WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2010; **10**:405–425
Published online 28 April 2009 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/wcm.774

Policy-based scanning with QoS support for seamless handovers in wireless networks

Sang-Jo Yoo[1*,†] and Nada Golmie[2]

[1]*Inha University, 253 Yonghyun-dong, Nam-gu, Incheon, Korea*
[2]*National Institute of Standards and Technology, Gaithersburg, MD 20899-8920, U.S.A.*

Summary

Supporting seamless handovers between different wireless networks is a challenging issue. One of the most important aspects of a seamless handover is finding a target network and point of attachment (PoA). This is achieved by performing a so-called channel scanning. In most handovers, such as between universal mobile telecommunications system (UMTS), wireless local area network (WLAN), and worldwide interoperability for microwave access (WiMAX), channel scanning causes severe service disruptions with the current PoA and degrades the quality of service (QoS) during the handover. In this paper, a new architecture for QoS supported scanning that can be generalized to different wireless networks is proposed. It employs two techniques. The first is for determining a policy-based order for the channel scanning sequence. With this technique, depending on the network costs and user requirements, the policy engine determines the channel scanning order for different network types and sets up a scanning sequence of PoAs for a given network type. This policy-based scanning order provides a faster discovery of the target PoA that meets the QoS demands of the user. The second technique consists of a QoS supported dynamic scanning algorithm where the scanning frequency and duration are determined based on the user QOS requirements. Most importantly, the scanning duration is scheduled to guarantee the user QoS requirements while the scan progresses. Simulation results show that the proposed mechanism achieves relatively short service disruptions and provides the desired quality to users during the scanning period. Copyright © 2009 John Wiley & Sons, Ltd.

KEY WORDS: scanning; policy-based; seamless handover; quality of service; scheduling

1. Introduction

The rapid progress in the development of wireless networking and communication technologies over the last decade has meant that different types of wireless communication systems, such as IEEE 802.11 wireless local area network (WLAN), worldwide interoperability for microwave access (WiMAX), universal mobile telecommunications system (UMTS), and Bluetooth for personal area networks have been deployed. These networks are complementary to each other and can be integrated to realize unified next-generation wireless networks. This would allow users to communicate without the geographical coverage limitations of individual communication systems and to choose an optimum wireless network interface and point of attachment (PoA) in accordance with their desired service needs. In this context of ubiquitous connectivity, it

*Correspondence to: Sang-Jo Yoo, Inha University, 253 Yonghyun-dong, Nam-gu, Incheon, Korea.
†E-mail: sjyoo@inha.ac.kr

is an important and challenging issue to support seamless mobility management.

Handovers typically consist of three phases: (i) neighbor discovery *via* scanning, (ii) a handover decision and a Layer 3 fast handover execution, and (iii) link switching and association to the new PoA. In order to locate the target PoA that best fits the mobility path and quality of service (QoS) requirements of the user, a wireless mobile station (MS) needs to scan multiple channels. Once the target network (horizontal or vertical) and PoA have been determined, the MS may perform a Layer 3 handover before a Layer 2 handover to the new PoA using a network layer mobility support protocol such as IETF Fast Mobile IPv6 (FMIPv6) [1]. The MS then switches the current link to the new PoA and authentication and association procedures are followed. During a handover, a certain level of service disruption characterized by delays and packet losses is inevitable. Specially, scanning and link switching are the main contributors to the disruptions of running services. As channel scanning can be a relatively time-consuming procedure, QoS degradation during a scan is a critical issue. Therefore, to support user QoS demands and seamless service, service disruptions during the scanning procedure should be controlled and scheduled effectively.

Most of the previous scanning algorithms focused on reducing the scanning time [2–9]. Using the previous channel selection history or information regarding the neighboring network topology, they aim at scanning fewer channels (i.e., selective scan) in an effort to reduce the handover latency. One common approach is that an individual MS determines the channels that may have working PoAs and scans only those channels. Another technique involves the current PoA reporting its neighboring PoA information to the MS. In some wireless medium access control (MAC) protocols such as WLAN [10,11] and WiMAX [12,13], the current PoA provides a specified level of information about the same network type neighbor PoAs to the MSs. With WLAN, the Neighbor Report frame from the current access point (AP) includes a list of the neighbor APs, and with WiMAX, a neighbor advertisement (MOB_NBR-ADV) message is sent by the current base station (BS) at specified intervals to identify the neighbor network systems and to define the characteristics of neighboring BSs. This enables a selective scan. The neighbor network information can be also obtained by the information service of the IEEE 802.21 MIHF [14], which provides a query/response mechanism for information transfers about neighbor networks. It contains both static (e.g., the neighbor network topology) and dynamic (e.g., the QoS conditions) information for all types of neighbor networks.

In this paper, a new architecture for QoS support scanning is proposed. With the benefits of the neighbor discovery protocol specified by the aforementioned standards, an MS can obtain a set of candidate channels (or PoAs). For a given candidate set, a scan-ordering policy is proposed that can determine which network type and PoA should be scanned first. This policy-based scan sequence leads to a faster discovery of the target PoA that satisfies user preferences and QoS demands. To provide the desired QoS level to user applications while minimizing MAC layer delays and the packet loss ratio, the entire scanning period is scheduled based on the scanning policy and measured QoS level. The proposed scheduling method divides the entire required scanning period into several smaller scan times termed service interruption times (SITs). After each short scan during an SIT, the MS reverts to normal data transmission conditions. The SIT and interleaving interval are dynamically determined in accordance with the desired and measured QoS. In this paper, to determine which (and how many) scanning-related operations can be performed during each SIT scanning time, the new concept of the *independent scanning piece* is introduced, in which two independent scanning pieces are not correlated with a common timer. If the SIT time is less than the required time to capture one independent scanning piece, a dynamic adjustment of the Layer 2 system parameters is followed in the proposed scanning procedure.

The remainder of this paper is organized as follows: Section 2 presents related work concerning channel scanning for handovers in wireless networks. In Section 3, the new policy-based QoS support scanning architecture is introduced. In Section 4, the policy-based scan ordering and scheduling mechanisms are presented. In Section 5, simulation results for WLAN handover scenarios demonstrate the dynamic nature of the proposed scanning mechanisms for various network and QoS requirements, and the delay and service disruption time are compared with an existing consecutive scanning method. This paper concludes with Section 6.

2. Related Work

Most research concerning channel scanning has focused on WLAN systems. In WLAN networks, scanning implies a set of actions (e.g., a change of radio channels and exchanges of signaling messages) which often interrupts the data transmission with the serving

DOI: 10.1002/wcm

AP. The IEEE 802.11 [10] specifies two scanning methods. For a passive scan, the wireless station switches to a new channel and waits for beacon frames from neighboring APs. Therefore, a large amount of time may be required to discover all neighboring APs. With an active scan, the wireless station broadcasts a probe request frame on a selected new channel and waits for a probe response frame from neighboring APs operating on that same channel. In IEEE 802.16e WiMAX networks [13], when the serving BS has obtained downlink/uplink channel descriptor (DCD/UCD) information *via* the wired backbone networks, it broadcasts this information periodically using MOB_NBR-ADV messages. To start the scanning process, the MS sends MOB_SCN-REQ to its serving BS and waits for a response MOB_SCN-RSP that includes the synchronization parameter set of the neighbor BS.

Existing scanning algorithms can be classified into two categories. While the first category aims to reduce the total required channel scanning time by reducing the number of channels to scan, the goal of the second category is to minimize the QoS degradation during the scanning period.

Several studies have attempted to reduce the scanning time in WLAN. In a recent study [2], the MS stops scanning a channel once all expected probe responses have been received before a pre-determined maximum waiting time. The actual scanning time is adjusted according to the number of probe response collisions. In References [3,4], each AP keeps information about neighboring APs in a neighbor graph data structure that is then made available to wireless stations. In Reference [5], each MS stores scan results in an AP cache for future use. When the MS moves to a previously visited location, the channels that have APs can be recognized through a check of the AP cache. Reference [6] proposes a procedure that selectively scans only one probable channel based on a weighted channel list. A special agent gathers handover experiences from all selective scan-aware MSs, which increases the probability of having an AP at each channel. Most history-based scanning algorithms often lead to poor handover performance since most handovers will target a fewer number of recently visited APs. To reduce the probe delay for WLAN, the spatiotemporal approach (DeuceScan) was proposed in Reference [7], in which an MS maintains a spatiotemporal triangle list containing time and location data of candidate APs. If an MS re-enters a location it has previously traversed, it can extract the relevant spatiotemporal triangle list and begin scanning for candidate APs. However, the candidate set decision is based only on the signal strength and requires precise location information in order to acquire a proper triangle list. In Reference [8], a different reply channel for probe responses is used. The scanning node leaves its current channel to send probe requests on all channels to be scanned, and then returns directly to its original channel and can continue to send and receive data packets while awaiting probe responses that are routed back *via* backbone networks. Reference [9] proposes a low-cost scanning technique (SyncScan) for the continuous tracking of nearby BSs by synchronizing short listening periods at the client using periodic transmissions from each BS. This method can reduce the costs of continuous full scanning and leads to better handover decisions by continuously monitoring the signal quality of multiple APs. However, it requires precisely controlled beacon transmission scheduling and synchronization between neighboring APs. For WLAN networks operated individually, this may not be practical.

To overcome serious quality degradations during scanning, Reference [15] proposes an adaptive channel scanning (ACS) algorithm that determines the duration and frequency of channel scanning to facilitate the discovery of neighboring BSs and handovers across multiple IEEE 802.16 networks. In ACS, a BS determines the scanning schedules of mobile nodes based on the delay requirements. One channel scanning time is simply divided into multiple scanning intervals. In Reference [16], ProactiveScan was proposed, in which time-consuming channel scan is decoupled from the actual handover. This method separates the handover-triggering step into a proactive scan trigger and the actual handover trigger. To scan channels, the scanning phase is divided into scan times and they are interleaved into normal ongoing data traffic. The scan time is designed to be small enough (average 10 ms) so that it introduces no human-detectable interruptions to ongoing VoIP sessions. In Reference [16], the scan interval is determined from the several fixed values. The scan time and scan interval are not dynamically adjusted in order to guarantee different QoS requirements. In Reference [17], the voice activity of a two-way VoIP conversation is taken into account, and channels are scanned during mutual silence periods. Scanning is interrupted if the MS has VoIP packets to send to a peer user. As it is not possible to know whether or not the peer user has packets to send, packets transmitted from the peer user can be lost during the scan. Some channels may be skipped in the second round if no beacons are detected in the first round. With passive scanning, as the MS can recognize the beacon interval of each AP after the first round, the channel with the nearest beacon arrival time is selected as the

DOI: 10.1002/wcm

next channel to be scanned. Reference [18] proposes a smooth MAC layer handover scheme for IEEE 802.11 WLAN networks, in which 11 wireless LAN channels were divided into groups. Instead of consecutively scanning all channels, the wireless station temporarily halts the channel scanning operation and switches back to its normal data-transmission mode. The wireless station switches to the channel-scanning mode again and discovers APs working on another group of channels. To adjust the threshold for the channel scanning operation dynamically, an adaptive threshold control mechanism is proposed. However, in Reference [18], scanning-channel grouping methods or a time-interval decision mechanism that switches between scanning and data transmission modes was not presented in details.

