國立交通大學

電信工程學系碩士班

碩士論文

在無接縫垂直交接情況下的網路成本最小化研究

Network Cost Minimization for Seamless Vertical Handover

研究生：陳怡中　　Student: Yi-Chung Chen

指導教授：廖維國　博士　　Advisor: Dr. Wei-Kuo Liao

中華民國九十三年六月

在無接縫垂直交接情況下的
網路成本最小化研究

研究生：陳怡中　　　　　　　　指導教授：廖維國博士

國立交通大學電信工程學系碩士班

民國93年6月

中文摘要

在下一代的行動電話系統當中，使用者會希望系統業者提供一種連線方式，讓其即時的傳輸服務可以總是保持在最佳的網路上，除此之外，使用者也預期在進行垂直交接的時候不會有延遲的情況發生。因此在此篇論文中，我們針對現有的3G UMTS 以及 WLAN 來提供使用者這樣的需求。我們提出一種新的方案叫做“Designated Crossover Point”（DCP）來支援快速的垂直交接，接下來，我們讓DCP可以交接到使用者所在的路由器上，以增進路由的效率，最後，根據VHE user profiles 所提供的移動特性，我們可以利用馬可夫決定過程計算出最佳的交接方式，並且這些交接方式只需要網路最小的成本。在模擬結果當中，我們考慮了兩種情況：1)只考慮一個3G UMTS 以及一個WLAN的結合，2)考慮一個3G UMTS 以及兩個WLAN的結合，不管是在哪種情況之下，我們都可以為系統業者找出最佳的策略去執行DCP的交接，並且將網路的成本最小化，此外，我們也將討論不同參數對於交接策略的影響。

Network Cost Minimization for Seamless Vertical Handover

Student: Yi-Chung Chen Advisor: Wei-Kuo Liao

INSTITUTE OF COMMUNICATION
ENGINEERING NATIONAL CHIAO TUNG
UNIVERSITY

June, 2004

ABSTRACT

In the next generation of mobile system, it is expected to support the concept of *Always Best Connected* (ABC). Besides, from user's perspective, the seamless vertical handover for real-time service is also very important. Therefore, in this thesis, we focus on the integration of 3G UMTS and WLAN through loosely-coupled interworking architecture, and propose a scheme called "*Designated Crossover Point*" (DCP) to execute vertical handover in a fast way. Next, we let the DCP to handover to increase routing efficiency, and finally, by the value iteration method of Markov Decision Process, we can find the optimal DCP handover policy when we can get the mobility characteristic form VHE user profiles in advance. According to the policy the network operator can decide when to perform a DCP handover, which can minimize the network cost. We consider two cases: 1) there are only one UMTS and one WLAN, 2) there are one UMTS and two WLANs. In both case we can always minimize the network cost by the best DCP handover policy. Furthermore, we discuss the effect of tuning the system parameters.

Keywords: Always Best Connected (ABC), IP end-to-end QoS architecture, Virtual Private Network (VPN), VHE user profiles, Markov Decision Process, value iteration, UMTS/WLAN.

誌謝

碩士班兩年的時間已經接近尾聲，首先要謝謝我的指導老師：廖維國教授辛苦的指導，不管是在課業方面，或是生活上，都給我許我寶貴的建議，使我獲益良多。還要感謝網路分析實驗室812b的各位學長：邦欣學長、永瑋學長、家宏學長；各位同學：政澤、怡翔、國瑋、柏谷；以及學弟：于彰、健智、憲良、賢宗；這段時間有你們的陪伴讓我的碩士班兩年留下很好的回憶。最後，感謝我的父母給我這樣的機會與環境，使我可以無後顧之憂的專心學習，謝謝你們。

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The next generation of mobile systems will support multiple radio access technologies. As a result, for a user who wants to connect to a service, she or he is expected to dynamically choose access technologies in a way that best suits his or her needs, and change when something better becomes available. Such a dynamic way of radio access selection is called *always best connected* (ABC) [1]. Besides, user would like to experience *seamless vertical handover* for real-time services, i.e., no service interruption is perceived when changing the radio access technologies. In our work we focus on supporting ABC in 3G UMTS/WLAN integration with seamless handover.

Figure 1.1 shows a possible configuration where WLAN forms small cells within large UMTS cells. In the configuration, a 3G UMTS operator can benefit from the high-speed connection provided by the WLAN in some hot-spot areas, e.g. in internet coffee shop, within an office building, and apartment building.

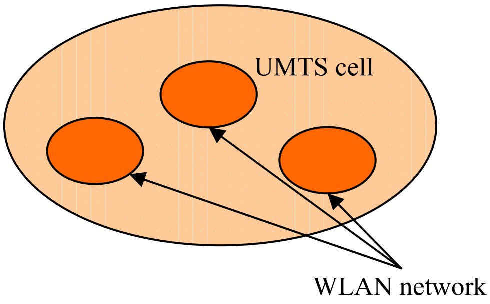

Figure 1.1 Hot-spot scenario between UMTS and WLAN

In order to supporting seamless handover for real-time service under this heterogeneous network, from the user's perspective, one of the important issues is how to limit the delay to an acceptable level when the mobile is performing a vertical handover. While the radio-level handover latency only depends on the circuitry of the access device and can be improved simply by circuit design, the majority concern of handover delay will come from the network side.

In this thesis, we therefore discuss the forementioned problems from the network architectural aspect. We will review *Virtual Private Network* (VPN), Mobile IP, and the IP end-to-end QoS architecture, and then form a promising architecture to support fast vertical handover based on the existing infrastructure. In addition, by the concept of *Virtual Home Environment* (VHE) *user profiles* defined in 3GPP, we can calculate the network cost, which could be the *network implied cost* for maximizing the network utilization or service charge, by formulating our problem to a *Markov Decision Process*, therefore, mobile operators can select the best policy to minimize their network cost.

The rest of the thesis is organized as follows: In the chapter 2, we discuss the scope of this problem and give overview to our system architecture. In the chapter 3, we propose a novel scheme called "Designated Crossover Point" (DCP) to speed up the vertical handover, and using the VHE user profiles to minimize network cost. In the chapter 4, we describe the necessary background knowledge about value iteration method of Markov Decision Process for simulation. And the simulation results are shown in the chapter 5. In chapter 6, we make conclusion and list the future work.

Chapter 2

Scope

In this chapter the 3G UMTS and the WLAN are briefly introduced, and we will show two interworking architectures that integrate UMTS and WLAN together. Since there are pre-allocated pipes across the IP backbone for each UMTS-WLAN pairs to supporting the vertical handover, we employ the concept of Virtual Private Network (VPN) to achieve this. The VPN connection model will be introduced. The Resource Reservation Protocol (RSVP) and Mobile IP are used to guarantee the mobile user will receive at least the same QoS level when moving from one domain to another. By using Designated Crossover Point (DCP) the handover latency can be reduced, and the network cost can be minimized.

2.1 Vertical handover system architecture

In the 3G UMTS network there are two landbased network segments: the UMTS radio access network (UTRAN) and the core network (CN), and the 3G W-CDMA air interface. The standard interfaces and components of a 3G UMTS network are outlined in TS 23.002 and illustrated in figure 2.1 [2]. Together, they form the *administrative domain* of the mobile operator. The CN itself is further divided into the circuit-switched and packet-switched domains. The UMTS had been designed for connecting to Internet ubiquitously.

A mobile user's equipment (UE) can communicate with multiple base stations which is called *Node Bs* in UMTS over the wireless Uu interface. The outgoing (uplink) user-level packets will be segmented by the UE into radio network layer (RNL) frames, called *transport blocks*. Then these transport blocks are transmitted over the air interface, using the wideband CDMA (W-CDMA) access and modulation techniques, to the Node B which serves mobile user. Each Node B encapsulates a set of transport blocks into a single frame and forwards the frame to its *radio network controller* (RNC) over the Iub interface. The RNC can be considered as a *base station controller* (BSC) in GSM.

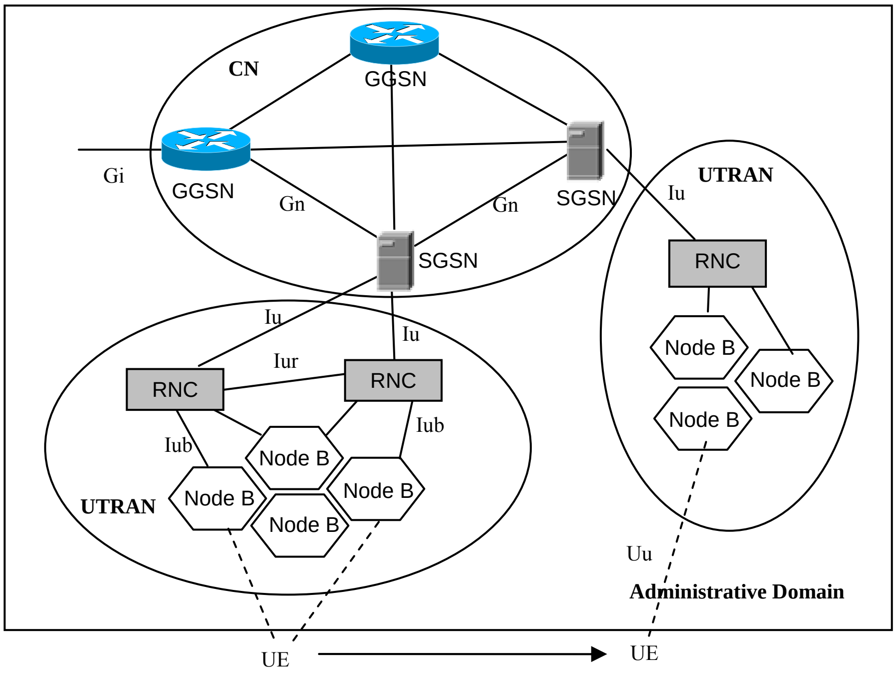

Figure 2.1. UMTS network architecture

When the multiple Node Bs serving a mobile user have different controlling RNCs, one of the RNCs will act as the *serving* RNC for that UE. The serving RNC of the mobile user is responsible for frame selection among the multiple received copies of the same transport block, processing the other sublayers of the RNL, and reassembling the user-level packet. It also maintains the link layer state for the UE, that is, it maps the host identity with the identities of the Node Bs and the communication channels within each Node B that currently serves that host. The transport network between the Node Bs and the RNCs has been traditionally composed of T1 lines.

The packet-switched portion of the core network in UMTS consists of two types of Generalized Packet Radio Service (GPRS) support nodes (GSNs) , the *serving GPRS support node* (SGSN) and the *gateway GPRS support node* (GGSN). The SGSN is responsible for mobility management (location update, page etc.), session management, BSS queue management, producing billing information, and routing packets to correct RNC. The GGSN is responsible for interworking between PDN and GPRS PLMN, packet screening, address mapping, and PDU tunneling, it is like an IP gateway and a border router. To communicate with the data network, the mobile host have to register with the CN by performing a *GPRS attach* operation. This action results in the creation of two GPRS tunneling protocol (GTP) sessions, specific to that host: between the RNC and the SGSN on the Iu interface, and between the SGSN and the GGSN on the Gn interface. The user-level packets are encapsulated into GTP frames and are forwarded between the RNC and the GGSN over a chosen transport network, like ATM.

In general the SGSN controls the inter-RNC mobility management of the mobile user, while the GGSN handles the inter-SGSN mobility management. Through GPRS attachment, there will be a mapping created at the RNC between the host identity and the GTP session between the RNC and the SGSN. Besides, a record is saved at the GGSN which contains the mapping between the host's IP address and the GTP session with the corresponding SGSN. When the serving RNC of the mobile changes, as long as the new RNC is within the region of the same SGSN, it only redirects the GTP session between the SGSN and the RNC. The session between the SGSN and GGSN remains the same. On the other hand, both host-specific GTP sessions are reestablished if mobility results in a different point of GPRS attachment. In addition to mobility management the GSNs also perform various billing and security functions that do not affect the underlying network architecture.

The WLAN, e.g., IEEE 802.11 or HYPERLAN [3], is deployed to achieve hotspot coverage. IEEE 802.11 standard defines the protocol to interconnect the portable devices with equipment through the radio or infrared medium in a LAN by using the *Carrier Sense Multiple Access protocol with Collision Avoidance* (CSMA/CA). The basic service set (BSS) is the basic building block of an IEEE 802.11 network. The 802.11 protocol supports wireless LANs in two different modes, namely *infrastructure-based* or *ad-hoc networking*. In adhoc networking, IEEE 802.11 WLAN is formed without preplanning, and it is supported by the Distributed Coordination Function (DCF). Infrastructure-based LAN is supported by both DCF and Point Coordination Function (PCF), and an access point (AP) provides an access to the wireline backbone network, which is called Distributed System (DS). The whole interconnected WLAN including different cells, their respective APs and the DS, is seen to the upper layers of the OSI model, and is called Extended Service Set (ESS). The figure 2.2 is a typical 802.11 WLAN with the components described previously. The connection between AP and the DS can be broadband wireless link, wireline connection, or radio connection.

