Study on using Design Patterns to Implement a Simulation System for WiMAX Network

Hsin-Hung Hsieh[1], Bih-Hwang Lee[1], Huai-Kuei Wu[2], Wen-Pin Hsu[1] and Hung-Chi Chien[1]
[1] Department of Electrical Engineering, National Taiwan University of Science and Technology, Taiwan
[2] Department of Electronic Engineering, Oriental Institute of Technology, Taiwan

Abstract: In the study of the wireless network, the network simulator always is an important tool for researchers to evaluate their theories. If a selected simulator inherently has poor architecture, the maintenance, will become more difficult in the future. To increase the flexibility of a simulator's architecture, the use of design patterns as the norms of system architecture design is suggested in this paper. In this paper, we propose a network simulator architecture, named as CCGns, which is a discrete-event virtual network simulator and follows the IEEE 802.16-2009 standard. The main contributions are proposed three architectures include SM^{3}PA, AMRNA, and TSMVBA. By using mathematic calculation to verify the simulation results, CCGns has been proven to possess excellent fidelity.

Keywords**-- simulator, design pattern, IEEE 802.16, object oriented design**

1. Introduction

To study wireless network, a network simulator has been an important tool. By using software to construct a simulator, there have the advantages of having more convenience to build simulation scenarios and establish monitoring procedure. In order to quickly modify the scenario and revise the study concepts, simulation software has become the main method to fulfill the researchers' study concepts. Because of the natures of innovation and excellence in academic study, simulation software frequently encounters that the existing modular functions are inadequate and need to be appended or modified. How to find the most suitable simulation software always is a difficult and important issue, which may be determined by some evaluation metrics, such as fidelity, suitability, extensibility, scalability, user-support, learning-time, implementation-friendly, performance, cost, etc.

In order to make the simulator's architecture to have the extensibility, we proposed to use the design patterns as the norm of system architecture design and implement. We surveyed the six most used simulators, i.e., J-Sim [1], NS-2 [2], OMNeT++ [3], OPNET [4], QualNet [5], and NS-3 [6], to ponder their system architectures and design concepts from their source codes and the related literatures of the modular functions. We propose a network simulator architecture, named as CCGns, based on the IEEE 802.16-2009 standard. CCGns is a discrete-event virtual network simulator coded by Java language and uses Eclipse as develop tool, which obeys the object oriented design (OOD) principles to design the function modules. CCGns uses design paradigm to design system architecture and comprises of eight packages: physical (PHY) layer, medium access control (MAC) layer, network (NET) layer, devices (DEV), topology (TPY), events (EVT), scheduler (SCH), and report (RPT) packages. The main contribution of CCGns is to propose three aspects: a scalable MAC messages management and processing architecture (SM^{3}PA), an applicable for multi-hop relay network architecture (AMRNA), and a two-stage minimum variance bandwidth allocation (TSMVBA) algorithm. In SM^{3}PA, users can arbitrarily define the management message types to focus on the interested messages and save the needed real simulation time. In AMRNA architecture, users can use a unified interface to simulate four different network topology architectures. TSMVBA is responsible to provide the optimal frame structure for the uplink (UL) and downlink (DL) subframe using the different subcarrier permutation mode.

2. Related Works

Several literatures have been proposed to evaluate the simulation software, which can be summarized into three categories: The first category is for overview and introduction [7,8]. The second category is the evaluation and comparison for the inner system architecture, such as fidelity, suitability, scalability, and extensibility [9–12]. The third is the evaluation and comparison for the outer user experiences, such as learning time, user support, interface user-friendly, operate convenience, and performance [13–15].

In the previous studies, many researchers provide their contribution on how to add or modify the WiMAX module, e.g., some NS-2 WiMAX modules are proposed by Networks and Distributed System Laboratory (NDSL) [16], National Institute of Standards and Technology (NIST) [17], and Light WiMAX (LWX) [18], respectively. In addition, some NS-3 WiMAX modules are proposed [19, 20, 21]. In [22], the authors propose to use strategy pattern to implement the scheduling module. From these literatures, we found that any two simulation software do not implement the same control message processing procedure. Hence, it would be different for the performance expression of time-related evaluation metrics in different simulation software, e.g., the influence of packet delay-time. In order to investigate how the management message processing procedure affects the time-related evaluation metrics, a system architecture should allow users to arbitrarily define his management message types and processing procedure, then we can obtain a more objective system performance evaluation results.

