# Proxy-Based Wireless Data Access Algorithms in Mobile Hotspots

Sangheon Pack, *Member, IEEE*, Humphrey Rutagemwa, *Member, IEEE*, Xuemin Shen, *Senior Member, IEEE*, Jon W. Mark, *Life Fellow, IEEE*, and Kunwoo Park

***Abstract*—In this paper, we investigate efficient wireless data access algorithms in mobile hotspots. We introduce a proxy cache (PC) and propose PC-based poll-each-read (P-PER) and PC-based callback (P-CB) data access algorithms to reduce the transmission cost over wireless links in mobile hotspots. An analytical model is developed, and extensive simulations are conducted to demonstrate the performance of P-PER and P-CB. It is shown that P-PER and P-CB can improve cache hit performance and significantly reduce transmission cost. A tradeoff between P-PER and P-CB suggests the need to use a hybrid proxy-based approach to attain optimal performance of wireless data access in mobile hotspots.**

***Index Terms*—Callback (CB), mobile hotspots, poll-each-read (PER), proxy cache (PC)-based CB (P-CB), proxy cache (PC)-based PER (P-PER), wireless data access.**

## I. Introduction

MOBILE hotspots are novel service platforms that extend Wi-Fi hotpot services to moving vehicles (e.g., subways, buses, trains, vessels, and airplanes) [1], [2]. Mobile hotspots are perceived as important service concepts for realizing ubiquitous computing, and they pose many research challenges on various issues: mobility management [3], [4], quality-of-service (QoS) support [5], link-layer transmission techniques [6], [7], gateway architectures [8], [9], and testbed implementation [10], [11]. Fig. 1 shows a network scenario for emerging applications in mobile hotspots [12]. The mobile terminal (MT) can access the application server (AS) through the access point (AP) and the base station (BS). The first wireless hop from the MT is the wireless local area network (WLAN) link, and the second wireless hop is the wireless wide area network (WWAN) link.

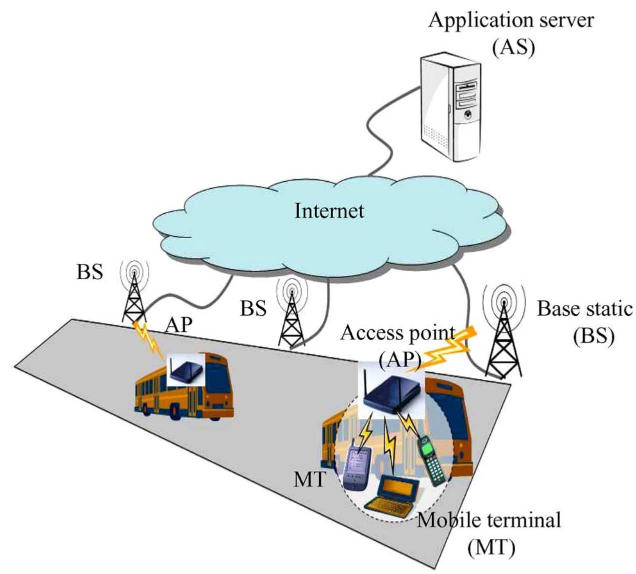



Fig. 1. Wireless data access in mobile hotspots.

Within a moving *platform*, which will be referred to as *vehicle* throughout this paper, a WLAN is used to connect a number of MTs to an AP. At the same time, a WWAN is employed for the connection between the AP and the BS, which is, in turn, connected to the AS through a wireline link. The WLAN–WWAN integrated link provides several advantages in mobility and resource management for data access in mobile hotspots.

Data caching is one of the promising techniques that can be used to enhance the latency and throughput performance of wireless data access applications running over mobile hotspots. The use of data caching requires the identification of the nodes where data can be cached, the appropriate cache replacement policy, and the degree of cache consistency. Although the data object can be cached at any node along the data path (e.g., the MT, AP, and BS in Fig. 1), only nodes that can significantly improve the performance should be considered. Usually, all nodes in a data path have limited memory to store a cached data object. Therefore, an efficient policy is required to determine the cached data object, which can be overwritten when the reserved memory space allocated for data caching is full. The cached data object may be required to maintain strong or weak consistency, depending on the nature of data applications. For weak consistency, a stale copy of the data may be returned to the user, whereas for strong consistency, the consistency between cached and original copies is always enforced, and no stale copy of the modified data is allowed to be used by the user. As shown

Manuscript received September 2, 2007; revised December 15, 2007 and January 2, 2008. This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada under Strategic Grant STPGP 257682 and in part by the IT R&D Program of the Ministry of Information and Communication/Institute of Information Technology Assessment under Project 2007-F-038-01 (Fundamental Technologies for the Future Internet). The review of this paper was coordinated by Dr. J. Misic.

S. Pack is with the School of Electrical Engineering, Korea University, Seoul 136-701, Korea (e-mail: shpack@korea.ac.kr).

H. Rutagemwa is with the Communications Research Centre, Ottawa, ON K2H 8S2, Canada (e-mail: humphrey.rutagemwa@crc.ca).

X. Shen and J. W. Mark are with the Centre for Wireless Communications, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: xshen@bbcr.uwaterloo.ca; jwmark@bbcr.uwaterloo.ca).

K. Park is with the School of Computer Science and Engineering, Seoul National University, Seoul 151-742, Korea (e-mail: kwpark@mmlab.snu.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2008.917238

in Fig. 1, since data access applications in mobile hotspots run over a different network model, new data caching algorithms need to be established and thoroughly studied.

In this paper, we consider wireless data access applications that require a strongly consistent data cache in mobile hotspots. Different types of applications such as the mobile address book, news report, and stock information belonged to these applications [13], [16]. We introduce a proxy cache (PC) and propose two enhanced strongly consistent data access algorithms: PC-based poll-each-read (P-PER) and PC-based callback (P-CB). Through extensive simulations, we evaluate the performance of P-PER and P-CB in terms of the cache hit probability and the transmission cost. In addition, the effects of the cache size, access pattern, and data size are investigated. Our major contributions in this paper are twofold: 1) We propose two proxy-based wireless data access algorithms in mobile hotspots; and 2) we develop an analytical model to evaluate the performance of strongly consistent wireless data access algorithms and carry out extensive simulations to validate the analytical model. To the best of our knowledge, this is the first research in the open literature that focuses on the strongly consistent wireless data access algorithms in mobile hotspots.

The remainder of this paper is organized as follows: Overviews of two strongly consistent data access algorithms, namely PER and CB, are given in Section II. The system model and proxy-based data access algorithms are proposed in Sections III and IV, respectively. Performance analysis of the algorithms is presented in Section V, and extensive simulation results are given in Section VI, followed by concluding remarks in Section VII.

## II. Strongly Consistent Data Access Algorithms

PER [14] and CB [15] are two strongly consistent wireless data access algorithms reported in the literature. These algorithms have been analyzed in [16] and [17] and extended in [18] and [19]. In this section, we first introduce the basic terminologies and then describe the operations of PER and CB, where an MT with cache capability accesses data objects in an AS.

Let $O_i$ be the $i$th data object. $O_i$ is associated with a time sequence number $t$ ($t > 0$), which is assigned in an increasing manner (i.e., $O_i$ with $t + \Delta$ ($\Delta > 0$) is a more recent data object than $O_i$ with $t$). Each MT is identified by an identifier $j$, and a vehicle has a group identifier $k$. For instance, MT $j$ that resides in vehicle $k$ can be uniquely identified by $(j, k)$. Four terminologies are defined for wireless data access algorithms.

1) $\mathbf{Access}(i, t)$: This message requests an access of object $i$. For PER, $t > 0$ specifies the current sequence number for a cached object, whereas $t = 0$ indicates that there is no object in the cache. For CB, invalidation is always performed by the AS, and therefore, $t$ is always set to zero.
2) $\mathbf{Send}(i, t, F)$: This message is used to send a data object or to confirm $\mathbf{Access}(i, t)$ in PER. $i$ and $t$ denote the object index and the sequence number, respectively. $F$ is a flag indicating whether the data object is included in the message or not. If the object $i$ is transmitted with this message, $F$ is set to one. On the other hand, if only a confirmation message is sent, $F$ is set to zero.

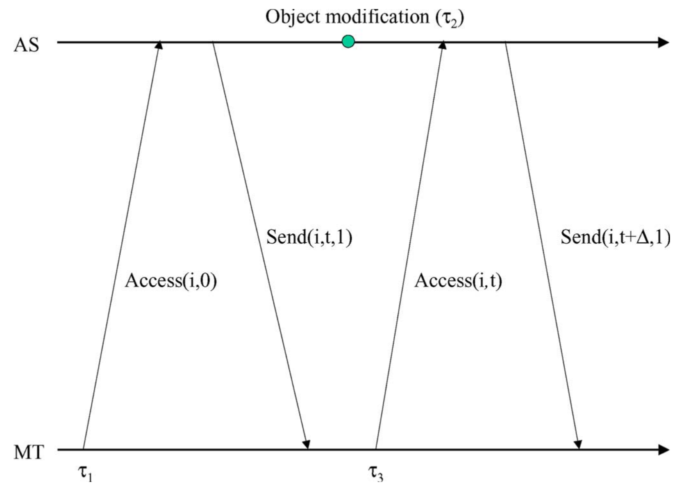



Fig. 2. PER operation.

3) $\mathbf{Update}(i, j, k)$: This message invalidates the object $i$ in the cache of MT $j$ located in vehicle $k$.
4) $\mathbf{Ack}(i, j, k)$: This message acknowledges the receipt of $\mathbf{Update}(i, j, k)$.