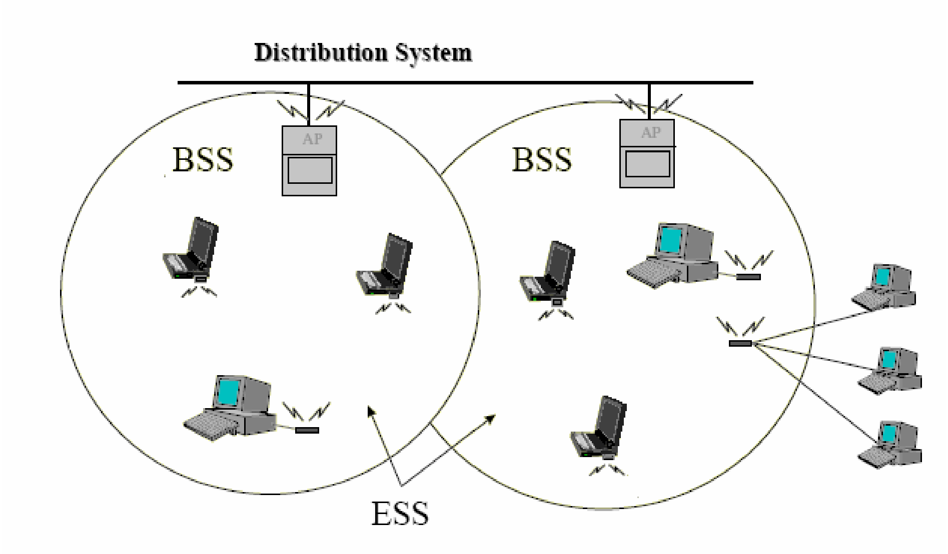

Figure 2.2. WLAN network architecture

In this work we study the case where the WLAN coverage is mostly overlapped with the UMTS coverage. Consequently, when the mobile user crosses the WLAN coverage, it may perform the vertical handovers between UMTS and WLAN back and forth to let its real-time services best connected.

We will refer to the heterogeneous wireless access services consisting of less than ten UMTS networks and thousands of WLANs, and all are connected to an IP backbone. In each UMTS network, there are thousands of base stations, tens of RNCs, and very few SGSNs and GGSNs.

2.1.1 Loosely-coupled and tightly-coupled interworking

Depending on the degree of inter-dependence that one is willing to introduce between the UMTS network and WLAN, there are two different ways of integrating the two wireless technologies [4] [5]. We define them as *tightly-coupled interworking* and *loosely-coupled interworking*.

Tightly-coupled Interworking

The WLAN 1 and UMTS in figure 2.3 show an example of the tightly-coupled interworking. In the tightly-coupled interworking the WLAN network is connected to the UMTS core network in the same manner as UTRAN, the Interworking Unit (IWU) of WLAN is connected to SGSN. The WLAN network emulates functions natively available in UMTS access networks. In this architecture the IWU appears to the UMTS core network as an RNC, and hides the details of the WLAN from the UMTS core network, it also implements the required UMTS protocols. Mobile Nodes (MNs) have to run the UMTS protocol stack on top of their standard 802.11 network interface cards and switch from one physical layer to the other as vertical handover happens. All 802.11 client traffic is injected using UMTS protocols into the core network. The networks share the same authentication, signaling, transport and billing infrastructures.

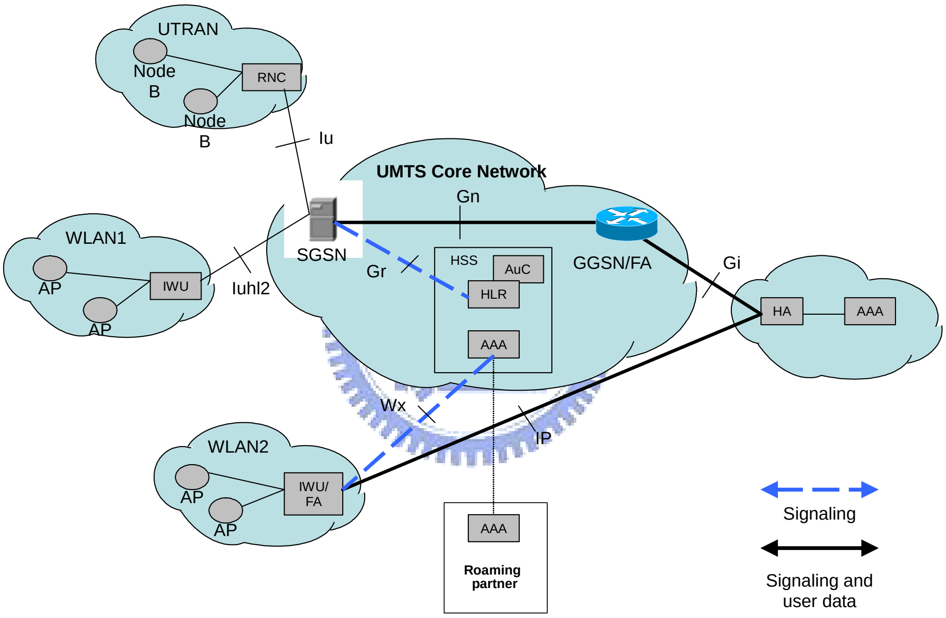

Figure 2.3. Two scenarios to interwork the UMTS with WLAN

This architecture presents several disadvantages. First, exposing the UMTS interface may incur security problem. Second, injecting WLAN traffic directly into the UMTS core network will upsets the carefully-engineered capacity and configuration of the network. The CN setup – including SGSN, router and GGSN – have to be re-engineered to support the increased load and differing traffic characteristics. On the other hand, the configuration of user device also has to be modified. The authentication mechanism also forces WLAN operators to interconnect to the UMTS operator’s SS7 network.

Loosely-coupled Interworking

The WLAN 2 and UMTS in figure 2.3 display an example of the loosely-coupled interworking. However, in this architecture the IWU connects to the Internet and does not have any direct link to the UMTS core network. The WLAN may serve users visiting from other networks as well as local subscribers. This approach completely separates the data paths in WLAN and UMTS networks, the WLAN data flow is never injected into the UMTS core network, but the mobile user still receives seamless handover.

In the architecture different mechanisms and protocols can be responsible for authentication, billing and mobility management in the UMTS and WLAN. However, for seamless handover with UMTS, this requires that the WLAN edge router, or gateway, provides mobility management for access across both network, as well as authentication, authorization, accounting (AAA) services to interwork with the UMTS networks. This enables the UMTS operator to collect the WLAN accounting information and produce a unified billing statement indicating usage and various price schemes for both networks.

There are several advantages for the loosely-coupled interworking. First, it allows the independent deployment and traffic engineering of WLAN and UMTS networks. UMTS operators can benefit from other operators' WLAN deployments without extensive capital investments. At the same time, they can continue to deploy 3G networks using well-established engineering techniques and tools. Furthermore, by roaming agreements with many partners can result in widespread coverage, including hot-spot areas, users benefit from having just one subscription for all network access. They no longer have to create separate accounts with mobile operators in different regions or covering different access technologies.

The loosely-coupled approach clearly offers several architectural advantages over the tightly-coupled approach with virtually no drawbacks. Therefore, in this work we choose the loosely-coupled interworking as the preferred architecture for the integration of WLAN and UMTS networks.

2.1.2 Virtual Private Networks (VPN) connection model

A virtual private network (VPN) is the extension of a private network that involves various links across shared or public networks like the Internet, it is a network tunnel created for data transmission between two or more authenticated parties. A VPN enables you to send data between two computers across a shared or public internetwork by emulating the properties of a point-to-point private link.

In order to emulate a point-to-point link, data should be encapsulated with a header that includes routing information allowing it to traverse the shared or public internetwork to reach its endpoint. To emulate a private link, the data ready to be sent is encrypted. Packets that are intercepted on the shared or public network are hard to break without the encryption keys. We call the part of the connection in which the private data is encapsulated as *a tunnel*. The part of the connection in which the private data is encrypted is known as the *virtual private network (VPN) connection* (figure 2.4).

VPN connections allow users working at home to connect to a remote corporate server using the infrastructure provided by a public internetwork like the Internet. Users see the VPN connection as a point-to-point connection between their computers and a corporate server. The nature of the intermediate internetwork is irrelevant to the user because it appears as if the data is being sent over a dedicated private link [6].

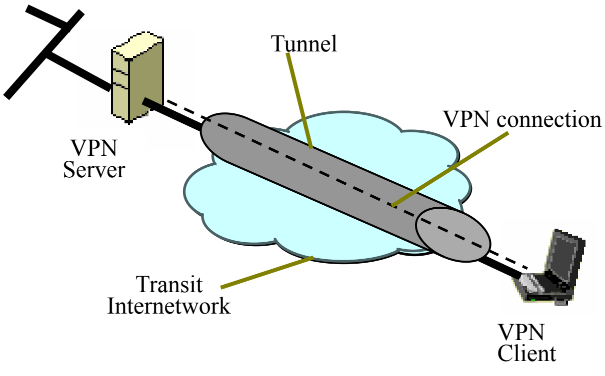

Figure 2.4. VPN connection model

VPN technology also allows a corporation to connect to branch offices or to other companies over a public internetwork, such as the Internet, in a secure way. The VPN connection across the Internet logically operates as a wide area network

(WAN) link between the sites.

In both of these cases, from user's perspective, the secure connection across the internetwork can be seen as a private network communication, despite the fact that this communication occurs over shared or public internetwork, hence we give it the name *virtual private network*.

The hub-and-spoke topology is the most well-known VPN network topology, in which a number of remote offices (spokes) are connected to a central site (hub), similar to the setup in figure 2.5. The remote offices usually can exchange data, but the amount of data exchanged between them is negligible. The hub-and spoke topology is used typically in organizations with strict hierarchical structures.

With increased redundancy requirements, the simple hub-and spoke topology is often enhanced with an additional router at the central site or with a backup central site, which is then connected with the primary central site through a higher-speed connection.

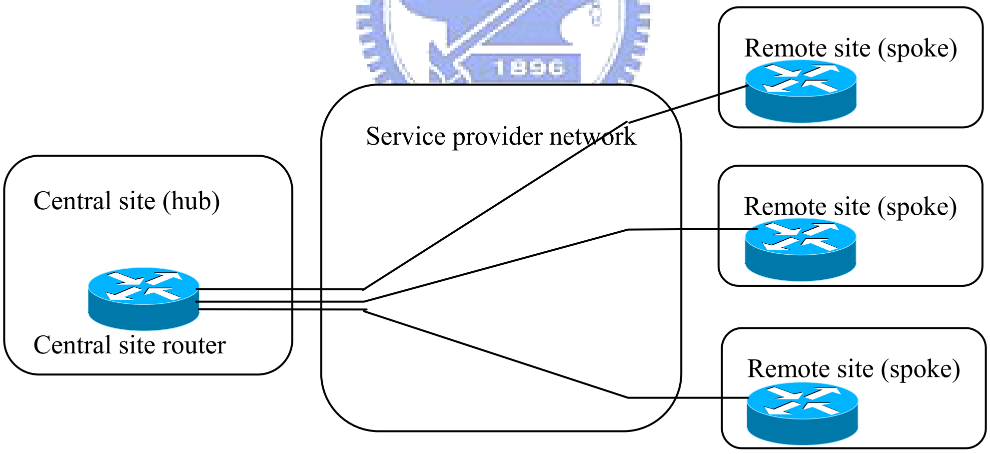

Figure 2.5. Hub-and-spoke topology

In our work we advocate hub-and spoke with redundancy topology where the hubs are the GGSNs and spokes are the customer's edge (CE) router of WLANs to pre-allocate pipes between them. As depicted in figure 2.6, there are two UMTS networks, UMTS 1 and UMTS 2, three WLANs, indicated by WLAN i, the pipes across the IP backbone are pre-allocated for each UMTS-WLAN pair to support the vertical handover.

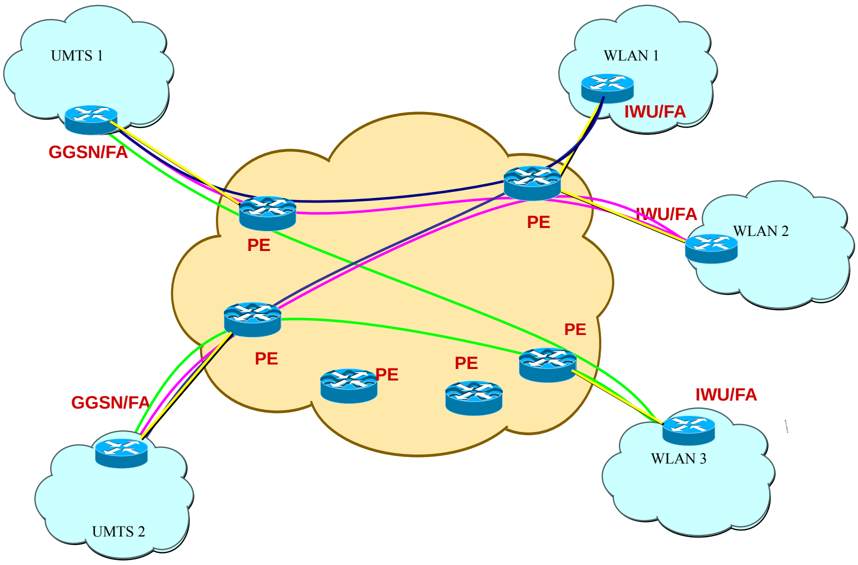

Figure 2.6. Pipe pre-allocation for handover management in WLAN/UMTS

2.2 Supporting macro-mobility through Mobile IP and QoS by RSVP

Mobile IP

It is generally agreed that IP terminal mobility can be broken into two parts – macro-mobility and micro-mobility – and that these need two different solutions. The macro-mobility means "mobility over a large area" and the micro-mobility means "mobility over a small area". More precisely, we can consider macro-mobility as "handover between different access network" and micro-mobility as "handover within the same access network". Here we will give an overview to Mobile IP which is the best-known proposal for handling macro-mobility handover.

In Mobile IP [7], a mobile host is always identified by its home address, regardless of its current point of attachment to the Internet. When the mobile user is away from its home network, the mobile also has another address, called a "Care-of Address" (CoA), which is associated with the mobile's current location. Mobile IP solves the mobility problem by storing a dynamic mapping between the home IP address, which acts as its permanent identifier, and the CoA, which acts as its temporary locator.

