Software Agent Framework for Dynamic Handoff Decision

Cheng-shian Juo
Department of Communications Engineering, National Chung Cheng University
Chia-Yi, Taiwan, Republic of China
Email: kusojuo@gmail.com

Jen-yi Pan
Department of Communications Engineering, National Chung Cheng University
Chia-Yi, Taiwan, Republic of China
E-mail: jypan@comm.ccu.edu.tw

***Abstract*— This paper proposes a framework which supports dynamic handoff decision strategy (DHDS) and software agent (SA) for highly overlapping wireless networks. SA can dynamically download SA from local mobility strategy server to mobile node, and hence adapt the handoff decision to local environment and situations. In our experiment, a comprehensive comparison has been conducted to show that the proposed framework outstands from traditional unchangeable strategies in different situations and criteria based on our DHDS notion.**

Keywords: Dynamic Handoff Strategy, Application Oriented Handoff, Software Agent.

I. INTRODUCTION

The popularity of wireless network is because of its convenience. In our life, users can roam through varieties of public spaces, restaurants, cafe, mass rapid transmit and so on. The overlapping wireless networks provide service anywhere. The coexistence of heterogeneous wireless networks (HWN) aims to construct complete services for a large number of users. It is important to design a dynamic handoff decision strategy (DHDS) across different networks concerned with numerous factors on handoff decision

Handoff processes consist of three phases: handoff decision, radio link transfer and channel assignment [1]. In this paper, we focus on handoff decision and consider which the best is in candidate networks.

Recently, many handoff decision strategies have been proposed, including a lot of factors we noticed and considered in. First, using the received signal strength (RSS) is well known measurement [2] [3]. Many strategies improve quality of service for roaming users. Heavy traffic may induce long delay and more packets loss. In light of network, reference [4] designs a load-balancing scheme for wireless LANs. Reference [2] mainly focused on small delay and few lost packets using signal/noise ratio system load and association time. Besides, economic factors may be taken into account, e.g. cost and battery.

But now and future, based on integrated networks or heterogeneous networks, vertical handoff decision strategies like [5] [6] [7] [8] [9], reference [7] takes account of service type, monetary cost, network conditions, system performance, mobile terminal conditions and user preferences to make a strategic decision. Moreover, integrated decision criteria including [8] [9], reference [8] proposed a smart model, using the weight functions. Reference [9] presented a vertical handoff decision algorithm that enables a wireless access network not only to balance the overall load among all network but also to maximize the collective battery life time of Mobile Nodes. On the other way, it focuses on developing the algorithm based on suitable optimization criteria. However, the users' application and the network need a more intelligent and adaptive approach on handoff strategy and decision.

Contrary to the others methods, reference [10] proposed active application oriented (AAO) scheme that considers application requirements. It reduced the power consumption caused by unnecessary interface activation. Reference [11] provides a multi-application scheme that considers requirements of various applications and performs handoff process at the right time. In [7], the authors present a tutorial on the design and performance issues for vertical hand-off in fourth-generation mobile environment.

Providing perfect services over heterogeneous wireless networks is expected to be a crucial problem. Heterogeneous wireless networks pose challenges in the handoff design. Next generation network integrates wireless LAN, WiMAX, 3G and even other new access technology. Meanwhile, user's applications grow day by day with diverse network requirements, e.g. voice communication demanding constant bit rate and bounded delay, data transmission demanding huge bandwidth, and web service demanding reliability. Thus, it's hard to use only one general strategy or predetermined attributes to decide where to hand off without modification.

For example, in terms of the capacity of the network, the network traffic or network load was considered in the first. But in the viewpoint of users, the best service is more important than others. Higher throughput or stable communication is what the users want. Various factors have to be considered in. We believed that network architecture may change and the diversity of network service grows day by day. In commercial, we consider the cost of the network service. The major consideration is monetary cost. In academic, quality and fairness should be taken into account. However, it is not complete enough to design a perfect algorithm faced with the future network environment.

Furthermore, it's hard to be complete in both respects. For example, we want power-save and bandwidth. But the relation between them is inversely proportional. The larger bandwidth of application will consume more power at higher data rate. We can say that bandwidth and power-saving are complementary factors. On the other hand, the larger the number of MN attached to the same AP, the more the power is consumed by each MN because the MN gets lower rate and hence needs to connect to the associated AP longer. Contrarily, some factors are positive correlated. The number of users affect throughput, packet delay and packet loss. The distance correlates closely with signal strength. However, we conclude from these observations. Table I shows the factors on handoff decision.