### A. PER

In the PER algorithm, whenever an MT accesses an object, it first polls the AS to check the validity of its cached object. If the cached object is up to date, the AS sends a confirmation message without any data object. Otherwise, the AS sends a reply message with the updated data object to the MT. The message flow of the PER algorithm is illustrated in Fig. 2. Let $\tau_i$ denote the time instant when an event occurs, and let $\tau_i < \tau_{i+1}$. At time $\tau_1$, an MT first tries to access an object $O_i$. Since this is the first time that the MT tries to access the object $O_i$ (and, therefore, it does not have any cached version of the object), the MT sends the message $\mathbf{Access}(i, 0)$ to the AS. The AS responds with $\mathbf{Send}(i, t, 1)$, where the data object $O_i$ is included. At time $\tau_2$, $O_i$ is modified, and therefore, the cached object becomes invalid. At time $\tau_3$, a data access to the object $O_i$ is requested, and the MT sends $\mathbf{Access}(i, t)$ to check the validity of its cached object. Then, the AS returns $\mathbf{Send}(i, t + \Delta, 1)$, where $\Delta > 0$, with the data object ($F$ is set to one) to the MT, and the MT updates its local cache.

### B. CB

Unlike PER, the CB algorithm satisfies the strong consistency by invalidation procedures. When an object in the AS is modified, the MTs with the corresponding cached objects are notified, and the cached objects are invalidated. When an MT wants to access an object in the AS, it first checks the availability of a cached object. If a cached object exists, the MT uses the cached object with no transmission cost. Otherwise, the MT contacts the AS to obtain the updated object. Fig. 3 illustrates the CB operations. After the initial access to $O_i$ at time $\tau_1$, the MT maintains the up-to-date $O_i$ in its cache. Therefore, no transmission cost incurs when the cached data object is accessed at time $\tau_2$. When the object $O_i$ is updated

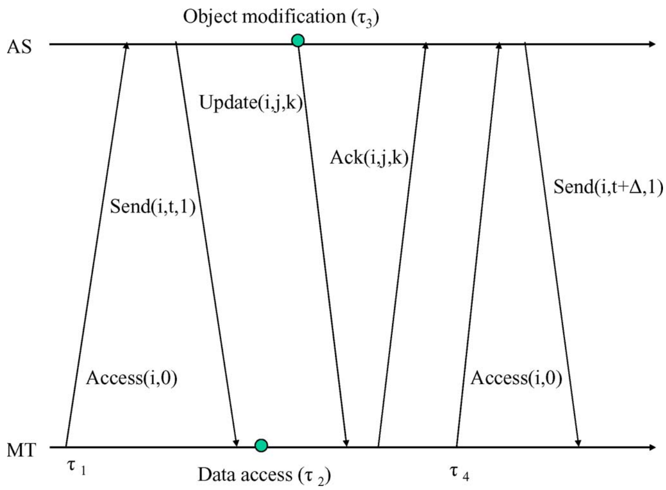



Fig. 3. CB operation.

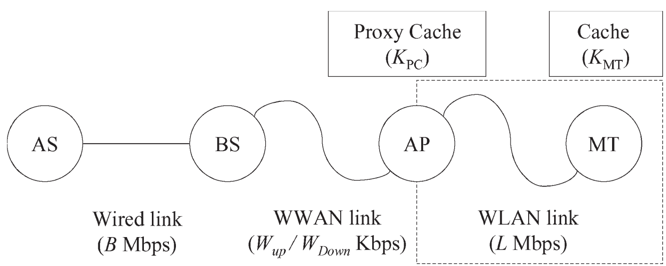



Fig. 4. System model for wireless data access in mobile hotspots.

at time $\tau_3$, the AS transmits $\mathbf{Update}(i, j, k)$ to invalidate the cached object, and then, the MT sends an acknowledgement message $\mathbf{Ack}(i, j, k)$ to the AS. When the MT accesses $O_i$ after the invalidation (i.e., at time $\tau_4$), the MT contacts the AS to get the updated $O_i$, and then, the AS responds with $\mathbf{Send}(i, t + \Delta, 1)$.

## III. System Model

Fig. 4 shows the system model for the proxy-based wireless data access applications in mobile hotspots. An MT connects to an AS through the WLAN–WWAN integrated link. The WLAN supports a higher data rate than the WWAN, but it has a smaller service coverage area than the WWAN. Consequently, the WLAN is used to connect a number of MTs to the AP, whereas the WWAN is used to connect the AP and the BS. With the integrated wireless link, the WWAN provides an extended service coverage to the mobile vehicle, and the WLAN can accommodate more users without excessive usage of the resource in the WWAN. The wireline link is used to connect the BS to the Internet. We assume that data access applications are running over reliable transport and/or data link protocols. Therefore, at the application layer, the WWAN and WLAN links can apparently be characterized as having negligible packet losses. In addition, the time interval between two events that access the same object is relatively large compared to the channel delay variations so that it is reasonable to further consider the WWAN and WLAN links with fixed but arbitrary bandwidths. Consequently, the wireline and WLAN links are considered to have symmetrical bandwidths of $B$ and $L$ (in megabits per second), respectively. On the other hand, the WWAN link is considered to have asymmetrical links, where the uplink and downlink bandwidths are $W_{\mathrm{Up}}$ and $W_{\mathrm{Down}}$ (in kilobits per second), respectively.

We assume that all modifications to objects are only made by the AS. The MT has a cache with a limited size $K_{\mathrm{MT}}$. The AP is installed in a vehicle (e.g., bus or train), and the PC of size $K_{\mathrm{PC}}$ is colocated with the AP. Therefore, the PC can reduce the number of accesses over the WWAN link, which is a bottleneck link in a mobile hotspot. The PC is shared by multiple MTs, and thus, $K_{\mathrm{PC}}$ is much larger than $K_{\mathrm{MT}}$. Throughout this paper, we only consider wireless data access applications where MTs in a vehicle access data objects in the AS (see Fig. 1). However, this system model can be easily extended to vehicular ad hoc networks or vehicle-to-vehicle communications [20]. In these situations, a group of vehicles forms a WLAN, and a PC can be installed at an anchor vehicle to maintain connectivity with the BS.

## IV. Proxy-Based Wireless Data Access Algorithms

From the system model, the WWAN link is a potential bottleneck due to its limited bandwidth. Therefore, the PC is installed at the AP to reduce the transmission cost over the WWAN link, which allows a two-tier caching architecture. At the first tier, the MT and the PC act as a client and a server, respectively, and the cache in the MT is utilized. At the second tier, the PC plays the role of a client, whereas the AS acts as a server. In the following sections, we describe the operations of the proposed P-PER and P-CB algorithms and then present the cache replacement schemes in P-PER and P-CB.

### *A. P-PER*

To describe the operations of P-PER, we consider four possible cases: 1) when there are no cached objects in both the MT and the PC; 2) when there is no cached object in the MT, but there is one in the PC; 3) when there is a cached object in the MT but not in the PC; and 4) when there are cached objects in both the MT and the PC.

*1) When there are no cached objects in both the MT and the PC:* Fig. 5 shows the P-PER operation. The MT first sends $\mathbf{Access}(i, 0)$ to the PC. Since the PC also does not have any cached object for $O_i$, it relays $\mathbf{Access}(i, 0)$ to the AS. After that, the AS delivers the up-to-date $O_i$ via $\mathbf{Send}(i, t, 1)$, and the PC and the MT maintain the object in their caches.

*2) When there is no cached object in the MT, but there is one in the PC:* In Fig. 6, the MT has no cached object for $O_i$; thus, it sends $\mathbf{Access}(i, 0)$ to the PC. Assume that the PC has a cached object and its time sequence is $t_{\mathrm{PC}}$. After receiving $\mathbf{Access}(i, 0)$ from the MT, the PC converts the message into $\mathbf{Access}(i, t_{\mathrm{PC}})$ and transmits $\mathbf{Access}(i, t_{\mathrm{PC}})$ to the AS. If there is no modification to $O_i$ after $t_{\mathrm{PC}}$, $\mathbf{Send}(i, t_{\mathrm{PC}}, 0)$ is returned; otherwise, $\mathbf{Send}(i, t_{\mathrm{PC}} + \Delta, 1)$ is sent to the PC. When the PC receives $\mathbf{Send}(i, t_{\mathrm{PC}}, 0)$, it is confirmed that the PC's cached object is the latest object. Therefore, no transmission cost for the data object occurs between the AS and the PC, and

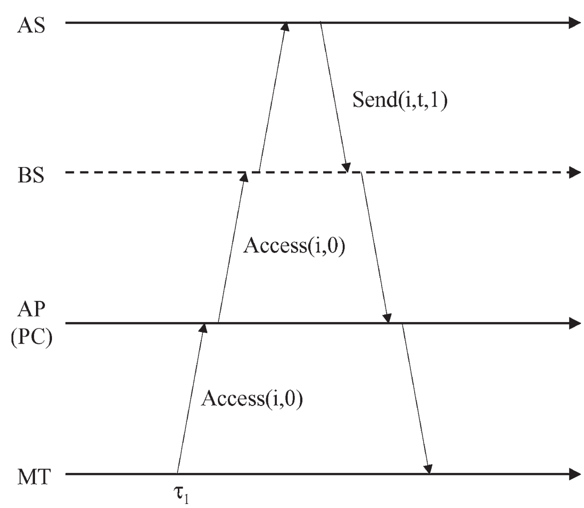



Fig. 5. P-PER operation. There are no cached objects both in the MT and the PC.

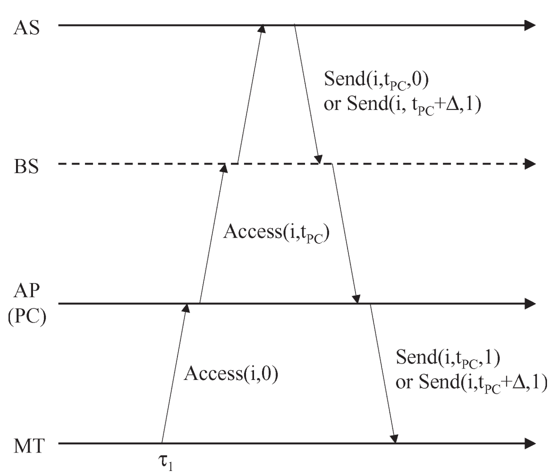



Fig. 6. P-PER operation. There is no cached object in the MT, but there is one in the PC.

the PC delivers its cached data object to the MT. On the other hand, if the PC receives $\mathbf{Send}(i, t_{\rm PC} + \Delta, 1)$, the PC updates its cache as the new object and relays it to the MT.