The key functional entity in Mobile IP is the Home Agent (HA), which is a specialized router that maintains the mapping between a mobile's home and CoA. Each time the mobile user moves to a new subnet, it obtains a new CoA form the Foreign Agent (FA) which is another specialized router in the visited network and registers it with HA. Mobile IP enables that a Correspondent Node (CN) can always send packets to the mobile: the correspondent addresses them to the mobile's home address – so the packets are routed to the home link – where the HA intercepts them and uses IP-in-IP encapsulation to tunnel then to the Mobile's CoA. In other words, the HA creates a new packet, which the new header containing the CoA and the new data part consisting of the complete original packet including the original header. At the other end of tunnel, the original packet can be extracted by removing the outer IP header which is called decapsulation.

Note that Mobile IP is only concerned with traffic to the mobile. In the reverse direction, packets are sent directly to the correspondent host, which is assumed to be at home. If it is not, mobile IP have to be used in that direction as well. Transparency is a key feature of Mobile IP, we can continue to use the same IP address because the HA routes them transparently to the mobile's current CoA.

In the section 2.1.1 we advocate loosely-coupled interworking architecture as our integration scenario. With the Mobile IP, we can manage macro-mobility and if so, the IWU and GGSN then act as FA, and the packets sent to the mobile will be transferred to the HA first and then relayed to the FA. An optional extension to Mobile IP, called *Route Optimization*, allows a correspondent node to send packets directly to a mobile node by informing the address of FA to the CN.

IP end-to-end QoS architecture

In order to supporting QoS we review the IP end-to-end QoS architecture.

Figure 2.7 shows an example of interworking between different customer networks, which could be UMTS networks or WLANs, and an IP backbone. As shown in the figure, both the customer's network and IP backbone include *QoS servers*, which is called QoS server in customer's network (QCS) and QoS server in provider's network (QNS), respectively. There are three main purpose of QoS server, to accommodate the new signaling protocols, to interworking between customer's network and IP backbone, and to enforce the policy [8]. Note that in the customer's network, the resource reservation is through the RSVP-TE (*tunnel extension*), and in the IP backbone the resource reservation is through the RSVP-E2E (*end-to-end flow*). As a result, if we want to reserve the resource over the IP backbone, it requires the *QCS-QNS negotiation*, which may be via the protocol *COPS*.

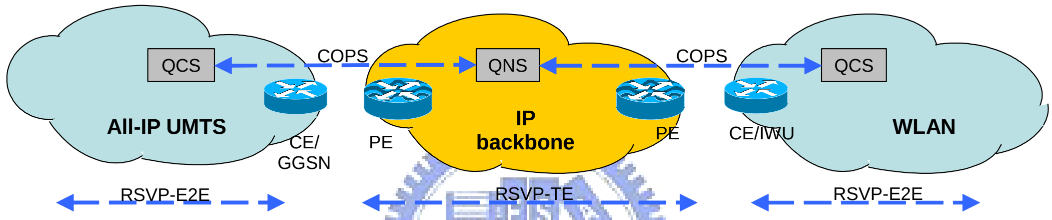

Figure 2.7. A reference IP end-to-end QoS architecture.

The delay induced by QCS-QNS negotiation is expected to be the major factor contributing to the connection setup latency. And since the core routers in IP backbone only recognize the aggregated flows under Diffserv architecture, if we want to enhance these core routers to recognize micro flow, there will be an issue on scalability.

To simplify our discussion, we let IWU be CE router in WLAN and let GGSN be CE router in UMTS.

2.3 Business model

In this thesis, we talk about network cost minimization problem based on

Mobile IP between UMTS and WLAN in the loosely-coupled intrworking architecture for real-time service. Whether the 3G UMTS operator has its own WLANs or the 3G UMTS operator interoperates with WLAN through roaming agreements, our architecture can always be used to discuss the problem.

Chapter 3

Proposed Scheme

3.1 Fast vertical handover via Designated Crossover Points (DCP)

In the chapter 2 we know that the mobile will change its association of the network and CoA, when a vertical handover happens. As a result, if without proper arrangement, the QCS-QNS negotiation for the resource reservation over IP backbone to the CE of the mobile induces a high latency of vertical handover and serious interruption of real-time services.

The latency hiding techniques would be useful to relieve the problem mentioned above. There are two approaches of such techniques possible: (a) *delaying the vertical handover* , which means enabling the vertical handover after the new path is established, and (b) *multicasting the streams to the base stations* where the mobile currently and possibly visits [9] [10]. Under the IP end-to-end QoS architecture, the efficiencies of both approaches depend on the location to join the old and new paths.

We can do the join operation in the endpoint (CN or HA) or crossover router. When joining the old and new path in the end point, the latency induced by QCS-QNS negotiation will seriously affect the real-time service. To cut down the latency of the vertical handover, we can simply join the old and new paths at the crossover router [11] [12]. The original proposals on such a solution are to join at the optimal crossover point [13] when dealing with macro-mobility or a fixed gateway when dealing with the micro-mobility problem. However, within the IP end-to-end QoS architecture, more considerations are needed to be taken for the vertical handover.

There are three possible crossover points to perform the join operation under the reference IP end-to-end QoS architecture in figure 3.1. As shown, if the mobile moves from the network 15.5.0.0 to 15.6.0.0. The optimum crossover point is (a), which is a core router (P router). Secondly, (b) and (c) are a PE router and CE router, respectively. For the cases (b) and (c), the new path should be *derouted*, performed by PE attaching to and CE residing in the network 15.6.0.0, respectively, to meet the *designated crossover point* (DCP). At the first glimpse, placing the crossover point at (a) induces best routing efficiency. However, the case (a) or (b) raises other problems. The reservation needs the QCS-QNS negotiation, which will induce longer delay. The router in the provider network should be modified to recognize the micro flow, and in some case there will be multiple reservations in the network, which will waste the system resource.

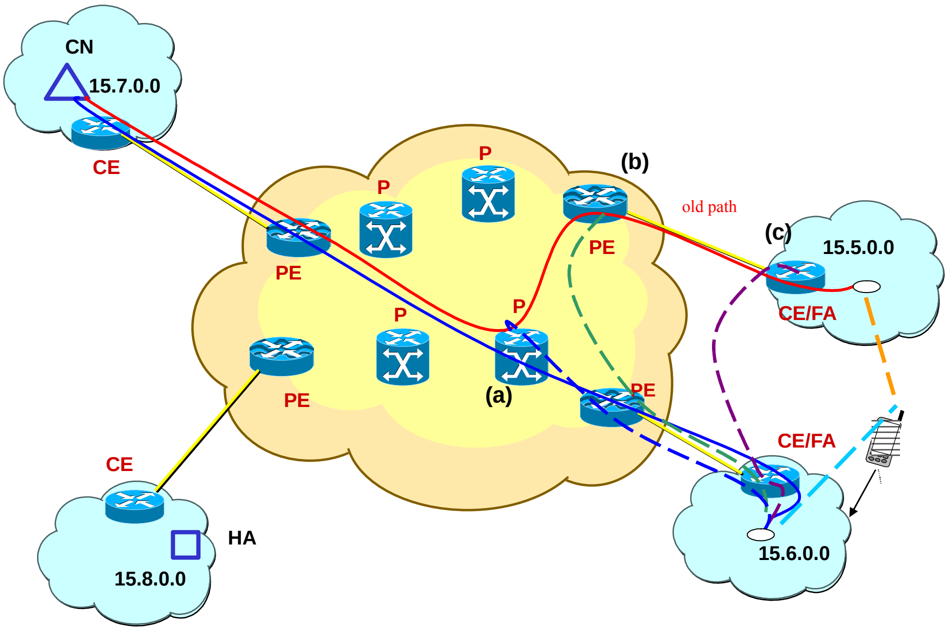

Figure 3.1. Three possible locations for placing the designated crossover point (DCP)

With the above reasons, we focus on placing the DCP in CE router, as in case (c). Furthermore, to overcome the resultant routing inefficiency, we let the DCP to handover to CE of mobile. It is important to stress that the “DCP handover” is not the *radio-level handover*. Rather, it is proposed to *re-route* the reservation path over the IP backbone to enhance the routing efficiency. When incorporating with Mobile IP, the address of CE router where the DCP is located is always the CoA registered and bound at the HA and CN, respectively. A DCP handover then includes the operation that the change of CoA due to the last vertical handover is reflected at both HA and CN.

3.2 cost minimization by VHE user profile

Virtual Home Environment (VHE) is a concept for Personal Service Environment (PSE) portability across network boundaries and between terminals. The concept of VHE is such that users are consistently presented with the same personalized features, User Interface customization and services in whatever network and whatever terminal, and wherever the user may be located.

A user's VHE is enabled by user profiles. A user may have a number of user profiles which enable her to manage communications according to different situations or needs, for example being at work, in the car or at home. Each user profile consists of two kinds of information – user interface related information and services related information [14] [15].

The user interface profile may include information such as: menu setting and terminal setting. The user services profile may include a list of favorite user services, as well as the personal configuration information for each of the above services and the service status whether the service is active or not. We can obtain the user profile from the network operator's database, e.g., HSS. In the personal configuration of the user service profile we assume that there will be the moving characteristics and vertical handover transition probability (figure 3.2), so that we can minimize network cost and find the optimal policy according to the information (see in chapter 5). Since all of the computation is done off-line, the real-time service will not be affected.

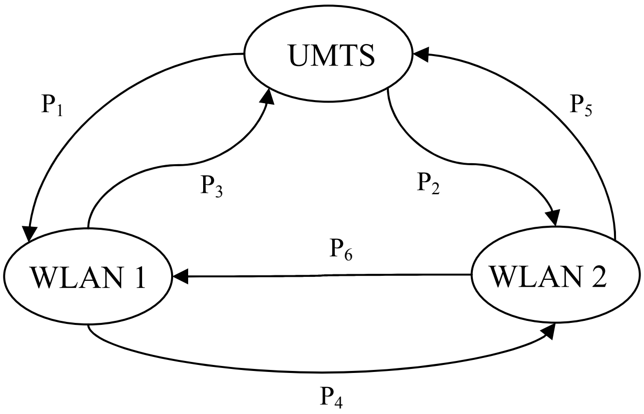

Figure 3.2. Vertical transition probabilities

We assume the mobile user moving form UMTS to WLAN 1 has the probability P_1, and moving from UMTS to WLAN 2 has the probability P_2. If the user is in the WLAN 1, we have the probability P_3 for moving to UMTS, and have the probability P_4 for moving to WLAN 2. If the user is in the WLAN 2, we have the probability P_5 for moving to UMTS, and have the probability P_6 for moving to WLAN 1. We show these vertical handover transition probabilities in the figure 3.2.

Chapter 4

Background Knowledge of Markov Decision Process

In this chapter we will introduce the basic concept of Markov Decision Process [16] with rewards and give formal definition to reward, alternative, expected cost value, and policy. We also introduce the value iteration method, which will lead us to find the optimal policy for the system.

4.1 The solution of the sequential decision process by value iteration

Suppose that an N-state Markov process earns r_{ij} dollars when it makes a transition form state i to state j. We call r_{ij} the "reward" associated with the transition from i to j. The set of rewards for the process may be described by a reward matrix R with elements r_{ij}. The rewards need not be in dollars, they could be voltage level, units of production, or any other physical quantity relevant to the problem.

The Markov process now generates a sequence of rewards as it makes transitions from state to state. The reward is thus a random variable with a probability distribution governed by the probabilistic relations of the Markov process. One question we might ask concerning is: What will be the player's expected winning in the next n transitions if the process is now in state i? To answer this question, let us define $v_i(n)$ as the expected total earnings in the next n transitions if the system is now in state i.

Some reflection on this definition allows us to write the recurrence relation

$$v_i(n) = \sum_{j=1}^{N} p_{ij}[r_{ij} + v_j(n-1)] \qquad i = 1, 2, \cdots, N \qquad n = 1, 2, 3, \cdots \tag{4.1}$$

Where the p_{ij} is the transition probability of state i to state j. If the system makes a transition from i to j, it will earn the amount r_{ij} plus the amount it expects to earn if it starts in state j with one move fewer remaining. As shown in Eq. 4.1, these earnings from a transition to j must be weighted by the probability of such a transition, p_{ij}, to obtain the total expected earnings.

Notice that Eq. 4.1 may be written in the form

$$v_i(n) = \sum_{j=1}^{N} p_{ij} r_{ij} + \sum_{j=1}^{N} p_{ij} v_j(n-1) \qquad i = 1, 2, \cdots, N \qquad n = 1, 2, 3, \cdots \tag{4.2}$$

so that if a quantity q_i is defined by

$$q_i = \sum_{j=1}^{N} p_{ij} r_{ij} \qquad i = 1, 2, \cdots, N \tag{4.3}$$

Eq. 4.1 takes the form

$$v_i(n) = q_i + \sum_{j=1}^{N} P_{ij} v_j(n-1) \qquad i = 1, 2, \cdots, N \qquad n = 1, 2, 3, \cdots \tag{4.4}$$

The quantity q_i may be interpreted as the reward to be expected in the next transition out of state i; it will be called the expected immediate reward for state i. Rewriting Eq. 4.1 as Eq. 4.4 shows us that it is not necessary to specify both a **P** matrix and an **R** matrix in order to determine the expected earnings of the system. All that is needed is a **P** matrix and a **q** column vector with N components q_i. The reduction in data storage is significant when large problems are to be solved on a digital computer. In vector form, Eq. 4.4 may be written as

$$\mathbf{v}(n) = \mathbf{q} + \mathbf{P}\mathbf{v}(n-1) \qquad n = 1, 2, 3, \cdots \tag{4.5}$$

where $v(n)$ is a column vector with N components $v_i(n)$, called the total-value vector.

