Service Charge and Energy-Aware Vertical Handoff in Integrated IEEE 802.16e/802.11 Networks

Youngkyu Choi and Sunghyun Choi
School of Electrical Engineering and INMC
Seoul National University, Seoul, Korea
Email: ykchoi@mwnl.snu.ac.kr, schoi@snu.ac.kr

Abstract—This paper considers two issues arising in an integrated 802.16e/802.11 network: 1) finding a possible network, which mobile station (MSTA) can switch to, and 2) making a decision whether to execute a vertical handoff (VHO). For this purpose, we propose that 802.16e Base Stations (BSs) periodically broadcast the information about the density of 802.11 access points (APs) within their cell coverage. Based on this information, we develop a novel model, which predicts the successful scan probability during a given scan time. Using this analytical model, we devise an energy-efficient scan policy (ESP) algorithm, which enables an MSTA to decide 1) whether to attempt to discover APs in the current 802.16e cell, and 2) if so, how to set the 802.11 active scan interval considering the energy consumption. For the VHO decision, we mainly consider the impact of the service charge. Especially, a practical service fee models, i.e., flat pricing for WLAN and partially-flat pricing for 802.16e network, are considered. Under this service charge plan, we need to control the usage of the 802.16e network to minimize the user’s payment. To this end, we propose a scheme, which intelligently postpones the delivery of the delay-tolerant traffic within a certain time limit combined with ESP algorithm.

Index Terms—vertical handoff, IEEE 802.16e, IEEE 802.11, active scan, service pricing, mathematical modeling

I. INTRODUCTION

Vertical handoff (VHO), i.e., handoff across heterogeneous access networks, is considered a key feature to bring the next-generation wireless communication era. From the late 1990s, vertical handoff (VHO) has been studied mainly focusing on the interworking between 3G and wireless local area network (WLAN). When we categorize the previous work according to three generalized steps of handoff, namely, 1) finding candidate networks, 2) deciding a handoff, and 3) executing a handoff, we see that the research on the third step, i.e., handoff execution, has been actively conducted along with the interest in mobile IP by Internet Engineering Task Force (IETF) and Third Generation Partnership Project (3GPP). On the other hand, the study on VHO criteria, by which a handoff execution is triggered, has begun recently. Recent research results suggest that more than the received signal strength, a typical attribute for a horizontal (i.e., across the same type of networks) handoff decision, should be considered as attributes to decide VHO. Such attributes include available bandwidth, latency, packet error rate, monetary cost, power consumption, mobile speed, user preference, security, and so on [7]–[11].

There are different ways in making a VHO decision based on these attributes. One of the popular ways is to use a weighted sum of attributes [7]–[10]. A handoff is made to the network, which yields the highest magnitude of a weighted sum. Both the weighted product [9] or a weighted logarithm [7] of attributes can also be classified into this type. Since a specific weight reflects the relative importance of the corresponding attribute, it is important to adapt the weights toward achieving the desired behavior of VHO. Other algorithms proposed in the literature include a policy-based decision [12], a fuzzy logic-based decision [6], and a utility-based decision [15]. In the policy-based decision, a network with attributes best-matched with a pre-specified policy is selected as the target network. In the fuzzy logic-based decision, a fuzzy rule, which is inferred the closest to the ideal decision, is chosen. In the utility-based decision, a utility function, which is designed considering the service fee and the transfer completion time, is used to select the most appropriate network accounting for a user preference.

In the meantime, when discussing VHO between WLAN and other radio access networks such as 3G system, the discovery of the available WLANs has not taken much attention. It is typically assumed that mobile station (MSTA) always monitors available WLANs by attempting to detect the WLAN signal or even that MSTA is located at the specific region, where all the access networks of interest are accessible. However, this issue can be practically critical in that MSTA is usually battery-powered. Since WLANs are often deployed for hotspot services with spotty service coverages, keeping the WLAN interface turned on can be often translated into an ineffective use of energy [8]. For a smart discovery of WLANs, the authors of [8], [16] propose the notion of a location service server, which stores the information of WLAN access point (AP) locations, and hence, can inform the MSTA of the list of available WLANs when an MSTA sends a request message.

IEEE 802.16e [2] for mobile broadband wireless access (MBWA) was standardized in 2005, and the first 802.16e-based commercial service commenced under the brand name of “WiBro” by Korea Telecom (KT) in Korea in June, 2006. Along with the introduction of the 802.16e service, the interworking with WLANs, of which a commercial service has been operational for years, also attracts high interest. As 802.16e provides a larger link speed compared with that of existing 3G systems, it may not be valid any longer to say that a VHO into WLANs is always advantageous with respect
to throughput. In other words, maximizing the utilization of WLAN is not equivalent to optimizing VHO any longer.

In this paper, we do not propose a new cost function for the VHO decision, but provide practical solutions, which can resolve many issues expected when realizing VHO in an integrated IEEE 802.16e/802.11 network. More specifically, we use MSTA's speed to decide whether to regard a VHO to WLANs as a viable option, and to adapt the strategy for discovering WLAN APs. For the VHO decision, we mainly consider the impact of the service charge. In order to focus on the target network discovery and handoff decision, in this paper, we do not consider many protocol issues related to the VHO execution itself.

The rest of the paper is organized as follows. In Section II, we describe the system model, which requires a slight modification of the current 802.16e specification to support MSTA's discovery of WLAN APs. Then, in Section III, we propose our 802.11 scanning algorithm, which minimizes the amount of energy consumed during scan trials while guaranteeing the successful scan probability within a given scan time. In Section IV, we describe the impact of the service charge on the VHO, and then propose a VHO decision algorithm, which intelligently postpones the delivery of delay-tolerant traffic using the proposed scanning algorithm. Finally, we conclude with the summary of our contributions in Section V.