*3) When there is a cached object in the MT but not in the PC:* As mentioned before, the PC is shared by multiple MTs. Therefore, a cached object in the PC can be replaced by accesses of other objects. In this case, only the MT has a cached object. Fig. 7 shows the P-PER operation for this situation. The MT sends $\mathbf{Access}(i, t)$ to the PC, and then, the PC performs a conversion of the message. This conversion is required because the PC should cache the up-to-date $O_i$ by accessing the AS. If $\mathbf{Access}(i, t)$ is sent to the AS without conversion and no modification occurs after $t$, the PC cannot cache the recent $O_i$. Therefore, the PC sends $\mathbf{Access}(i, 0)$ instead of $\mathbf{Access}(i, t)$. When the AS receives $\mathbf{Access}(i, 0)$, it responds with $\mathbf{Send}(i, t, 1)$ or $\mathbf{Send}(i, t + \Delta, 1)$. For both cases, the PC caches the received data object. On the other hand, the reception of $\mathbf{Send}(i, t, 1)$ represents that the MT has the up-to-date object, and hence, the PC sends $\mathbf{Send}(i, t, 0)$ to the MT, which saves the transmission cost between the AP and the MT. For $\mathbf{Send}(i, t + \Delta, 1)$, since there is a modification to $O_i$ after the MT's

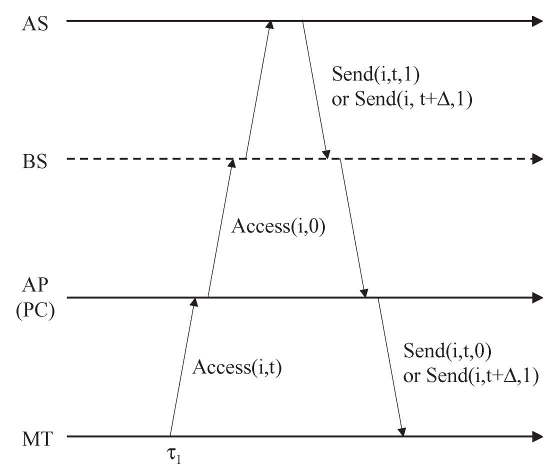



Fig. 7. P-PER operation. There is a cached object in the MT but not in the PC.

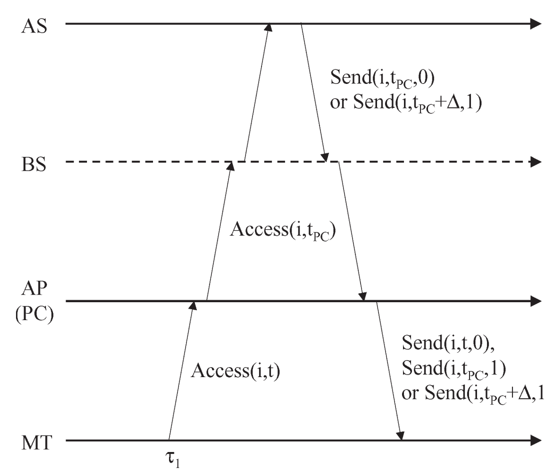



Fig. 8. P-PER operation. There are cached objects in both the MT and the PC.

data access, the PC delivers the recent data object $O_i$ via $\mathbf{Send}(i, t + \Delta, 1)$.

*4) When there are cached objects in both the MT and the PC:* Fig. 8 indicates the P-PER operation when both the MT and the PC have cached objects for $O_i$. Let $t_{\rm PC}$ and $t$ be the time sequences of the cached object in the PC and the MT, respectively. Intuitively, $t_{\rm PC}$ is always equal to or larger than $t$ so that the PC should check whether $t_{\rm PC}$ is the latest time sequence of $O_i$. Therefore, when the PC receives $\mathbf{Access}(i, t)$ from the MT, it converts $\mathbf{Access}(i, t)$ into $\mathbf{Access}(i, t_{\rm PC})$ and delivers $\mathbf{Access}(i, t_{\rm PC})$ to the AS. If $O_i$ is modified after $t_{\rm PC}$, $\mathbf{Send}(i, t_{\rm PC} + \Delta, 1)$ will be returned; otherwise, $\mathbf{Send}(i, t_{\rm PC}, 0)$ is sent to the PC. If the PC receives $\mathbf{Send}(i, t_{\rm PC} + \Delta, 1)$, it caches the latest $O_i$ and then relays the data object to the MT. On the other hand, if the PC receives $\mathbf{Send}(i, t_{\rm PC}, 0)$ and $t_{\rm PC} = t$, the PC does not need to deliver the data object to the MT. Hence, $\mathbf{Send}(i, t, 0)$ is sent to the MT, and then, the MT uses its cached object for its data access. However, if $t_{\rm PC} > t$ and $\mathbf{Send}(i, t_{\rm PC}, 0)$ is received, the PC delivers the up-to-date $O_i$ from its cache to the MT using $\mathbf{Send}(i, t_{\rm PC}, 1)$.

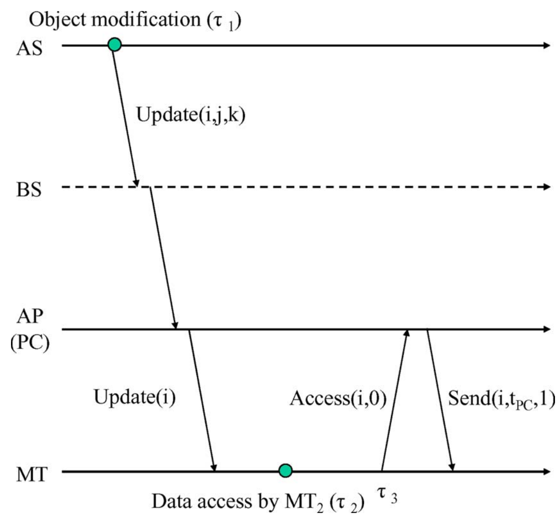



Fig. 9. P-CB operation. PC has a cached object.

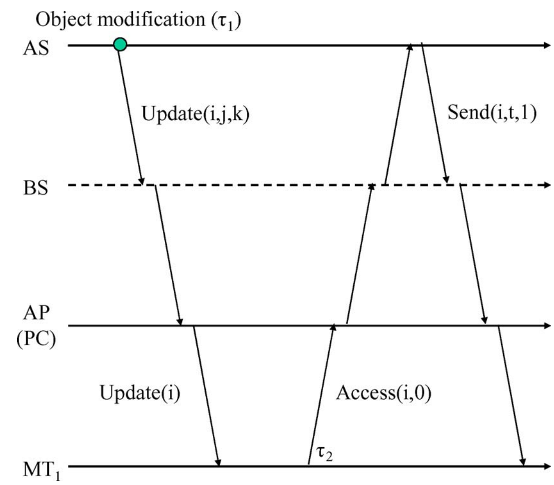



Fig. 10. P-CB operation. PC has no cached object.

### B. P-CB

We describe the P-CB operations by considering two cases: 1) when the PC has a cached object for $O_i$ and 2) when the PC does not have a cached object for $O_i$.

*1) When the PC has a cached object:* Fig. 9 illustrates the operation of P-CB when the PC has a cached object for $O_i$. Let $j$ and $k$ be the indexes of the MT and the vehicle where the MT resides, respectively. At time $\tau_1$, an object $O_i$ is modified, and hence, the AS sends $\mathbf{Update}(i, j, k)$ to invalidate $O_i$ maintained by MT $j$. When the PC receives $\mathbf{Update}(i, j, k)$, it also invalidates its cached object. In addition, the PC invalidates the cached object at the MT by sending $\mathbf{Update}(i)$. In a WLAN, link-layer broadcast can be supported. Therefore, the PC broadcasts $\mathbf{Update}(i)$ within the WLAN, where the corresponding MT locates via the broadcast address. Since broadcast is used, the indexes of the MT and the vehicle (i.e., $j$ and $k$, respectively) are not needed. The broadcast-based invalidation is more effective when multiple MTs in vehicle $k$ have cached objects. After the invalidation, MT $j$ sends an acknowledgement message $\mathbf{Ack}(i, j, k)$, which is omitted in Fig. 9. For reliable transmission in invalidation, the PC rebroadcasts $\mathbf{Update}(i)$ if it does not receive an acknowledgement message from MT $j$. At time $\tau_2$, another MT, i.e., $\text{MT}_2$, accesses $O_i$. As a result, the PC has a cached object with the time sequence $t_{\text{PC}}$. Therefore, the PC can resolve $\mathbf{Access}(i)$ sent by the MT at time $\tau_3$ by referencing its cache, and then, the PC sends its cached object to the MT via $\mathbf{Send}(i, t_{\text{PC}}, 1)$. Consequently, data object transmission only occurs in the WLAN link. This is a representative advantage that can be achieved by the PC-based wireless data access algorithms. In other words, although an MT does not access an object after the object update, the MT can reduce the transmission cost by contacting the PC if there is a data access by other MTs. This benefit becomes significant as the popularity of the object increases, which will be investigated in Section VI.

*2) When the PC has no cached object:* Fig. 10 shows the P-CB operation when the PC has no cached object for $O_i$. This situation can happen if 1) no MT within the vehicle accesses $O_i$ after the invalidation at time $\tau_1$, or 2) any MT within the vehicle accesses $O_i$ after $\tau_1$, but the cached object is replaced. Since both the MT and the PC have no cached objects, $\mathbf{Access}(i)$ is transmitted to the AS, and the AS replies with $\mathbf{Send}(i, t, 1)$. Then, the PC and the MT maintain the data object $O_i$ in their caches until it is invalidated or replaced.