Introduction of Alternatives

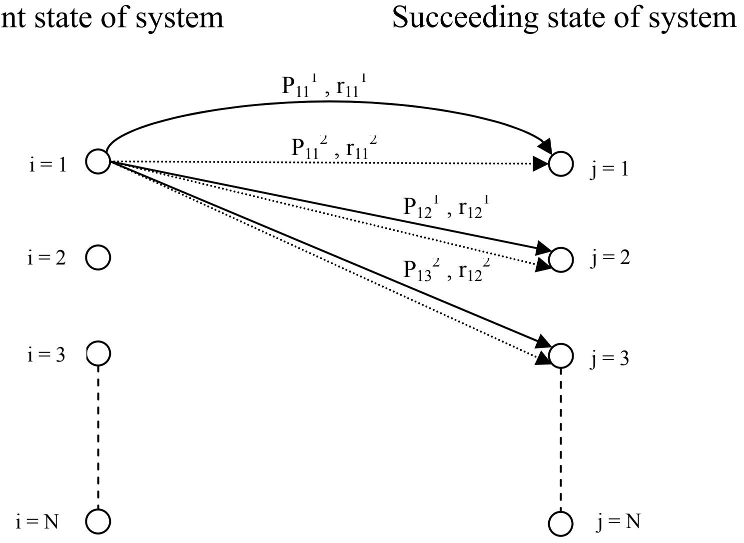

Figure 4.1. Diagram of states and alternatives

The concept of alternative for an *N*-state system is presented graphically in figure 4.1. In this diagram, two alternatives have been allowed in the first-state. If we pick alternative 1(k = 1), then the transition from state 1 to state 1 will be governed by the probability p_{11}^{1}, the transition from state 1 to state 2 will be governed by p_{12}^{1}, from 1 to 3 by p_{13}^{1}, and so on. The rewards associated with these transitions are $r_{11}^{1}, r_{12}^{1}, r_{13}^{1}$, and so on. If the second alternative in state 1 is chosen (k = 2), then $p_{11}^{2}, p_{12}^{2}, p_{13}^{2}, \cdots, p_{1N}^{2}$ and $r_{11}^{2}, r_{12}^{2}, r_{13}^{2}, \cdots, r_{1N}^{2}$, and so on, would be the pertinent probabilities and rewards, respectively. In Fig. 3.1 we see that, if alternative 1 in state 1 is selected, we make transitions according to the solid lines; if alternative is chosen, transitions are made according to the dashed lines. The number of alternatives in any state must be finite, but the number of alternatives in each state may be different from the numbers in other states.

We shall define $d_i(n)$ as the number of the alternative in the *i*th state that will be used at stage *n*. We call $d_i(n)$ the "decision" in state *i* at the *n*th stage. When $d_i(n)$ has been specified for all *i* and all *n*, a "policy" has been determined. The optimal policy is the one that maximizes total expected return for each *i* and *n*.

To analyze this problem, let us redefine $v_i(n)$ as the total expected return in n stages starting from state i *if an optimal policy is followed*. It follows that for any n

$$v_i(n+1) = \max_k \sum_{j=1}^{N} p_{ij}{}^k [r_{ij}{}^k + v_j(n)] \qquad n = 0, 1, 2, 3, \cdots \tag{4.6}$$

Suppose that we have decided which alternatives to follow at stage n, n - 1, … , 1 in such a way that we have maximized $v_j(n)$ for j = 1, 2, … , N. We are at stage n + 1 and are seeking the alternative we should follow in the ith state in order to make $v_i(n+1)$ as large as possible; this is $d_i(n+1)$. If we used alternative k in the ith state, then our expected return for n + 1 stages would be

$$\sum_{j=1}^{N} p_{ij}{}^k [r_{ij}{}^k + v_j(n)] \tag{4.7}$$

We are seeking the alternative in the ith state that will maximize Eq. 4.7. For this alternative, $v_i(n+1)$ will be equal to Eq. 4.7; thus we have derived Eq. 4.6, which we may call the value iteration equation. Equation 4.6 may be written in terms of the expected immediate rewards from each alternative in the form

$$v_i(n+1) = \max_k \left[q_i^k + \sum_{j=1}^{N} p_{ij}{}^k v_j(n) \right] \tag{4.8}$$

The use of the recursive relation Eq. 4.8 will tell us which alternative to use in each state at each stage and will also provide expected future earnings at each stage of the process. To apply this relation, we have to specify $v_j(0)$ the boundary condition for the process, therefore, we shall assign the value 0 to $v_j(0)$.

Evaluation of the Value-Iteration Approach

The method that has just been described for the solution of the sequential process may be called the value-iteration method because the $v_i(n)$ or "value" are determined iteratively. This method has some important limitations. For the most part, systems operate on an indefinite basis with no clearly defined end point. It does not seem efficient to have to iterate $v_i(n)$ for n = 1, 2, 3, and so forth, until we have a sufficiently large n that termination is very remote. We would much rather have a method that directed itself to the problem of analyzing processes of indefinite duration, processes that will make many transitions before termination.

Recall that, even if we were patient enough to solve the ling-duration process by value iteration, the convergence in the best alternative in each state is asymptotic and difficult to measure analytically. Even though the value-iteration method is not particularly suited to long-duration processes, it is relevant to those systems that face termination in a relatively short time.

Chapter 5

Numerical Analysis and Simulation Results

In this chapter, we will formulate our problem to Markov Decision Process, and find the best policy by using the value iteration method mentioned in the chapter 4. We assume that we can get the VHE user profile form operator's database in advance, then finding the optimal policy could be done off-line. The speed of computing will not be an issue.

We formulate the expected cost over any time interval for delaying the DCP handover. First, we assume that there are pre-allocated pipes engineered between UMTS and WLAN or between different WLAN.

The followings show the related parameters for delaying the DCP handover by apply *dwell timer*:

- τ: Duration to smooth the DCP handover.
- μ: Initial dwell timer value.
- L_1: Connection over pipe between CE of CN and CE of UMTS or WLAN.
- L_2: Connection over the pre-allocated pipe between GGSN of UMTS and CE of WLAN.
- L_3: Connection over the pre-allocated pipe between different CE of WLAN.

5.1 One UMTS and One WLAN

If we consider only one UNTS and one WLAN, we can depict the state chart for apply dwell timer to DCP handover in the figure 5.1. The name of the state means the number of connections of certain types in the IP backbone being used for a mobile user's real-time session. For example, the state "L_1+L_2" stands for the session currently using one L_1 connections and one L_2 connection. The name of transition represents the event that triggers the transition. The initial state is in "L_1", which means the DCP will be placed in the CE of mobile and thus only the connection L_1 is used over the IP backbone. When a vertical handover happens, an extra connection L_2 is added to deroute the traffic. One of the actions in this transition is to set the dwell timer to be μ. If there is another vertical handover prior to the expiration of dwell timer, which means the mobile switches back its association to the wireless network where DCP is located, the connection for derouting the traffic is removed immediately. As soon as the dwell timer counts down to zero, the DCP handover is invoked. During the execution of DCP handover, pipes over IP backbone between CE of CN and CE of mobile are being set up. To tolerate the setup delay, which is smaller than τ, the original connections will not be removed until τ time units later to smooth the DCP handover. Notice that during the DCP handover, i.e., in the state "$2L_1+L_2$", the events of vertical handovers will be conceptually queued and then trigger the transitions as long as the operating state is "L_1" or "L_1+L_2".

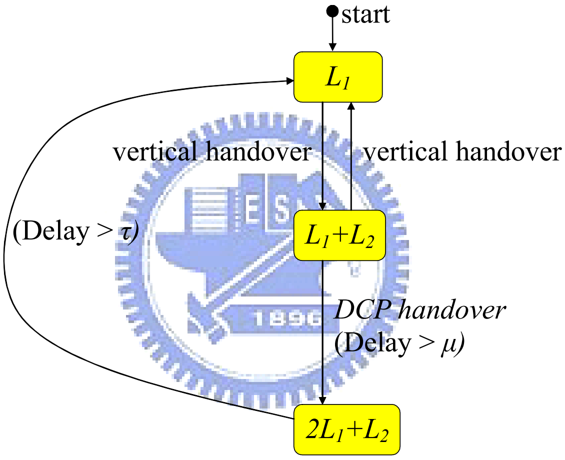

Figure 5.1. The state transition for DCP handover

5.1.1 Formulating Markov chain

By the state chart above, we can derive our Markov chain as figure 5.2. There are four states in our Markov chain, in each state we have four parameters: *mobile location/ DCP location/ pre-allocated pipe status/ state cost*. Taking state 1 for example, the mobile user is in 3G UMTS domain, the DCP is also located in CE of UMTS, and the pre-allocated pipe between GGSN of UMTS and CE of WLAN is off; so we can know that at this state it takes only $C(L_1)$ which means the cost of connection L_1 to setup and maintain the session.

If a vertical handover happens, which means the mobile move to WLAN form UMTS, the Markov process will enter state 2. At this state the mobile user is now in WLAN with DCP still located in 3G UNTS, the pre-allocated pipe will be turned on to support the connection between CN and mobile user; so the state cost should be $C(L_1) + C(L_2)$ at state 2. If the system stays at state 2 for time μ, the DCP handover will be taken and the process will enter state 3. As shown in the figure 5.2, the DCP is located in WLAN and the pre-allocated pipe is no more needed, this makes its pre-allocated pipe status is off; so the state cost is $C(L_1)$ again. We can see that when a DCP handover happens, there will be a transition cost $C(L_1) + C(L_2)$, because of the original connection L_1 and temporary pre-allocated pipe L_2.

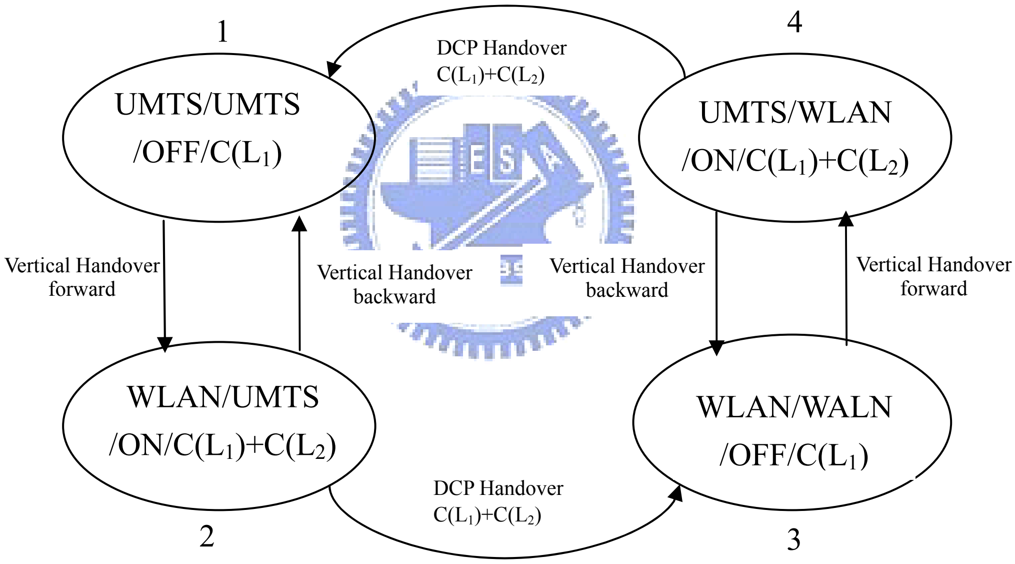

Figure 5.2. Markov chain for one UMTS and one WLAN

5.1.2 Finding minimal overhead

According to the simplified rule in the reference [17], we take the dwell time value either zero or infinity. This will reduce computation overhead when calculating the average overhead and it will also approximates the real case. Now if we take the dwell time value μ in UMTS and WLAN either zero or infinity, this will produce four different strategies as follows:

strategy	μ_UMTS	μ_WLAN
1	0	0
2	0	∞
3	∞	0
4	∞	∞

Table 5.1. Different strategies for μ in UNTS and WLAN

For strategy 1, because μ_UMTS and μ_WLAN are both equal to zero, the DCP handover will be taken immediately once the vertical handover is performed. The vertical handover backward will never occur, the Markov process can be modified to figure 5.3.

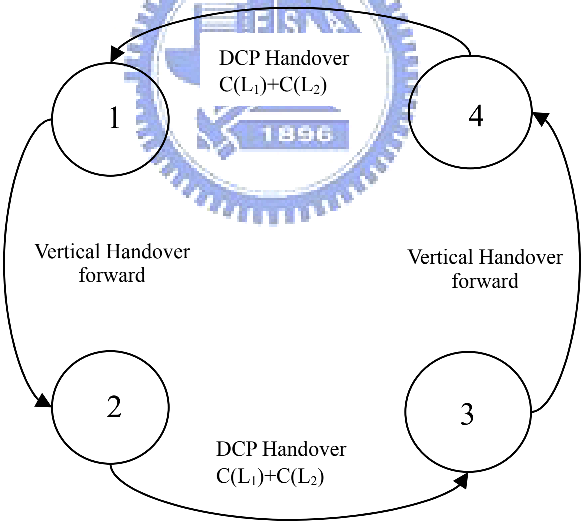

Figure 5.3. Markov chain for strategy 1

By using the renewal theory, the expected cost and the expected overhead for each state can be formulated as follows:

$$\mathrm{E}_1[\mathrm{cost}] = \frac{\lambda_1 C(L_1) + \lambda_3 C(L_1) + 2\tau[C(L_1) + C(L_2)]}{\lambda_1 + \lambda_3}$$

$$E_1[\text{overhead}] = \frac{2\tau[C(L_1)+C(L_2)]}{\lambda_1+\lambda_3}$$

The parameter λ_i presents the average life time staying at each state *i*. In this Markov chain state 2 and state 4 are both transient, so the expected cost for state 1 will not include the state cost of state 2 and 4, there will also two times of transition cost due to the DCP handover. The overhead means the additional efforts we should make except for the cost of original connection, it is equal to the value that subtracting $C(L_1)$ from $E_1[\text{cost}]$.

For strategy 2, because μ_UMTS is equal to zero and μ_WLAN is equal to infinity, when the mobile user moves from UMTS to WLAN, the DCP handover will never be performed; on the other hand, when the mobile user moves from WLAN to UMTS the DCP handover will be taken immediately after vertical handover occurred. The Markov process can be modified to figure 5.4. We can also find the expected cost and the expected overhead for state 2.