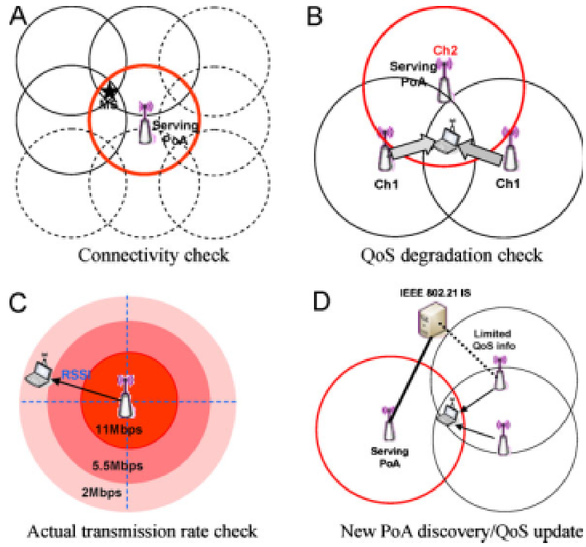


Fig. 1. Scanning objectives after obtaining the neighbor information. (a) Connectivity check, (b) QoS degradation check, (c) actual transmission rate check, and (d) new PoA discovery/QoS update.

3. Channel Scanning for Seamless Handovers

In this section, we present the general framework for policy-based scanning with support for QOS. We first discuss the motivations for scanning and review the different types of channel scanning before we describe the main components of the proposed architecture.

3.1. Motivation

Channel scanning is generally needed in order to find a suitable target network. Although some information about neighboring networks including QOS conditions may be obtained prior to scanning (as will be discussed in greater details in the proposed framework) and can actually help expedite the scanning, scanning may still be required in order to achieve some of the following objectives:

(i) *Check PoA connectivity* (Figure 1a): The MS may not to able to connect to certain PoAs if it is outside of the radio ranges of these PoAs. As shown in an example in Figure 1a, although the serving PoA has eight neighboring PoAs, only three PoAs are reachable by the MS.

(ii) *Confirm QoS levels* (Figure 1b): The current QoS level of a PoA (network wide) can differ from the actual QoS received by the MS. The MS may have multiple PoAs (in WLAN-like networks) that are operating on the same channel. In this case, if the MS is located in an overlapping area of these PoAs, it will experience strong interference, packet collisions, and long packet transmission delays, implying that the actual QoS will be worse than the network-wide QoS.

(iii) *Verify the actual transmission rate* (Figure 1c): The actual transmission rate depends on the signal strength (location dependent) in multi-rate networks. The transmission rate is one of the most important QoS parameters, but it is typically unknown before a scan is performed.

(iv) *Discover new PoAs and update their QoS conditions* (Figure 1d): By scanning, the MS can discover new PoAs that were not known by the IEEE 802.21 IS or by the serving PoA. New or updated QoS information can be also obtained by scanning.

3.2. Channel Scanning Types: Horizontal *Versus* Vertical Scanning

Mainly, there are two types of channel scanning. One type of scanning occurs simultaneously while data are being transmitted, whereas another type of scanning has to interrupt the transmission of data before it can proceed. Let vertical scanning refer to channel scanning performed in parallel with data transmission since this type of scanning often leads to a vertical handover. Similarly, let horizontal scanning refer to scanning that interrupts the transmission of data.

Vertical scanning assumes that scanning is independent of current data transmission and that one or more network interfaces are available. The networks scanned

DOI: 10.1002/wcm

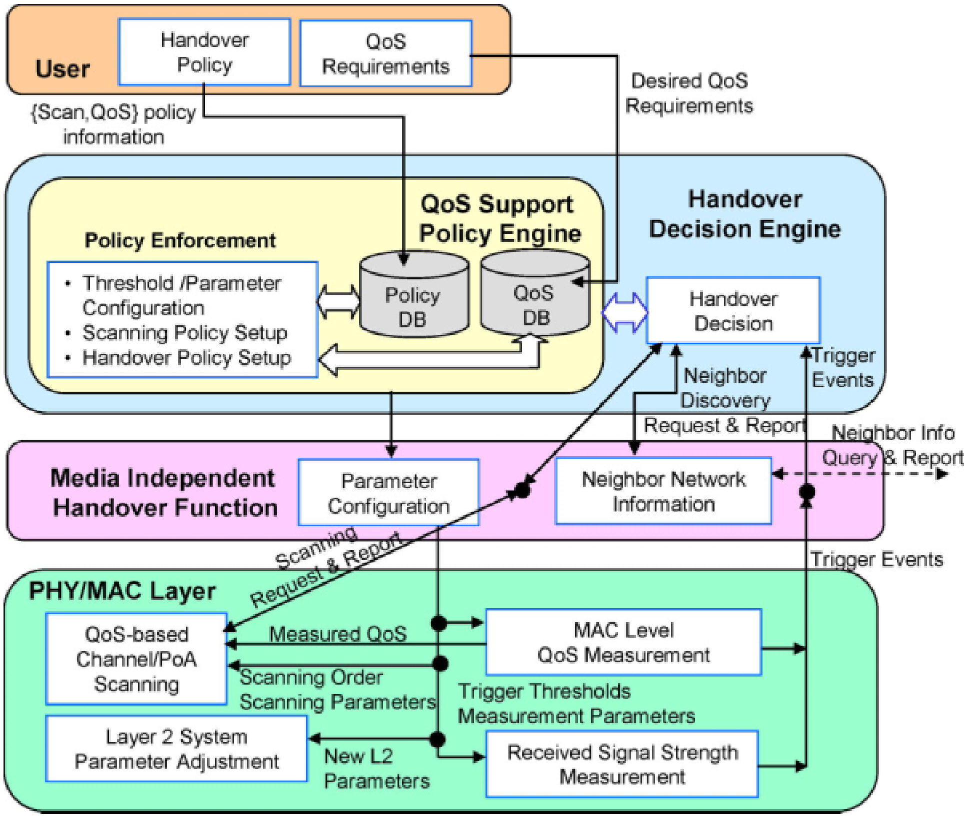


Fig. 2. General framework for QOS supported scanning and handovers.

are generally of different types, as in the case of multiple radio devices, but not necessarily. In horizontal scanning, the MS needs to switch to other channels in order to track the presence of other POAs. This may cause severe delays, jitter, and packet losses on the current service. Note that the framework we propose works with either type of channel scanning.

Generally the horizontal and vertical scanning to discover a target network and PoA can be performed periodic or aperiodic (on demand) manner. In this paper, we mainly focus on aperiodic scanning. If the cost (defined in Section 4) of the current network is not acceptable, then to find out other cost effective network the vertical scanning is activated. When the current PoA's measured QoS level is less then the pre-determined value, the horizontal scanning (QoS support scanning or urgent non-QoS support scanning) is initiated. In Section 4, we have defined horizontal or vertical handover policies for various network conditions.

3.3. QoS Supported Scanning Architecture

In this section, the architecture for QOS-supported scanning and handover is presented. Figure 2 shows the proposed QoS support scanning architecture based on a cross-layer design for seamless handovers. Given this architecture and considering some of the main components that affect scanning, namely, policy, neighbor information, QOS requirements, and network measurements, we observe that the first two components affect what sequence of channels to scan, while the following two determine when a scan is needed (or triggered) and how long and how often the scanning should be performed. Both aspects of scanning will be investigated in details in the next section.

As shown in Figure 2, the user sets up scanning and QoS policies and configures them into a policy database. When an application session is initialized, the user also installs the desired QoS requirements on the QoS database. The QoS support policy engine determines thresholds for link-layer triggering and scanning rules. Through IEEE 802.21 MIHF [14] primitives, physical (PHY) and MAC layer parameters are delivered to PHY/MAC-layer functional modules. The MS monitors the PHY- and MAC-layer QoS performance based on the delivered measurement parameters, including the QoS metric, measurement interval, and measurement methods. If the measured QoS satisfac-

DOI: 10.1002/wcm

tion degree crosses pre-determined scanning thresholds, the trigger events are notified to a handover decision engine. The handover decision engine requests the neighbor network status information from the MIHF layer. With the neighbor network information, the QoS support policy engine determines the network and PoA scanning order. The scanning order and policy is then delivered to the scanning functional module at the PHY/MAC layer; based on the current QoS measurement and required QoS level, the scanning module dynamically sets up the scanning time and interval. The scanning results are reported to the handover decision engine. If the scan time is too short to perform any type of scan, this is reported to the handover decision engine and the decision engine can request Layer 2 system adjustments to the Layer 2 parameters or can decide to begin vertical scanning immediately. When all of the scanning procedures are finished, the handover decision engine selects a target network and PoA and performs either a horizontal or a vertical handover.

4. Proposed Policy-based Scanning with QOS Support (PSQS)

In this section, QoS support link triggering mechanism, establishment of scan sequence, and QoS support scan frequency and duration decision mechanism are presented. To support the required QoS during scanning, several new approaches are considered as follows:

(i) *The performance criterion for handover triggering and the selection of a target PoA*: In this paper, a new criterion (QoS satisfaction degree) that is used to monitor the current service quality, to initiate a number of link layer triggers, to set up the scanning order of neighboring PoAs, and to determine a target PoA is defined.
(ii) *Layer 2 triggering for scanning and handovers*: QoS-based handovers should be finished before the QoS provided by the currently serving PoA crosses a pre-determined link-down QoS level. In the proposed approach, depending on the time-critical levels of the handover actions, different link level triggers are defined.
(iii) *Network and PoA scan ordering*: The network and PoA scan order is determined in accordance with user preferences, the expected level of QoS, and the degree of possible interference. This enables rapid discovery of the target PoA.
(iv) *QoS supports scan time scheduling*: To provide the desired level of QoS in terms of delays and loss ratios, a short scanning time and normal data transmissions are interleaved. Based on the required and measured QoS, both the scanning time and length of the interleaving interval are dynamically adjusted.
(v) *Action grouping for one scanning time*: During each short scanning time, the actions (i.e., scanning message exchanges) that can be grouped together and the number of channels or PoAs that can be scanned are automatically determined. For this action grouping, a new concept of 'independent scanning piece' is proposed.
(vi) *System parameter adjustment*: When the determined scanning time is too short for scanning, a Layer 2 parameter (timer or counter) adjustment can be attempted. If the desired parameter adjustment is not available, vertical scanning is immediately performed.

Given the many variables used in this paper, a glossary of variable names and definitions is provided in Table I.

4.1. Link Layer Triggers for Scanning

For QoS-based scanning and handovers, the '*QoS satisfaction degree*' is defined as a link quality metric in this paper. It is a function of the QoS metrics defined in Equation (1). Θ_i represents ith QoS component and q is the number of components. Different set of QoS components can be defined in accordance with QoS definition policy. Typical QoS components include delay, delay jitter, packet loss ratio, and transmission rate. The QoS satisfaction degree can be defined as the minimum value from all the QoS components or a weighted average, as shown in Equation (2), depending on the user requirements. $\mathrm{QoS}_c^{n,k}(t) \geq 1$ is desired. The measurement is performed at the MAC layer, and most wireless network standards, such as IEEE 802.11k [11], support these QoS measurements.

$$\mathrm{QoS}_c^{n,k}(t) = F\left(\Theta_1, \ldots, \Theta_q\right) \tag{1}$$

$$\begin{aligned} F(\bullet) \quad & = \min\left\{\frac{R\_\Theta_c^1}{M\_\Theta_c^1(t)}, \frac{R\_\Theta_c^2}{M\_\Theta_c^2(t)}, \ldots, \frac{R\_\Theta_c^q}{M\_\Theta_c^q(t)}\right\} \\ \text{or} \quad & = w_c(\Theta_1)\frac{R\_\Theta_c^1}{M\_\Theta_c^1(t)} + w_c(\Theta_2)\frac{R\_\Theta_c^2}{M\_\Theta_c^2(t)} \\ & \quad + \cdots + w_c\left(\Theta_q\right)\frac{R\_\Theta_c^q}{M\_\Theta_c^q(t)} \\ & \quad w_c(\Theta_1) + w_c(\Theta_2) + \cdots + w_c\left(\Theta_q\right) \\ & = 1\left(\forall c \in \bar{C}\right) \end{aligned} \tag{2}$$

DOI: 10.1002/wcm

Table I. Glossary of variable definitions.