### C. Cache Replacement Scheme

Since the AP and the MT have limited caches, a suitable cache replacement scheme needs to be investigated. In the literature, several cache-replacement policies have been studied [e.g., least frequently used (LFU) and least recently used (LRU)]. In wireless data access applications, the object update rate and the object access rate are important factors affecting cache efficiency. Therefore, a new cache-replacement scheme based on the update-to-access ratio is proposed in [18]. However, the update rate can only be accurately measured by the AS because object modifications occur at the AS. For P-PER and P-CB, we consider an LFU scheme for the PC and MT caches. In addition, the object update rate is estimated without the help of the AS and used as a secondary criterion for cache replacement.

In P-PER, every data access request is delivered to the AS. Therefore, both the MT and the PC can accurately measure the access rate. On the other hand, the MT and the PC can detect an object update event when they receive a $\mathbf{Send}$ message with $F = 1$. The $\mathbf{Send}$ message is only returned when the MT (or the PC) contacts the AS. Therefore, the time interval between two $\mathbf{Send}$ messages with $F=1$ is equal to or larger than the real update interval, which indicates that the update interval measured in P-PER is not an accurate value. Hence, we only use the update interval as a secondary criterion for cache replacement.

Let $\lambda_j^i$ be the access rate for $O_i$ of MT $j$. Then, the access rate for $O_i$ at the PC is $\lambda_{\text{PC}}^i = \sum_{j=1}^{M} \lambda_j^i$, where $M$ is the number of MTs in a vehicle. In addition, let $\mu_{\text{MT}}^i$ and $\mu_{\text{PC}}^i$ be the update rates of $O_i$ measured by the MT and the PC, respectively. At the cache of MT $j$, when a cache replacement is required, an object with the smallest $\lambda_j^i$ is searched. If there exist multiple objects with the smallest $\lambda_j^i$, their update rates are compared, and the object with the largest $\mu_{\text{MT}}^i$ is replaced. Similar procedures are

TABLE I
TRANSMISSION COSTS

| Message | Wired | WWAN | WLAN |
|---|---|---|---|
| **Access** | $H \cdot S_{access}/B$ | $S_{access}/W_{Up}$ | $S_{access}/L$ |
| **Update** | $H \cdot S_{update}/B$ | $S_{update}/W_{Down}$ | $S_{update}/L$ |
| **Ack** | $H \cdot S_{ack}/B$ | $S_{ack}/W_{Up}$ | $S_{ack}/L$ |
| **Send** w/o data object | $H \cdot S_{send}/B$ | $S_{send}/W_{Down}$ | $S_{send}/L$ |
| **Send** w data object | $H \cdot (S_{send} + S_{data})/B$ | $(S_{send} + S_{data})/W_{Down}$ | $(S_{send} + S_{data})/L$ |

performed at the PC, except that $\lambda_{\rm PC}^i$ and $\mu_{\rm PC}^i$ are used instead of $\lambda_j^i$ and $\mu_{\rm MT}^i$, respectively.

In P-CB, invalidation procedures for all object updates are performed by the AS. Therefore, the measurement of the update rate can be supported more efficiently than P-PER. Namely, the MT and the PC calculate the update rate as the inverse of the interval between two invalidation messages. However, an MT can receive invalidation messages only when it has a cached object. In addition, the PC can only receive invalidation messages when at least one MT within the vehicle has a cached object. Therefore, similar to P-PER, the update rate in P-CB is only used as a secondary criterion.

The cache-replacement operations of the MT cache are the same as those in P-PER. A data access at the PC in P-CB only occurs when there is no cached object in the MT cache. Let $\omega$ be the probability that there is no cached object in the MT cache. Then, the access rate for $O_i$ at the PC is $\lambda_{\rm PC}^i = \omega \sum_{j=1}^{M} \lambda_j^i$. For cache replacement in P-CB, the PC first checks an object with the smallest $\lambda_{\rm PC}^i$. If more than two objects are found, an object with the largest update rate is finally selected for replacement.

## V. PERFORMANCE ANALYSIS

In this section, we derive the analytical expressions for the transmission costs of P-PER and P-CB, which can be found by considering the traffic volume for a data access event. In [21], the transmission cost is calculated as the product of the message size and the hop distance. Since mobile hotspots are characterized with heterogeneous links (i.e., wired, WWAN, and WLAN links), the aforementioned approach cannot be directly used. In this analysis, we define the weighted transmission cost as the transmission cost divided by the corresponding link bandwidth, and its unit is bytes $*$ hops/Mb/s. Let $S_{\rm access}$, $S_{\rm update}$, and $S_{\rm ack}$ denote the sizes of **Access**, **Update**, and **Ack** messages, respectively. $S_{\rm send}$ and $S_{\rm data}$ represent the size of a **Send** message without a data object and the size of a **Send** message with a data object, respectively. The WWAN and WLAN links are one-hop links, whereas the wired link is $H$ hops. The weighted transmission cost in each link can be calculated, as shown in Table I, with the bandwidth defined in Section III. The weighted total transmission cost $C_T$ can be obtained by the sum of the weighted transmission costs in the wired, WWAN, and WLAN links. For instance, when an **Access** message is sent by the MT and the message is resolved by the PC using the cached object, the transmission cost for this case is $S_{\rm access}/L + (S_{\rm send} + S_{\rm data})/L$.

Let $access_i$, $update_i$, $ack_i$, $send_i$, and $data_i$ denote the weighted transmission costs for **Access**, **Update**, **Ack**, **Send** without a data object, and **Send** with a data object, respectively, where $i$ is the link index, i.e., the indexes of the WLAN, WWAN, and wired links are 1, 2, and 3, respectively. $ACCESS$, $UPDATE$, $ACK$, $SEND$, and $DATA$ are defined as $ACCESS = \sum_{i=1}^{3} access_i$, $UPDATE = \sum_{i=1}^{3} update_i$, $ACK = \sum_{i=1}^{3} ack_i$, $SEND = \sum_{i=1}^{3} send_i$, and $DATA = \sum_{i=1}^{3} data_i$. We analyze the transmission cost based on four assumptions.

1) The interobject update time $t_u$ for $O_i$ follows an exponential distribution with rate $\mu_i$.
2) The interobject access time $t_a$ for $O_i$ by a tagged MT follows an exponential distribution with rate $\lambda_i$.
3) MTs in a vehicle are independent and identically distributed. Therefore, the interobject access time $t_o$ for $O_i$ by MTs other than a tagged MT follows an exponential distribution with rate $(M-1)\lambda_i$, where $M$ is the number of MTs in a vehicle.
4) The size of a PC is sufficiently large, and therefore, the cached object in the PC is not replaced due to cache overflow.

Let $\alpha$ and $\beta$ be the probability that there exists an object update between two access events of a tagged MT and the probability that MTs other than the tagged MT access the updated object before the tagged MT's access, respectively. In addition, let $\gamma$ be the probability that a cached object at the MT is replaced due to cache overflow. To derive the transmission cost in P-PER and P-CB, we consider the following four cases: 1) No object update occurs between two access events, and the cached object at the MT is not replaced [probability $(1-\alpha)(1-\gamma)$]; 2) no object update occurs between two access events, but the cached object is replaced (probability $(1-\alpha)\gamma$); 3) an object update occurs, and other MTs access the updated object and cache it (probability $\alpha\beta$); and 4) an object update occurs, and no other MTs access the updated object [probability $\alpha(1-\beta)$].

In P-PER, the MT contacts the AS for confirmation and receives an acknowledgement message from the AS for case 1). Therefore, the transmission cost is $ACCESS + SEND$. For cases 2) and 3), the MT first contacts the PC, and the cached object at the PC is confirmed by the AS. The PC then delivers its cached object to the MT. Therefore, the transmission cost is $ACCESS + send_3 + send_2 + data_1$. For case 4), since the MT should contact the AS for confirmation and receive the up-to-date object from the AS, the transmission cost is $ACCESS + DATA$. Consequently, the expected transmission cost in P-PER can be computed as

$$\begin{aligned} C_{\text{P-PER}} = &(1-\alpha)(1-\gamma)(ACCESS + SEND) \\ &+ (1-\alpha)\gamma(ACCESS + send_3 + send_2 + data_1) \\ &+ \alpha\beta(ACCESS + send_3 + send_2 + data_1) \\ &+ \alpha(1-\beta)(ACCESS + DATA). \end{aligned} \quad (1)$$

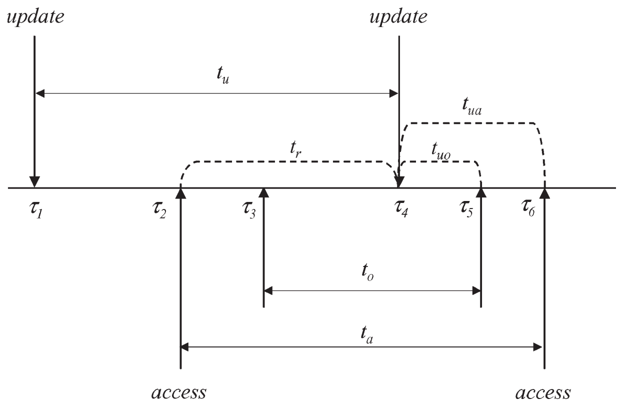



Fig. 11. Timing diagram.