A programming is a series of thinking process to convert an abstract problem description into a realistic code entity. Compare with the program-oriented design manner that based on the functions capability and executions order, the object-oriented design manner has capability to receive, process, and deliver independent data unit as design foundation, i.e., object, hence the system architecture has more flexible and easier to maintain. The design patterns are exactly in the basis of object-oriented design manner to get rid of bad one and keep good one from the frequent occurrence solutions. The purpose of design patterns is to organize these solutions in a simple and easy way, and they make our programming more flexible, more modularized, more reusable, and easier to understand. Design patterns do not teach us how to code, but they are the discussion schemes to solve the problems under different situations. The pattern in design patterns means a useful solution has been proven, that can be used to solve the recurring problems under a special scenario. Based on the Gof literatures [23], there are twenty-three types of design patterns, and are divided into three categories, i.e., creational, structural, and behavioral.

3. CCG Network Simulator

It is difficult to quantify and have a consistent assessment metric for the system architecture flexibility and expansion convenience, hence we propose to use design patterns as the criteria of system architecture design. In this section, we introduce how CCGns applies design patterns in the system architecture of network simulation software. CCGns follows IEEE 802.16-2009 standard and consists of eight packages, namely PHY, MAC, NET, DEV, TPY, EVT, SCH, and RPT packages.

PHY package is responsible for the physical layer functions, the main capability is using the frame structure form to provide link capacity, the modular function includes the various wireless technical frame structures. MAC package is responsible for the MAC layer functions, whose main capability is to process and execute the management messages. NET package is responsible for the network and upper layers functions, whose main capability is to implement the traffic generator. DEV package is responsible for network device, whose main capability is to facilitate the definition and implementation of the real world network devices. TPY package is responsible for real world network topology, whose main capability is conveniently to manage and use the network topology. EVT package is responsible for the real world events, whose main capability is to define the event types and the sub-procedure operation interfaces inside the event process procedures, and the code entity of execution detail is provided by the device instance. SCH package is responsible for scheduling mechanism, whose main capability is the implementation of scheduler algorithm, and the goal is to make the reuse of system scheduler codes to maximize. RPT package is responsible for the output of system information and performance evaluation metrics, whose main capability is to make the system information output module independent of other packages.

In order to prove practicable, we define three command flows, i.e., connection contention mechanism, bandwidth request and grant mechanism, and data transmit and receive mechanism. The design of command flow is based on hybrid multi-hop relay network (HMRN), and HMRN is described in next chapter.

4. Simulation Experiments

In order to demonstrate the above-mentioned design patterns for simulation, we propose a scenario as shown in Fig. 1, which shows a base station (BS) surrounded by three transparent relay stations (tRSs) and six non-transparent RSs (ntRSs) located at the corresponding positions as shown in this figure. In the hexagonal cell structure, the system service range is the inscribed circle of radius R, which is divided into the center region and the peripheral region. The center region is a circular area of radius 2/3R; the users located in the center area are served by BS. The peripheral region is an annular area of width 1/3R; every 120 degrees deploys one RS; the users located in the annular area are served by the RS. In hybrid relay network architecture, we deploy tRS and ntRS on the annular area and the vertex of hexagonal cell, respectively, where ntRS needs tRS's help to forward data to BS.

Table 1 System parameters for simulation

Parameters	Values
Node Distribution Type	Proportional
Schedule Type	Uniform
PHY Specification	SOFDMA 10MHz
Permutation	DL PUSC and UL PUSC
Frame Duration	5 ms
Modulation and Coding Scheme	16-QAM 3/4
Bandwidth Resource Ratio	DL : UL = 1 : 1
Packet Size	5 slots
Packet mean interarrival time	1.0 ms
Length of Packet Queue	50 Packages
Number of BS, tRS, and ntRS	1, 3, 6
Total Number of SS	20

The relevant system parameters are listed in Table 1. Among them, the proportional user distribution type means the serviced user amount of a serving station is decided by the ration between its service area and overall service area. The uniform scheduling mechanism means the serving station assigns resources to a user according to this user's total amount of the assigned resource (TAAR) in the past, where smaller TAAR has higher priority, which is to let all users fairly use the bandwidth resource. The partial usage of subcarriers (PUSC) permutation mode are also used in both UL and DL directions.