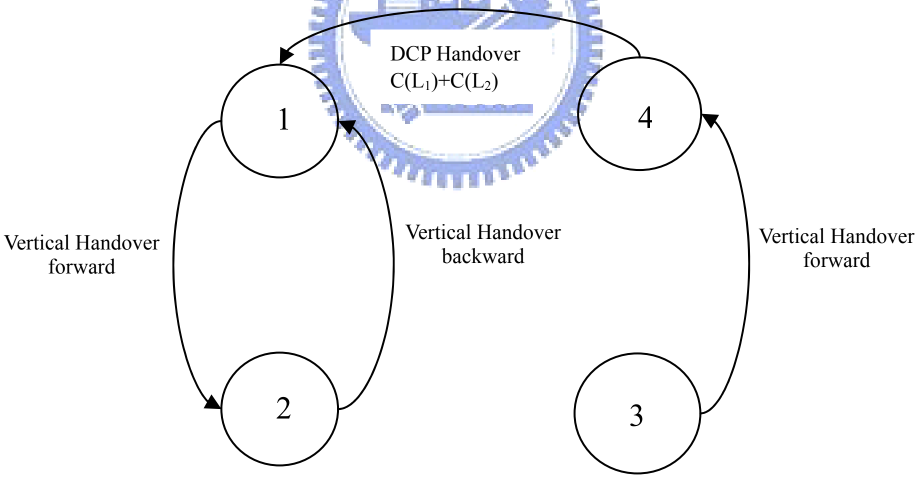

Figure 5.4. Markov chain for strategy 2

$$E_2[\text{cost}] = \frac{\lambda_1 C(L_1)+\lambda_2[C(L_1)+C(L_2)]+\tau[C(L_1)+C(L_2)]}{\lambda_1+\lambda_2}$$

$$E_2[\text{overhead}] = \frac{\lambda_2 C(L_2)+\tau[C(L_1)+C(L_2)]}{\lambda_1+\lambda_2}$$

For strategy 3, because μ_UMTS is equal to infinity and μ_WLAN is equal to zero, when the mobile user moves from UMTS to WLAN, the DCP handover will be taken immediately after vertical handover occurred; on the other hand, when the mobile user moves from WLAN to UMTS, the DCP handover will never be performed. The Markov process can be modified to figure 5.5. We can also find the expected cost and the expected overhead for state 3.

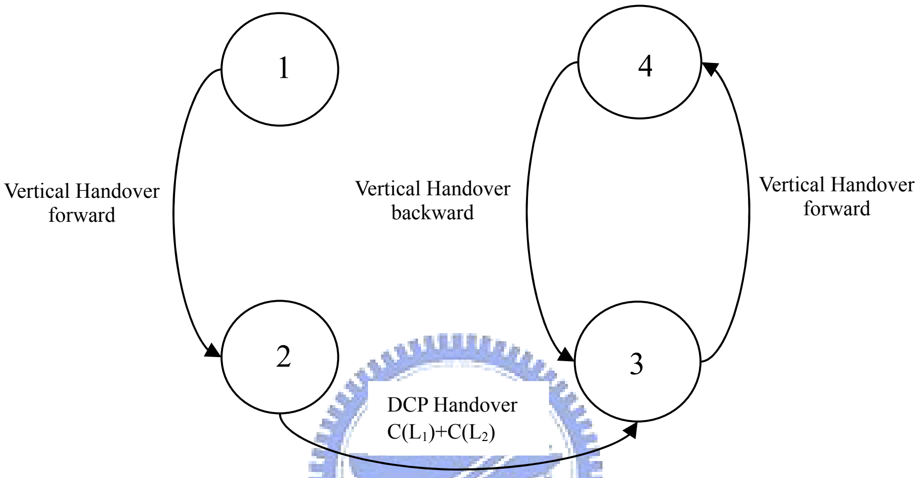

Figure 5.5. Markov chain for strategy 3

$$\mathrm{E}_3[\mathrm{cost}] = \frac{\lambda_3 C(L_1) + \lambda_4[C(L_1) + C(L_2)] + \tau[C(L_1) + C(L_2)]}{\lambda_3 + \lambda_4}$$

$$\mathrm{E}_3[\mathrm{overhead}] = \frac{\lambda_4 C(L_2) + \tau[C(L_1) + C(L_2)]}{\lambda_3 + \lambda_4}$$

For strategy 4, because μ_UMTS and μ_WLAN are both equal to infinity, the DCP handover will never be performed. The Markov process can be modified to figure 5.6. We can also find the expected cost and the expected overhead for state 4.

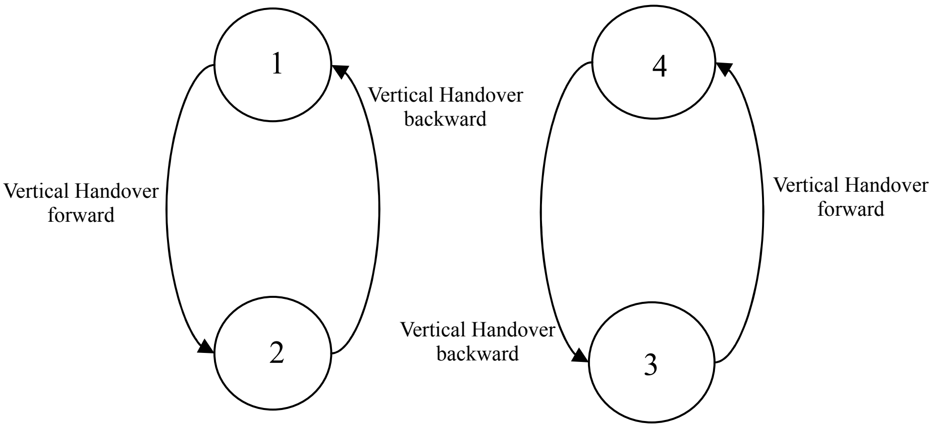

Figure 5.6. Markov chain for strategy 4

$$\mathrm{E}_4[\mathrm{cost}] = \begin{cases} \lambda_1 C(L_1) + \lambda_2 [C(L_1) + C(L_2)] \Big/ \lambda_1 + \lambda_2 , & \text{if starting at state 1 or state 2} \\ \lambda_3 C(L_1) + \lambda_4 [C(L_1) + C(L_2)] \Big/ \lambda_3 + \lambda_4 , & \text{if starting at state 3 or state 4} \end{cases}$$

$$\mathrm{E}_4[\mathrm{overhead}] = \begin{cases} \lambda_2 C(L_2) \Big/ \lambda_1 + \lambda_2 , & \text{if starting at state 1 or state 2} \\ \lambda_4 C(L_2) \Big/ \lambda_3 + \lambda_4 , & \text{if starting at state 3 or state 4} \end{cases}$$

In the beginning we assume that we can get the user profile form operator's database in advance, so for each user we can calculate the expected overhead of different strategies. When we get the four values of expected overhead, we select the strategy that makes E[overhead] minimal as our best policy. According to the policy, we can decide when a DCP handover should be taken will minimize the expected overhead.

5.2 One UMTS and two WLANs

In this section we will consider the interworking of one UMTS cell and two WLAN cells. Like the section 5.1, we can formulate our problem to a Markov chain. In this process we will have fifteen states and each state will have six parameters: *mobile location/ DCP location/ pipe1 status/ pipe2 status/ pipe3 status / state cost.*

We can find that these parameters are similar to which of the Markov chain in the figure 5.2, but now we have three pre-allocated pipes. Pipe 1 is the pre-allocated pipe between GGSN of UMTS and CE of WLAN 1, pipe 2 is the pre-allocated pipe between GGSN of UMTS and CE of WLAN 2, and pipe 3 is the pre-allocated pipe between CE of WLAN 1 and CE of WLAN 2. The mobile location, DCP location, and the state cost are the same as what we talked about before.

state	mobile location	DCP location	pipe 1 status	pipe 2 status	pipe 3 status	State cost
1	*UMTS*	*UMTS*	*OFF*	*OFF*	*OFF*	$C(L_1)$
2	W_1	*UMTS*	*ON*	*OFF*	*OFF*	$C(L_1) +C(L_2)$
3	W_2	*UMTS*	*ON*	*OFF*	*ON*	$C(L_1) +C(L_2)+ C(L_3)$
4	W_2	*UMTS*	*OFF*	*ON*	*OFF*	$C(L_1) +C(L_2)$
5	W_1	*UMTS*	*OFF*	*ON*	*ON*	$C(L_1) +C(L_2)+ C(L_3)$
6	W_1	W_1	*OFF*	*OFF*	*OFF*	$C(L_1)$
7	*UMTS*	W_1	*ON*	*OFF*	*OFF*	$C(L_1) +C(L_2)$
8	W_2	W_1	*ON*	*ON*	*OFF*	$C(L_1) +2C(L_2)$
9	W_2	W_1	*OFF*	*OFF*	*ON*	$C(L_1) +C(L_3)$
10	*UMTS*	W_1	*OFF*	*ON*	*ON*	$C(L_1) +C(L_2)+ C(L_3)$
11	W_2	W_2	*OFF*	*OFF*	*OFF*	$C(L_1)$
12	*UMTS*	W_2	*OFF*	*ON*	*OFF*	$C(L_1) +C(L_2)$
13	W_1	W_2	*ON*	*ON*	*OFF*	$C(L_1) +2C(L_2)$
14	W_1	W_2	*OFF*	*OFF*	*ON*	$C(L_1) +C(L_3)$
15	*UMTS*	W_2	*ON*	*OFF*	*ON*	$C(L_1) +C(L_2)+ C(L_3)$

Table 5.2. State parameters for one UMTS and two WLAN

In the table 5.2 we show the detail of each state. First, we assume that the diameter of WLAN is about 200 meters, it takes one user 5 minutes to go through the coverage of a WLAN, and assume the average time of a call is less than 10 minutes; so going through two WLANs takes 10 minutes which had been exceeded the average of a call. Second, we assume that the cost of connecting two WLANs by the scheme of local anchor is higher than the cost of re-establishing a new connection to where the mobile user located. Based on the two assumptions above, we will at most connect two WLANs by the local anchor scheme.

Taking state 3 for example, we know that the mobile user is now in the WLAN 2 with DCP located in the UMTS, and the pipe 1 and pipe 3 are both in use, so we can illustrate this state by the connection diagram figure 5.7_(3).

We can say that at the beginning the mobile user is at the UMTS domain, the user then moves to WLAN 1, and then also moves to WLAN 2 form WLAN 1. The vertical handover occurs twice here, moving form UMTS to WLAN 1 is the first time, and moving form WLAN 1 to WLAN 2 is the second time. So the pipe 1 and pipe 3 are ON, and the state cost is $C(L_1) + C(L_2) + C(L_3)$. All the other states can be explained in the same way, and we can plot the connection diagram for each state in the process.