Variable	Definition
N_w	Networks of network type w
$C(N_w)$	Cost of network N_w
N_{CP}	Controlled provisioning network
N_{UP}	Uncontrolled provisioning network
PoA_p	pth point of attachment (PoA)
$\mathrm{QoS}_c\left(\mathrm{PoA}_p\right)$	QoS satisfaction degree of PoA_p for class c
$\bar{P}_l$	Set of PoAs that use the channel indexed l
$\bar{G}_j$	Set of channels that have j PoAs
$\mathrm{SIT}_c(n)$	nth service interruption time of class c
$\mathrm{SII}_c(n)$	nth service interruption interval of class c
$\mathrm{MaxSIT}_c(n)$	Maximum $\mathrm{SIT}_c(n)$ bound
$\mathrm{MinSII}_c(n)$	Minimum $\mathrm{SII}_c(n)$ bound
R_delay_c, R_loss_c	Required delay and loss ratio for class c
$M\_\mathrm{delay}_c(t)$, $M\_\mathrm{loss}_c(t)$	Measured delay and loss ratio for class c at time t
$R\_\mathrm{delay}_c^S$, $R\_\mathrm{loss}_c^S$	Required delay and loss ratio for class c during the scanning period
R_d, R_l	Tolerable delay and loss degradation ratio during the scanning period
sp_i	ith scanning piece
$\bar{S}_l$	Set of scanning pieces to scan channel l
$\bar{S}$	Set of scanning pieces to scan all neighboring channels
T_k^i	kth timer value for sp_i
$T_k^{i(a)}$	Actual used time for a timer T_k^i operation
$\bar{T}_i$	Set of correlated timers for sp_i $\bar{T}_i = \left\{T_1^i, \ldots, T_R^i\right\}$
t_sp_i	Required time for sp_i
$t\_\mathrm{sp}_i^{(a)}$	Actual used time for a sp_i operation
$\bar{V}_n$	Set of scanning pieces for nth SIT

where c is a service class index from the service class set $\bar{C}$, n is a network type (e.g., WLAN or WiMAX), k is the current PoA index, and $w_c(\Theta_i)$ is a weighting factor for the QoS metric Θ_i of class c. $M\_\Theta_c^i(t)$ indicates the measured QoS value for the metric Θ_i for class c at time t. $R\_\Theta_c^i$ is the required QoS performance for the metric Θ_i for class c.

In order to separate the actual handover execution, which is the most important time-critical behavior, with QoS support scanning, two link-level scanning triggers are newly defined in addition to conventional link triggers. Link_Going_Down (LGD) and Link_Down (LD) triggers have been defined in most of network standards to trigger handover execution and to indicate actual link down event and they are very time-critical event triggers. The following scan triggers are fired before the LGD trigger.

- *QoS_Scan_Start trigger*: If the measured QoS crosses a pre-defined QoS_Scan_Start threshold, the link layer generates this trigger event to the upper layer (handover decision engine). After obtaining neighbor network and PoA information, the MS sets up the scanning order for candidate PoAs and performs QoS support scanning.
- *Vertical_Scan_Start trigger*: This is an optional trigger for vertical interface scanning. This trigger is generated when the measured QoS crosses a predefined Vertical_Scan_Start threshold. Before this triggering event, if the MS is unable to locate an available horizontal target PoA, the MS commences vertical scanning concurrently with the current horizontal scanning process. It should be noted that vertical scanning does not interrupt the current services.

Figure 3 shows the time sequence of the proposed approach. If the current measured QoS satisfaction degree is lower than the QoS_Scan_Start threshold, the MS sends query messages to the IEEE 802.21 information server (IS) to obtain neighbor PoA information along with their QoS supportabilities. The neighbor information can also be obtained from the currently serving PoA using neighbor advertisement messages in WLAN and WiMAX systems. The MS maintains a Neighbor PoA QoS table, as shown in Figure 4. Among all neighbor networks and PoAs, the PoAs with currently measured QoS levels that are lower than the user-desired QoS level are eliminated from the candidate list for scanning. If the MS cannot determine the

DOI: 10.1002/wcm

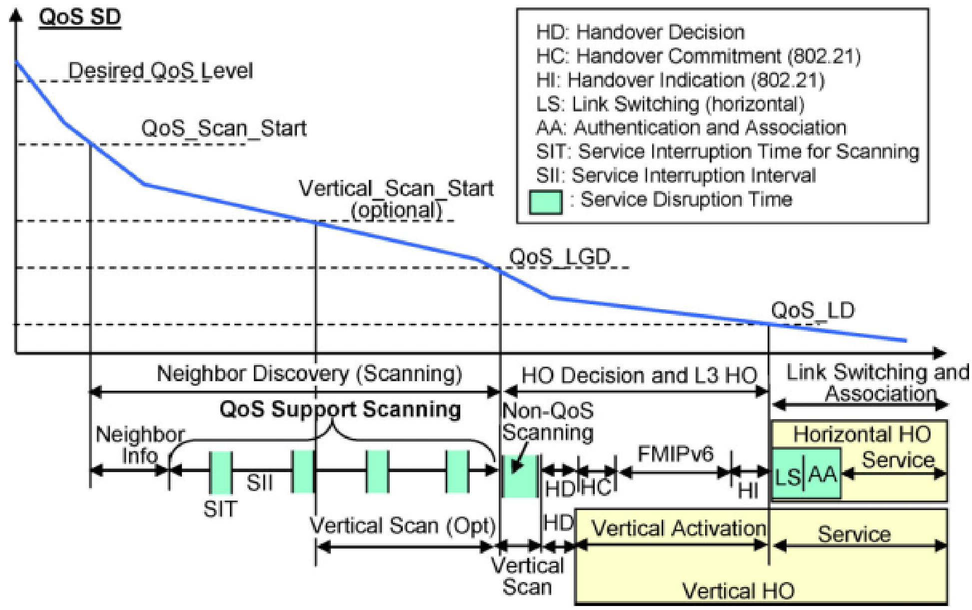


Fig. 3. QoS support scanning and handover procedure.

Type	PoA Address		Ch Num	QoS	Additional Admission Capacity
HI	PoA_1		1	QoS^{1}_{c}={delay,loss, rate}	Yes
	PoA_2		3	QoS^{2}_{c}={delay,loss, rate}	No
	PoA_3		4	NA	NA
	PoA_4		3	QoS^{4}_{c}={delay,loss, rate}	Yes
	...		...	...	...
	PoA_n		2	QoS^{n}_{c}={delay,loss, rate}	Yes
VI	VI 1	PoA_i	7	QoS^{i}_{c}={delay,loss, rate}	No
		...	...	...	...
		...	...	...	...
	VI k	PoA_j	9	QoS^{j}_{c}={delay,loss, rate}	Yes
		...	...	...	...

HI: Horizontal Interface, VI: Vertical Interface, NA: Not Available

Fig. 4. Example of the neighbor PoA QoS table.

QoS levels for some PoAs because they are not connected to the IEEE 802.21 server or because the QoS information for the determination of the QoS satisfaction degree is not sufficient, the MS includes the PoAs in the candidate list.

When the measured QoS crosses the QoS Link_Going_Down (QoS_LGD) trigger threshold and if a target PoA that can serve the QoS requirements of the user has not been found *via* QoS support scanning, the MS performs non-QoS support horizontal scanning. With non-QoS support scanning, to find the target PoA, the MS consecutively scans all remained channels, as in this case the link down event is imminent and the scanning is a time-critical process. The QoS_LGD trigger also activates vertical scanning if it was not activated by the Vertical_Scan_Start trigger. If there is no target PoA for a horizontal handover or if an MS user prefers a vertical handover to a different network PoA, the MS initiates a vertical handover. The MS performs a Layer 3 fast handover before Layer 2 switching when the target PoA is not on the same subnet.

4.2. Establishing a Scanning Sequence

In the proposed scanning policy, scan ordering involves two steps: network scan ordering and PoA scan ordering. When scanning is required, the MS determines the sequence of networks to scan as a first step based on a pre-defined ordering rule. In the second step, within a network if there are multiple channels or PoAs to scan, the MS also determines the scan sequence of the channels or PoAs. To set up different ordering policies for different network types, two network classes are defined in this paper.

- *Controlled provisioning networks* (N_{CP}): These networks are well provisioned to avoid interference that can occur between neighbors. The same frequency channel is not used by adjacent PoAs and QoS is generally guaranteed once the MS is admitted, such as in WiMAX and cellular communication systems.
- *Uncontrolled provisioning networks* (N_{UP}): These networks can be installed in an uncoordinated manner, as with WLAN APs. The same channel can be used by adjacent PoAs. In these networks, the MS of the overlapping area may experience strong interference from multiple co-channel neighbor PoAs.

DOI: 10.1002/wcm

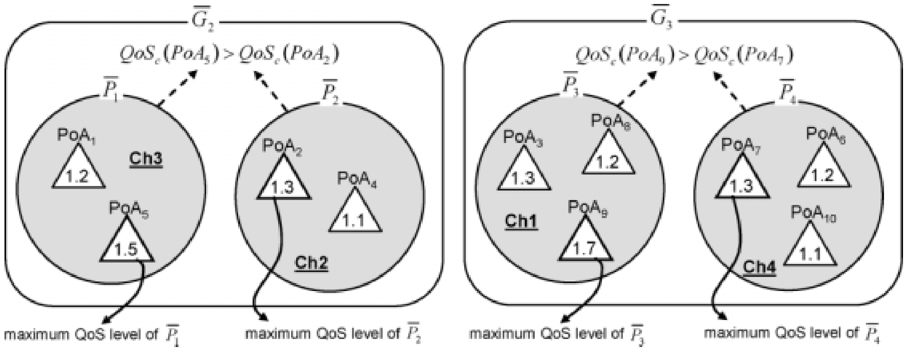


Fig. 5. Scan-ordering example for uncontrolled provisioning networks.

To determine the network scanning order, the network cost $C(N_w)$ for network type w is defined as a function of the monetary cost, average QoS, average power consumption, and user preference, as in Equation (3). QoS_{N_w} is determined as the average QoS level of the neighboring PoAs of network type w. If QoS information is not available for a specific network type, the expected QoS level can be used instead of the actual measured QoS. A user of the MS can define the function of Equation (3) in real environments.

$$C(N_w) = \mathbf{F}\left(\mathrm{Cost}_{N_w}, \mathrm{QoS}_{N_w}, \mathrm{Power}_{N_w}, \mathrm{User\_Pr\,eference}_{N_w}\right) \tag{3}$$

For horizontal and vertical network scanning, different scanning policies apply. As shown in 'Scanning Policy 1' below, horizontal scanning is always performed in an aperiodic manner (only when a scanning trigger is generated) while vertical scanning can be performed in either a periodic or an aperiodic manner.

SCANNING POLICY 1 - Horizontal and vertical scanning policy

i) Horizontal scanning

Condition 1: The current PoA QoS is less than the QoS_Scan_Start threshold.
Action 1: The MS determines the PoA scan order for the current network and performs QoS scanning.