Table I. Factors of handoff decision.

	Factors	Details of factor
Network provider	Bandwidth	Available bandwidth of a network
	Coverage	Coverage of a network
	Related service	Academic/ enterprise/ commercial
	Connections	Support numbers of user
	Mobility	Support mobility management or not
	Network structure	WLAN/3G/3.5G/WiMAX
User	Application	The application running on mobile node(MN)
	Power	Power consumption of MN
	Price	Offers service's fee to ISP or network provider
	RSS	Received Signal Strength from base stations
	Bandwidth	The minimum bandwidth that user needed
	Packet loss	Packet loss of the connecting network

Therefore, this paper provided a DHDS framework to adapt the involved problem. This framework is suited for any purpose. The remaining part of this paper consists of four parts. Section 2 will introduce the framework. In section 3, the DHDS is explained. The experiments are presented in section 4 and section 5 evaluates the proposed framework. Finally, we conclude the paper in section 6.

II. Network Environment

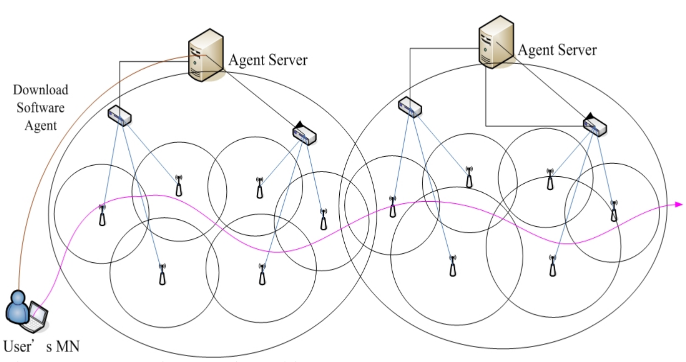

Figure 1. Proposed Network Architecture

We propose an efficient scheme to solve the pre-mentioned situation by designing software agent. In Figure 1 .The Agent Server is introduced to provide the information, such as network condition and network topology. Every Agent Server covers a wide range area. It manages nearby base stations (BS) and access points in its coverage. Therefore, Agent Server collects every APs or BSs information and constructs topology map, such that Agent Server can build integral environment information.

We focused on the Software Agent's (SA) work and the DHDS. In [7], the authors use the agent-like software in their MN. Based on different purpose and requirement, we design numerous decision-cores, which are stored in Agent Server. MN may download a suitable decision-core and then the SA, regarding to several aspects, decides whether the MN should handoff or not. We implement the SA in Java language since Java software can works across different platforms. Therefore, the SA can execute on any MN if it supports Java Runtime Environment.

Besides, SA should be aware of the network environment and the network condition. MNs can request information from the Agent Server. Then the network information can be considered in the decision process. Basically, the important information includes each network interface's bandwidth, loading, and the topology of the environment.

From our observations, there are already many decision models. We summarize these to weight functions or cost functions. In detail, there are many factors, but the basic idea is to find out the highest score in the available networks. So they can hand off to the best network. But these functions are restricted by several factors. In this case, our framework not only makes better decision but also emphasizes the flexibility. The core part of this paper is DHDS. The proposed DHDS framework can fits every aspect by our software agent notion.