Fig. 2 shows the uplink subframe output data (USOD) for various device types, in kilobit (kb), where the abscissa and Y-axis represent system time and output data, respectively. The USOD of BS subscribe (bSS), tRS subscribe (tSS), ntRS subscribe (ntSS), RS, and BS have been shown by the curves in different colors of red, blue, green, pink and orange, respectively. In Fig. 2, the USOD of BS is 30.96 kb, which is composed of the BS user data 10.8 kb and the forwarded user data by RS 20.16 kb, while the latter part are from tSS and ntSS with the forwarded user data 12.96 kb and 7.2 kb, respectively. Because that the access zone (AZ) size for one hop users is 105 slots ($3T_{slots}$ ×35CHs), of which 75 slots are allocated to the BS to receive user data, the BS AZ output data is 10.8 kb (75×48×4×3/4=10.8 kb). The size of transparent relay zone (tRZ) is 140 slots ($4T_{slots}$ ×35CHs). BS uses all tRZ slots to receive the data forwarded by tRS, so the output data is 20.16 kb. As to non-transparent relay zone (ntRZ), because it merely supplies forwarded data from ntRS to tRS, so the output data is 0 kb. It's worth mentioning that, in order to guarantee an RS can forward data to its destination as soon as possible, we adopted the relay first resource allocation strategy (RFRAS). RFRAS gives RS packets having higher priority when the service object includes RS and SS packets, then resources will be assigned according to their priority order. In the meantime, to prevent service objects with lower priority from being starved, each service object can only be assigned with one resource at a time.

Fig. 3 shows the average packet delay times for various device types. In Fig. 3, the delay times of bSS, tSS, ntSS, RS, BS with the values of 2.5 ms, 7.5 ms, 127 ms, 49.1 ms, and 32.8 ms, respectively. The value of red line is 2.5 ms, because BS users can upload data to the BS directly.

Fig. 4 shows the average packet queue lengths for all device types. In Fig. 4, we first see the growing up trends of bSS and tSS are similar. Since they have identical packet arrival rate and approximately the same packet upload rate, their growing curves are similar. Secondly, we can see that ntSS maintained at about 10 packets at the initial stage, then reached full loading at 400 ms. At the initial stage, because the tRS queue is under light loading, tRS has the maximum capability to receive serving user packets. However, as the ability of BS receiving packets is lower than the ability of the whole tRSs transmitting packets, which causes the packet queue length to gradually increase until full.

Fig. 5 shows the average packet waiting times (APWT) for various device types. In AZ, each bSS has 9.375 slots, i.e., 1.875 packet resource usable in average, and bSS queue often keeps at 48 packets, the APWT of the last packet is 25.6 frame duration, i.e., 128 ms. Similarly, tSS has 10 slots usable in average, APWT of the 48th packet is 24 frame duration, i.e., 120 ms. For tSS, every ntSS can upload 3.5 packets in average, therefore, the 47th packet needs to wait for 13.4 frame duration, i.e., 67 ms. Therefore the estimated results are close to the simulation results.

Fig. 6 shows the average packet drop ratios for all device types. In Fig. 6, in terms of bSS, the packet drop ratio is 58% by simulation, and it is lower than 62.5% by calculation with average manner. Because at the initial stage of simulation, there are plenty bandwidth resources, so the drop ratio is small. The smaller drop ratio dilutes the late stage's larger drop ratio, therefore, the simulation drop ratio is lower than the average drop ratio. Similar situation also happens in tSS and ntSS. In terms of tSS, the simulation and average drop ratios are 55.4% and 60%, respectively, and the ntSS simulation and average drop ratios are 22.5% and 30% respectively.

5. Conclusions

In order to make the simulator architecture to have the flexibility and the simulation results are believable. The design patterns are used as the norms of system architecture design, and the creative ideas inspired by the most use simulator design ideas and related module expansion literatures to construct our system architecture. In the simulation results, we use the most complex hybrid relay network topology as an example, and also use the mutual verifying manner for the average calculated values and the simulation results to prove excellent fidelity for the system throughput, packet average delay time, packet average wait time, packet average queue length, and packet average drop ratio. Although we are not the first to propose the design patterns on the wireless network simulator architectures, but both the amount and types of using design patterns are the most. In the future, we will also implement the LTE protocol simulation system by using this architecture.

Acknowledgement

This work was supported in part by the Ministry of Science and Technology (MOST) of Taiwan under Grant No.: MOST 102-2221-E-011-069.