II. SYSTEM MODEL

In this paper, the 802.16e network is assumed to be always reachable by MSTAs while the 802.11 WLAN is opportunistically reachable since the base station (BS) of the 802.16e is deployed to support seamless mobility. For this reason, it is enough only to consider how to discover WLANs for the detection of the target systems. Basically, we assume that the 802.16e network can assist MSTAs to search available 802.11 networks. To this end, we propose to slightly revise the specification of the 802.16e in Section II-B. Before this, we discuss the difficulties originated by MSTA's mobility, and suggest a practical solution to deal with this problem in the following Section II-A.

A. Handling MSTA's Mobility-Related Issues

The velocity of an MSTA is one of the important factors in VHO, especially, for the handoff into WLANs [10]. For the MSTA traveling at a high speed, a handoff into a WLAN is followed by returning to the 802.16e after a short time. This causes so-called the ping-pong effect, which could also happen due to the received signal strength fluctuation as in horizontal handoff cases. Furthermore, the knowledge about target APs, which is obtained via an 802.11 scanning, could be outdated as the MSTA changes its location. The concept of the dwell timer, used in [8], [14], is effective to assure that the selection of a target network is really correct. However, as a more time interval is consumed to examine the handoff condition by the dwell timer, the total time, for which WLANs can be utilized, is also reduced.

We can easily understand that residing in an 802.16e network is a better choice when an MSTA moves too fast considering the overhead and cost incurred by searching target APs and executing VHO. Assuming that an MSTA can estimate its own speed, we propose that MSTA does not consider a VHO into WLANs when its speed exceeds a certain threshold $v_{\text{max}}$. There are many ways to estimate MSTA's speed, but we do not discuss them since they are outside the scope of this paper. Our proposed solution is intuitively simple, but quite reasonable. Now, the problem is how to determine $v_{\text{max}}$. The following proposition says that $v_{\text{max}}$ should be low enough to yield a sufficiently-long sojourn time in a WLAN considering the time overhead consumed during a VHO execution.

**Proposition 1:** Let us define $\zeta$ as a VHO execution time (due mainly to layer-3 (L3) operations, e.g., mobile IP). We propose that $v_{\text{max}}$ should satisfy the condition, $\text{Prob}(s > \kappa\zeta) \geq \phi$, where $s$ is a random variable representing the MSTA's sojourn time in a WLAN, $\kappa$ is a constant chosen by the user, and $\phi$ is the target probability controllable by the user as well. Under this policy, $v_{\text{max}}$ is given by

$$v_{\text{max}} = \frac{2r}{\kappa\zeta}\sqrt{1 - \phi^2},$$

where $r$ is the radius of the WLAN coverage assuming a circular shape of the coverage.

**Proof:** It is straightforward to prove this proposition using the probability density function of $s$ shown in Appendix I.

From Proposition 1, we can statistically guarantee the utility obtainable via the VHO into WLANs. For example, when $r = 50$ m, $\zeta = 500$ ms, $\kappa = 100$, and $\phi = 0.9$, $v_{\text{max}} \simeq 0.87$ m/s. In this example, the constraint is that the MSTA's sojourn time in a WLAN be longer than $100\zeta = 5$ sec with a probability larger than 0.9. In this case, the MSTA attempts to perform a VHO only when it moves at a speed under 0.87 m/s. Depending on $\kappa$, this policy enables MSTAs to attempt VHOS aggressively or conservatively. Basically, $\kappa$ is determined in a distributed manner, but it is also possible to enforce using a specific value determined by a central controller. Using the centralized controller, it may be possible to control the number of MSTAs participating in VHOS. We leave this issue as the future work.

B. IEEE 802.16e-Assisted 802.11 Scanning

As mentioned earlier, we assume that the 802.16e network can give MSTAs the information useful to find 802.11 APs. Otherwise, MSTAs should keep its WLAN interface turned on or switch it on periodically at a pre-determined time interval to detect a WLAN signal [8]. Even though this is quite simple, the use of a more sophisticated method is encouraged because it is more and more emphasized to manage multiple interfaces intelligently for energy-efficient communications.

Now, we consider which information the 802.16e network will provide to aid MSTAs to discover available 802.11 APs. Obviously, the more detailed the information is, the more efficient scheme we can devise. As the case of the location
service server [16], if MSTA can obtain such knowledge as the exact location of 802.11 APs near the location of MSTA along with the available bandwidth and latency supportable by each AP, then MSTA can possibly find WLANs more efficiently and also decide the handoff more intelligently. However, we should note that providing much abundant information via the 802.16e network incurs the extravagant consumption of precious bandwidth resources of the 802.16e network, even though the cost of the air resource of the 802.16e is known typically cheaper than that of other conventional cellular systems. In addition, a unicast protocol operating in a request-response manner is also required to make it feasible.

Therefore, the size of such aid-information should be compromised not to consume the air resource too much. To this end, we consider an 1-octet overhead indicating the 802.11 AP density, which is defined as the ratio of the number of APs deployed within an 802.16e single-cell to its cell coverage. Compared with the information provided by the location service server, the AP density is not heavy, and also easy to maintain the validity from the operator’s perspective.