III. DHDS Model and Procedure

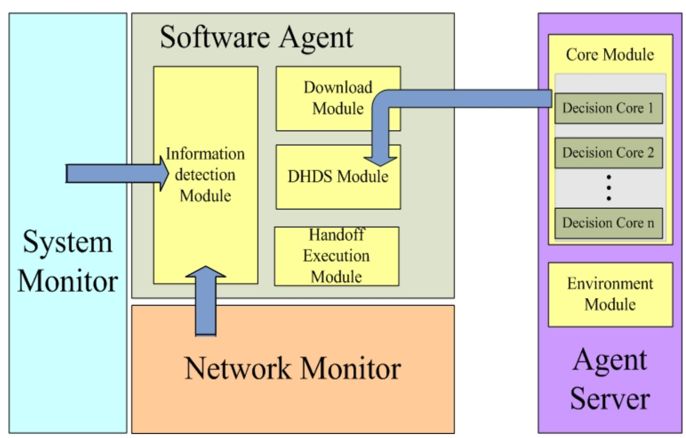

Figure 2. DHDS Architecture

In this section, we present the proposed DHDS architecture. In Figure2, DHDS is composed of four parts, including System monitor, Network monitor, Software agent and Agent Server. System monitor can observes system device, system information. System information includes the applications we launch, current remaining battery and so on. The Network monitor can check the availability of network services such as WLAN, WiMAX. Also, it can observe the network condition, e.g. received signal strength. System monitor and Network monitor keep observing and feeding the information to the software agent running on the MN. Agent Server is composed of two parts, including environment module and core module. Environment module collects every APs and network's information. Core module constructs a core-mapping table and stores numerous decision-cores. Core-mapping table checks the requirement and then maps to a decision-core. For example, if MN changes to power saving mode, then SA's information detection module links up with Agent Server to download a power-saving core according to the core-mapping table. So every MN's SA can link up with Agent Server, and then download suitable decision-core to assist handoff decision.

From Figure 3, we can know that SA's procedure. In the beginning, an MN goes into the Agent Server coverage. The MN links up with Agent Server and downloads SA at the first term. Then SA keeps running in MN. The system monitor and network monitor continuously acquire information and then bind to SA, give the information to information detection module. Therefore, information detection module evaluates the environment in which the MN locates, and connects to Agent Server to download a suitable decision-core by the core-mapping table. After the SA downloads the suitable decision-core, SA runs this core to select the best choice from candidate networks. Then SA eventually executes handoff to the most proper candidate network.

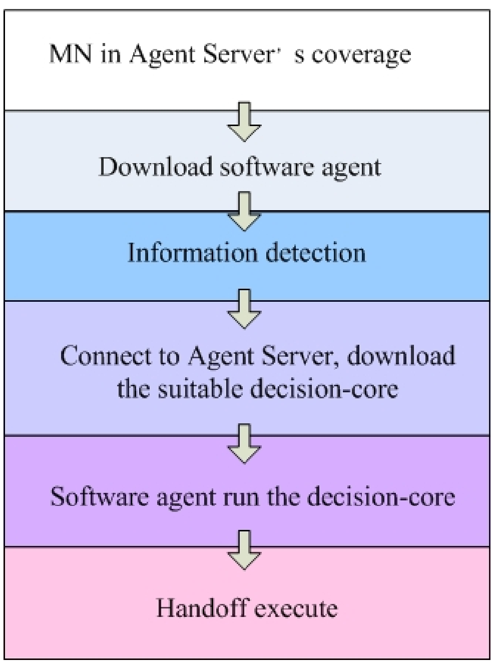

Figure 3. Software Agent's procedure

IV. EXPERIMENT

Most recent researches show their effectiveness by simulation because simulation not only takes less real equipments to carry out large scale experiments but also can be easily controlled and reproduced. Instead of simulation, we implemented our framework and demonstrate how it works. The reasons why we did so are listed as follows: 1. We can demonstrate the framework that can run different types of decision-core and select the suitable core in our scenario. 2. Based on the framework and several instances of decision-core, we can easily design other kinds of decision-core and add them into the proposed framework.

The bandwidth of each access point is assumed to be 2Mb/s. Each AP has its own coverage, which depends on nearby environment.

Without loss of generality, we construct a simple environment with only few WiFi APs in rooms, to demonstrate that the framework can work while considering only some characteristics of wireless networks, such that the readers can easily understand operation of our framework. Besides, our framework can also accommodate more complex environments, such as heterogeneous networks with both WiFi and WiMAX, and thus conform to our objective. Our experiment demonstrates the proposed architecture and easily changes the decision-core within it. Comparing with the non-changeable decision algorithm, our DHDS performs better in the experiment. Two decision-cores are implemented respectively: 1. RSS-based core, selecting a network with stronger signal as target network. 2. High bandwidth demanded core, also called load-based core. The throughput is the most important performance indicator. The network with the least load is selected as target network.