References

[1] J-Sim, [Online] Available: https://sites.google.com/site/jsimofficial/

[2] NS-2, [Online] Available: http://www.isi.edu/nsnam/ns/

[3] OMNeT++, [Online] Available: http://www.omnetpp.org/

[4] OPNET WiMAX, [Online] Available: http://www.opnet.com/WiMAX/index.html

[5] QualNet, [Online] Available: http://web.scalable-networks.com/content/qualnet

[6] NS-3 [Online] Available: http://www.nsnam.org/

[7] L. Hogie, P. Bouvry, and F. Guinand, "An Overview of MANETs Simulation," In Electronic Notes in Theoretical Computer Science, Proc. of 1st International Workshop on Methods and Tools for Coordinating Concurrent, Distributed and Mobile Systems (MTCoord'05), LNCS, pp.81-101, Namur, Belgium, April 2005. Elsevier.

[8] S. Siraj, A. K. Gupta, and R. Badgujar, "Network Simulation Tools Survey," International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), Vol.1, Issue 4, pp-201-210, June 2012.

[9] S. Duflos, G. L. Grand, A. A. Diallo, C. Chaudet, A. Hecker, C. Balducelli, F. Flentge, C. Schwaegerl, and O. Seifert, "List of available and suitable simulation components," Technical report, Ecole Nationale Superieure des Telecommunications (ENST), Sep. 2006.

[7] L. Begg, W. Liu, L. Pawlikowski, S. Perera, and H. Sirisena, "Survey of simulators of Next Generation Networks for Studying Service Availability and Resilience," Technical Report TR-COSC 05/06, Department of Computer Science & Software Engineering, University of Canterbury, Christchurch, New Zealand, February 2006.

[8] N. I. Sarkar and S. A. Halim, "A Review of Simulation of Telecommunication Networks: Simulators, Classification, Comparison, Methodologies, and Recommendations," Cyber Journals: Multidisciplinary Journals in Science and Technology. Special Issue, Journal of Selected Areas in Telecommunications (JSAT), vol.2(3), pp.10 – 17, Mar. 2011.

[9] A. Kumar, S. K. Kaushik, R. Sharma, and P. Raj, "Simulators for Wireless Networks: A Comparative Study," International Conference on Computing Sciences (ICCS), pp.338-342, 2012.

[13] J. Lessmann, P. Janacik, L. Lachev, and D. Orfanus, "Comparative Study of Wireless Network Simulators," Seventh International Conference on Networking, pp.517-523, 2008.

[14] E. Weingartner, H. vom Lehn, and K. Wehrle, "A Performance Comparison of Recent Network Simulators,"

IEEE International Conference on Communications (ICC), pp.1-5, June 2009.

[15] A. R. Khan, S. M. Bilal, and M. Othman, “A performance comparison of open source network simulators for wireless networks,” in IEEE International Conference on Control System, Computing and Engineering (ICCSCE), pp.34-38, Nov. 2012.

[16] NDSL (Networks & Distributed Systems Lab), [Online] Available: http://ndsl.csie.cgu.edu.tw/ .

[17] N. I. of Standards and Technology, “The Network Simulator NS-2 NIST add-on – IEEE 802.16 model (PHY+MAC),” Technical Report, Jan. 2009.

[18] Y.-C. Lai, and Y.-H. Chen, “Designing and Implementing an IEEE 802.16 Network Simulator for Performance Evaluation of Bandwidth Allocation Algorithms,” 2009 11th IEEE International Conference on High Performance Computing and Communications (HPCC '09), pp.432-437, 2009.

[19] J. Farooq, and T. Turletti, “An IEEE 802.16 WiMAX Module for the NS-3 Simulator,” in Proc. of the 2nd International Conference on Simulation tools and Techniques, pp.1-11, 2009.

[20] M. A. Ismail, G. Piro, L. A. Grieco, and T. Turletti, “An Improved IEEE 802.16 WiMAX Module for the NS-3 Simulator,” in Proc. SIMUTools, pp.63:1-63:10, Mar. 2010.

[21] W. P. Furlong, and R. Guha, “OFDMA Extension of NS-3 WiMAX Module,” In UKSim Fourth European Modeling Symposium on Computer Modeling and Simulation, pp.426-431, 2010.

[22] G. Pedreno, J. J. Alcaraz, and F. Cerdan, “Using Design Patterns in a HSDPA System Simulator,” 3rd International Symposium on Wireless Communication Systems (ISWCS), pp.679-683, Sept. 2006.

[23] E. Gamma, R. Helm, R. Johnson and J. Vlissides, “Design Patterns: Elements of Reusable Object-Oriented Software”, 1995.