If we exploit the existing broadcast message of the 802.16e to notify this 1-octet information, we do not need to define a new protocol in the specification of the 802.16e. Bearing this in mind, we decide to modify a MAC (Medium Access Control) management message, called MOB_NBR-ADV, which is originally defined to convey the information about the system configuration of neighboring BSs in order to support horizontal handoffs among 802.16e BSs. Since the VHO aid-information represents the system configuration of WLANs, the integration with the MOB_NBR-ADV is fairly a natural approach. According to the specification of the 802.16e [2], a MOB_NBR-ADV message can be delivered via broadcast or via a primary management connection, which implies that it can also be delivered via unicast if an MSTA requests. The modified format of the MOB_NBR-ADV message is shown in Table I, where the proposed revisions are highlighted with bold face. Utilizing a reserved bit of the existing optional field bitmap, the MOB_NBR-ADV message can accommodate our revision optionally. Currently, the revision informs only the information about WLANs underlying a single cell coverage of the 802.16e. However, it can be extended to specify per-sector information, and even include such time-varying information as the available capacity of each WLAN at the expense of the 802.16e air resource. This issues are also left to the future work.

III. Scanning Rule for WLANs

In this section, we discuss how to discover 802.11 APs intelligently. Denoting the AP density by $\rho$, we propose a simple criterion used to decide whether to attempt to scan WLANs:

**Proposition 2:** If $\rho > \rho_{\text{min}}$, an MSTA attempts to scan WLANs. Otherwise, turn off the WLAN interface until $\rho$ is newly given by another 802.16e BS, which is met after a forthcoming horizontal handoff across the 802.16e cells.

Here, we emphasize that $\rho_{\text{min}}$ can be determined by each individual MSTA. In this paper, we take into account the energy consumption, the target probability of successful scan, and scan time to determine $\rho_{\text{min}}$.

IEEE 802.11 supports two different mechanisms to detect nearby APs: passive and active scan. Passive scan is conducted by listening to the beacons broadcast periodically by APs. Typically, passive scan is known to take longer time than active scan [17], and hence, we consider only the active scan in this paper. Indeed, the active scan is the main mechanism used by WLAN stations to detect APs. In this section, we discuss how to determine $\rho_{\text{min}}$, and how to optimize the scan interval such that the energy consumed for the scan operations is minimized while satisfying the target probability of successful scan for a specified scan time. In Section III-A, we briefly review the active scan specified by IEEE 802.11 standard [4], and then take a look at the energy consumption during active scans. In Section III-B, we build an analytical model to describe the successful scan probability considering MSTA’s mobility. Then, we finally present how to determine $\rho_{\text{min}}$ and how to optimize the scan interval by using a mathematical formulation in Section III-C.

A. 802.11 Active Scan and Energy Consumption

An active scan trial operates as follows:

1) After broadcasting a Probe Request frame at a specific frequency channel, if the channel stays idle for minChannelTime, declare the channel as empty.

2) If the channel becomes busy, waits for maxChannelTime in the receiving MAC state, and then handles Probe Response frames received afterwards.

3) Moving to the next channel, repeat the same procedures until there is no frequency channel to scan.

Here, both minChannelTime and maxChannelTime have the unit of time unit (TU), which is 1,024 $\mu$s. Typically, minChannelTime should be larger than DIFS + aCW_{min} × aSlotTime = 670 $\mu$s in the 802.11b PHY [5], and hence reasonably set to 1 TU [17]. Since both Probe Request and Probe Response frames are transmitted using distributed coordinate function (DCF) in the same way as data frame,

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Size</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mgmt Message Type = 53</td>
<td>8 bits</td>
<td>Bit[0]= 1, omit Operator ID field</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bit[4]= 1, omit VHO fields</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bit[5]-Bit[7]: reserved</td>
</tr>
<tr>
<td>VHO info</td>
<td>8 bits</td>
<td>The 802.11 AP density in the 802.16e cell coverage</td>
</tr>
<tr>
<td>N_NEIGHBORS</td>
<td>8 bits</td>
<td></td>
</tr>
</tbody>
</table>
the transmission latency depends on the degree of loading in the basic service set (BSS). It is reported that 10 (TUs) is a reasonable value for maxChannelTime when 50% offered load is imposed on the network composed of 10 stations [17]. If a nonempty channel is examined, the scan time $T_u$ is

$$T_u = T_{p,req} + \text{maxChannelTime}. \quad (2)$$

On the contrary, if an empty channel is examined, the scan time $T_e$ is

$$T_e = T_{p,req} + \text{minChannelTime}. \quad (3)$$

Under the 802.11b PHY [5], $T_{p,req}$ can be computed as follows: When SSID (Service Set IDentifier) is set to maximally 32 octets and all four transmission rates of the 802.11b PHY are supported, the transmission time of a Probe Request frame shown in Fig. 1 is calculated by

$$T_{p,req} = \text{DIFS} + \frac{aCW_{\text{min}}}{2} + 192 + \frac{(24+34+6+4)8}{\text{Mbits/s}} = 1,096 \mu s, \quad (4)$$

assuming the transmission rate of 1 Mbits/s.

For total $n$ frequency channels, each channel is sequentarily examined by transmitting Probe Request frame. A scan trial is completed if $n$ sequential scans are completed. If at least a Probe Response frame is received during a scan trial, the corresponding scan trial is regarded successful. Unless any Probe Response is received, another scan trial begins after a scan interval denoted by $\tau_s$. Here, we assume that the impact of MSTAs’ movement during a scan trial can be ignored since $\tau_s$ is much longer than the time spent during a scan trial. For this reason, $\tau_s \gg nT_e \simeq 23 \text{ ms}$ for $n = 11$ since $T_e \simeq 2.12 \text{ ms}$ from Eqs. (3) and (4). By the rule of thumb, we consider that $\tau_s$ is lower-bounded by

$$\tau_s \geq \tau_s^{\text{min}} = 0.23 \text{ sec}. \quad (5)$$

For $\tau_s$, the energy consumption is minimized by changing MSTAs’ MAC state to sleep or off state. Figs. 2(a) and (b) illustrates the failed and successful scan trials, respectively.