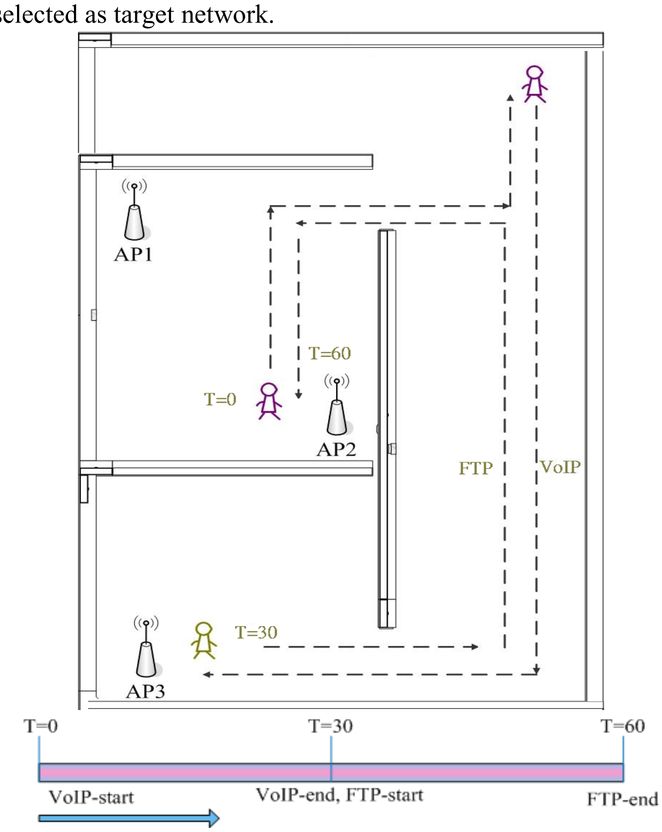

Figure 4. Experiment Scenario

Figure 4 depicts our Experiment Scenario. In the start of experiment (T=0), MN uses the VoIP application, e.g. K-phone in Linux. Upon SA's information detection model detecting a running VoIP application, SA downloads the RSS-based-core into the MN. After that, MN walks to another place. After 30 seconds (T=30), the user hangs up the VoIP and invokes a FTP application. At this time, system monitor detects the FTP application and then changes decision core from the RSS-based core to the load-based core. Finally MN returns back to its origin after exactly 60 seconds, and the experiment is hence finished.

We use the Java API-JPcap to emulate distinct AP load. It generates packets and transmits through APs in our environment. In this way, these different APs can support distinct bandwidth and QoS. The more AP loading leads that MNs get the less throughput. Then the SA collects AP's loading from Agent Server's environment module. It helps MN calculates these APs' loading at current time. Figure 5 shows the load of the APs in our Experiment.

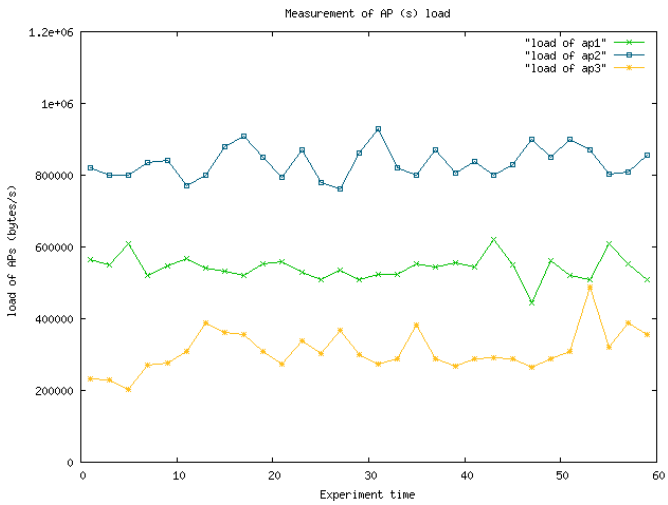

Figure 5. The load of AP1, AP2 and AP3.

V. Evaluation

The scenario of our experiment includes two stages: the first stage from 0 to 30 seconds, where a VoIP application runs on the MN, and the second stage from 30 to 60 seconds, where the VoIP session ends and an FTP application is consecutively invoked. We measure the VoIP's packet loss and FTP throughput to evaluate DHDS performance in this experiment.