Condition 2: The current PoA QoS is lower than the QoS_LGD threshold and no target PoA is determined.
Action 2: Non-QoS support consecutive scanning is performed for the remaining PoAs

ii) Vertical scanning

Condition 1: The current PoA network cost is allowable by the user and the current PoA QoS is lower than the Vertical_Scan_Sart threshold.
Action 1: Vertical interfaces are activated sequentially according to the network scan order and consecutive PoA scanning is performed by the PoA scan order until the MS finds a target PoA.

Condition 2: The current PoA network cost is allowable by the user, the current PoA QoS is lower than QoS_LGD threshold, vertical scanning is not activated, and no target PoA has been determined.
Action 2: Vertical interfaces are activated sequentially according to the network scan order and consecutive PoA scanning is performed by the PoA scan order until the MS finds a target PoA.

Condition 3: If the current network cost is not allowable.
Action 3: The MS performs vertical scanning periodically according to the network and PoA scan order to locate a better network type and PoA.

DOI: 10.1002/wcm

The network scan-ordering rule is given as Rule 1.

Rule 1 (Network scan ordering): For all network types in which the MS has network interfaces, $\forall N_w, (1 \le w \le n)$ the network scan order is determined as

$$Network\_Scan\_Order = \{N_1, N_2, \cdots, N_w, N_{w+1}, \cdots N_n\} \quad \text{where } C(N_w) \le C(N_{w+1}) \tag{4}$$

Within a single network type, there can be multiple neighboring PoAs to scan. The PoA scan order is determined by Rule 2, in which scanning is performed for the neighboring PoAs of the scanning candidate list $\bar{L}$. Any neighbor PoA in which the measured QoS level is larger than or equal to the required QoS $\left(R\_\mathrm{QoS}_c\right)$ or in which the current QoS information is not available is included in PoA scanning list $\bar{L}$. In uncontrolled provisioning networks, some neighboring PoAs can use the same channel and can operate in the same region. If the MS is moving to or located in overlapping area by PoAs that use the same channel, the MS may experience an unacceptable degree of QoS degradation. Therefore, for uncontrolled provisioning networks, the channels with higher numbers of multiple neighboring PoAs are scanned later. Figure 5 shows an example of PoA scan ordering in an uncontrolled provisioning network. The $\bar{G}_2$ group is scanned before the $\bar{G}_3$ group as it has fewer PoAs using the same channel. In this example, the scan order is $\bar{P}_1 \to \bar{P}_2 \to \bar{P}_3 \to \bar{P}_4$ (i.e., the channel order is $\mathrm{Ch}_3 \to \mathrm{Ch}_2 \to \mathrm{Ch}_1 \to \mathrm{Ch}_4$).

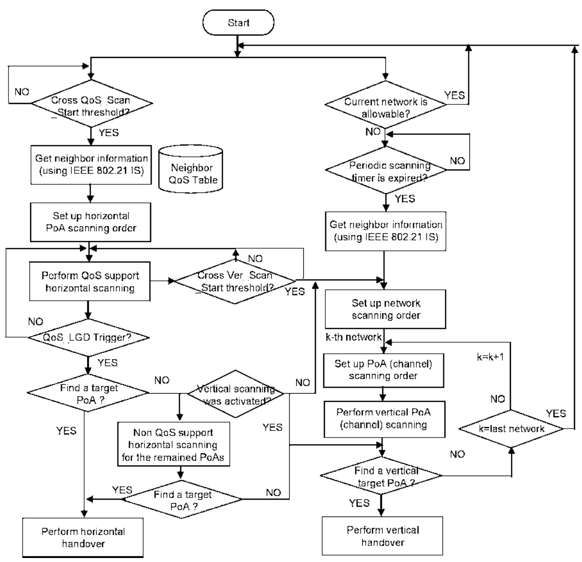


Fig. 6. Proposed horizontal and vertical scanning procedure.

Rule 2 (PoA scan ordering): For $\forall PoA_p : \left[QoS_c(PoA_p) \ge R\_QoS_c\right]$ or $\left[QoS_c(PoA_p) = \text{Not Available}\right]$

$$N_{CP} = \{PoA_1, PoA_2, \cdots, PoA_p, PoA_{p+1}, \cdots, PoA_P\}, \quad \text{where} \, QoS_c(PoA_p) \ge QoS_c(PoA_{p+1}) \quad \forall p \in \bar{L} \tag{5}$$

$$N_{UP} = \left\{ \underbrace{\bar{P}_1, \bar{P}_2, \cdots, \bar{P}_{g_1}}_{\bar{G}_1} \underbrace{\bar{P}_{g_1+1}, \bar{P}_{g_1+2}, \cdots, \bar{P}_{g_1+g_2}}_{\bar{G}_2} \underbrace{\quad \cdots \quad}_{\cdots} \underbrace{\bar{P}_{g_1+\cdots+g_{k-1}+1}, \bar{P}_{g_1+\cdots+g_{k-1}+2}, \cdots, \bar{P}_{g_1+\cdots+g_{k-1}+g_k}}_{\bar{G}_k} \right\}$$

$$\text{where } \bar{P}_l = \text{set of PoAs that use } l\text{-th channel index}, \ \bar{G}_j = \text{set of channels that have } j \text{ PoAs} \tag{6}$$

$$\text{for } \forall j, \quad \max_{\forall PoA_p \in \bar{P}_l} QoS_c(PoA_p) \ge \max_{\forall PoA_p \in \bar{P}_{l+1}} QoS_c(PoA_p), \ \forall \bar{P}_l, \bar{P}_{l+1} \in \bar{G}_j$$

Figure 6 shows the overall proposed horizontal and vertical scanning procedure.

4.3. Determining a Scanning Frequency and Duration

In conventional horizontal channel scanning that uses the same network interface type as the current serving PoA, candidate neighboring channels are generally scanned consecutively. A service interruption of several hundred milliseconds occurs, during which wireless stations cannot send or receive data packets. The main objective of the proposed QoS support scanning is to minimize this type of disruptive scanning effect on application traffic and to guarantee the user QoS demands during scanning procedures. Instead of consecutively scanning all channels, the length of every scanning time and interval between scans are dynamically determined in the proposed scanning mechanism based on the QoS requirements and current

DOI: 10.1002/wcm

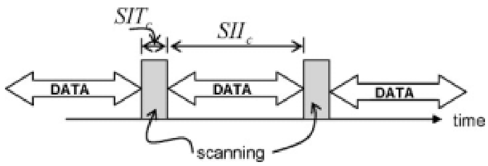


Fig. 7. SIT and SII definition.

QoS measurements. A group of scanning actions that can be executed during each scanning time is adaptively selected by the proposed grouping rule.

Two scanning parameters are defined for each service class c, SIT and service interruption interval (SII), as shown in Figure 7. SIT and SII indicate the actual scanning time (during which the current service is interrupted) and the time from the end of the current SIT to the beginning of the next SIT, respectively.

The desired QoS level in terms of the delay and loss ratio during a scanning period ($R\_\text{delay}_c^S$ and $R\_\text{loss}_c^S$) can be set by the user to be slightly lower than the QoS level under normal operations, as in Equation (7). In Equation (8), R_d and R_l represent tolerable delay and loss degradation ratios, respectively.

$$R\_\text{delay}_c^S \geq R\_\text{delay}_c,\ R\_\text{loss}_c^S \geq R\_\text{loss}_c \tag{7}$$

$$\begin{cases} R\_\text{delay}_c^S = R_d \times R\_\text{delay}_c \\ 10\log_{10}\left(R\_\text{loss}_c^S\right) = \frac{10\log_{10}(R\_\text{loss}_c)}{R_l},\ R_d, R_l \geq 1 \end{cases} \tag{8}$$

To derive scanning parameter values adaptively at any time t, constant bit rate (CBR) applications are assumed. As shown in Figure 8a, during the SIT, the MS cannot send data to the currently serving PoA. The data packets are stored in the buffer of the MS and they can be sent after the SIT has finished. Therefore, the SIT generates additional delay. The delay during the scanning period should be kept lower than the desired delay $R\_\text{delay}_c^S$, as in

$$\left(\text{SIT}_c(n) + M\_\text{delay}_c(t_1)\right) \leq R\_\text{delay}_c^S \tag{9}$$

where the nth SIT starts at time t_1 and $M\_\text{delay}_c(t_1)$ is the measured delay at time t_1. Therefore, the maximum bound of the SIT is given as in Equation (11).

$$\text{SIT}_c(n) \leq \left(R\_\text{delay}_c^S - M\_\text{delay}_c(t_1)\right) \tag{10}$$

$$\text{MaxSIT}_c(n) = \left(R\_\text{delay}_c^S - M\_\text{delay}_c(t_1)\right) \tag{11}$$

A management frame is not used in a WLAN to inform the currently serving AP of a pending scan by the MS; thus, all packets from the AP can be lost, as shown in the case in Figure 8b. For other networks such as WiMAX, a notification message regarding a scan is sent to the current PoA, allowing it to start buffering during the scanning time. However, if many MSs exist while the scan takes place and if the PoA buffer size is too small to store packets from all MSs, packets can be lost during SIT. When the SIT starts at t_1 and finishes at t_2, as in Figure 8b, in order to guarantee that the packet loss ratio during $(t_1, t_1 + \text{SIT}_c(n) + \text{SII}_c(n)]$ is less than or equal to the desired loss level $\left(R\_\text{loss}_c^S\right)$, $\text{SII}(n)$ should be determined using Equation (12) and the minimum value used in this case is given in Equation (14).

$$\frac{\text{Average lost packets during } (t_1, t_1 + \text{SIT}_c(n) + \text{SII}_c(n)]}{\text{Transmitted packets during } (t_1, t_1 + \text{SIT}_c(n) + \text{SII}_c(n)]} \leq R\_\text{loss}_c^S$$

$$\frac{C \times \text{SIT}_c(n) + C \times \text{SII}_c(n) \times M\_\text{loss}_c(t_2)}{C \times (\text{SIT}_c(n) + \text{SII}_c(n))} \leq R\_\text{loss}_c^S \tag{12}$$

$$\text{SII}_c(n) \geq \frac{\left(1 - R\_\text{loss}_c^S\right) \times \text{SIT}_c(n)}{R\_\text{loss}_c^S - M\_\text{loss}_c(t_2)} \tag{13}$$

$$\text{MinSII}_c(n) = \frac{\left(1 - R\_\text{loss}_c^S\right) \times \text{SIT}_c(n)}{R\_\text{loss}_c^S - M\_\text{loss}_c(t_2)} \tag{14}$$

where, $M\_\text{loss}_c(t_2)$ is the measured loss ratio at time t_2 and C is the packet transmission rate (packets/s). In the proposed mechanism, the measurement periods for $M\_\text{delay}(t)$ and $M\_\text{loss}(t)$ do not include the SITs. Therefore, in Figure 8b, $M\_\text{loss}_c(t_2) = M\_\text{loss}_c(t_1)$.