In order to consider the energy consumption, we assume that the power consumption at different MAC states is given as shown in Table II. When a scan trial failed, the energy consumed during a scan trial $\delta_{\text{fail}}$ is represented as

$$\delta_{\text{fail}} = (\text{DIFS} + \frac{aCW_{\text{min}}}{2} + \text{minChannelTime})P_l + (T_{p,req} - \text{DIFS} - \frac{aCW_{\text{min}}}{2})P_t. \quad (6)$$

Similarly, when a scan trial was successful, we can quantify the energy consumption $\delta_{\text{success}}$. However, since $\delta_{\text{success}}$ depends on both the number of nonempty channels and the number of Probe Response frames received, we do not present the exact expression for $\delta_{\text{success}}$. Anyway, we know that $\delta_{\text{success}} > \delta_{\text{fail}}$, obviously. When we denote by $\mathcal{E}(k)$ the energy consumption after $k$ scan trials, $\mathcal{E}(k)$ is also lower-bounded by $\mathcal{E}_{\text{fail}}(k)$, i.e., the energy consumption when all $k$ trials failed.

Assuming that the MAC state is changed to sleep state during scan interval $\tau_s$, $\mathcal{E}_{\text{fail}}(k)$ is represented as

$$\mathcal{E}_{\text{fail}}(k) = (n\delta_{\text{fail}} + \tau_s P_s)k. \quad (7)$$

When scanning is done during time $t$, the number of scan trials, $k$, is approximated by

$$k = \left[ \frac{t}{nT_e + \tau_s} \right], \quad (8)$$

where $[\cdot]$ is the operator that yields the minimum integer number greater than or equal to given value.

**Proposition 3:** At a given time $t$, $\mathcal{E}_{\text{fail}}(k)$ is a decreasing function of $\tau_s$.

**Proof:** Approximating $k$ with $\frac{t}{nT_e + \tau_s}$, inserting $k$ into Eq. (7), and differentiating $\mathcal{E}_{\text{fail}}$ with $\tau_s$, we obtain

$$\frac{d\mathcal{E}_{\text{fail}}}{d\tau_s} = -\frac{nt}{(nT_e + \tau_s)^2} (\delta_{\text{fail}} - P_sT_e). \quad (9)$$

From Eqs. (3) and (6), $P_sT_e = 0.085 \text{ mJ}$ and $\delta_{\text{fail}} = 2.579 \text{ mJ}$, respectively. Accordingly, $\frac{d\mathcal{E}_{\text{fail}}}{d\tau_s} < 0$.

If the MAC state is changed to off state during the scan interval, Proposition 3 is obviously trivial. However, under the specifications shown in Table II, Proposition 3 shows that it is still energy efficient to use longer scan interval even in case that MAC state is changed to sleep state during the scan interval.

**B. Analytical Model for Scanning in Mobile Environment**

Let us denote the transmission range of WLAN by $r$, and denote a single-cell coverage of the 802.11e by $A$. When 802.11 APs are randomly placed in the area $A$, the probability density function of distance $\xi$ from MSTAs to the nearest

<table>
<thead>
<tr>
<th>MAC States</th>
<th>Power Consumption</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off</td>
<td>0 mW</td>
<td>Turns off the WLAN interface completely</td>
</tr>
<tr>
<td>Sleep</td>
<td>$P_s = 40 \text{ mW}$</td>
<td>Turns off most parts of circuit except for critical circuit</td>
</tr>
<tr>
<td>Listen</td>
<td>$P_l = 800 \text{ mW}$</td>
<td>Keeps sensing the carrier</td>
</tr>
<tr>
<td>Receive</td>
<td>$P_r = 900 \text{ mW}$</td>
<td>Performs the receive operations such as demodulation</td>
</tr>
<tr>
<td>Transmit</td>
<td>$P_t = 2,000 \text{ mW}$</td>
<td>Transmits the frame to the air</td>
</tr>
</tbody>
</table>

Table II: Power consumption depending on the MAC states of WLAN [18].
802.11 AP can be characterized by Poisson point process as follows [19]:

\[ f(\xi) = 2\pi \rho \xi e^{-\rho \pi \xi^2}. \]  

(10)

The assumption of random deployment of APs may not sound reasonable since network operators often do the cell-planning before installing APs. However, we should pay attention to a trend in public WLAN service field where the existing WLAN APs are integrated under a big umbrella to enable the ubiquitous access [22]. In this situation, it is likely that the existing APs had been deployed under the lack of control, and then our assumption, i.e., the random distribution of AP deployment, can be justified. For a given \( \rho \), the probability that an MSTA finds any 802.11 AP at the first scan trial is represented as

\[ P(\xi \leq r) = 1 - e^{-\rho \pi r^2}. \]  

(11)

In general, when an area \( A_0 \) in the system plane \( A \) has been examined, the probability of finding \( n_0 \) nodes in area \( A_0 \) is given by [19]

\[ P(n_0) = \binom{\rho A_0}{n_0} n_0! e^{-\rho A_0}. \]  

(12)

We assume that an MSTA moves linearly with the constant speed of \( v \) (\( 0 \leq v < v_{\text{max}} \)). Therefore, at every scan trial, the MSTA additionally examines the area \( \Delta C \), which is not overlapped by the area examined previously. We denote by \( \Delta C(k) \) the additional area examined at the \( k \)-th scan trial. When we consider random waypoint model [19] as a mobility pattern, it is more likely to examine the area again, which have been already examined at previous scan trials as the time goes by. Unfortunately, it is impossible to characterize \( \Delta C(k) \) by an accurate mathematical expression. For this reason, we propose a model of \( \Delta C(k) \) based on the following observations, and then verify the model using the data measured by simulations.