In general, the VoIP application would tolerate packet loss rates of up to 3 %. So we set the threshold of packet loss in our evaluation. In Figure 6, we calculate (total packet loss/ total packet transmitted) every second. The performances of DHDS's and RSS-based strategy's are very close because when the VoIP runs in the MN, the DHDS's SA also runs RSS-based-core. But we can see the packet loss in load-based strategy is intolerable. Load-based strategy not considering RSS, packet loss occurred in the beginning of experiment because the AP3's RSS is not good. After about 15 seconds MN approaches AP3, so the total packet loss does not increase anymore. On the other hand, the MN with RSS-based strategy and DHDS always connect to the AP with the strongest signal. Packets get lost only when the experiment starts or when the MN hands off to the candidate AP.

In the second stage, we measure the FTP throughput of RSS-based strategy, load-based strategy and DHDS as shown in Figure 7. The throughput of DHDS's and Load-based strategy's are very close because when the FTP runs in MN, the SA also runs the Load-based-core. It shows the SA considering AP's loading, then keep connecting with the AP3 from T=30 to T=60.

Moreover, the figure reveals that the throughput decreases from 200Kbytes/sec to 100Kbytes/sec at about T=15 since the distance between AP3 and MN increases. In contrast with RSS-based strategy, although AP2's and AP1's RSS is the highest after 40 seconds, throughput is still lower than AP3's since the AP1 and the AP2 suffer from higher load, especially when MN connects with the AP2, whose transmission rate is even down to nearly 60K bytes/sec.

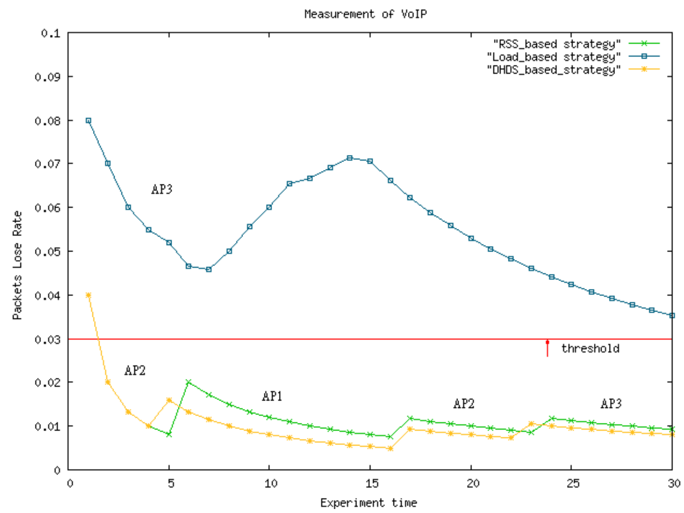

Figure 6. Packet loss of VoIP

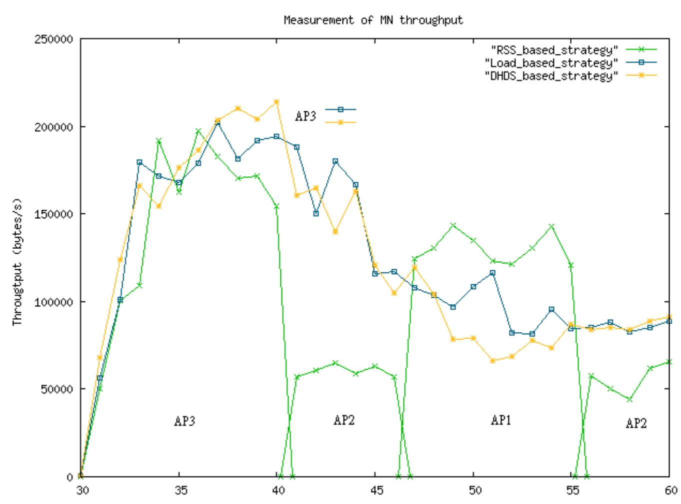

Figure 7. Throughput of FTP

Finally, this paper summarizes the result in the Table II. The readers can easily see that our framework combines the RSS-based-core and Load-based-core. Results show that different strategy connects to different APs in overall experiment. Considering VoIP packet loss rate and average throughput, the DHDS shows the better result than the non-changed strategy.

Table II. Summarized experiment results.

