Functional Model of Algorithms for Optimal Data Transmission Over Air Interface

Vasil Shterev[1] and Georgi Iliev[2]

Abstract – **This paper proposes a model of architecture for optimal transmission in a wireless channel. The nature of this type of channels is random and stochastic; hence algorithms and methods of control are more difficult and complicated. Our goal is to achieve optimal performance and maximum throughput for data transfer over this type of channels. The model is implement in an UML (Unified Modeling Language) environment. We investigate the wireless channel and mitigate the negative effects in it. In our model, we take into account most important parameters that influence the characteristic of the channel. On the other hand these parameters must be easy and fast to estimate in order to achieve effective and feasible management of our system.**

Keywords – **Teletraffic, Network Control, Data Communication.**

I. Introduction

The main problem in a wireless communication system is the stochastic channel. Many existing systems [2] try to mitigate the negative effects when we exploit a radio channel to transfer information (data, voice and video). Popular well-known examples of these systems are IEEE 802.11 (Wi-Fi) and IEEE 802.16 (WiMAX). They play a scenario in which most reconfigurable parameters are separately tune due to changes in the environment. In the IEEE 802.16 system, designers have include three basic values from the OSI MAC layer. These parameters are: output power level; adaptive modulations scheme and control of coding rate [2].

Further, we develop a model in the UML environment, where an exact sequence of actions is taken. The reason to select UML as tool is because: it is demonstrative and easy to understand; it is a standardize general-purpose modeling language and it combines the best techniques from data modeling [1].

II. A Brief Overview of UML

UML has many versions (from ver. 1.0 to 2.2 nowadays). The latest version has 14 types of diagrams divide into two categories [4]. Seven diagram types represent the structural information, and the other seven represent general types of behavior, including four that represent different aspects of interactions.

[1]Vasil Shterev is with the Faculty of Telecommunications at Technical University of Sofia, 8 Kl. Ohridski Blvd, Sofia 1000, Bulgaria, E-mail: vas@tu-sofia.bg

[2]Georgi Iliev is with the Faculty of Telecommunications at Technical University of Sofia, 8 Kl. Ohridski Blvd, Sofia 1000, Bulgaria, E-mail: gli@tu-sofia.bg

UML does not restrict UML element types to a certain diagram type. In general, every UML element may appear on almost all types of diagrams. This flexibility is very important for developers.

We will concentrate on three types of diagrams: one of structural information and two of behavioral category. From the first category, we chose a class diagram, because it depicts how different components are connect together to form larger components.

Our second choice is to use a use case diagram, because it presents a graphical overview of the functionality provide by a system in terms of actors, their goals (represent as use cases), and any dependencies between those use cases. The actors in the use case diagram represent functionality which have a relation to the functionality present in use case and this functionality is not depict in detail [3]. These diagrams are very important when we model the behavior of a system.

The third type is a sequence diagram. It is shown as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchang between them and in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

III. The UML Diagrams

In this section we propose three types of UML diagrams. We will start with use case, then follow class diagram and finally is sequence one.

A. Use Case Diagram

Use case diagram is depict in Fig.1. This figure depicts two main types of actors: operator and customer. The Operator has connections (also call communication association) with different use cases. In our case, they are: system startup, effective usage of system, maximization of profit, illegal usage and user account. Operator has also connection to a billing system, which in turn is connect with session use case. The last one has communication association with second actor – the customer. A Session is communication with connection, which in turn has a connection with QoS requirements use case and allocation channel. Power management and free resource unit are connect to an allocation channel. In turn the free resource unit has a communication with the free time slot and the free bandwidth use case. This is an overview of interaction between different use cases.

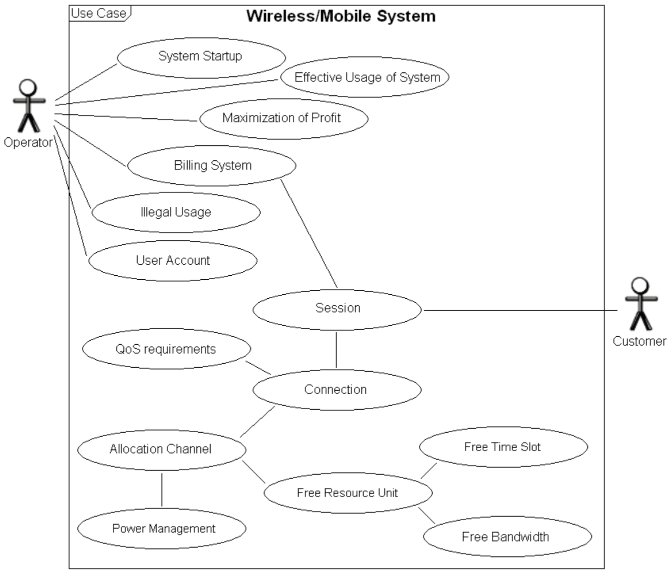

Fig. 1. Use case diagram of wireless/mobile system.