**Observation 1: Properties of \( \Delta C(k) \)**

1) \( \Delta C(k) \) is always less than or equal to \( B \), which represents the area of the region overlapped by both a ring made by two concentric circles with radius \( r \) and \( r + v r_s \), where \( r_s = n T_c + v_s \), respectively and another circle with radius \( r \), which has its center \( v r_s \) apart from those concentric circles. Depending on \( v r_s \), \( \Delta B \) is shown as

\[ \Delta B = \begin{cases} r^2 \left( \pi - 2 \arccos \left( \frac{v r_s}{r} \right) + \frac{v r_s}{r} \sqrt{1 - \left( \frac{v r_s}{r} \right)^2} \right) & \text{if } 0 \leq v r_s \leq 2r, \\ \pi r^2, & \text{if } v r_s > 2r. \end{cases} \]  

(13)

2) When \( \Delta C(k+1) \) is modulated by a function \( \varphi(\cdot) \), i.e., \( \Delta C(k+1) = \varphi(\cdot) \Delta B \), \( \varphi(\cdot) \) is a decreasing function of \( \frac{\sum_{i=1}^{k} \Delta C(i)}{A} \), ranging from 0 to 1 depending on \( \frac{\sum_{i=1}^{k} \Delta C(i)}{A} \), which also ranges from 0 to 1. In order to model \( \varphi(x) \), we introduce a sigmoidal function of

\[ \varphi(x) = \frac{h e^{-\sigma x}}{1 + h e^{-\sigma x}}, \]  

(14)

where \( h \) is set to quite a large value so that \( \varphi(x) \) yields 1 when \( x \) is very small. Later, we set \( h \) to 10,000, and then find \( \sigma \) comparing with the simulation results. From the viewpoint of the filter theory, \( \varphi(x) \) can be seen as a low-pass filter. Since the inflection point of the curve of \( \varphi(x) \) is \( x = \frac{\ln h}{\sigma} \), for a given \( h \), we can easily control the passband by selecting the value of \( \sigma \) properly. When \( h = 10,000 \), \( \sigma > 4 \) so that the inflection point is between 0 and 1.

Using \( \varphi \), \( \Delta C(k+1) \) for \( k \geq 1 \) is represented by

\[ \Delta C(k+1) = \varphi \left( \frac{\sum_{i=1}^{k} \Delta C(i)}{A} \right) \Delta B, \quad \Delta C(1) = \pi r^2. \]  

(15)

Letting \( S(k) = \sum_{i=1}^{k} \Delta C(i) \), we obtain

\[ S(k) = \Delta C(1) + \sum_{i=2}^{k} \varphi \left( \frac{\sum_{j=1}^{i-1} \Delta C(j)}{A} \right) \Delta B, \]  

(16)

where \( I(\cdot) \) is the indicator function yielding 1 if given condition is true, otherwise zero. Given the initial condition, i.e., \( S(1) = \Delta C(1) \), \( S(k) \) can be solved iteratively. The computational complexity is \( O(k) \). Finally, the cumulative probability \( P_s(k) \) of a successful scan after \( k \) trials can be represented from Eq. (12) as

\[ P_s(k) = 1 - P(0) = 1 - e^{-\rho S(k)}. \]  

(17)

In order to find the parameter \( \sigma \), we obtain \( P_s \) via simulation by varying the scan interval \( \tau_v \) when the scan is done for the time duration \( t_{\text{max}} \). The profiles of the simulation are summarized as follows:

1) A square, whose sides have the length of \( 10^3 \sqrt{\pi} \) m, and hence its area \( A = 10^6 \pi \) m\(^2\), is assumed to be the coverage of the 802.16e single-cell, and the left-bottom point of the square is assumed as the origin \((0,0)\) of a coordinate system.

2) At time \( t = 0 \), \( M \) APs and an MSTA are placed randomly in the square, and at the same time, the MSTA conducts the first scan trial. As mentioned earlier, if there exists at least an AP located within the range of \( r = 50 \) m from the MSTA, it is regarded as successful scan trial.

3) Otherwise, the MSTA starts to move according to the random waypoint model [19], and hence the MSTA moves linearly with the constant speed of \( v \) towards randomly chosen destination. If the MSTA arrives at the destination, the coordinate of destination is chosen again randomly.

4) A run of simulation is finished if the scan trial is successful or the simulation time reaches \( t_{\text{max}} \). While the simulation is repeated as many as 10,000 runs, we count the number of runs, where at least an AP is found. For \( M = 10 \), we found that the results from our modeling, i.e., Eq. (17), is fit into the simulation results by using \( \sigma = 15 \). Then, we apply the same \( \sigma \) to other cases of \( M = 40, 120, \) and 200, respectively. Fig. 3 shows that the analytical results for other values of \( M \) is also fairly identical to the simulation
results. Interestingly, we observe from the simulation results that there is a limit in increasing \( P_s \) merely by scanning more often. For this reason, we can justify the rule of thumb used to determine \( \tau_s \) in Eq. (5). Also, for other values of \( v \), there is little difference between two approaches, though we do not present the figure due to the space limit. One thing we should mention is that our modeling depends on the specific assumption of both the mobility pattern and the distribution of deployed APs.