	Connected AP		In VoIP	In FTP
Experiment	In VoIP	In FTP	Packet Loss Rate	Avg. Throughput
RSS-based strategy	AP2->AP1->AP2->AP3	AP3->AP2->AP1	0.83%	88.5KB/s
Load-based strategy	AP3	AP3	2.40%	127.6KB/s
DHDS strategy	AP2->AP1->AP2->AP3	AP3	0.80%	123.0KB/s

VI. CONCLUSION AND FUTURE WORK

In this paper, we presented DHDS framework, which can dynamically convey handoff decision strategies. We also demonstrated DHDS, system monitor, and network monitor's prototype as well as how the framework can change decision-core for different scenarios.

Our experiment results show that comparing with the non-changeable decision algorithm, our DHDS performs better in the experiment. The SA dynamic changes its decision-core according to different application changed in MN. This framework has shown the features and advantages: fully adaptive structure and runtime strategy-binding. As in Table I, we can see that every factor can be considered in the handoff decision. So if we want to propose the decision strategy, we just put in adaptive decision-core in our MN.

An interesting aspect of DHDS is the concept of developing decision-core considering other aspects such as different wireless access technologies, e.g. 3G, WiMax. Based on the framework we can slightly modify the network monitor and add the core in to SA. Although we showed that the framework is capable of supporting different environments, better results can be obtained by supporting more details and experiment evaluation.

ACKNOWLEDGEMENTS

The authors would like to thank the Industrial Technology Research Institute and the Ministry of Economic Affairs (Contract No. MOEA97-EC-17-A-03-R7-0619) of the Republic of China, Taiwan, for financially supporting this research. This work was also supported partially by National Science Council of Republic of China under Grants NSC 97-2221-E-194-025 and 96-2219-E-194-008.

REFERENCES

[1] I. F. Akyildiz, J. Ho, and W. Wang, "Mobility Management in Next-Generation Wireless Systems", *Proceeding of IEEE*, 1999, vol. 87, no. 8, pp. 1347–1384.

[2] K. Yang, B. Qiu, and L. S. Dooley, "Using SINR as Vertical Handoff Criteria in Multimedia Wireless Networks", IEEE International Conference on Multimedia and Expo, 2007, pp. 967-970.

[3] C. Guo, Z. Guo, Q. Zhang, and W. Zhu. "A Seamless and Proactive End-to-End Mobility Solution for Roaming Across Heterogeneous Wireless Networks", IEEE Journal of Selected Areas in Communications, 2004, vol.22, no.5, pp. 834-848.

[4] H. A. Velayos, and V. G. Karlsson, "Load Balancing in Overlapping Wireless LAN Cells", IEEE International Conference on communication, 2004, pp. 3833-3836.

[5] W. Shen and Q. Zeng "A Novel Decision Strategy of Vertical Handoff in Overlay Wireless Networks Network Computing and Applications", Fifth IEEE International Symposium on Network Computing and Applications, 2006, pp. 227-230.

[6] F. Zhu and J. McNair, "Optimizations for vertical handoff decision algorithms", IEEE Wireless Communications and Networking Conference, 2004, vol. 2, pp. 867-872.

[7] J. McNair and Z. Fang, "Vertical handoffs in fourth-generation multi-network environments", IEEE Wireless Communication, 2004, vol. 3, pp. 8-15.

[8] L. J. Chen, T. Sun, B. Chen, V. Rajendran, and M.Gerla, "A smart Decision Model for Vertical Handoff", The 4th ANWIRE International Workshop on Wireless Internet, 2004, pp. 1-5.

[9] M . Bernaschi, F. Cacace and G. Iannello, "Vertical Handoff Decision Algorithm providing Optimized Performance in Heterogeneous Wireless Networks", IEEE Global Telecommunications Conference, 2004, pp. 100-107.

[10] W. Chen and Y. Shu, "Active application oriented vertical handoff in next-generation wireless networks Wireless Communications and Networking Conference", IEEE Wireless Communications and Networking Conference, 2005, vol. 3, pp. 1383-1388.

[11] Y. Shi, S. Chen, Y. Li, L. Ye, and D. Dong, "An Application-Oriented Cooperative Vertical Handoff Decision Method for Multi-interface Mobile Terminals Telecommunications", 7th International Conference on ITS, 2007, pp. 1-5.