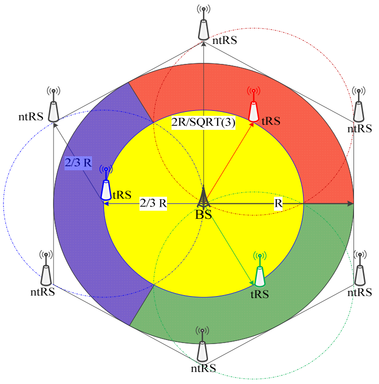

Fig. 1 Single cell simulation environment

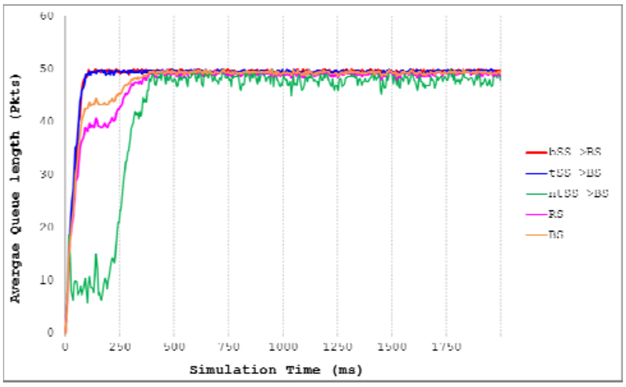

Fig. 4 The average queue length for all device types

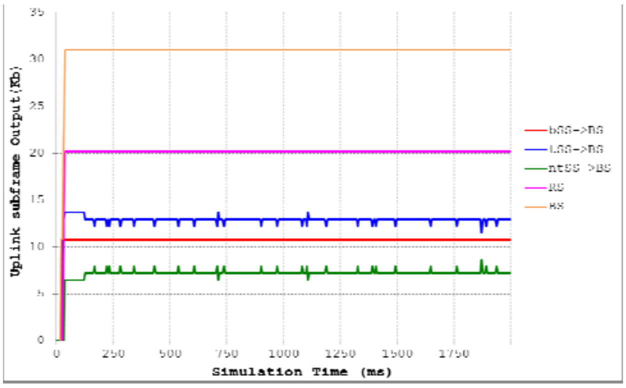

Fig. 2 The UL subframe output data of various device types.

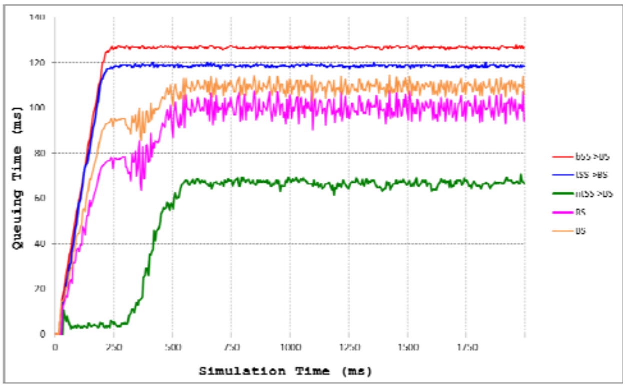

Fig. 5 The average waiting time

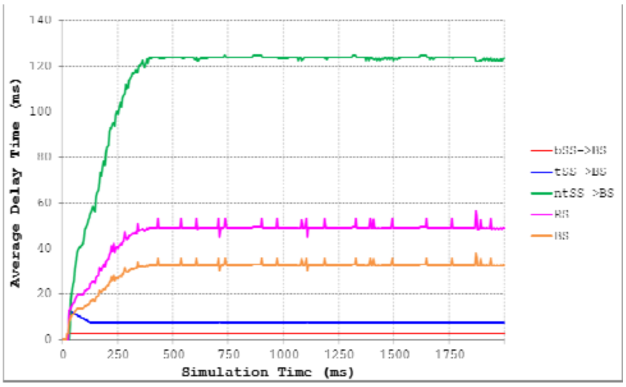

Fig. 3 The delay time of all device types

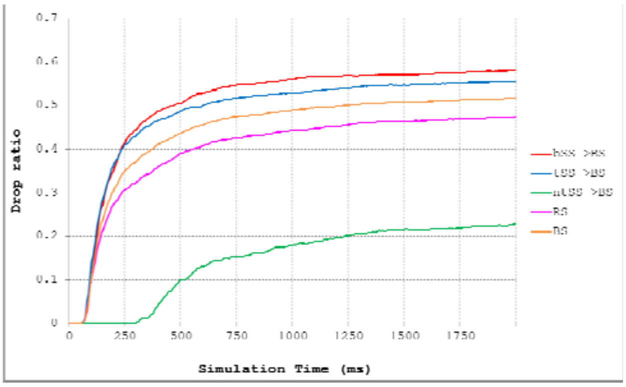

Fig. 6 The average packet drop ratio