimizing Signaling Costs in Mobile IP," *IEEE Trans. Mobile Computing*, vol. 1, no. 3, pp. 163-175, July-Sept. 2002.

[20] T. Yoshimura, Y. Yonemoto, T. Ohya, M. Etoh, and S. Wee, "Mobile Streaming Media CDN Enabled by Dynamic SMIL," *Proc. 11th Int'l World Wide Web Conf.*, pp. 651-661, May 2002.

[21] A. Zimmermann, "User Manual 3.0," *TimeNET: A Software Tool for the Performability Evaluation with Stochastic Petri Nets*, TU Berlin, 2001.

Ing-Ray Chen received the BS degree from the National Taiwan University, Taipei, Taiwan, and the MS and PhD degrees in computer science from the University of Houston, University Park, Houston, Texas. He is a professor in the Department of Computer Science at Virginia Tech. His research interests include mobile computing, pervasive computing, multimedia, distributed systems, real-time intelligent systems, and reliability and performance analysis. He has served on the program committees of numerous conferences and currently serves on the editorial board of *Wireless Personal Communications* and as associate editor for *The Computer Journal* and *International Journal on Artificial Intelligence Tools.* He is a member of the IEEE Computer Society and the ACM.

Baoshan Gu received the BS degree from the University of Science and Technology of China, Hefei, China, in 1992 and the MS degree in computer science from the Institute of Computing Technology, Chinese Academia of Science, Beijing, China, in 1995. From 1995 to 2000, he was a research and development engineer at the Institute of Computing Technology, Chinese Academia of Science. He is currently pursuing the PhD degree in the Department of Computer Science, Virginia Tech, where he is a research assistant in the Systems and Software Engineering Laboratory. His research interests include next-generation wireless system architectures, design and evaluation of location and service management schemes in mobile computing environments, and mobile multimedia systems.

Sheng-Tzong Cheng received the BS (1985) and MS (1987) degrees in electrical engineering from the National Taiwan University, Taipei, Taiwan. He received the MS (1993) and PhD (1995) degrees in computer science from the University of Maryland, College Park. He was an assistant professor of computer science and information engineering at the National Dong Hwa University, Hualien, Taiwan, in 1995 and became an associate professor in 1996. He is a professor in the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan. His research interests are in design and performance analysis of mobile computing, wireless communications, multimedia, and real-time systems.