Now, having verified the accuracy of the proposed model, we investigate the impact of both \( \tau_s \) and \( v \) in Fig. 4 using the analytical model. Other parameters are maintained the same as used at the previous evaluation. At the first glance, we observe that when the AP density is higher, i.e., higher \( M \), it is more probable to discover the 802.11 AP(s) for given \( \tau_s \). However, the impact of the mobile speed on \( P_s \) is different depending on the scan interval \( \tau_s \). More specifically, when \( \tau_s = 5, 25, \) and 100 sec, \( P_s \) is higher at \( v = 0.8 \) m/s than at \( v = 0.5 \) m/s. However, for other \( \tau_s \)'s, \( v \) does not make any difference. This is because \( \Delta H \) in Eq. (13) depends on \( v \) only when \( v \Delta H \leq 2r \). We can confirm that the inequality is satisfied when \( \tau_s = 5, 25, \) and 100 sec for both \( v = 0.5 \) and 0.8 m/s. Another thing is that we can obtain the information of how quickly the 802.11 AP can be discovered with certain probability. For example, if an MSTA moving with \( v = 0.5 \) m/s scans AP with \( \tau_s = 25 \) sec when \( M = 200 \), the 802.11 AP can be found with the probability of 0.9 taking the scan time less than 580 sec. Therefore, we can provide to the system the information by when the WLAN interface can be available statistically. Consequently, VHO decision algorithm can be improved with more intelligent manner. We discuss this in detail in Section IV.

C. Optimizing Scanning Rule

Now, we consider a problem to optimize the scan operation. Our goal is to minimize the energy consumption when we conduct scanning for a given time \( \tau_s \). Here, we minimize the lower bound of \( \mathcal{E} \), i.e., \( \mathcal{E}_{fail} \) in Eq. (7) instead of \( \mathcal{E} \).

**Problem 1:** Scanning Problem in WLAN

\[
\begin{align*}
\min & \quad \mathcal{E}_{fail} \\
\text{s.t.} & \quad P_s \geq P_{\text{target}},
\end{align*}
\]

where the constraint is interpreted that the scan should succeed with a probability larger than \( P_{\text{target}} \). If \( \tau_s \) is determined according to the application's needs, \( \tau_s \) represents the maximum time the application can wait until the communication via WLAN interface is available. In this context, each individual MSTA can set \( \tau_s \) independently.

Here, we define \( \rho_{\text{min}} \) as the minimum value of \( \rho \) which belongs to the feasible region of Problem 1. Proposition 4 shows how \( \rho_{\text{min}} \) can be determined.

**Proposition 4:** \( \rho_{\text{min}} \) is the minimum \( \rho \), which satisfies Eq. (18) using \( \tau_s = \tau_s^{\text{min}} \).

**Proof:** Obviously, \( P_s(k) \) is proportional to \( k \) from Eq. (16). To satisfy \( P_s \geq P_{\text{target}} \) even at \( \rho_{\text{min}} \), \( k \) should be maximized. The scanning policy maximizing \( k \) is to use \( \tau_s = \tau_s^{\text{min}} \).

Therefore, \( \rho_{\text{min}} \) is given by

\[
\rho_{\text{min}} = \frac{\ln (1 - P_{\text{target}})}{S \left( \frac{\tau_s^{\text{min}}}{nT + \tau_s^{\text{min}}} \right)}. \tag{19}
\]

Table III represents \( \rho_{\text{min}} \) computed for \( \tau_s = 30, 100, 500, \) and 1,000 sec and for \( v = 0.5 - 5.0 \) m/s, respectively, when \( P_{\text{target}} = 0.95 \). We see that \( \rho_{\text{min}} \) becomes large as \( \tau_s \) and \( v \) are smaller.

For a given \( \rho \geq \rho_{\text{min}} \), we solve Problem 1 using the energy-efficient scan policy (ESP) algorithm:

**Algorithm 1:** Energy-efficient Scan Policy (ESP)

1. Eq. (7) shows that a small \( \tau_s \) is preferred for a given \( k \) to minimize \( \mathcal{E} \). Therefore, from \( k = \frac{\tau_s^{\text{max}}}{nT + \tau_s} \), we set \( \tau_s = \min \left\{ \frac{\tau_s^{\text{max}}}{nT + \tau_s}, \tau_s^{\text{max}} \right\} \).
2. According to Proposition 3, \( \rho \) should be minimized to maximize \( \tau_s \). For this reason, we find \( k_{\text{min}} \), which satisfies the constraint in Eq. (18) for given AP density \( \rho \), informed by the 802.16e BS:

\[
k_{\text{min}} = \min \left\{ k | S(k) \geq \frac{\ln (1 - P_{\text{target}})}{\rho} \right\}. \tag{20}
\]

3. Finally, the optimal scan interval \( \tau_s^{\ast} \) is represented as

\[
\tau_s^{\ast} = \max \left\{ \frac{\tau_s^{\text{max}}}{k_{\text{min}}} - nT, \tau_s^{\text{min}} \right\}. \tag{21}
\]

In ESP algorithm, \( k_{\text{min}} \) is ensured to exist, i.e., \( k_{\text{min}} < \infty \) if \( \rho \geq \rho_{\text{min}} \). Since \( (k) \) is monotonically increasing with \( k \), \( k_{\text{min}} \) can be found fast by using an efficient search algorithm such as binary searching [23].