In P-CB, since the MT has a valid cached object for case 1), no transmission cost occurs. On the other hand, for case 2), the MT should contact the PC and receive a data object. Thus, the transmission cost is $access_1 + data_1$. For case 3), the transmission costs for object invalidation and the cached object from the PC to the MT are required, and thus, the transmission is $UPDATE + ACK + access_1 + data_1$. For case 4), since both the MT and the PC have no object, the transmission cost from the AS to the MT and the object invalidation cost occur. Thus, the transmission cost is $UPDATE + ACK + ACCESS + DATA$. Therefore, the expected transmission cost in P-CB can be obtained as follows:

$$\begin{aligned} C_{\text{P-CB}} = &(1-\alpha)\gamma(access_1 + data_1) \\ &+ \alpha\beta(UPDATE + ACK + access_1 + data_1) \\ &+ \alpha(1-\beta)(UPDATE + ACK \\ &\qquad + ACCESS + DATA). \end{aligned} \tag{2}$$

To derive $\alpha$ and $\beta$, we use a timing diagram, as shown in Fig. 11. At $\tau_1$ and $\tau_4$, two object update events occur, and at $\tau_2$ and $\tau_6$, two object access events by a tagged MT occur. On the other hand, at $\tau_3$ and $\tau_5$, two object accesses by MTs other than the tagged MT are triggered. Then, $t_u$ and $t_a$ represent the interobject update time and the interobject access time by a tagged MT, respectively. In addition, $t_o$ refers to the interobject access time by MTs other than the tagged MT. Let $t_r$ be the time from the access event to the update event, which follows an exponential distribution with rate $\mu_i$ by the random observer property. Then, $\alpha$ is derived as

$$\begin{aligned} \alpha = \Pr(t_a > t_r) &= \int_0^\infty \Pr(t_a > \tau) \cdot \mu_i e^{-\mu_i \tau} d\tau \\ &= \frac{\mu_i}{\mu_i + \lambda_i} = \frac{1}{1+\rho_i} \end{aligned} \tag{3}$$

where $\rho_i = \lambda_i/\mu_i$ is the access-to-update ratio for $O_i$ of a tagged MT. Let $t_{\text{ua}}$ and $t_{\text{uo}}$ be the time from the object update event to the object access event by a tagged MT and the time from the object update event to the object access event by MTs other than the tagged MT, respectively. Then, the probability $\beta$ that there exists an advance object access event by other MTs, except the tagged MT, can be computed as

$$\beta = \Pr(t_{\text{ua}} > t_{\text{uo}}). \tag{4}$$

By the random observer property, $t_{\text{ua}}$ and $t_{\text{uo}}$ follow exponential distributions with rates $\lambda_i$ and $(M-1)\lambda_i$, respectively. Therefore, (4) reduces to

$$\beta = \Pr(t_{\text{ua}} > t_{\text{uo}}) = \frac{(M-1)\lambda_i}{(M-1)\lambda_i + \lambda_i} = \frac{M-1}{M}. \tag{5}$$

The probability that a cached object in the MT is replaced due to cache overflow $\gamma$ can be approximated as follows: Let $\lambda_i$ and $\lambda_o$ be the access rates for a tagged object $O_i$ and other objects except $O_i$, respectively. Then, $\lambda_o = \sum_{j \neq i} \lambda_j$. By the superposition property, the access process for other objects, except $O_i$, follows a Poisson process with rate $\lambda_o$. Let $\theta(k)$ be the probability that there are $k$ access events during an interobject access time for $O_i$. For tactical analysis, we consider the average interobject access time for $O_i$, i.e., $1/\lambda_i$. Then, $\theta(k)$ is given by

$$\theta(k) = \frac{e^{-\lambda_o/\lambda_i}(\lambda_o/\lambda_i)^k}{k!}. \tag{6}$$

A cached object is replaced when there are more than $K_{\text{MT}}$ access events for other objects except $O_i$. Consequently, $\gamma$ can be obtained from

$$\gamma = \sum_{k=K_{\text{MT}}}^{\infty} \theta(k). \tag{7}$$

## VI. Simulation Results

To evaluate the performance of P-PER and P-CB and validate the analytical model, we have developed an event-driven simulator and run extensive simulations. $N$ data objects are assumed to be in the AS, and the relative frequency for data objects follows a Zipf-like distribution, where the relative probability of a request for the $i$th most popular object is proportional to $1/i^\kappa$ [22], where $\kappa$ determines the skewness in the Zipf-like distribution. For instance, for $\kappa = 1$, the access probability of a data object is strictly proportional to its popularity ranking (i.e., Zipf's law). On the other hand, for $\kappa = 0$, the access probabilities for all data objects are the same. Let $O_i$ be the $i$th most popular object $(1 \leq i \leq N)$. Then, the probability of accessing $O_i$ is given by

$$p_i = \frac{\Omega}{i^\kappa} \tag{8}$$

where $\Omega = (\sum_{i=1}^{N}(1/i^\kappa))^{-1}$.

In the simulations, the interobject access time for $O_i$ follows an exponential distribution with rate $\lambda_i$, which is given by $p_i\lambda$, where $\lambda$ is the net access rate to data objects. The interobject update time is drawn from a Gamma distribution with mean $1/\mu_i$ and variance $\nu$. Then, the aggregate access-to-update ratio for objects, i.e., $\rho$, can be obtained from

$$\rho = \frac{\sum_{i=1}^{N}\lambda_i}{N\mu_i} = \frac{\lambda}{N\mu_i}. \tag{9}$$

The performance metrics are the total cache hit probability $p$ and the weighted transmission cost $C_T$ defined in Section V. The total cache hit probability is defined as the probability that a

TABLE II
DEFAULT PARAMETER VALUES FOR SIMULATION

| $S_{access}$ | $S_{update}$ | $S_{ack}$ | $S_{send}$ | $S_{data}$ | $N$ | $\kappa$ |
|---|---|---|---|---|---|---|
| 45 bytes | 45 bytes | 45 bytes | 45 bytes | 727 bytes | 100 | 0.8 |
| $W_{Down}$ | $W_{Up}$ | $B$ | $L$ | $K_{MT}$ | $K_{PC}$ | $H$ |
| 384 Kbps | 96 Kbps | 100 Mbps | 11 Mbps | 20 | 60 | 10 |

TABLE III
WEIGHTED TRANSMISSION COST (IN bytes * hops/Mb/s): SIMULATION (S) VERSUS ANALYTICAL (A) RESULTS

| $\rho$ | P-PER (A) | P-PER (S) | Error (%) | P-CB (A) | P-CB (S) | Error (%) |
|---|---|---|---|---|---|---|
| 0.01 | 2616 | 2619 | 0.14 | 3706 | 3805 | 2.6 |
| 0.1 | 2456 | 2452 | 0.19 | 3350 | 3445 | 2.8 |
| 1 | 1652 | 1649 | 0.18 | 1696 | 1741 | 2.6 |
| 10 | 848 | 851 | 0.31 | 362 | 372 | 2.8 |
| 100 | 689 | 686 | 0.32 | 106 | 106 | 0.15 |

data access can be resolved by a cached object at the MT cache or the PC. Let $p_{\mathrm{MT}}$ and $p_{\mathrm{PC}}$ be the MT cache hit probability and the PC cache hit probability, respectively. $p_{\mathrm{MT}}$ is computed as $N_{\mathrm{hit}}^{\mathrm{MT}}/N_a$, where $N_{\mathrm{hit}}^{\mathrm{MT}}$ is the number of data accesses resolved by the MT cache, and $N_a$ is the number of total data accesses. $p_{\mathrm{PC}}$ is determined by $N_{\mathrm{hit}}^{\mathrm{PC}}/N_a$, where $N_{\mathrm{hit}}^{\mathrm{PC}}$ is the number of data accesses that cannot be resolved by the MT cache but can be resolved by the PC. Since only the MT cache is employed in PER and CB, $p_{\mathrm{MT}}$ is the total cache hit probability. On the other hand, the total cache hit probability in P-PER and P-CB can be expressed as

$$p = p_{\mathrm{MT}} + (1 - p_{\mathrm{MT}}) \cdot p_{\mathrm{PC}}. \quad (10)$$

The default parameter values for the simulations are derived from [16] and summarized in Table II.

### A. Validation of Analytical Model

In Table III, it can be seen that the analytical results are mostly consistent with the simulation results and that the errors in the analytical results are less than 3% in all cases. These errors come from the approximation for the PC size. Since we assume that the PC size is infinite in the analytical model, no PC overflow occurs, and MT cache misses can be resolved at the PC. On the other hand, a finite-buffer PC in the simulations can incur higher transmission costs because the MT should contact the AS for the PC overflow. That is, the infinite-buffer PC can result in underestimation of the transmission cost in P-PER and P-CB. It can also be seen that P-CB has higher errors than P-PER. This is because P-CB is more sensitive to the PC size than P-PER. That is, no transmission cost occurs for a cache hit in P-CB, whereas a confirmation cost is needed even for a cache hit in P-PER.

### B. Effects of the Access-to-Update Ratio $\rho$

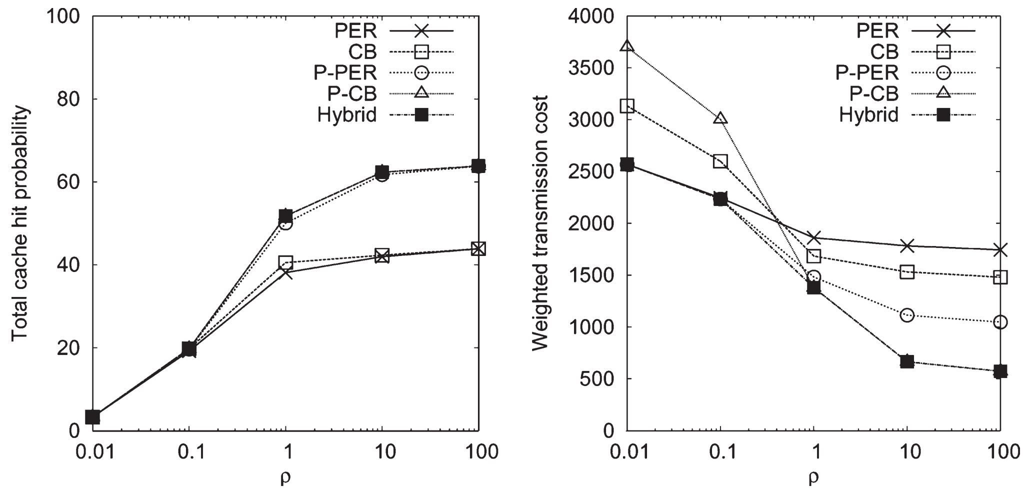
Fig. 12 illustrates the effect of the access-to-update ratio $\rho$ on total cache hit probability and weighted transmission cost. It can be seen that the total cache hit probability increases as $\rho$ increases, but the weighted transmission cost decreases as $\rho$ increases. This can be explained as follows: When $\rho$ is low, the update rate dominates the access rate. Therefore, the possibility that the MT or the PC cache has a stable data object is high. This reduces the effectiveness of the cache. On the other hand, when $\rho$ is high, the access events occur more frequently than the object update events, and thus, the cache can be actively referenced. As shown in Fig. 12, P-PER yields the best performance when $\rho$ is low, and P-CB gives the best performance when $\rho$ is high. In other words, there is no single algorithm that always yields the best performance for all possible values of $\rho$. This means that it is possible to minimize the weighted transmission cost by integrating P-PER and P-CB. Based on this observation, we have proposed a hybrid wireless data access in [23]. As shown in Fig. 12, the hybrid wireless data access algorithm achieves an adaptive optimization in transmission cost over a wide range of access-to-update ratios. That is, the transmission costs of the hybrid wireless data access algorithm approach those of P-PER and P-CB when the access-to update ratios are low and high, respectively.