In this paper, the scanning actions that can be performed during an SIT time are considered. A SIT can be too short to handle all actions for single channel scanning or it can be large enough to scan multiple channels. The new concept of '*independent scanning pieces*' is defined to address this. If two procedures (message exchanges) are not correlated (controlled by the same timer) or if the correlated timer values for the next message transmissions at the MS side are relatively long compared with the SII, they are considered as independent scanning pieces. For example, for active scanning in a WLAN, probe request and probe response messages are not independent and should be handled as a set. Figure 9 shows an example of this. In order

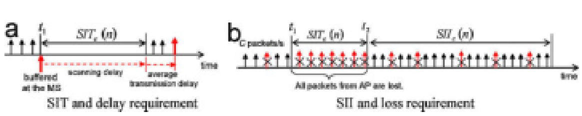


Fig. 8. SIT and SII derivation. (a) SIT and delay requirement, (b) SII and loss requirement.

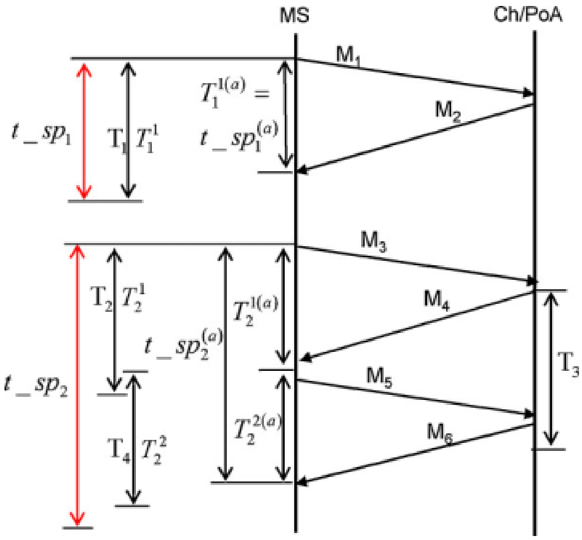


Fig. 9. Independent scanning pieces.

to scan one channel (or PoA), six message exchanges are required. The message exchange M_1 and M_2 are controlled by timer T_1 and messages from M_3 to M_6 are controlled by correlated timers T_2, T_3, and T_4. It is assumed that the time requirement to transmit M_3 message after receiving M_2 is larger than SII value. Therefore, in this example we have two independent scanning pieces.

Let define sp_i be the ith independent scanning piece, t_sp_i be the required time for independent scanning piece sp_i, T_i^j be the jth timer of the ith independent scanning piece, and $\bar{T}_i = \{T_i^1, T_i^2, \ldots, T_i^R\}$ be a set of correlated timers for sp_i on the MS side. In Figure 9, $\bar{T}_2 = \{T_2, T_4\}$. The required time for the ith independent scanning piece (t_sp_i) is derived as

$$t\_\mathrm{sp}_i = \sum_{k=1}^{R} T_i^k \tag{15}$$

The actual scanning time for sp_i can be less than t_sp_i because the MS can finish the required message exchanges before the related timer expirations. Let $T_i^{k(a)}$ be the actual time used by timer T_i^k operations. The actual time used by sp_i, $t\_\mathrm{sp}_i^{(a)}$, is given as

$$t\_\mathrm{sp}_i^{(a)} = \sum_{k=1}^{R} T_i^{k(a)} \tag{16}$$

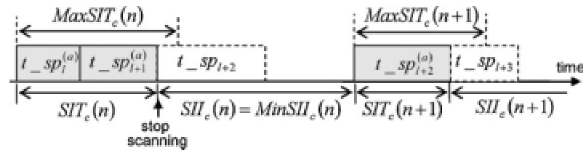


Fig. 10. Dynamic SIT and SII determination.

As in Equations (11) and (14), $\mathrm{MaxSIT}_c(n)$ and $\mathrm{MinSII}_c(n)$ are determined at the nth SIT and SII start time, respectively. However, $\mathrm{SIT}_c(n)$ and $\mathrm{SII}_c(n)$ are dynamically determined based on the actually used time for each independent scanning piece. If N is the number of channels (or PoA) to scan, the set of independent scanning pieces to scan the all required channels $\bar{S}$ is given as

$$\bar{S} = \left\{ \underbrace{\mathrm{sp}_1, \mathrm{sp}_2, \ldots, \mathrm{sp}_{S_1}}_{\bar{S}_1}, \underbrace{\mathrm{sp}_1, \mathrm{sp}_2, \ldots, \mathrm{sp}_{S_2}}_{\bar{S}_2}, \ldots, \underbrace{\mathrm{sp}_1, \mathrm{sp}_2, \ldots, \mathrm{sp}_{S_N}}_{\bar{S}_N} \right\} \tag{17}$$

where $\bar{S}_l = \{\mathrm{sp}_1, \mathrm{sp}_2, \ldots, \mathrm{sp}_{S_l}\}$ is the set of scanning pieces for channel (or PoA) l, and S_l is the number of independent scanning pieces for scanning channel l. The sequence of channels of $\{\bar{S}_1, \bar{S}_2, \ldots, \bar{S}_N\}$ is ordered by the channel scan-ordering rule. In general, for a given network type, the number of scanning pieces necessary to scan each channel (or PoA) is constant, as $S_1 = S_2 = \cdots = S_N = S$, implying that set $\bar{S}$ can be rearranged, as in

$$\bar{S} = \{\mathrm{sp}_1, \mathrm{sp}_2, \ldots, \mathrm{sp}_{N \cdot S}\} \tag{18}$$

where, sp_i is the ith sp in $\bar{S}$.

The durations for $\mathrm{SIT}_c(n)$ and $\mathrm{SII}_c(n)$, which both start at time t_1 and t_2, respectively, are derived by Rule 3. As shown in Figure 10, $\mathrm{SIT}_c(n)$ is determined to capture all of independent pieces that can be performed within the maximum SIT bound. If the time required to perform the next independent piece is longer than the time remaining in the budget of $\mathrm{MaxSIT}_c(n)$, the current $\mathrm{SIT}_c(n)$ finishes and normal data transmissions are resumed. $\mathrm{SII}_c(n)$ is determined with $\mathrm{MinSII}_c(n)$.

DOI: 10.1002/wcm

Rule 3 (Dynamic SIT and SII determination): For the set of $\overline{S}=\{sp_1, sp_2, \cdots, sp_{N\cdot S}\}$:

If during $SIT_c(n-1)$, independent scanning pieces up to sp_{l-1} have been performed, then

$\overline{V}_n=\{\phi\}$, $next=l$, $MaxSIT_c(n)=\left(R\_delay_c^S - M\_delay_c(t_1)\right)$

where $\overline{V}_n$ is the set of independent scanning pieces for n-th SIT.

while $\left(\mathrm{sum}_{\forall sp_i \in \overline{V}_n}\{t\_sp_i^{(a)}\}+t\_sp_{next} \le MaxSIT(n)\right)$ {

Perform sp_{next}, $sp_{next} \rightarrow \overline{V}_n$, $next = next+1$;

}

Finish n-th SIT scanning, $$SIT_c(n)=\mathrm{sum}_{\forall sp_i \in \overline{V}_n}\{t\_sp_i^{(a)}\} \quad (19)$$

The n-th SII time is $$SII_c(n)=MinSII_c(n)=\frac{\left(1-R\_loss_c^S\right)\times SIT_c(n)}{R\_loss_c^S - M\_loss_c(t_2)} \quad (20)$$

To finish each independent scanning piece successfully with QoS guarantee, the required time of each independent piece for channel or PoA scanning should be less than or equal to the SIT maximum bound, as in

$$t\_\mathrm{sp}_i \le \mathrm{MaxSIT}_c(n), \ \forall i \in \bar{S} \quad (21)$$

If t_sp_i is greater than $\mathrm{MaxSIT}_c(n)$, the following Layer 2 system parameter adjustment policy is applied:

Scanning Policy 2 - Layer 2 system parameter adjustment policy

Condition 1: If $t\_sp_i > MaxSIT_c(n)$, $\forall i \in \overline{S}$

Action 1-1: Adjust some of timer values of $\overline{T}_i=\{T_i^1, T_i^2, \cdots T_i^R\}$ to fit $t\_sp_i=\sum_{k=1}^{R} T_i^k \le MaxSIT_c(n)$

Action 1-2: If timer adjustment is not available, a vertical handover procedure begins.

5. Performance Results

In this section, the scanning performance of the proposed QoS support scanning mechanism is evaluated. It was assumed that the MS performs horizontal handovers in WLAN networks and for each channel to be scanned there exist at least one AP. First, numerical analysis is performed to evaluate the impact on the scanning parameters by different QoS requirements and network conditions. Second, we have simulated the real QoS performances during the scanning period such as the measured delay and the packet transmission statistics using a WLAN NS-2 network simulator [19].

5.1. QoS Support Scanning in WLAN

The IEEE 802.11 WLAN standard [10] provides a set of functions for MAC-layer handovers, such as active/passive scanning, an authentication process, and a re-association process. As indicated in several research papers [9,18,20], among all operations in the handover process of a WLAN, the time required for scanning to discover the target network amounts to as much as 90% of the overall handover latency. Therefore, QoS support during the scanning time in a WLAN is a very important issue. At the beginning of a scan, the MS can obtain the neighbor PoA QoS information from the IEEE 802.21 IS. During the actual scan, by receiving probe response frames from neighbor APs during an active scan, the MS can ensure that genuine connectivity to the neighbor APs exists, can estimate the possible transmission rate, and can recognize the QoS conditions of the neighbor APs. The probe response frame includes the basic service set (BSS) channel utilization, the available admission capacity, the average access delay of the BSS, and other parameters. After the MS finishes the proposed QoS support scanning, it can select a target AP that currently provides highest level of QoS.

In a passive scan, the MS switches to a channel and simply waits for beacon frames from APs. As the default beacon interval is 100 ms, 100 ms × 11 = 1.1 s is required to scan all APs in 11 channels. Most current WLAN cards use active scanning to probe available APs [21]. In an active scan, as shown in Figure 11, the MS switches to each channel of the scanning list and waits for the indication of an incoming frame

DOI: 10.1002/wcm

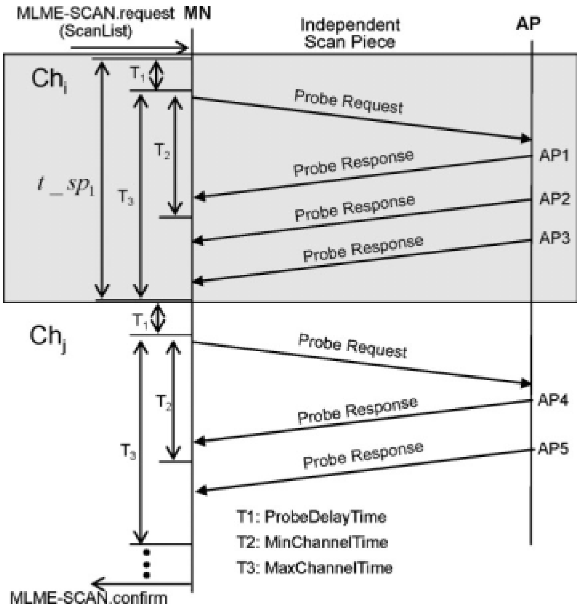


Fig. 11. Scanning operations in a WLAN.

or for the ProbeDelay time (T_1) to expire. It then broadcasts a probe request frame on one channel after contending to the medium and starts a probe timer. If no activity is detected in the wireless media when the probe timer reaches MinChannelTime (T_2), the station determines that no AP is working in that channel and scans another channel. If the station detects that the channel is not idle, it will wait for probe response frames from working APs until the probe timer reaches MaxChannelTime (T_3). An empirical measurement shows that the ProbeDelay time is a few microseconds, MinChannelTime is approximately 20 ms, and that MaxChannelTime ranges from 30 to 40 ms [20,22]. Therefore, in a WLAN, there exists only one independent piece, $\bar{S}_i = \{\mathrm{sp}_1\}$, that exchanges probe request and probe response frames. The required time for the independent piece is given by

$$\begin{aligned} t\_\mathrm{sp}_1 &= \mathrm{ProbeDelayTime} + \mathrm{MaxChannelTime} \\ &= T_1 + T_3 \end{aligned} \tag{22}$$

As there may exist no AP that responds in certain channels, the actual used time for the sp_1 can be

$$t\_\mathrm{sp}_1^{(a)} = \begin{cases} T_1 + T_2, & \text{if there is no Probe response} \\ T_1 + T_3, & \text{if there is at least one Probe response} \end{cases} \tag{23}$$

When $\mathrm{MaxSIT}_c(n)$ is given, the minimum number of channels that the MS can scan during the nth SIT is derived using

$$N_{\min}^{(n)} = \left\lfloor \frac{\mathrm{MaxSIT}_c(n)}{T_1 + T_3} \right\rfloor \tag{24}$$

In WLAN networks, if the current $\mathrm{MaxSIT}_c(n)$ is too small to capture t_sp_1, the following rule for the Layer 2 parameter adjustment is applied in the proposed scanning mechanism. Δ is the smallest marginal time to receive a probe response.