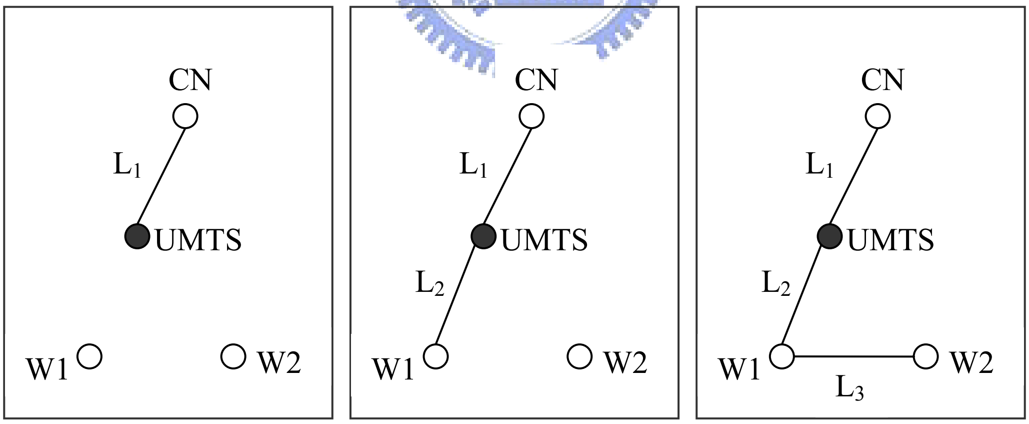

Figure 5.7_(1).(2).(3) Connection diagrams for state 1, 2, and 3

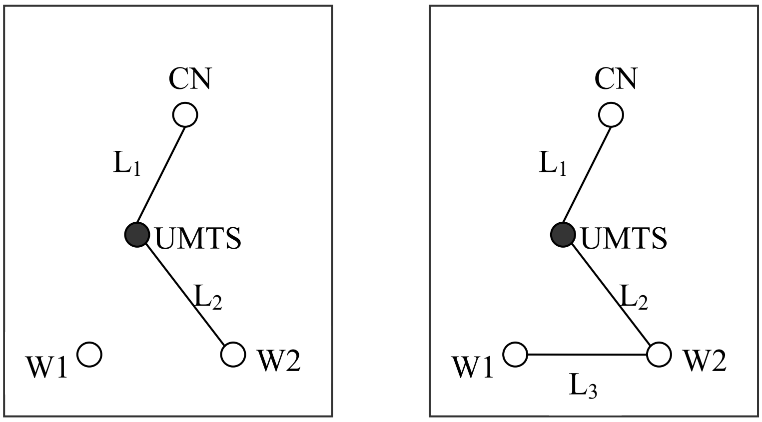

Figure 5.7_(4).(5) Connection diagrams for state 4 and 5

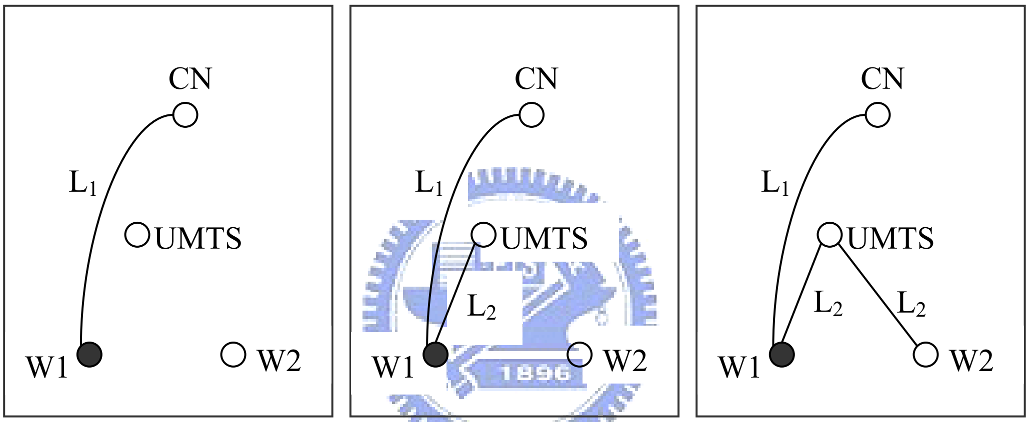

Figure 5.7_(6).(7).(8) Connection diagrams for state 6, 7, and 8

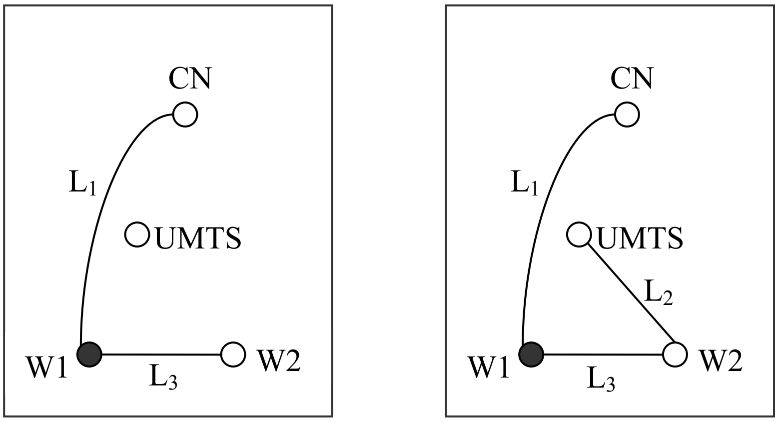

Figure 5.7_(9).(10) Connection diagrams for state 9 and 10

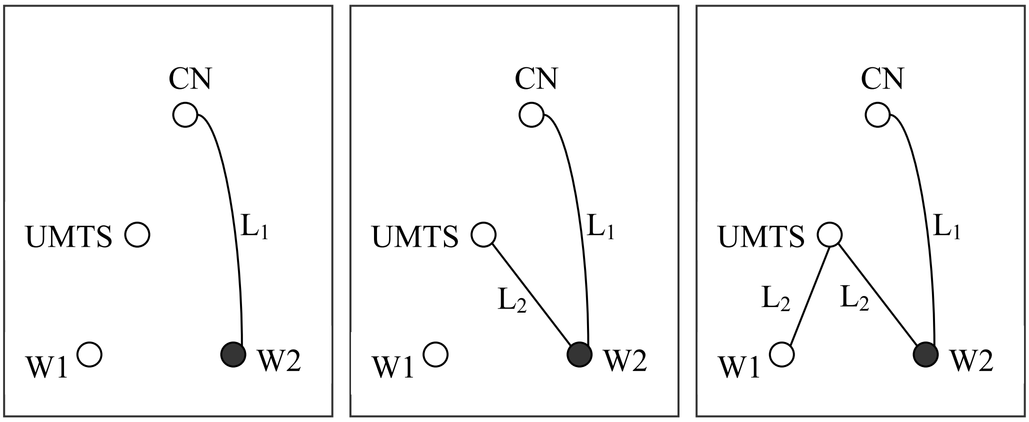

Figure 5.7_(11).(12)(13) Connection diagrams for state 11, 12, and 13

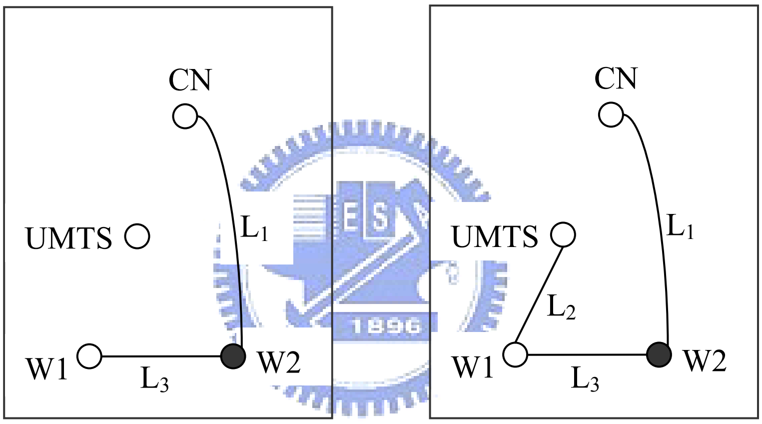

Figure 5.7_(14).(15) Connection diagrams for state 14 and 15

In the connection diagram we have four nodes: corresponding node, UMTS, WLAN 1, WLAN 2, if there is a real line between two nodes that means the session is using this connection. Taking state 3 for example again, the mobile user is using L_1 connection, L_2 connection and L_3 connection. The black node in the diagram show the location of DCP, so we can see that in the figure 5.7_(1) ~ figure 5.7_(5) the DCP is located in the UMTS, in the figure 5.7_(6) ~ figure 5.7_(10) the DCP is located in the WLAN 1, in the figure 5.7_(11) ~ figure 5.7_(15) the DCP is located in the WLAN 2.

5.2.1 Formulating Markov chain

Now we can have our Markov chain for interworking one UMTS and two WLANs as follows figure 5.8. There are fifteen states in the Markov process, we can find the meaning of each state by the connection diagram above. The real line represents the vertical handover, and the dotted line represents the DCP handover. We can find that the state 1, 6, 11 have only vertical handover transition, the vertical handover transition probability is given in the figure 3.2. The state 2, 4, 7, 9, 12, 14 have one more DCP handover transition in addition to vertical handover transition. The state 3, 5, 8, 10, 13, 15 have three DCP handover transition in addition to vertical handover transition.

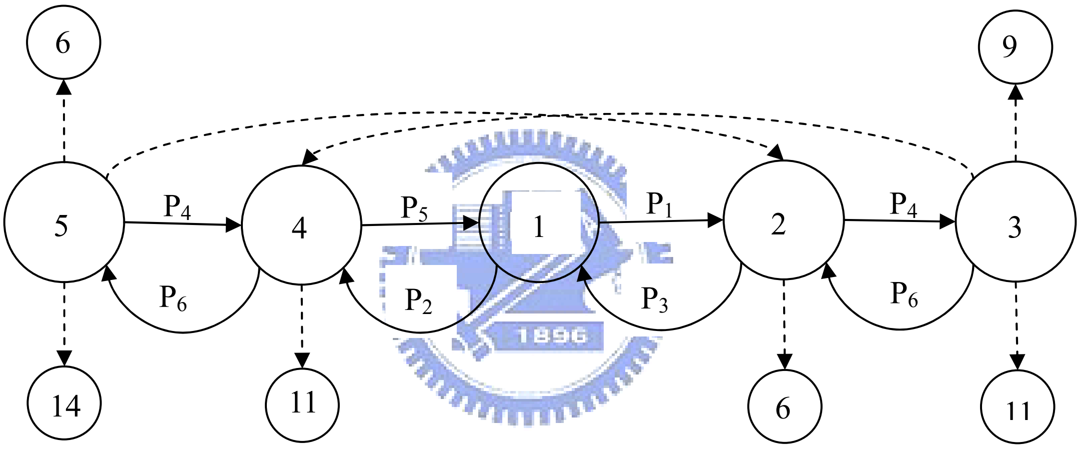

Figure 5.8_(1) state 1 ~ state 5 of Markov chain for one UMTS and two WLAN

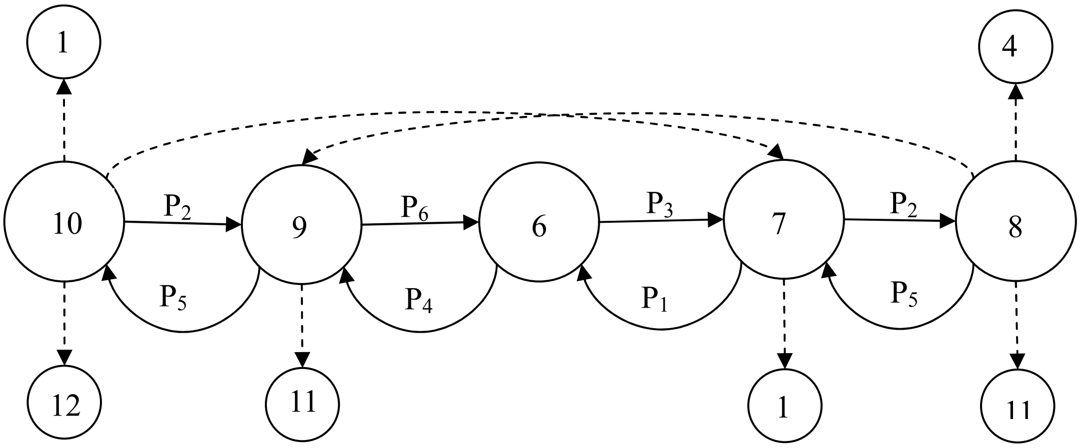

Figure 5.8_(2) state 6 ~ state 10 of Markov chain for one UMTS and two WLAN

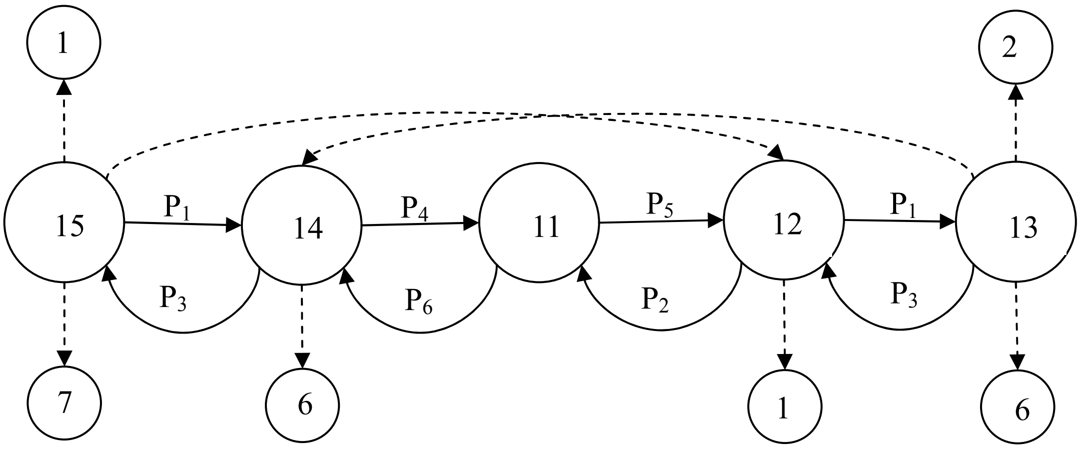

Figure 5.8_(3) state 11 ~ state 15 of Markov chain for one UMTS and two WLAN

Every transition of state can be seen as an action, we can only take one action as our decision at each time. Taking state 1 for example, it has two actions, one is taking the transition to alternative state 2 with probability P_2, the other is taking the transition to alternative state 3 with probability P_3. If taking state 3 for example, it has four alternative, they are transitions to state 2 with probability P_6, transition to state 4, state 9, and state 11 with probability 1 respectively. The first one action is vertical handover transition, and the others are DCP handover transition, if we decide to take DCP handover, it will have the probability one. By the state cost shown in the table 5.2, we can calculate the expected immediate reward, q_i^k, or say expected immediate cost for state *i* go to the alternative state *k*. For DCP handover there will be a time interval that the old connection should be kept until the direct connection is setup. Taking state 3 for example, if it takes the action to move to alternative state 9, the link between CN and UMTS and the link between UMTS and WLAN 1 should be kept for time τ to smooth the DCP handover, the expected immediate cost will be $[C(L_1) + C(L_2)]*\tau$, if it takes the action to move to alternative state 2, this means a vertical handover, all the connections should be kept for a time λ_i which represents the average life time in state *i*, therefore, the expected immediate cost will be $[C(L_1) + C(L_2) + C(L_3)]*\lambda$. Here we assume every state has the same average life time λ. In the table 5.3 we give the alternative, transition probability, and expected immediate reward.

state	alternative	transition probability															expected immediate reward
i	k	P_{i1}^{k}	P_{i2}^{k}	P_{i3}^{k}	P_{i4}^{k}	P_{i5}^{k}	P_{i6}^{k}	P_{i7}^{k}	P_{i8}^{k}	P_{i9}^{k}	P_{i10}^{k}	P_{i11}^{k}	P_{i12}^{k}	P_{i13}^{k}	P_{i14}^{k}	P_{i15}^{k}	q_i^k
1	2	$1-p_1$	p_1														$C(L_1)*\lambda$
	4	$1-p_2$			p_2												$C(L_1)*\lambda$
2	1	p_3	$1-p_3$														$[C(L_1)+C(L_2)]*\lambda$
	3		$1-p_4$	p_4													$[C(L_1)+C(L_2)]*\lambda$
	6						1										$[C(L_1)+C(L_2)]*\tau$
3	2		p_6	$1-p_6$													$[C(L_1)+C(L_2)+C(L_3)]*\lambda$
	4				1												$[C(L_2)+C(L_3)]*\tau$
	9									1							$[C(L_1)+C(L_2)]*\tau$
	11											1					$[C(L_1)+C(L_2)+C(L_3)]*\tau$
4	1	p_5			$1-p_5$												$[C(L_1)+C(L_2)]*\lambda$
	5				$1-p_6$	p_6											$[C(L_1)+C(L_2)]*\lambda$
	11											1					$[C(L_1)+C(L_2)]*\tau$
5	2		1														$[C(L_2)+C(L_3)]*\tau$
	4				p_4	$1-p_4$											$[C(L_1)+C(L_2)+C(L_3)]*\lambda$
	6						1										$[C(L_1)+C(L_2)+C(L_3)]*\tau$
	14														1		$[C(L_1)+C(L_2)]*\tau$