### C. Effects of the Cache Size

In Fig. 13, it can be observed that the performance of PER and CB can be significantly improved by increasing $K_{\mathrm{MT}}$. In PER and CB, if the cached object in the MT is not valid, the MT accesses the AS and therefore incurs a high transmission cost. Hence, if a large $K_{\mathrm{MT}}$ is used, the total cache hit probability can be increased, and the weighted transmission cost can be significantly reduced. The performance of P-PER and P-CB can also be improved as $K_{\mathrm{MT}}$ increases. However, as shown in Fig. 14, it can be seen that the effect of $K_{\mathrm{MT}}$ is not significant in P-PER and P-CB. In P-PER and P-CB, some cache misses at the MT cache can be resolved at the PC. Therefore, the total cache hit probability, considering both the MT cache hit and the PC cache hit, is not highly sensitive to $K_{\mathrm{MT}}$. On the other hand, Fig. 15 shows that the total cache hit probability and the weighted transmission cost are significantly affected by $K_{\mathrm{PC}}$ in P-PER and P-CB. Namely, as $K_{\mathrm{PC}}$ increases, the total cache hit probability increases, and the weighted transmission cost decreases. Unlike the MT cache, since the PC is installed in a vehicle with sufficient processing power, it is less sensitive to the cache size. Therefore, a larger $K_{\mathrm{PC}}$ can be easily deployed, and improved cache performance can be achieved by P-PER and P-CB in mobile hotspots.

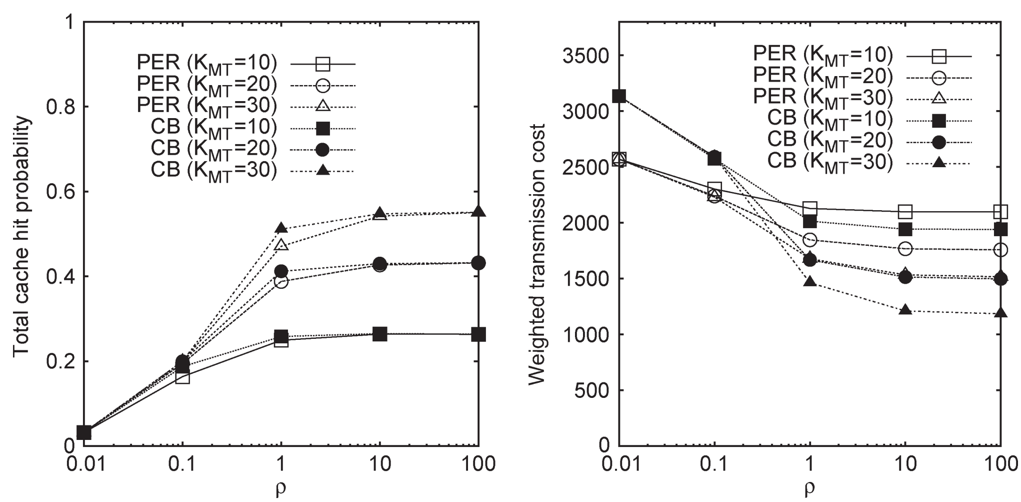
### D. Effects of $\kappa$

In mobile hotspots, different types of users can coexist in a moving vehicle, and their data access patterns are quite diverse. Therefore, we analyze the effect of $\kappa$ that determines the skewness in the data access pattern. As $\kappa$ approaches 1.0, a few specific data objects have a large portion of data accesses. On the other hand, $\kappa = 0$ represents that the access frequency for each data object is identical. Obviously, the performance of data access algorithms with cache can be improved as $\kappa$ increases. Figs. 16–18 indicate this trend. The effect of $\kappa$ is more apparent for PER and CB, as compared with P-PER and P-CB. This is because the effect of $\kappa$ in P-PER and P-CB can be mitigated by using two-tier caching. This result demonstrates that P-PER and P-CB can be used in a variety of applications with different access patterns, regardless of the value of $\kappa$.



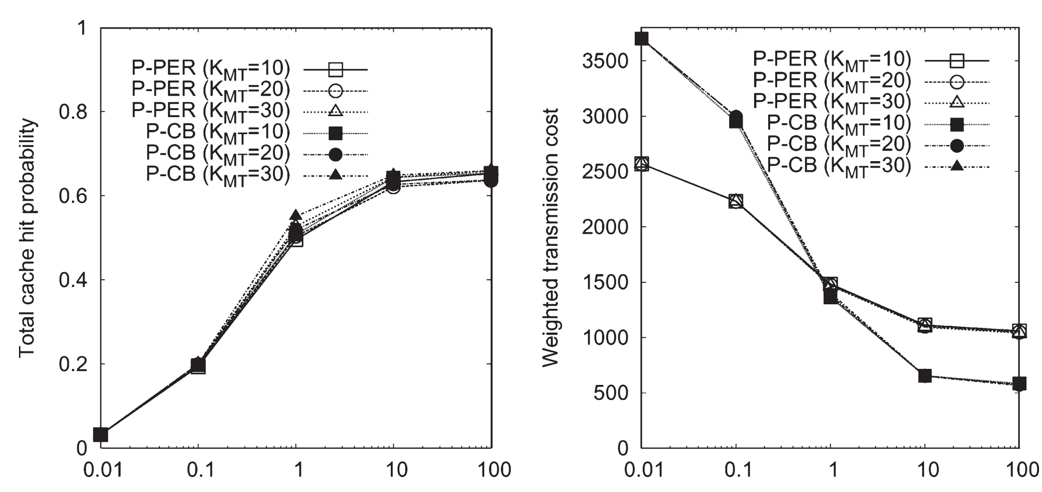

(a) (b)

Fig. 12. Effects of $\rho$. (a) Cache hit probability. (b) Weighted transmission cost.




(a) (b)

Fig. 13. Effects of $K_{\mathrm{MT}}$: PER and CB. (a) Cache hit probability. (b) Weighted transmission cost.




(a) (b)

Fig. 14. Effects of $K_{\mathrm{MT}}$: P-PER and P-CB. (a) Cache hit probability. (b) Weighted transmission cost.

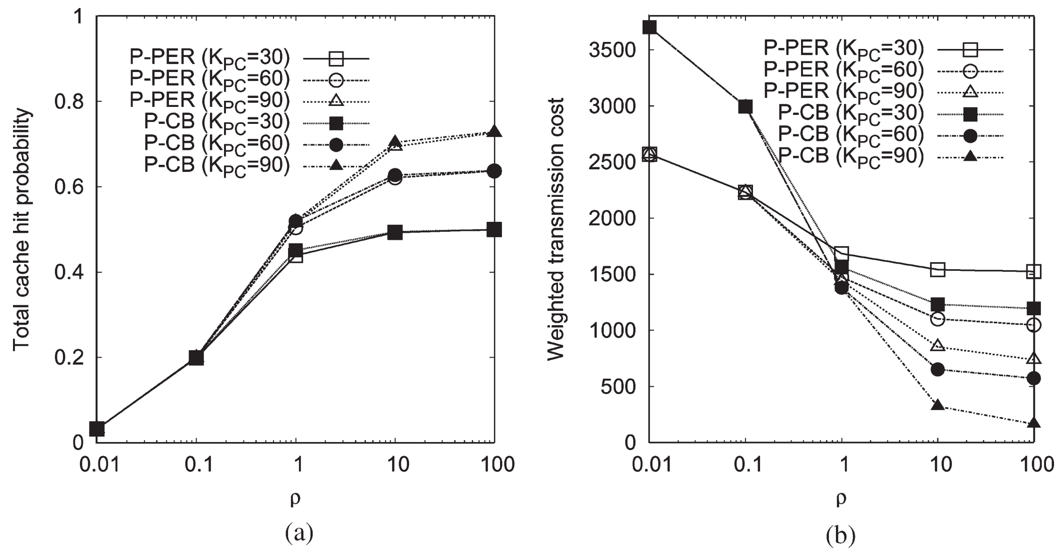



Fig. 15. Effects of $K_{\mathrm{PC}}$: P-PER and P-CB. (a) Cache hit probability. (b) Weighted transmission cost.