RULE 4 (Layer 2 parameter adjustment in WLAN):

When $t\_sp_1 = (ProbeDelayTime + MaxChannelTime) > MaxSIT_c(n)$

if $MaxSIT_c(n) \le (ProbeDelayTime + \Delta)$

Start vertical scanning and perform a vertical handover;

else if $(ProbeDelayTime + \Delta) < MaxSIT_c(n) < (ProbeDelayTime + MinChannelTime)$

$MinChannelTime = MaxSIT_c(n) - ProbeDelayTime;$

$MaxChannelTime = MinChannelTime;$

else if $(ProbeDelayTime + MinChannelTime) \le MaxSIT_c(n) < (ProbeDelayTime + MaxChannelTime)$

$MaxChannelTime = MaxSIT_c(n) - ProbeDelayTime;$

5.2. Numerical Results

In this numerical experiment, first the SIT and SII variations are evaluated based on various network conditions in terms of different measured QoS levels and different QoS classes. Second, the average number of channels that can be scanned during an SIT time and the total scanning period are analyzed for different QoS conditions. Finally, the maximum service disruption time is compared with conventional consecutive scanning with different numbers of neighboring channels. Table II shows the numerical analysis parameter values used in this section.

DOI: 10.1002/wcm

Table II. Numerical analysis parameters.

Parameter	Value
Number channels to scan (N)	1–10
Tolerable delay and loss degradation ratios (R_d, R_l)	(2, 2)
Delay QoS requirement $\left(R\_\mathrm{delay}_c\right)$	40 ms ($c = 1$), 80 ms ($c = 2$), 160 ms ($c = 3$)
Loss QoS requirement (R_loss_c)	10^{-2} ($c = 1$), 2×10^{-2} ($c = 2$), 4×10^{-2} ($c = 3$)
Delay and loss ratio $\left(R\_\mathrm{delay}_c / M\_\mathrm{delay}_c(t),\ R\_\mathrm{loss}_c / M\_\mathrm{loss}_c(t)\right)$	(1–0.8, 1–0.8)
ProbeDelay + MaxChannelTime	30 ms

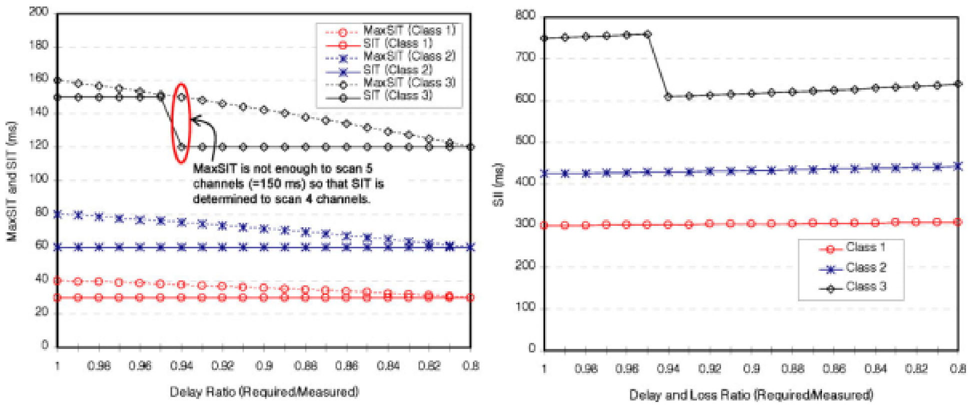


Fig. 12. MaxSIT, SIT, and SII variations for different QoS conditions. (a) MaxSIT and SIT, (b) SII.

Figure 12 shows the MaxSIT, SIT, and SII variations depending on different QoS conditions. As the measured QoS level (i.e., increasing the measured delay and loss ratio) decreases, the MaxSIT slowly decreases because a lower current QoS level requires a shorter scan. On the other hand, the SII slowly increases as the measured QoS level decreases. It is clear that, for a higher QoS class, a shorter MaxSIT is required. In Figure 12a, if the delay ratio is less than 0.95, the MaxSIT cannot include five independent scanning pieces. Hence, only four channels can be scanned.

Figure 13 shows the SIT and SII variations for the different QoS requirements. The required delay and loss ratio vary from 20 to 120 ms and from 10^{-3} to 10^{-2}, respectively when the delay and loss ratio are fixed at 0.9. The smaller required delay makes the smaller SIT and results in the smaller SII for a given loss ratio. For a given delay requirement, the smaller loss ratio requires the larger SII. The SII does not increase linearly as the required delay increasing due to the limitation of allowable number of independent scanning pieces.

For different QoS requirements and measured QoS levels, the number of channels scanned during an SIT time $\left(N_{(\mathrm{SIT})}\right)$ and the total required scanning time per channel $\left(T_{\mathrm{scn}/c}\right)$ were determined. As shown in Figure 14, with lower QoS requirements, a larger

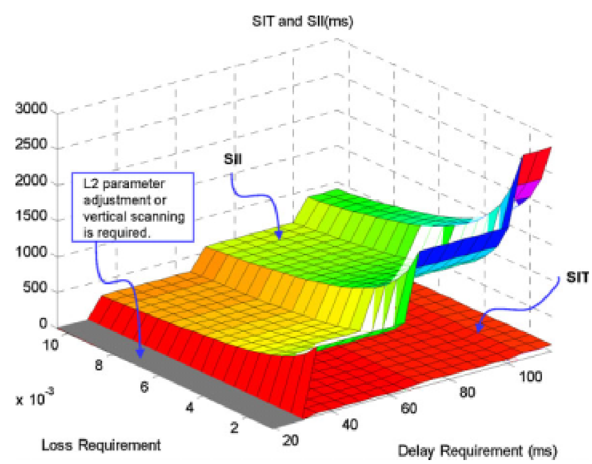


Fig. 13. SIT and SII for different QoS requirements.

number of channels can be scanned during an SIT time. Figure 15 shows the total scanning time per channel, as calculated by Equation (25). For a higher QoS class, a higher value of $T_{\mathrm{scn}/c}$ is required. The total scanning time depends on the required QoS classes rather than on the measured QoS levels. Figure 16 shows $T_{\mathrm{scn}/c}$ variations according to the delay and loss QoS requirements (the required/measured ratio is fixed here at 0.9), in which $T_{\mathrm{scn}/c}$ increases linearly when the

DOI: 10.1002/wcm

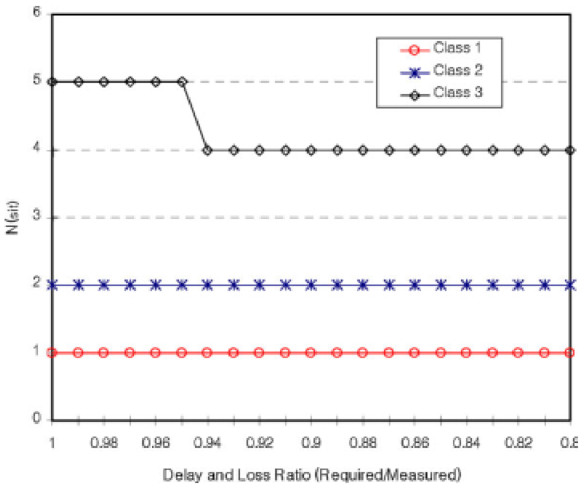


Fig. 14. The number of channels scanned during an SIT.

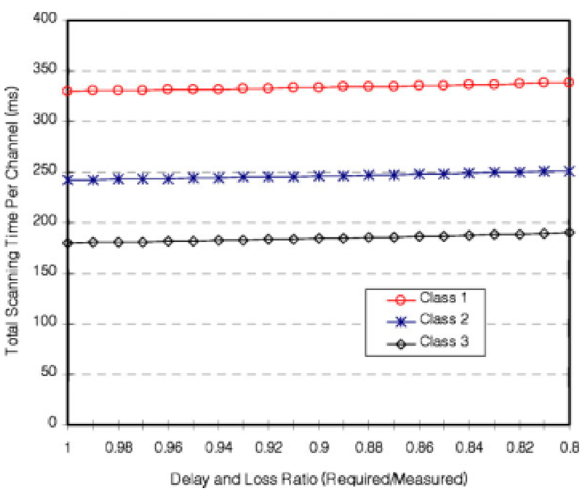


Fig. 15. The total scanning time per channel.

required delay is reduced but increases exponentially when the required loss ratio is decreased. $T_{\mathrm{scn}/c}$ is highly sensitive to loss ratio requirements, because the loss ratio determines the SII time, which is longer than the SIT time determined by delay requirements.

$$\text{Total scanning time per channel} = T_{\mathrm{scn}/c} = (\mathrm{SIT}_c + \mathrm{SII}_c)/N_{(\mathrm{SIT})} \qquad (25)$$

In real network environments, as the MS moves away from the currently serving PoA (decreasing the received signal strength), as the traffic volume of the current network increases, or as the number of wireless stations sending or receiving data increases, the QoS level measured by the MS decreases monotonically. In the study, this was modeled using exponential functions in time, as shown in Figure 17. The delay and loss satisfaction degree (required QoS/measured QoS) decreased exponentially during the scanning period. Figure 18 shows the total required scanning time to scan 10 neighboring channels with different QoS classes. As was previously evaluated, the scanning time is scarcely affected by the current measured QoS; instead, it depends mainly on the required QoS. Figure 19 shows the proposed QoS support scanning steps for the scanning of 10 neighbor channels. Here, the measured QoS degradation function $\exp(-0.06{*}t)$ is used. A lower QoS class results in a shorter total scanning time and a longer SIT time (i.e., more channels scanned during an SIT).

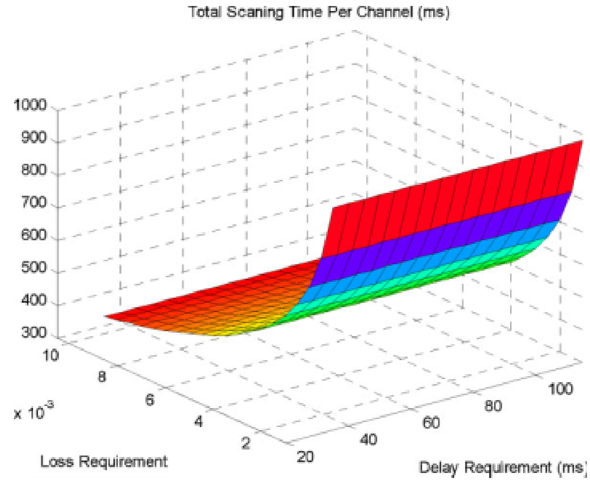


Fig. 16. The total scanning time per channel for different QoS requirements.