Table 5.3_(1) Transition probabilities for different alternatives for state 1 ~ state 5

state	alternative	transition probability															expected immediate reward
i	k	P_{i1}^{k}	P_{i2}^{k}	P_{i3}^{k}	P_{i4}^{k}	P_{i5}^{k}	P_{i6}^{k}	P_{i7}^{k}	P_{i8}^{k}	P_{i9}^{k}	P_{i10}^{k}	P_{i11}^{k}	P_{i12}^{k}	P_{i13}^{k}	P_{i14}^{k}	P_{i15}^{k}	q_i^k
6	7						$1-p_3$	p_3									$C(L_1)*\lambda$
	9						$1-p_4$			p_4							$C(L_1)*\lambda$
7	1	1															$[C(L_1)+C(L_2)]*\tau$
	6						p_1	$1-p_1$									$[C(L_1)+C(L_2)]*\lambda$
	8							$1-p_2$	p_2								$[C(L_1)+C(L_2)]*\lambda$
8	4				1												$[C(L_1)+C(L_2)]*\tau$
	7							p_5	$1-p_5$								$[C(L_1)+2C(L_2)]*\lambda$
	9									1							$2C(L_2)*\tau$
	11											1					$[C(L_1)+2C(L_2)]*\tau$
9	6						p_6			$1-p_6$							$[C(L_1)+C(L_3)]*\lambda$
	10									$1-p_5$	p_5						$[C(L_1)+C(L_3)]*\lambda$
	11											1					$[C(L_1)+C(L_3)]*\tau$
10	1	1															$[C(L_1)+C(L_2)+C(L_3)]*\tau$
	7							1									$[C(L_2)+C(L_3)]*\tau$
	9									p_2	$1-p_2$						$[C(L_1)+C(L_2)+C(L_3)]*\lambda$
	12												1				$[C(L_1)+C(L_3)]*\tau$

Table 5.3_(2) Transition probabilities for different alternatives for state 6 ~ state 10

state	alternative	transition probability															expected immediate reward
i	k	P_{i1}^k	P_{i2}^k	P_{i3}^k	P_{i4}^k	P_{i5}^k	P_{i6}^k	P_{i7}^k	P_{i8}^k	P_{i9}^k	P_{i10}^k	P_{i11}^k	P_{i12}^k	P_{i13}^k	P_{i14}^k	P_{i15}^k	q_i^k
11	12											$1-p_5$	p_5				$C(L_1)*\lambda$
	14											$1-p_6$			p_6		$C(L_1)*\lambda$
12	1	1															$[C(L_1)+C(L_2)]*\tau$
	11											p_2	$1-p_2$				$[C(L_1)+C(L_2)]*\lambda$
	13												$1-p_1$	p_1			$[C(L_1)+C(L_2)]*\lambda$
13	2		1														$[C(L_1)+C(L_2)]*\tau$
	6						1										$[C(L_1)+2C(L_2)]*\tau$
	12												p_3	$1-p_3$			$[C(L_1)+2C(L_2)]*\lambda$
	14														1		$2C(L_2)*\tau$
14	6						1										$[C(L_1)+C(L_3)]*\tau$
	11											p_4			$1-p_4$		$[C(L_1)+C(L_3)]*\lambda$
	15														$1-p_3$	p_3	$[C(L_1)+C(L_3)]*\lambda$
15	1	1															$[C(L_1)+C(L_2)+C(L_3)]*\tau$
	7							1									$[C(L_1)+C(L_3)]*\tau$
	12												1				$[C(L_2)+C(L_3)]*\tau$
	14														p_1	$1-p_1$	$[C(L_1)+C(L_2)+C(L_3)]*\lambda$

Table 5.3_(3) Transition probabilities for different alternatives for state 11 ~ state 15

By the value iteration method described in the section 4.1, we can find the best decision for each state. There is one noteworthy point related to our value iteration. In the Eq. 4.8 it takes maximum over the value

$$q_i^k + \sum_{j=1}^{N} p_{ij}{}^k v_j(n)$$

this is because we want to make our reward the bigger the better. But here the value represents the total expected network cost, so we should take minimum instead of taking maximum. In the simulation we assume average life time λ is equal to 30 seconds and the time τ is equal to 2 seconds for smoothing the DCP handover. We also assume that we can get the transition probability from user profile and the connection cost for each link is reserved for mobile operator to define. Therefore, we can calculate the expected total cost $v_i(n)$ for each state.

Set	$C(L_1)$	$C(L_2)$	$C(L_3)$
A	1	1	1
B	3	2	1
C	4	3	8
D	2	7	1

Table 5.4. Different sets of connection cost

5.2.2 Finding optimal policy by value iteration method

In order to simulate our problem, we set the transition probability arbitrarily, $P_1 = 0.5$, $P_2 = 0.5$, $P_3 = 0.4$, $P_4 = 0.6$, $P_5 = 0.1$, $P_6 = 0.9$, and define different sets of connection cost for each link in the table 5.4. In the real case, we can get these probabilities form user profile and the cost of link will be defined by mobile operator. The expected total cost diagram is shown in figure 5.9; for different sets we can find the best policy in the fourth stage through decision matrix shown in figure 5.10.