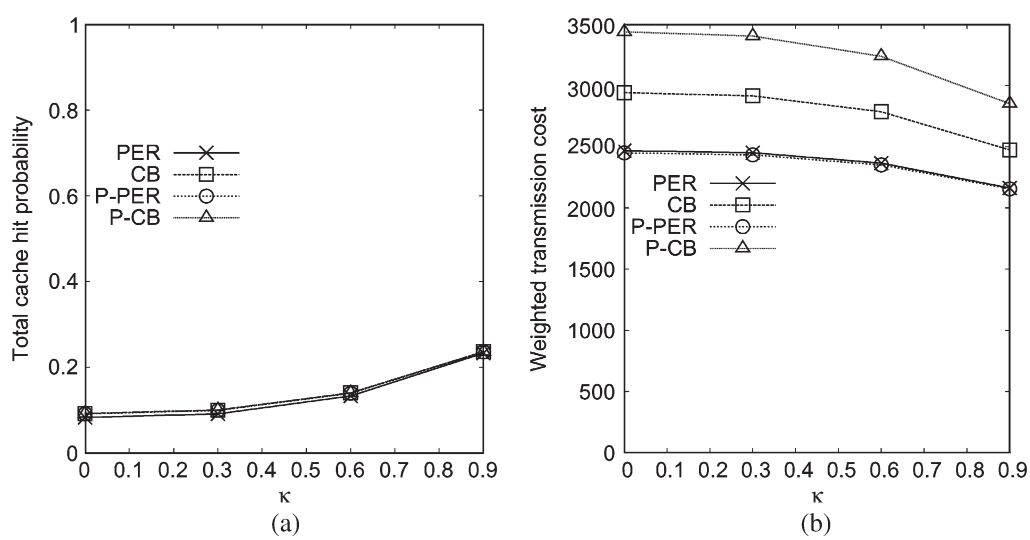



Fig. 16. Effects of $\kappa$ ($\rho = 0.1$). (a) Cache hit probability. (b) Weighted transmission cost.

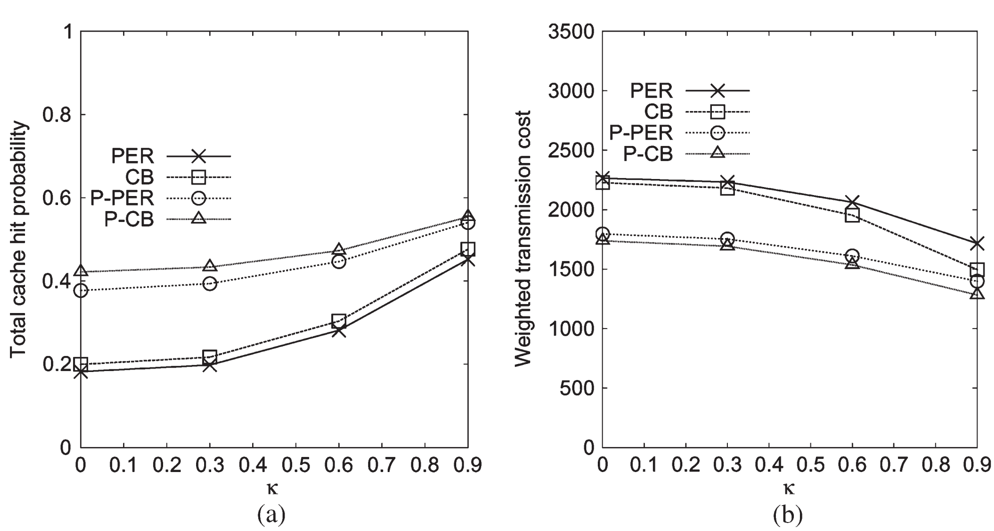



Fig. 17. Effects of $\kappa$ ($\rho = 1.0$). (a) Cache hit probability. (b) Weighted transmission cost.

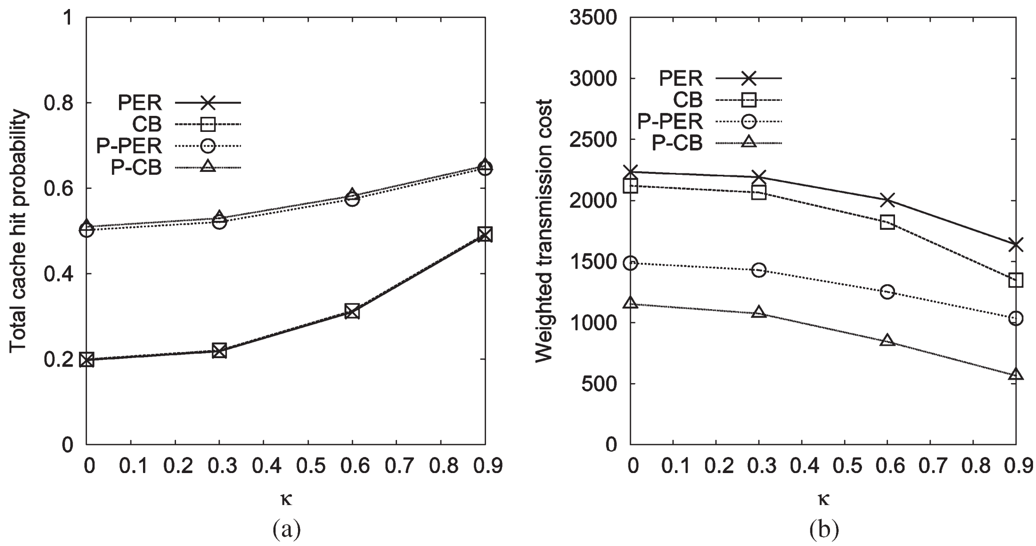



Fig. 18. Effects of $\kappa$ ($\rho = 10$). (a) Cache hit probability. (b) Weighted transmission cost.

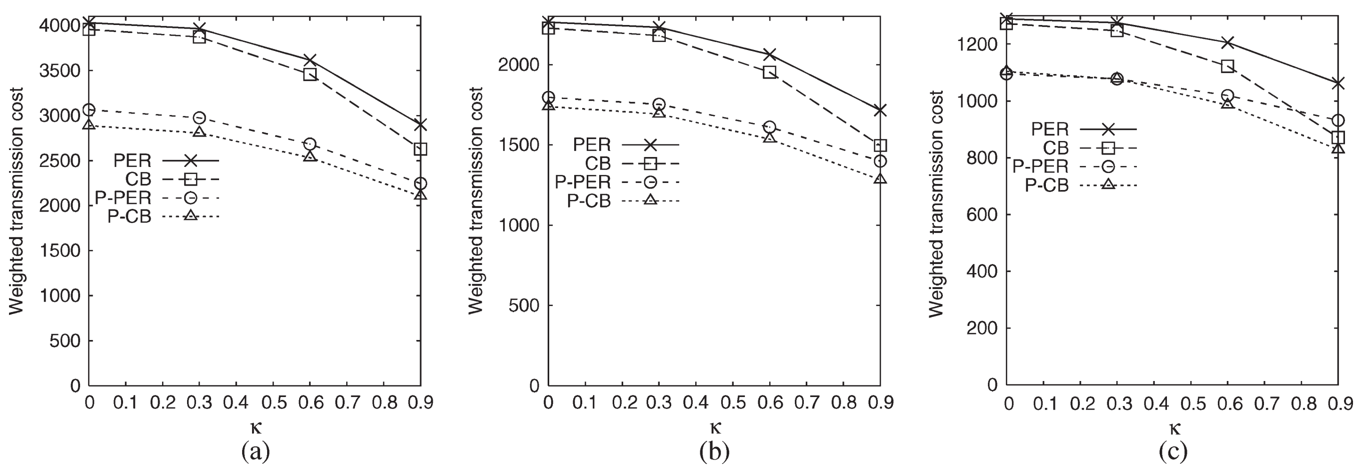



Fig. 19. Effects of $S_{\mathrm{data}}$. (a) $S_{\mathrm{data}} = 300$ bytes. (b) $S_{\mathrm{data}} = 727$ bytes. (c) $S_{\mathrm{data}} = 1500$ bytes.

In addition, it can be seen that the effect of $\kappa$ exhibits a greater effect when $\rho = 10$, i.e., the access rate dominates the update rate.

### *E. Effects of the Data Object Size $S_{\mathrm{data}}$*

It is expected that mobile hotspots will support different types of wireless data access applications (e.g., traffic/road/weather information search, real-time news reports, and multimedia contents delivery). These heterogeneous applications have different data object sizes. Since the weighted transmission cost is largely dependent on the data object size $S_{\mathrm{data}}$, we investigate the performance of P-PER and P-CB over a wide range of data object sizes. Fig. 19 demonstrates that the gains of P-PER and P-CB increase as $S_{\mathrm{data}}$ increases. This is because P-PER and P-CB can reduce the transmission cost over the WWAN with the fewest radio resources by introducing the PC installed in a vehicle (i.e., AP). Therefore, P-PER and P-CB are more suitable for applications with large objects (e.g., multimedia data objects).

## VII. Conclusion

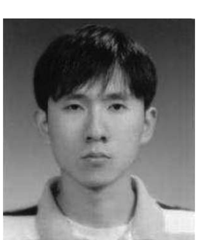
In this paper, we have proposed two proxy-based wireless data access algorithms, namely, P-PER and P-CB, for mobile hotspots. We have demonstrated that P-PER and P-CB can improve cache hit performance and significantly reduce transmission cost. Furthermore, P-PER and P-CB can be easily applied to other application scenarios in mobile hotspots (e.g., vehicle-to-vehicle communications). A tradeoff between P-PER and P-CB in response to the access-to-update ratio suggests that an adaptive approach can further improve the performance of wireless data access applications in mobile hotspots. In future work, we will extend our research to wireless data access applications that run over unreliable data paths and only require a cached data object with weak consistency.

## References

[1] J. Ott and D. Kutscher, "Drive-thru Internet: IEEE 802.11b for 'automobile' users," in *Proc. IEEE INFOCOM*, Mar. 2004, p. 373.

[2] V. Mancuso and G. Bianchi, "Streaming for vehicular users via elastic proxy buffer management," *IEEE Commun. Mag.*, vol. 42, no. 11, pp. 144–152, Nov. 2004.