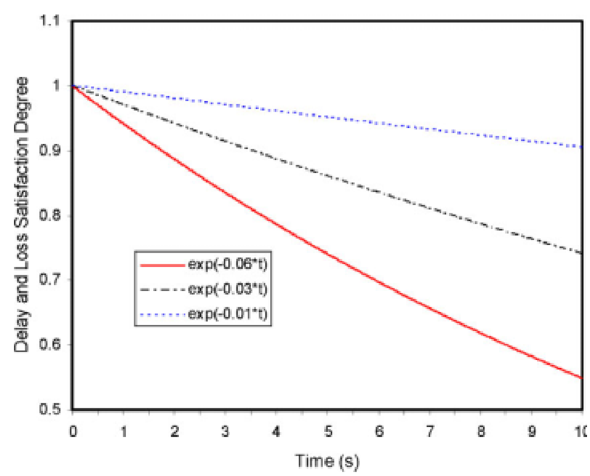


Fig. 17. Exponential measured QoS decrease.

DOI: 10.1002/wcm

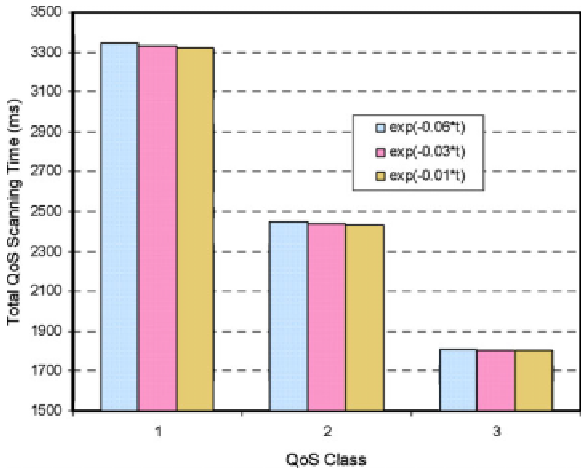


Fig. 18. The total scanning time for 10 channels.

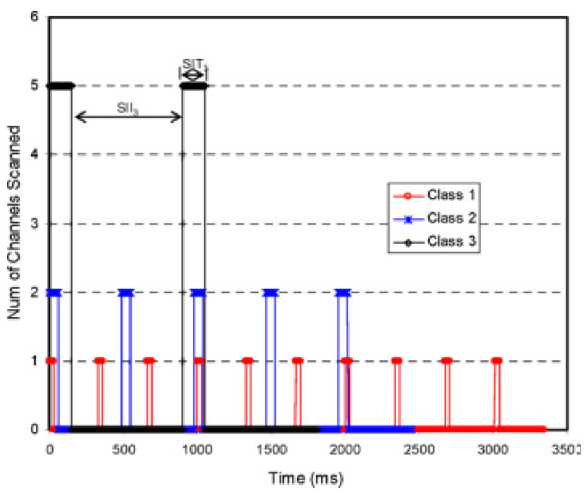


Fig. 19. QoS support scanning.

Lastly, the maximum service disruption time during a scan was compared with a conventional consecutive channel scanning method. For the measured QoS degradation function, exp($-0.06*t$) was used. As shown in Figure 20, as the number of neighboring channels to scan increases, the maximum service disruption time of the conventional non-QoS support scanning increases linearly. However, with the proposed method, the maximum service disruption times are bounded and are less than the desired delay QoS values (class $1 = 40$ ms, class $2 = 80$ ms, class $3 = 160$ ms).

5.3. NS-2 Simulation Results

In this simulation experiment, we consider an MS that is communicating with a corresponding node (CN) through a WLAN AP and an internet access router

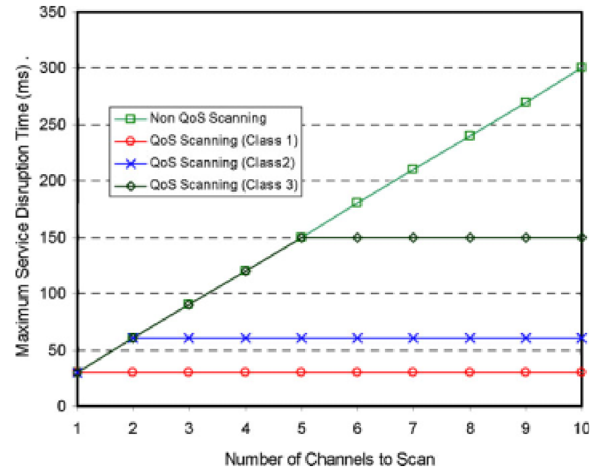


Fig. 20. Maximum service disruption time comparisons.

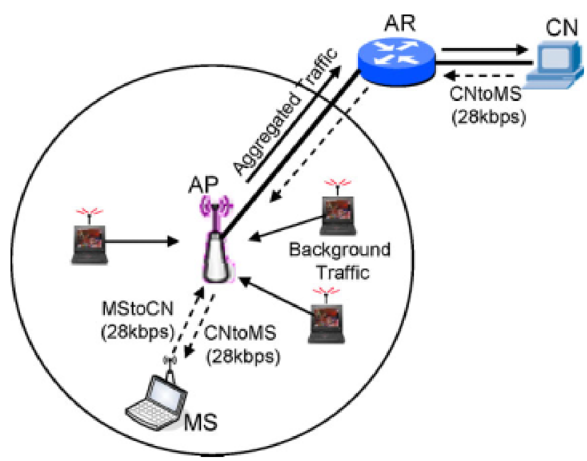


Fig. 21. Simulation network topology.

(AR). The MS requires a certain level of delay and loss QoS. When the measured QoS satisfaction degree at the MS is less than the QoS_Scan_Start threshold, the MS starts the proposed QoS scanning. In this simulation, the QoS satisfaction degree is defined in Equation (26). We assumed that there are 10 neighbor channels to scan. Figure 21 shows the network topology used in the simulation. There exist other 22 MSs in the current AP and they generate background traffic. The aggregated background traffic increased until the total network traffic triggered the QoS scanning. Table III shows the simulation parameters.

$$\mathrm{QoS}\,(t) = \min\left\{\frac{R\_\mathrm{delay}_c}{M\_\mathrm{delay}_c\,(t)}, \frac{R\_\mathrm{loss}_c}{M\_\mathrm{loss}_c\,(t)}\right\} \quad (26)$$

Since we used RTS/CTS mechanism for data transmission in this WLAN simulation, the packet loss

DOI: 10.1002/wcm

Table III. Simulation parameters.

Parameter	Value
Number channels to scan	10
Tolerable delay and loss degradation ratios (R_d, R_l)	(2, 2)
Delay and loss QoS requirements	40 ms, 10^{-2}
QoS_Scan_Start threshold	0.9
Wireless channel capacity	2 Mbps
Wired network data rate	100 Mbps
Data packet size	200 bytes
Data transmission rate	28 kbps (Poisson packet generation)

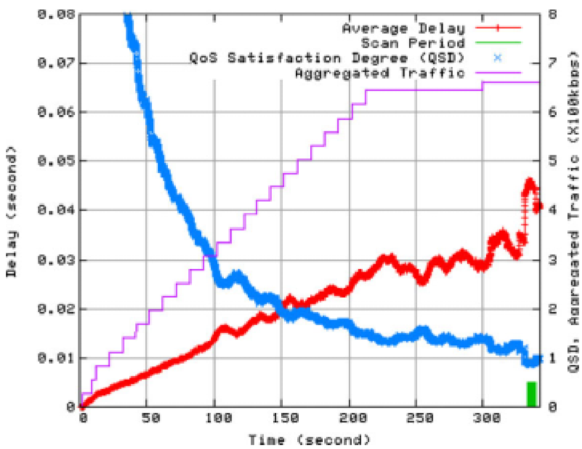


Fig. 22. QoS satisfaction degree variations.

ratio was not dominant component compared with packet transmission delay in the QoS satisfaction degree computation. Figure 22 shows the observed QoS satisfaction degree variations in time when we increase the aggregated network traffic. As increasing the network traffic, the measured average delay at the MS is also increasing. Around 332 s, the QoS satisfaction degree crossed the QoS_Scan_Start threshold and the MS initiated the QoS scanning. For performance comparison, we also implemented consecutive scanning, in which the MS scans all channels of the neighbor networks consecutively without interruptions.

Figure 23 shows the measured delay for different scanning methods. Delay is defined as the time from receiving a data from the upper layer to receiving the corresponding ACK from the CN. The MS performs two rounds of scanning, in which a round is for scanning all the neighboring channels (in this simulation, 10 channels). We have two types of consecutive scanning. The consecutive scanning A starts the second round at the same time for the second round in the

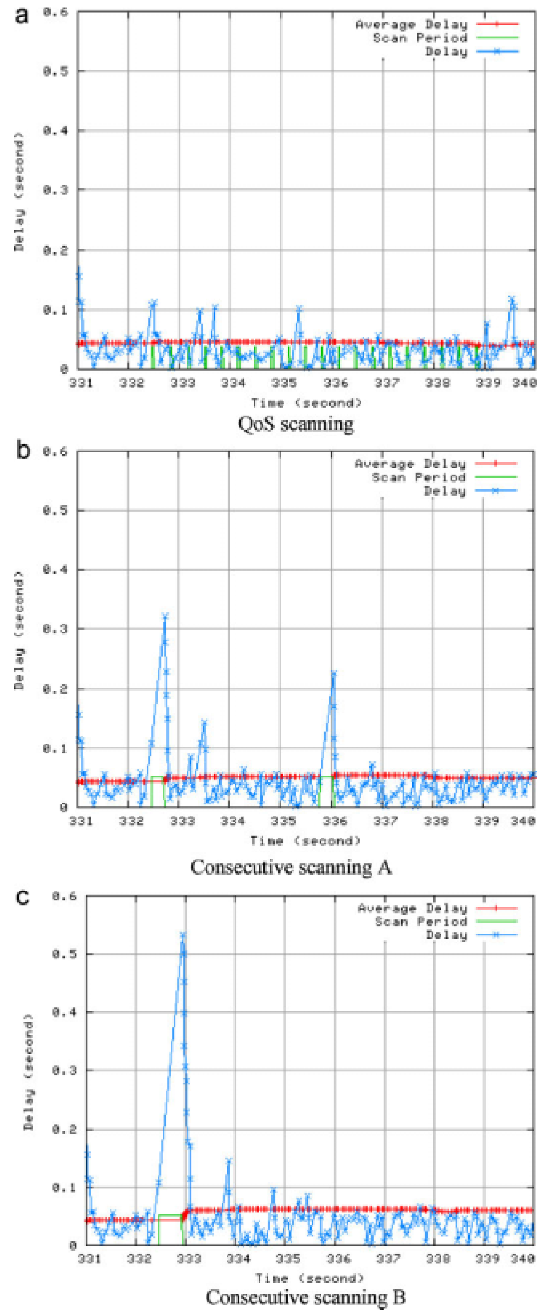


Fig. 23. Measured delay. (a) QoS scanning, (b) consecutive scanning A, and (c) consecutive scanning B.