Fig. 5 shows the scan interval \( \tau_s^{\ast} \) determined by ESP algorithm depending on the VHO information, i.e., the number of APs, when an MSTA moves with the constant speed of 3.0 m/s. Here, since we assume \( A = 10^3 \pi m^2 \), for \( \tau_s^{\ast} \)

\[TABLE III
\begin{tabular}{|c|c|c|c|c|}
\hline
\( v(\text{m/s}) \) & \( t_{\text{max}}(\text{sec}) \) & 30    & 100   & 500   & 1,000 \\
\hline
0.5       & 32.05  & 25.32 & 9.122 & 5.179 &         \\
1.0       & 27.63  & 16.80 & 5.180 & 2.778 &         \\
2.0       & 21.66  & 10.77 & 2.279 & 1.442 &         \\
3.0       & 17.81  & 7.227 & 1.899 & 0.973 &         \\
4.0       & 15.12  & 6.271 & 1.442 & 0.735 &         \\
5.0       & 13.14  & 5.187 & 1.162 & 0.590 &         \\
\hline
\end{tabular}
users sensitive to service price can get economic benefits using 802.16e in terms of monetary cost. Most think that VHO is still advocated in that WLANs possess throughputs’ perspective is somewhat weakened. However, we consider the 802.16e network, the most distinguished point is that the peak rate of the 802.16e becomes comparable to 100 Mbps Ethernet, in this paper, we only consider WLANs with the 802.11b PHY, which has been already widely deployed.

More specifically, we assume that the service fee for WLAN is charged according to flat pricing [20] while the service fee for the 802.16e is charged according to partially-flat pricing, where the service fee begins to increase in proportion to the network usage exceeding certain threshold \( u_{th} \). Moreover, we assume that the maximum of the total charge for a specific user \( i \) is limited by \( \nu_{\text{max},i} \). This assumption is practically reasonable since there is a restriction such that the communication fee can not be imposed infinitely by law or by public restriction. Indeed, we see that such a policy is popularly used by many commercial operators, and \( \nu_{\text{max},i} \) is often determined by contract between the user and the service operator. Finally, when the service fee for both WLAN and the 802.16e networks is imposed with integrated manner, the total service fee \( \nu_i \) charged to user \( i \) is modeled as

\[
\nu_i(u) = \min \{ \nu_{\text{max},i} \nu, c + d(u - u_{th})I_{u > u_{th}} \},
\]

where \( c \) is the minimum fee which the user should pay irrespective of the network usage, \( u \) is the total amount of network usage via the 802.16e network for a subscription interval, and \( d \) is the charging rate when \( u > u_{th} \). Naturally, \( c \leq \nu_{\text{max},i} \) for the non-triviality of pricing model. And, we define \( u^* \) as

\[
u^* = \min \{ u \mid \nu_i(u) = \nu_{\text{max},i} \}.
\]

To increase the revenue, the operator reduces the service rate or lowers the service quality of the user whose network usage of the 802.16e exceeds \( u^* \).

From the user’s perspective, it is the goal for users to use two access networks freely as to maximize their satisfaction. If the total amount of network usage for a subscription interval is surely less than \( u_{th} \), the monetary cost is not the attribute impacting on user’s VHO decision. In this case, any decision criteria proposed by the related researches [6]–[15] can be employed. However, if \( u \) is larger than \( u_{th} \), the service fee which the user should pay starts to increase, and further if \( u \) is also larger than \( u^* \), the user will even experience the degradation of service quality provided by the 802.16e network until the service period is renewed. Recall that the network operator would give penalty for the traffic exceeding \( u^* \). Therefore, the user may need to balance the service amount via the 802.16e network in advance in order to keep the quality
of service maintained in the future critical time, i.e., when the 802.16e network is the only option to use.

For this purpose, we propose the following strategy for the MSTA: If the traffic delivery can be deferred, postpone it until WLANs are available. Mainly, this policy is applicable to latency-insensitive applications such as file transfer and e-mail delivery, which can also be conducted by background process. In order to realize this policy, a smart middle-ware is required since the traffic delivery should be controlled depending on the connectivity status (on/off) of specific network interface. Sun et al. [12] proposed an architecture for connectivity manager, which has the functionality of Disconnection treatment, and thus the application can be be suspended when the connectivity is not available and also be resumed when available. However, if WLAN is not available so long time that the traffic delivery of the application is pending over certain threshold of time, the user may want to resume the communication via the 802.16e network at the risk of service fee increase or even the service degradation possible at the future time.

B. Delayed Traffic Delivery (DTD)+ESP Algorithm

The ESP algorithm presented in Section III provides the statistical guarantees on discovering WLANs whenever \( \rho \) informed by the 802.16e BS is larger than \( \rho_{\text{min}} \). Using this ESP algorithm, we can easily decide in advance whether to suspend the traffic delivery of delay-tolerable applications until WLANs are available. Motivated by this idea, we propose a VHO decision algorithm named ‘DTD+ESP’. Let us assume that a user is satisfied if a movie file with size of \( Q \) is downloaded no later than time \( T \).

Algorithm 2: DTD+ESP algorithm

1) Set \( t_{\text{max}} = \frac{T}{N} - \Delta \), where both \( N \) and \( \Delta \) are controllable parameters.
2) From Eq. (4), determine \( \rho_{\text{min}} \).
3) Given the AP density \( \rho \), which can be known by the broadcasting information provided by the 802.16e network, attempt to scan WLANs with \( \tau^* \) which is given by Algorithm 1.
4) Check the data amount received at every \( t = i \frac{T}{N} - \Delta \) \( (1 \leq i \leq N - 1) \). If it is less than \( i \frac{Q}{N} \), receive the data from the 802.16e network during \( \Delta \). In other words, it is guaranteed that at least \( i \frac{Q}{N} \) of data is delivered by \( i \frac{T}{N} \).