[3] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, *Network Mobility (NEMO) Basic Support Protocol*, IETF RFC 3963, Jan. 2005.
[4] S. Pack, X. S. Shen, J. W. Mark, and J. Pan, "Mobility management in mobile hotspots with heterogeneous multi-hop wireless links," *IEEE Commun. Mag.*, vol. 45, no. 9, pp. 106–112, Sep. 2007.
[5] A. Iera, A. Molinaro, S. Polito, and G. Roggeri, "End-to-end QoS provisioning in 4G with mobile hotspots," *IEEE Network*, vol. 19, no. 5, pp. 26–34, Sep./Oct. 2005.
[6] D. Ho and S. Valaee, "Information raining and optimal link-layer design for mobile hotspots," *IEEE Trans. Mobile Comput.*, vol. 4, no. 3, pp. 271–284, May/Jun. 2005.
[7] G. Korkmaz, E. Ekici, and F. Ozguner, "A new high throughput Internet access protocol for vehicular networks," in *Proc. ACM VANET*, Sep. 2005, pp. 89–90.
[8] P. Rodriguez, R. Chakravorty, J. Chesterfield, I. Pratt, and S. Banerjee, "MAR: A commuter router infrastructure for the mobile Internet," in *Proc. ACM MOBISYS*, Jun. 2004, pp. 217–230.
[9] R. Wakikawa, K. Okada, R. Koodli, A. Nilsson, and J. Murai, "Design of vehicle network: Mobile gateway for MANET and NEMO converged communication," in *Proc. ACM VANET*, Sep. 2005, pp. 81–82.
[10] K. Lan, H. Petander, E. Perrera, L. Libman, C. Dwertman, and M. Hassan, "MOBNET: The design and implementation of a network mobility testbed for NEMO protocol," in *Proc. IEEE LANMAN*, Sep. 2005, pp. 1–6.
[11] R. Kuntz, K. Mituya, and R. Wakikawa, "Performance evaluation of NEMO basic implementations," in *Proc. WONEMO*, Jan. 2006.
[12] S. Pack, H. Rutagemwa, X. S. Shen, J. Mark, and L. Cai, "Performance analysis of mobile hotspots with heterogeneous wireless links," *IEEE Trans. Wireless Commun.*, vol. 6, no. 10, pp. 3717–3727, Oct. 2007.
[13] H. Chen and Y. Xiao, "Cache access and replacement for future wireless Internet," *IEEE Commun. Mag.*, vol. 44, no. 5, pp. 113–123, May 2006.
[14] J. Howard, M. Kazar, S. Menees, D. Nichols, M. Satyanarayanan, R. Sidebotham, and M. West, "Scale and performance in a distributed file system," *ACM Trans. Comput. Syst.*, vol. 6, no. 1, pp. 51–58, Feb. 1988.
[15] M. Nelson, B. Welch, and J. Ousterhout, "Caching in the sprite network file system," *ACM Trans. Comput. Syst.*, vol. 6, no. 1, pp. 134–154, Feb. 1988.
[16] Y. Lin, W. Lai, and J. Chen, "Effects of cache mechanism on wireless data access," *IEEE Trans. Wireless Commun.*, vol. 2, no. 6, pp. 1247–1258, Nov. 2003.
[17] Y. Fang and Y. Lin, "Strongly consistent access algorithms for wireless data networks," *Wirel. Netw.*, vol. 11, no. 3, pp. 243–254, May 2005.
[18] H. Chen, Y. Xiao, and X. S. Shen, "Update-based cache replacement policies in wireless data access," *IEEE Trans. Mobile Comput.*, vol. 5, no. 12, pp. 1734–1748, Dec. 2006.
[19] Y. Xiao and H. Chen, "Optimal callback with two-level adaptation for wireless data access," *IEEE Trans. Mobile Comput.*, vol. 5, no. 8, pp. 1087–1102, Aug. 2006.
[20] C. Evans-Pughe, "The connected car," *IEE Rev.*, vol. 51, no. 1, pp. 42–46, Jan. 2005.
[21] S. Lo, G. Lee, W. Chen, and J. Liu, "Architecture for mobility and QoS support in all-IP wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 4, pp. 691–705, May 2004.
[22] L. Berslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in *Proc. IEEE INFOCOM*, Mar. 1999, pp. 126–134.
[23] S. Pack, H. Rutagemwa, X. S. Shen, J. W. Mark, and K. Park, "A hybrid wireless data access algorithm in mobile hotspots," in *Proc. IEEE INFOCOM MOVE Workshop*, May 2007, pp. 37–42.



**Sangheon Pack** (S'03–M'05) received the B.S. (*magna cum laude*) and Ph.D. degrees in computer engineering from Seoul National University, Seoul, Korea, in 2000 and 2005, respectively.

Since March 2007, he has been an Assistant Professor with the School of Electrical Engineering, Korea University, Seoul. From July 2006 to February 2007, he was a Postdoctoral Fellow with Seoul National University. From 2005 to 2006, he was a Postdoctoral Fellow with the Broadband Communications Research Group, University of Waterloo, Waterloo, ON, Canada. He has also been a member of the Samsung Frontier Membership since 1999. He was a Visiting Researcher with Fraunhofer FOKUS, Berlin, Germany, in 2003. His research interests include mobility management, multimedia transmission, and QoS provision issues in the next-generation wireless/mobile networks.

Dr. Pack is a member of the Association for Computing Machinery. He was a recipient of the Student Travel Grant Award for the 2003 International Federation for Information Processing Personal Wireless Conference. He was also a recipient of the Korea Foundation for Advanced Studies Computer Science and Information Technology Scholarship from 2002 to 2005.

**Humphrey Rutagemwa** (S'03–M'07) received the B.Sc. degree (with first-class honors) in electronics and communications from the University of Dar es Salaam, Dar es Salaam, Tanzania, in 1998 and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2002 and 2007, respectively.

From January 1999 to August 2000, he was a System Engineer with CRDB Bank, Tanzania. From September 2003 to July 2007, he was a Research Assistant and then a Lecturer with the Department of Electrical and Computer Engineering, University of Waterloo. Since August 2007, he has been with the Communications Research Centre, Ottawa, ON, where he is a Research Scientist. His current research interests are in wireless communications and networking, with particular focus on performance modeling, design, and analysis of vehicular wireless networks and cognitive networks.

**Xuemin (Sherman) Shen** (M'97–SM'02) received the B.Sc. degree from Dalian Maritime University, Dalian, China, in 1982 and the M.Sc. and Ph.D. degrees from Rutgers University, New Brunswick, NJ, in 1987 and 1990, respectively, all in electrical engineering.

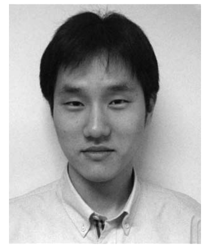
He is currently with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, where he is a Professor and the Associate Chair for Graduate Studies. He has coauthored two books and has published more than 200 papers and book chapters on wireless communications and networks, control, and filtering. His research interests include mobility and resource management in interconnected wireless/wireline networks, UWB wireless communications systems, wireless security, and ad hoc and sensor networks.

Dr. Shen is a Registered Professional Engineer in the Province of Ontario. He served as the Technical Program Committee Chair for the 2007 IEEE Global Communications Conference (Globecom); the 2007 IEEE Wireless Communications and Networking Conference (WCNC); the 2005 International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (Qshine); the 2005 IEEE International Conference on Broadband Networks (Broadnet); the 2005 International Conference on Wireless Networks, Communications, and Mobile Computing (WirelessCom); the 2005 International Federation for Information Processing (IFIP) Networking; the 2004 International Symposium on Parallel Architectures, Algorithms, and Networks (ISPAN); and the 2003 IEEE Globecom Symposium on Next-Generation Networks and Internet. He also serves as an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *ACM Wireless Networks*, *Computer Networks*, *Wireless Communications and Mobile Computing* (Wiley), etc., and was a Guest Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, *IEEE Wireless Communications*, and *IEEE Communications Magazine*. He was the recipient of the Premier's Research Excellence Award from the Province of Ontario, for demonstrated excellence of scientific and academic contributions in 2003 and the Outstanding Performance Award from the University of Waterloo for outstanding contributions to teaching, scholarship, and service in 2002.

**Jon W. Mark** (M'62–SM'80–F'88–LF'03) received the Ph.D. degree in electrical engineering from McMaster University, Hamilton, ON, Canada, in 1970.

Upon graduation, he joined the Department of Electrical Engineering (now Department of Electrical and Computer Engineering), University of Waterloo, Waterloo, ON, where he became a Full Professor in 1978 and served as the Department Chairman from July 1984 to June 1990. In 1996, he established the Centre for Wireless Communications, University of Waterloo, and has since been serving as the founding Director. He was on sabbatical leave at the IBM Thomas Watson Research Center, Yorktown Heights, NY, as a Visiting Research Scientist from 1976 to 1977; at AT&T Bell Laboratories, Murray Hill, NJ, as a Resident Consultant from 1982 to 1983; at the Laboratoire Méthodologie et Architecture des Systèmes Informatiques, Université Pierre et Marie Curie, Paris, France, as an Invited Professor from 1990 to 1991; and at the Department of Electrical Engineering, National University of Singapore, as a Visiting Professor from 1994 to 1995. He is a coauthor of the textbook *Wireless Communications and Networking* (Prentice-Hall, 2003). His current research interests are in wireless communications and wireless/wireline interworking, particularly in the areas of resource management, mobility management, and end-to-end information delivery with QoS provisioning.

Dr. Mark has served as a member of a number of editorial boards, including the IEEE TRANSACTIONS ON COMMUNICATIONS, *ACM/Baltzer Wireless Networks*, *Telecommunication Systems*, etc. He was a member of the Intersociety Steering Committee of the IEEE/ACM TRANSACTIONS ON NETWORKING from 1992 to 2003 and a member of the IEEE COMSOC Awards Committee during 1995–1998.



**Kunwoo Park** received the B.S. degree in computer engineering from Korea Advanced Institute of Science (KAIST), Daejeon, Korea, in 2004. He is currently working toward the Ph.D. degree with the School of Computer Science and Engineering, Seoul National University, Seoul, Korea.

His research interests include peer-to-peer networking, IPv6, and mobility management.