proposed QoS scanning. The consecutive scanning B executes the first and second rounds continuously without any time break. As shown in Figure 23(a), during the QoS scanning, the measured average and instance delay are maintained at the low level (less than the tolerable delay). In case of consecutive scanning, the MS cannot transmit packets during the long scanning time

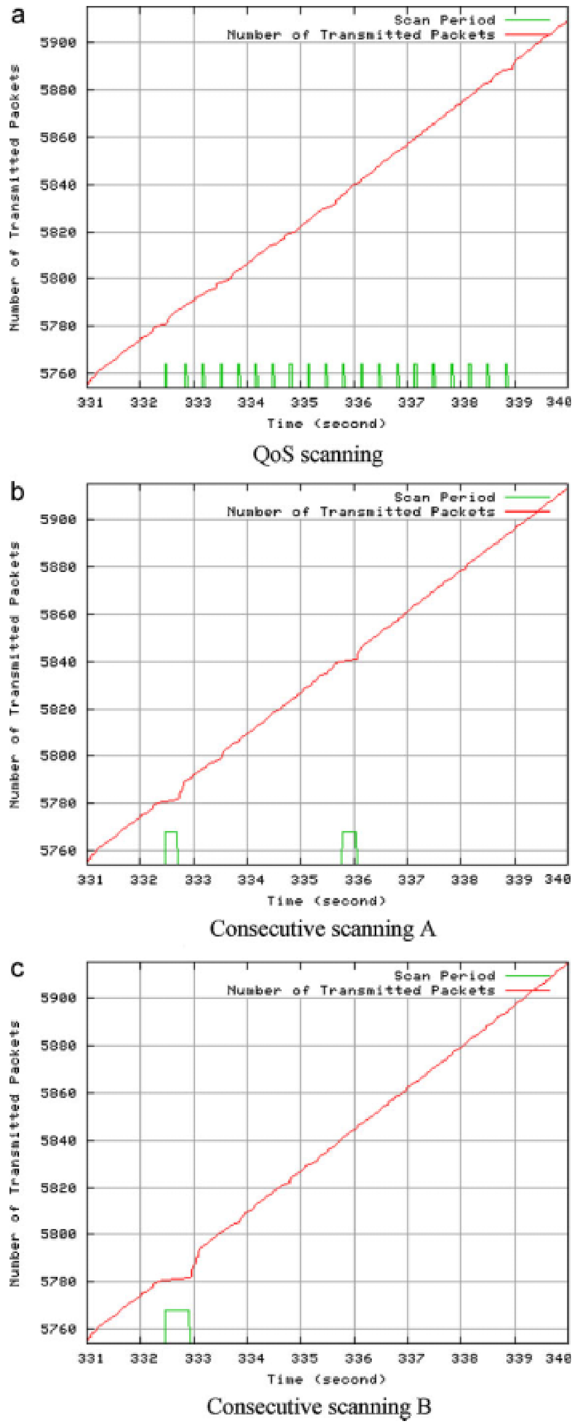


Fig. 24. Number of transmitted packets. (a) QoS scanning, (b) consecutive scanning A, and (c) consecutive scanning B.

the delay is rapidly increased. It should note that the scanning was initiated due to the low QoS satisfaction degree. However the consecutive scanning aggravated the provided QoS to the MS.

Figure 24 shows the total number of successfully transmitted packets by the MS. The MS is not able to send and receive data packets to and from the CN during the scanning periods. As we can see, since the proposed method interleaves short time scanning and normal data transmission in aware of possible delay and loss behaviors, the service disruption in terms of packet transmission is much smaller than the compared consecutive scanning methods. In the consecutive scanning, during the scanning period the packet transmissions from and to the MS are halted and the packets are buffered.

6. Conclusions

In this paper, new QoS support scanning architecture and mechanisms for seamless handovers are proposed. Service disruptions for scanning should be controlled effectively to guarantee the user QoS requirements. To trigger a QoS support scanning procedure, a new link-layer triggering mechanism is introduced. In the proposed scanning architecture, which takes into account user QoS demands and scanning policies, the network and PoA scanning order are dynamically determined. For controlled and uncontrolled provisioning network types, different scan-ordering policies are applied and provide a rapid target PoA discovery. To support the desired delay and loss ratio requirements during a scanning period, a short scanning time is interleaved after normal data transmissions. The actual scanning time (SIT) and its interval (SII) are adaptively determined by the QoS metric. All message exchange actions related to scanning are grouped as independent scanning pieces and only groups that can be handled during the current SIT time are executed in the proposed scanning mechanism. The timing and manner of adjustments to the Layer 2 system parameters can be altered to meet the scanning requirements.

A simulation of the proposed technique showed the adaptive scanning parameter decision results for different QoS requirements and network QoS conditions. The proposed scanning mechanisms were shown to be able to determine and adjust the scanning procedures dynamically to guarantee the required QoS. Higher QoS requirements need a shorter service disruption time and a longer scanning time interval. The maximum service disruption time during a scan is always bounded by the user requirements in the proposed mechanism. Compared to the conventional consecutive scanning method, the proposed QoS scanning provides low packet delay without discontinuity of packet

DOI: 10.1002/wcm

transmissions. The proposed scanning mechanisms are implemented only on the wireless MS side and do not require changes to APs or BSs, which is compatible with existing standards. Therefore, the proposed method can be applied widely to a range of wireless networks.

Acknowledgements

This research was supported by the NIST/Office of Law Enforcement Standards (OLES). This research was supported by the MKE, Korea, under the ITRC support program supervised by IITA (IITA-2008-C1090-0801-0019).

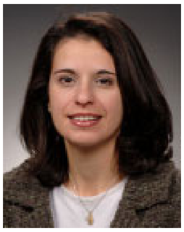
References

1. Koodli R. Fast Handover for Mobile IPv6. *IETF RFC 4068*, July 2005.
2. Kwon K, Lee C. A fast handoff algorithm using intelligent channel scan for IEEE 802.11 WLANs. *IEEE International Conference on Advanced Communication Technology*, 2004; 46–50.
3. Shin M, Mishra A, Arbaugh WA. Improving the latency of 802.11 hand-offs using neighbor graphs. *ACM International Conference on Mobile Systems, Applications, and Services (MOBISYS'04)*, 2004; 70–83.
4. Kim H-S, Park S-H, Park C-S, Kim J-W, Ko S-J. Selective channel scanning for fast handoff in wireless LAN using neighbor graph. *International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC'2004)*, 2004; 7F2P_29_1–7F2P_29_4.
5. Shin S, Forte AG, Rawat AS, Schulzrinne H. Reducing MAC layer handoff latency in IEEE 802.11 wireless LANs. *ACM International Workshop on Mobility Management & Wireless Access Protocols (MOBIWAC'04)*, 2004; 19–26.
6. Tommasi F, Molendini S, Tricco A. Experience-driven selective scan for 802.11 networks. *IEEE International Conference on Software in Telecommunications and Computer Networks (IEEE SoftCom'06)*, 2006; 137–141.
7. Chen Y-S, Chen C-K, Chuang M-C. DeuceScan: Deuce-based fast handoff scheme in IEEE 802.11 wireless networks. *IEEE Vehicular Technology Conference (IEEE VTC'06 Fall)*, 2006; 1–5.
8. Speicher S, Bunning C. Fast MAC-layer scanning in IEEE 802.11 fixed relay radio access networks. *IEEE International Conference on Networking, International Conference on Systems, and International Conference on Mobile Communications and Learning Technologies (IEEE ICN/ICONS/MCL'06)*, 2006; 144–150.
9. Ramani I, Savage S. SyncScan: practical fast handoff for 802.11 infrastructure networks. *IEEE INFOCOM'05*, 2005; 675–684.
10. Institute of Electrical and Electronics Engineers. Draft Standard for Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks- Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE P802.11/D9.0*, 2006.
11. Institute of Electrical and Electronics Engineers. Draft Standard for Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks- Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment: Radio Resource Measurement. *IEEE P802.11k/D7.0*, January 2007.
12. Institute of Electrical and Electronics Engineers. IEEE Standard for Local and Metropolitan Area Networks—Part 16: Air Interface for Fixed Broadband Wireless Access Systems. *IEEE Std. 802.16*, October 2004.
13. Institute of Electrical and Electronics Engineers. IEEE Standard for Local and Metropolitan Area Networks—Part 16: Air Interface for Fixed Broadband Wireless Access Systems, Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands. *IEEE Std. 802.16e*, February 2006.
14. Institute of Electrical and Electronics Engineers. Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services. *IEEE P802.21/D05.00*, April 2007.
15. Rouil R, Golmie N. Adaptive channel scanning for IEEE 802.16e. *IEEE Military Communications Conference (IEEE MILCOM'06)*, 2006; 1–6.
16. Wu H, Tan K, Zhang Y, Zhang Q. Proactive scan: fast handoff with smart triggers for 802.11 wireless LAN. *IEEE INFOCOM'07*, 2007; 749–757.
17. Huang J-J, Chen Y-H, Chang S-C, Ferng H-W. An efficient channel scan scheduling algorithm for VoIP handoffs in WLANs. *IEEE Vehicular Technology Conference (IEEE VTC'07 Spring)*, 2007; 1340–1344.
18. Liao Y, Cao L. Practical schemes for smooth MAC layer handoff in 802.11 wireless networks. *IEEE International Symposium on World of Wireless, Mobile and Multimedia Networks (IEEE WoWMoM'06)*, 2006; 181–190.
19. NS-2 Simulator, http://www.isi.edu/nanam/ns/
20. Mishra A, Shin M, Arbaugh W. An empirical analysis of the IEEE 802.11 MAC layer handoff process. *ACM SIGCOMM Computer Communication Review* 2003; **33**(2): 93–102.
21. Velayos H, Karlsson G. Techniques to reduce IEEE 802.11b MAC layer handover time. *IEEE International Conference on Communications (IEEE ICC'04)*, 2004; 3844–3848.
22. Hauang P-J, Tseng Y-C. A fast handoff mechanism for IEEE 802.11 and IAPP networks. *IEEE Vehicular Technology Conference (IEEE VTC'06 Spring)*, 2006; 966–970.

Authors' Biographies

Sang-Jo Yoo received the B.S. degree in electronic communication engineering from Hanyang University, Seoul, Korea, in 1988 and the M.S. and Ph. D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), in 1990 and in 2000, respectively. From 1990 to 2001, he was a member of technical staff at Korea Telecom Research and Development Group, where he worked in communication protocol conformance testing and network design fields. From September 1994 to August 1995 and from January 2007 to January 2008, he was a guest researcher at National Institute Standards and Technology (NIST), U.S.A. Since 2001, he has been with the Graduate School of Information Technology and Telecommunications, Inha University, where he is currently an associate professor. His current research interests include seamless network mobility control, wireless network QoS, cognitive radio network protocols, and wireless sensor networks.

DOI: 10.1002/wcm



Nada Golmie NADA GOLMIE (mailto:nada@nist.gov) received her Ph.D. in computer science from the University of Maryland at College Park. Since 1993, she has been a research engineer in the advanced networking technologies division at the National Institute of Standards and Technology (NIST). She is currently the manager of the high-speed network technologies group. Her research in media access control and protocols for wireless networks led to over 100 technical papers presented at professional conferences, journals, and contributed to international standard organizations and industry led consortia. She is the author of '*Coexistence in Wireless Networks: Challenges and System-level Solutions in the Unlicensed Bands*', published by Cambridge University Press (2006). She has served as vice-chair for the IEEE 802.15.2 Task Group on Coexistence and is currently a member of the IEEE 802.21 Media Independent Handover Working Group.

DOI: 10.1002/wcm