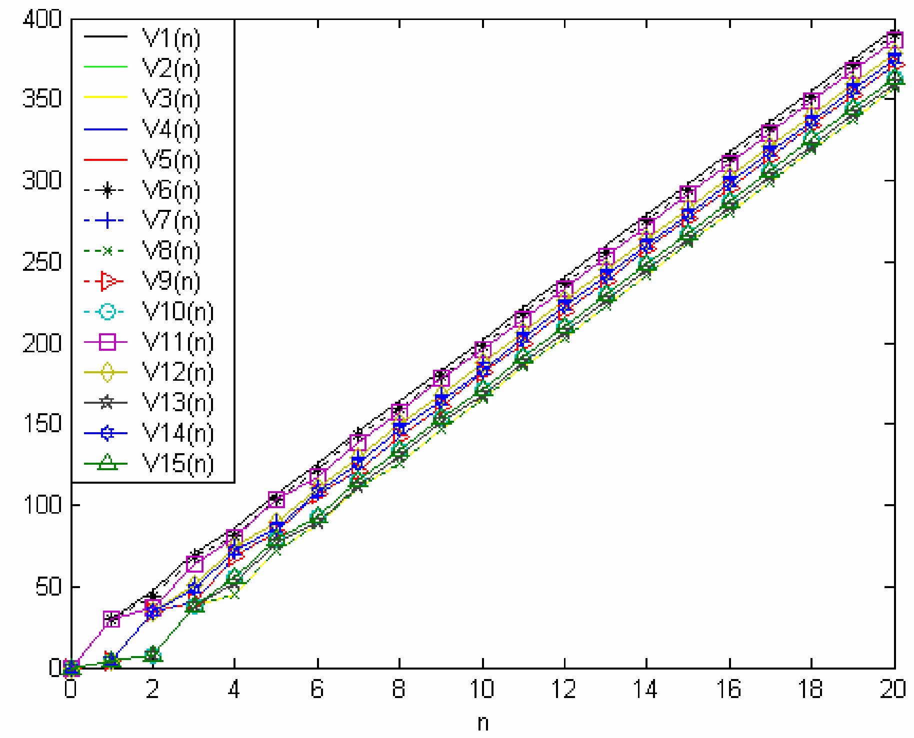

Figure 5.9_(a) Expected total cost for set A

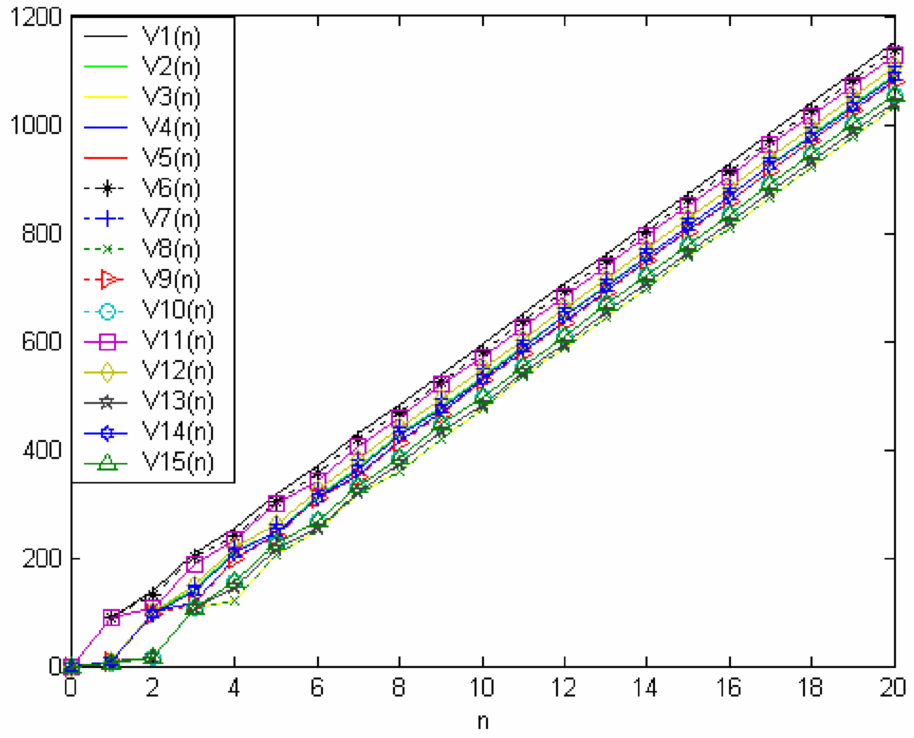

Figure 5.9_(b) Expected total cost for set B

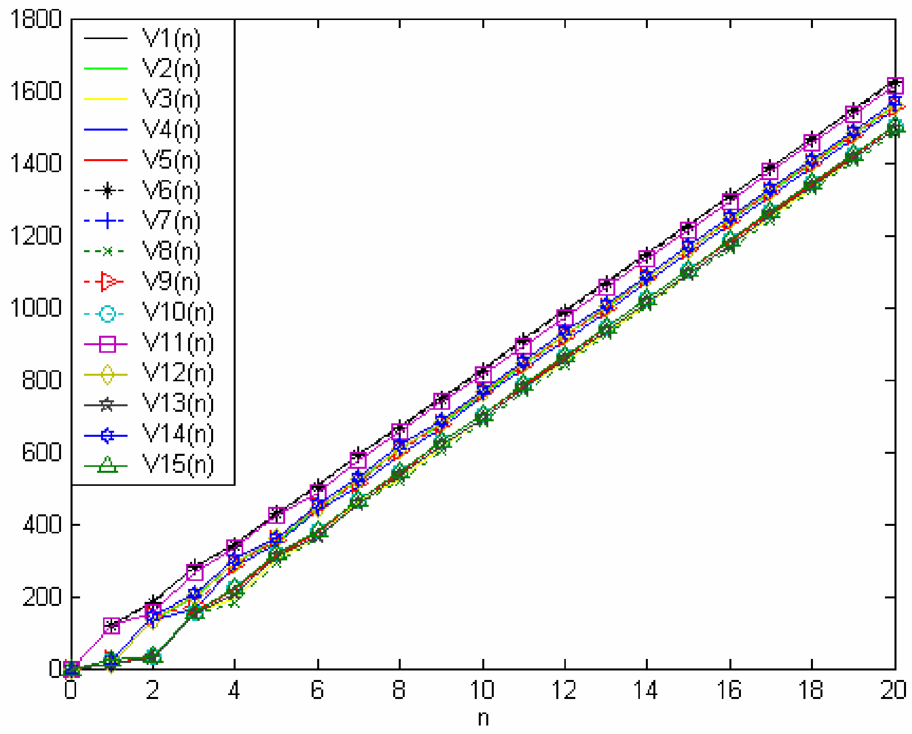

Figure 5.9_(c) Expected total cost for set C

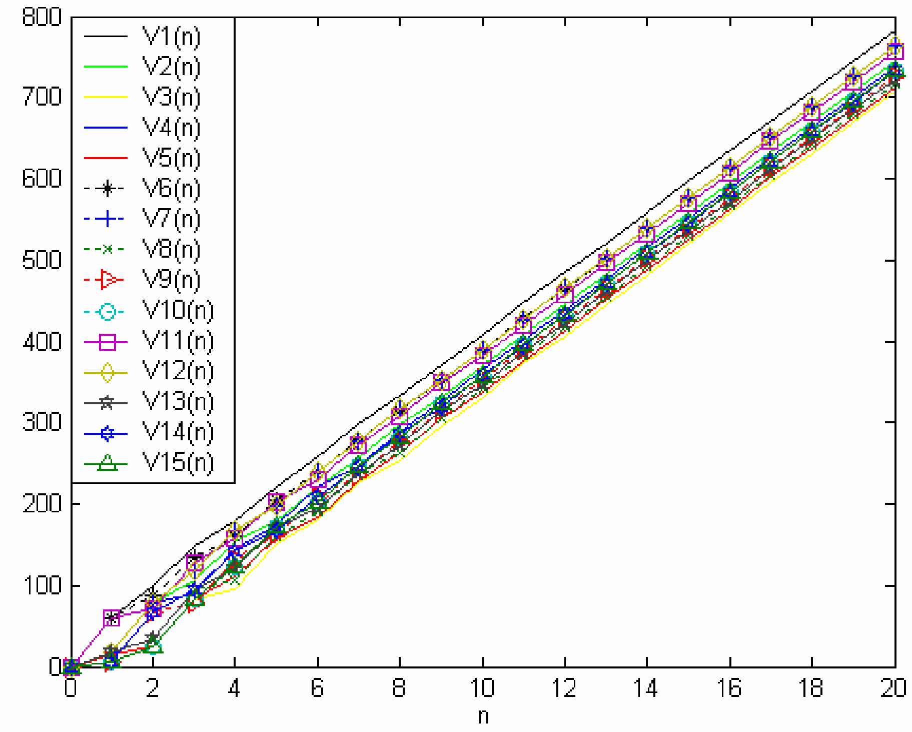

Figure 5.9_(d) Expected total cost for set D

$$\begin{array}{cc} n = & \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & \cdots \end{array} \\ & \begin{bmatrix} 2 & 2 & 2 & 4 & 4 & \cdots \\ 6 & 6 & 6 & 6 & 6 & \cdots \\ 4 & 4 & 4 & 4 & 4 & \cdots \\ 11 & 11 & 11 & 11 & 11 & \cdots \\ 2 & 2 & 2 & 2 & 2 & \cdots \\ 7 & 9 & 9 & 9 & 9 & \cdots \\ 1 & 1 & 1 & 1 & 1 & \cdots \\ 4 & 4 & 4 & 4 & 4 & \cdots \\ 11 & 11 & 11 & 11 & 11 & \cdots \\ 7 & 7 & 7 & 7 & 7 & \cdots \\ 12 & 14 & 14 & 14 & 14 & \cdots \\ 1 & 1 & 1 & 1 & 1 & \cdots \\ 2 & 2 & 2 & 2 & 2 & \cdots \\ 6 & 6 & 6 & 6 & 6 & \cdots \\ 7 & 7 & 7 & 7 & 7 & \cdots \end{bmatrix} \end{array}$$

(a) $C(L_1) =1; C(L_2)=1; C(L_3)=1$

$$\begin{array}{cc} n = & \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & \cdots \end{array} \\ & \begin{bmatrix} 2 & 2 & 2 & 4 & 4 & \cdots \\ 6 & 6 & 6 & 6 & 6 & \cdots \\ 4 & 4 & 4 & 4 & 4 & \cdots \\ 11 & 11 & 11 & 11 & 11 & \cdots \\ 2 & 2 & 2 & 2 & 2 & \cdots \\ 7 & 9 & 9 & 9 & 9 & \cdots \\ 1 & 1 & 1 & 1 & 1 & \cdots \\ 9 & 9 & 9 & 9 & 9 & \cdots \\ 11 & 11 & 11 & 11 & 11 & \cdots \\ 7 & 7 & 7 & 7 & 7 & \cdots \\ 12 & 14 & 14 & 14 & 14 & \cdots \\ 1 & 1 & 1 & 1 & 1 & \cdots \\ 14 & 14 & 14 & 14 & 14 & \cdots \\ 6 & 6 & 6 & 6 & 6 & \cdots \\ 12 & 12 & 12 & 12 & 12 & \cdots \end{bmatrix} \end{array}$$

(b) $C(L_1) =3; C(L_2)=2; C(L_3)=1$

$$\begin{array}{cc} n = & \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & \cdots \end{array} \\ & \begin{bmatrix} 2 & 2 & 2 & 4 & 4 & \cdots \\ 6 & 6 & 6 & 6 & 6 & \cdots \\ 9 & 4 & 4 & 4 & 4 & \cdots \\ 11 & 11 & 11 & 11 & 11 & \cdots \\ 14 & 2 & 2 & 2 & 2 & \cdots \\ 7 & 9 & 9 & 9 & 9 & \cdots \\ 1 & 1 & 1 & 1 & 1 & \cdots \\ 9 & 4 & 4 & 4 & 4 & \cdots \\ 11 & 11 & 11 & 11 & 11 & \cdots \\ 7 & 7 & 7 & 7 & 7 & \cdots \\ 12 & 14 & 14 & 14 & 14 & \cdots \\ 1 & 1 & 1 & 1 & 1 & \cdots \\ 14 & 2 & 2 & 2 & 2 & \cdots \\ 6 & 6 & 6 & 6 & 6 & \cdots \\ 12 & 12 & 12 & 12 & 12 & \cdots \end{bmatrix} \end{array}$$

(c) $C(L_1) =4; C(L_2)=3; C(L_3)=8$

$$\begin{array}{cc} n = & \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & \cdots \end{array} \\ & \begin{bmatrix} 2 & 2 & 2 & 4 & 4 & \cdots \\ 6 & 6 & 6 & 6 & 6 & \cdots \\ 4 & 9 & 9 & 9 & 9 & \cdots \\ 11 & 11 & 11 & 11 & 11 & \cdots \\ 2 & 14 & 14 & 14 & 14 & \cdots \\ 7 & 9 & 9 & 9 & 9 & \cdots \\ 1 & 1 & 1 & 1 & 1 & \cdots \\ 4 & 9 & 9 & 9 & 9 & \cdots \\ 11 & 11 & 11 & 11 & 11 & \cdots \\ 12 & 12 & 12 & 12 & 12 & \cdots \\ 12 & 14 & 14 & 14 & 14 & \cdots \\ 1 & 1 & 1 & 1 & 1 & \cdots \\ 2 & 14 & 14 & 14 & 14 & \cdots \\ 6 & 6 & 6 & 6 & 6 & \cdots \\ 7 & 7 & 7 & 7 & 7 & \cdots \end{bmatrix} \end{array}$$

(d) $C(L_1) =2; C(L_2)=7; C(L_3)=$

Figure 5.10. Decision matrix

Compare the decision matrix (a) with (b), we can find that the optimal policy is different in state 8, state 13, and state 15. When $C(L_1)$ =3, $C(L_2)$=2, and $C(L_3)$=1, state 8 will take the decision to alternative state 9, since the expected immediate cost q_8^9 which is equal to $2C(L_2)*\tau = 16$ is smaller than q_8^4 which is equal to $[C(L_1) + C(L_2)]*\tau = 20$. State 13 will take the decision to alternative state 14 because of the same reason. State 15 will take the decision to alternative state 12, since the expected immediate cost q_{15}^{12} which is equal to $[C(L_2) + C(L_3)]*\tau = 12$ is smaller than q_{15}^7 which is equal to $[C(L_1) + C(L_3)]*\tau = 16$. There are two more sets (c) and (d), we can explain the decision for each state by the same way. For different connection link cost we will have different optimal policy, the operator can decide when to take a DCP handover depending on the various connection models, and minimize the network cost for seamless vertical handover.

If changing the average life time λ, it will affect the stage we need to converge to the best policy. It takes at most 10 stages to obtain convergence when the average life time $\lambda = 1$. Since the smoothing time τ is equal to one, the expected immediate cost for each alternative state is much closer, and it will take more stages to reach convergence. For most case it only takes four stages to converge, the computation is not an issue. We show this in the figure 5.11.

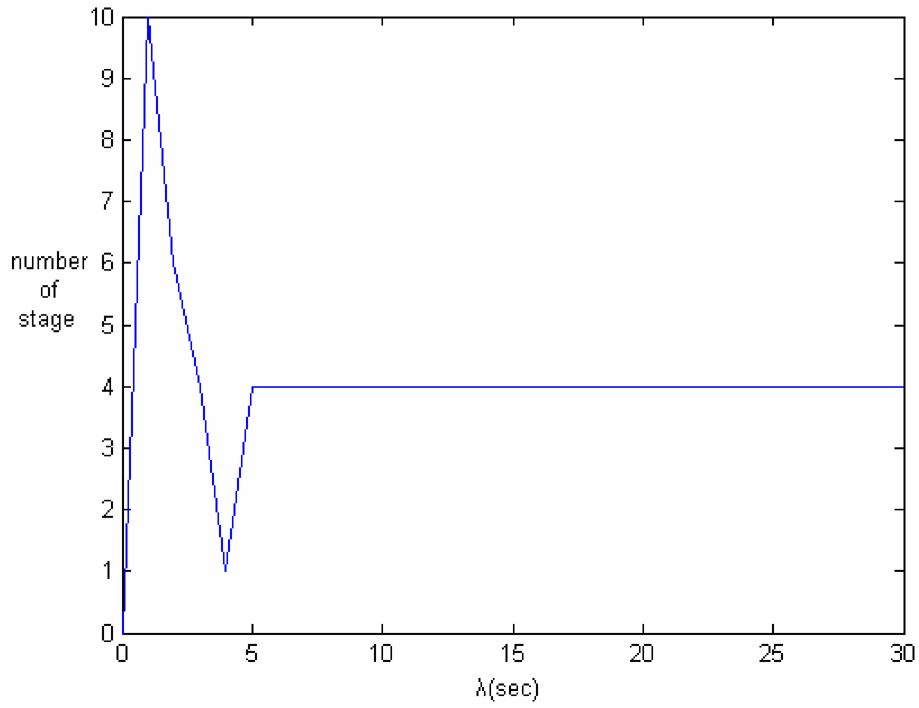

Figure 5.11. Numbers of stage over λ

Chapter 6

Conclusion

In this work we discuss the problem of minimizing the network cost of seamless handover for real-time service from the network architectural aspect. It identifies that the QCS-QNS negotiation is expected to be the major factor contributing to the latency of vertical handover. To overcome this bottleneck, we introduce the concept of Designated Crossover Point (DCP) to reduce the latency. By loosely-coupled interworking, VPN connection model, and Mobile IP, we can integrate UMTS and WLAN together and pre-allocate pipes between them to supporting DCP handover which will achieve lower cost for network operator. Our system architecture bases on the existing 3G UMTS and WLAN infrastructures, therefore, it is a viable scheme.

In our simulation, we can find the dwell timer μ and according to the timer value to perform the DCP handover. Here, we use simplified rule [17], therefore, the timer is either zero or infinity. If considering one UMTS and one WLAN, we just choose one strategy from table 5.1 as our optimal policy. If considering one UMTS and two WLAN, by using the value iteration method of Markov decision process and the VHE user profiles defined by 3GPP, in which we can find the mobility characteristic, we can calculate the network cost and derive the optimal policy that will minimize total expected network cost. That means, the network operator can select the best connection model to supporting the seamless vertical handover for real-time service by spending the least network resource, on the other hand, the user will continue his/her ongoing sessions without interrupting. Both customers and operators will benefit from our proposed scheme.

Reference

[1] *Fodor, G., Eriksson, A., and Tuoriniemi, A,* "Providing quality of service in always best connected networks" Communications Magazine, IEEE , Volume: 41 , Issue: 7 , July 2003

[2] *Chiussi, F.M., Khotimsky, D.A., and Krishnan, S.*, "Mobility management in third-generation all-IP networks" Communications Magazine, IEEE , Volume: 40 , Issue: 9 , Sep 2002

[3] ETSI TR 101 957 v1.1.1 (2001-08) "Broadband Radio Access Networks (BRAN); HIPERLAN TYPE 2; Requirements and Architectures for Interworking between HIPERLAN/2 and 3rd Generation Cellular Systems."

[4] *Jaseemuddin, M* , "An architecture for integrating UMTS and 802.11 WLAN networks" Computers and Communication, 2003. (ISCC 2003). Proceedings. Eighth IEEE International Symposium on , 30 June-3 July 2003

[5] *P M Feder, N Y Lee, and S Martin-Leon*, "A SEAMLESS MOBILE VPN DATA SOLUTION FOR UMTS AND WLAN USERS"Bell Laboratories - Mobility Solutions, Lucent Technologies Inc., USA, UK

[6] *Ivan Pepelnjak and Jim Guichard*, "MPLS and VPN architectures – a practical guide to understanding, designing and deploying MPLS and MPLS-enabled VPNs"

[7] *David Wisely, Philip Eardley, and Louise Burness*, "IP for 3G – networking technologies for mobile communications"

[8] *V. Fineberg*, "A Practical Architecture for Implementing End-to-End QoS in an IP Network", IEEE Communications Magazine, January 2002

[9] *W.T. Chen and L.C. Huang*, "RSVP Mobility Support: A Signaling Protocol for Integrated Services Internet with Mobile Hosts," Infocom 2000.

[10] *A. K. Talukdar, B.R. Badrinath, and A. Achaya*, “MRSVP: A Reservation Protocol for an Integrated Services Packet Network with Mobile Hosts,” Dept. Comp. Sci. tech rep. TR-337, Rutgers Univ.

[11] *A. De Carolis et al*, “QoS-Aware Handover: Secondary Home Agent”, IETF draft-decarolis-qoshandover-02.txt, April 2000.

[12] *S. Paskalis, A. Kaloxylos, E. Zernas, and L. Merakos*, “RSVP Mobility Proxy,” Internet draft, Dec. 2001.

[13] *Q. Shen et al.*, “On Providing Flow Transparency Mobility Support for IPv6-based Real-Time Services, “ Proc. MoMuC 2000, Tokyo, Japan, Oct. 2000.

[14] *Bougant, F.; Delmond, F.; Pageot-Millet, C.*, “The user profile for the virtual home environment” Communications Magazine, IEEE , Volume: 41 , Issue: 1 , Jan. 2003

[15] 3GPP TS 22.121 V5.3.0 (2002-03), “Technical Specification Group Services and System Aspects; Service Aspects; The Virtual Home Environment,” rel. 5.

[16] *Ronald A. Howard*, “Dynamic Programming and Markov Process”

[17] *Wei-Kuo Liao and Yi-Chung Chen*, “Supporting Vertical Handover between UMTS and WLAN for Real-Time Services” 3rd International Conference on Networking ICN'04