Given \( T = 4 \frac{Q}{\omega} \), where \( \omega \) represents the lowest service rate provided by the 802.16e network, if we set \( \Delta \) to \( \frac{Q}{\omega N} \), then DTD+ESP algorithm can guarantee the delivery of file \( Q \) within the time of \( T \). \( P_{\text{target}} \) is set to 0.9\( P_s \), where \( P_s \) is the successful scan probability achieved by using scan interval \( \tau_{\text{min}}^* \) during \( t_{\text{max}} \). For the purpose of performance comparison of DTD+ESP algorithm, we introduce two other algorithms as follows:

1) Periodic Scan (PS): The MSTA always attempts to scan with fixed scan interval. If the WLAN is available at a given time, receive the data through WLANs. Otherwise, receive the data via the 802.16e network.
2) DTD+PS: In this algorithm, the MSTA always wants to receive the data only via WLANs. Accordingly, it defers receiving data until the WLAN is available. The MSTA attempts to scan with fixed scan interval as well.

The reason why we consider above two options is that the PS algorithm looks quite similar to the behavior shown by the typical VHO algorithm, and the next one, i.e., DTD+PS, can minimize the network usage \( u \) counted in the 802.16e network, that is advantageous to minimize the service fee \( \nu_q \) or to avoid the possible penalty given by the 802.16e network. The scan interval for both PS and DTD+PS algorithms is set to \( \tau_{\text{min}}^* \).

Basically, we evaluate the performance under the mobile environment, which has been described in Section III-B. At a given random position of APs, three MSTAs start downloading the file simultaneously, and each of them performs different VHO algorithm, i.e., one algorithm out of DTD+ESP, PS, and DTD+PS. To make use of WLAN link, the MSTA should locate within coverage of at least an AP, and the WLAN interface of MSTA should be also active. Whenever the MSTA moves out of the coverage of WLAN, the WLAN interface becomes inactive until the next scan trial is successful. In order to simplify the evaluation, however, we consider some assumptions additionally as follows:

1) We ignore all kinds of handoff delay, i.e., 802.16e-WLAN and WLAN-WLAN. Accordingly, the maximum time how long the MSTA can communicate via WLAN is the same as the sum of sojourn time over seamless coverage among WLANs.
2) We consider fixed service rate for both WLAN and the 802.16e network. Assuming the data rate of wireless one hop is the bottleneck link, the service rate of WLAN is assumed \( R_{11} = 5.5 \text{ Mbits/s} \), and the service rate of the 802.16e is \( R_{16} = 4 \text{ Mbits/s} \) [24].
3) The MSTA has no other applications except for file downloading, and the size of file is \( Q \). In this case, the MSTA running DTD+PS algorithm is equivalent to a WLAN-only station.
4) \( u^* = \eta Q \) \( (0 < \eta < 1) \), and when \( u > u^* \), the service rate of the 802.16e is reduced to \( R_{16}^* = 0.5 \text{ Mbits/s} \) due to the penalization by the 802.16e network. Therefore, \( \omega = R_{16}^* \).
5) The energy consumed during scanning follows \( E_{\text{fluid}} \) as shown in Eq. (7).

Fig. 6 shows the result obtained by repeating the simulation as many as 200 times at a given number of APs \( M \) when \( Q = 2,400 \text{ Mbits}, N = 1,000, T = 19,200 \text{ sec} \), and \( \eta = 0.5 \). PS algorithm can finish receiving the file the most quickly, and hence the energy consumption during the scan is small. However, this is obtained at the expense of much abundant amount of the 802.16e network usage. Fig. 6 (a) shows that the network usage in PS algorithm exceeds \( u^* \) while DTD+ESP algorithm violates \( u^* \) only when \( M \) is small. On the other hand, DTD+PS algorithm never use the resource of the 802.16e network, but it may not guarantee the traffic delivery time as shown in Fig. 6 (b). Especially, when \( M \) is small, considerable amount of energy is also consumed ineffectively to find WLAN APs. In summary, DTD+ESP
algorithm based VHO can reduce the user’s service fee while guaranteeing the traffic delivery time and minimizing the energy consumed to find target WLAN APs.

V. CONCLUSION

In order to support an energy-efficient VHO between IEEE 802.16e and IEEE 802.11, we propose that the 802.16e BS broadcasts the AP density within its cell coverage. Considering the 802.11 active scan, we develop a mathematical model, which shows the successful scan probability in terms of the AP density and the scan interval. Based on this model, we propose an energy-efficient scan Policy (ESP) algorithm, which optimizes the scan interval such that the amount of the energy consumed for scan operation is minimized while satisfying the target successful scan probability. For VHO decision, we mainly consider the impact of the service charge on VHO. Under a practical service charge plan, it is beneficial for users to control the traffic of the 802.16e network. To this end, we propose a delayed traffic delivery scheme combined with our ESP algorithm, especially, for delay-tolerant applications’ traffic. By doing so, we can provide the economical efficiency to users while guaranteeing the traffic delivery time and minimizing the energy consumed due to frequent WLAN scan trials.

APPENDIX I

THE PROBABILITY DENSITY FUNCTION OF $s$

Lemma 1: When an MSTA with velocity $v$ moves linearly across a circular cell with radius $r$, the probability density function (pdf) of cell sojourn time $s$ is expressed by

$$f(s) = \frac{sv^2}{2r\sqrt{4r^2 - s^2v^2}} \quad 0 \leq sv \leq 2r.$$  

(23)

Here, a cell sojourn time is defined as how long the MSTA resides in the cell coverage.

Proof: We skip the proof due to the lack of space.

REFERENCES