B. Class Diagram

Next, we show a class diagram which depicts a relation between separate classes. The reason, why in Fig. 2 we miss to write attributes and methods, is clear - all attributes and methods are describe below Fig. 2.

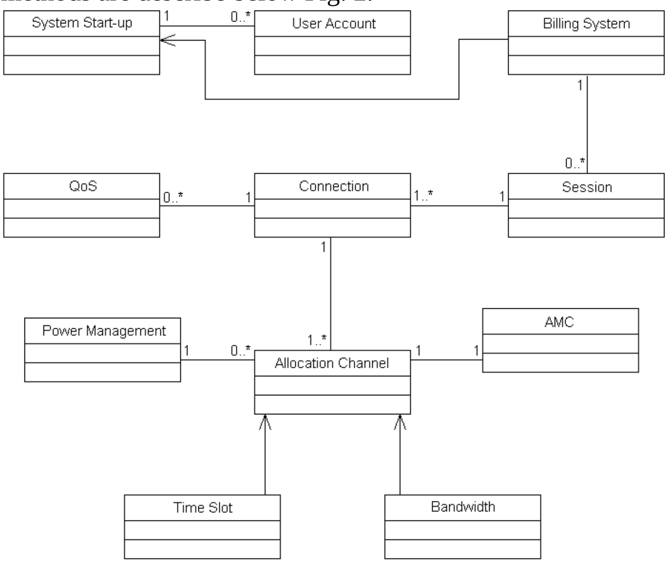

Fig. 2. Class diagram of wireless/mobile system. In second and third row, we intentionally miss to put the attributes and methods.

The first abstract class is System Start-up. It has a relationship with two other classes, which are User Account and Billing System. The association between User Account (UA) and System Start-up (SS) is such that exactly one SS can interact with 0, 1 or more UA. The SS has a dependency relationship with abstract class Billing System (BS). In turn, BS has an association with Session abstract class, with multiplicities of type one BS and multiple (including zero) Session.

QoS and Connection have the same multiplicities as previous ones, because it is possible that we do not have any QoS requirement for our connection. The opposite situation, where we need nearly all requirements is also possible. The relation between the Connection and the Session is little bit different because one Session must have at least one or more Connections. The situation is the same in the relationship between the Connection and the Allocation Channel (AC). Here one Connection must have at least one AC. The situation between Power Management (PM) and AC is as follows: one PM has the possibility to serve zero or more AC, because PM functions independently.

The abstract class AMC (adaptive modulation and coding) and AC have a one to one type of relationship, because we use a particular AMC scheme for each channel. Finally, the relationship between Time Slot (TS) and AC is the same as that of a Bandwidth (BW) and an AC. They are dependent on each other, because the AC cannot exist without TS and BW. Further we describe the attributes and methods of each class.

The SS is very important, because it is responsible for the start of the whole system and checks different components, so attributes assign to this abstract class are: system test, memory test and component test, while the methods and operations which correspond to this class are: check memory; check system block and check peripheral component. Next class is the UA, which is characterize by identification, user name and password as attributes; and check for illegal usage, check password and validation account for operations.

BS is also very important, especially for the operator of the system. This class has an attribute call identification and a method call update account record. The next class is QoS, which has the following attributes: delay, jitter, and bandwidth and packet loss ratio. The operations that correspond to this attributes are: watch delay, watch jitter parameter, look up for bandwidth and look up packet loss ratio. Further we examine class PM. It possesses the following attributes: transmitting power and maximal power. The operation here is call power control. The next abstract class is the AC, which has two attributes: network status and scheduling. Operations that correspond to this abstract class are: network status estimation and actual schedule scheme. The Session class has four methods: start session time, end session time, requirements and maximal profit. Methods are: set session start time and set session end time. Next abstract class is Connection. It has five attributes: start connection time, end connection time, bit rate, bit error rate (BER) and delay. Corresponding operations are: set connection start time, set connection end time, watch bit rate, watch BER and look up delay. The next class is call AMC and it has two attributes and methods. They are: modulation scheme, convolutional code rate and respectively adaptive modulation and code rate adjust. The last two abstract classes are exceptionally important for the correct operation of wireless / mobile systems. The first one is TS, which has the following attributes: time and free time slot. Methods are: system time,

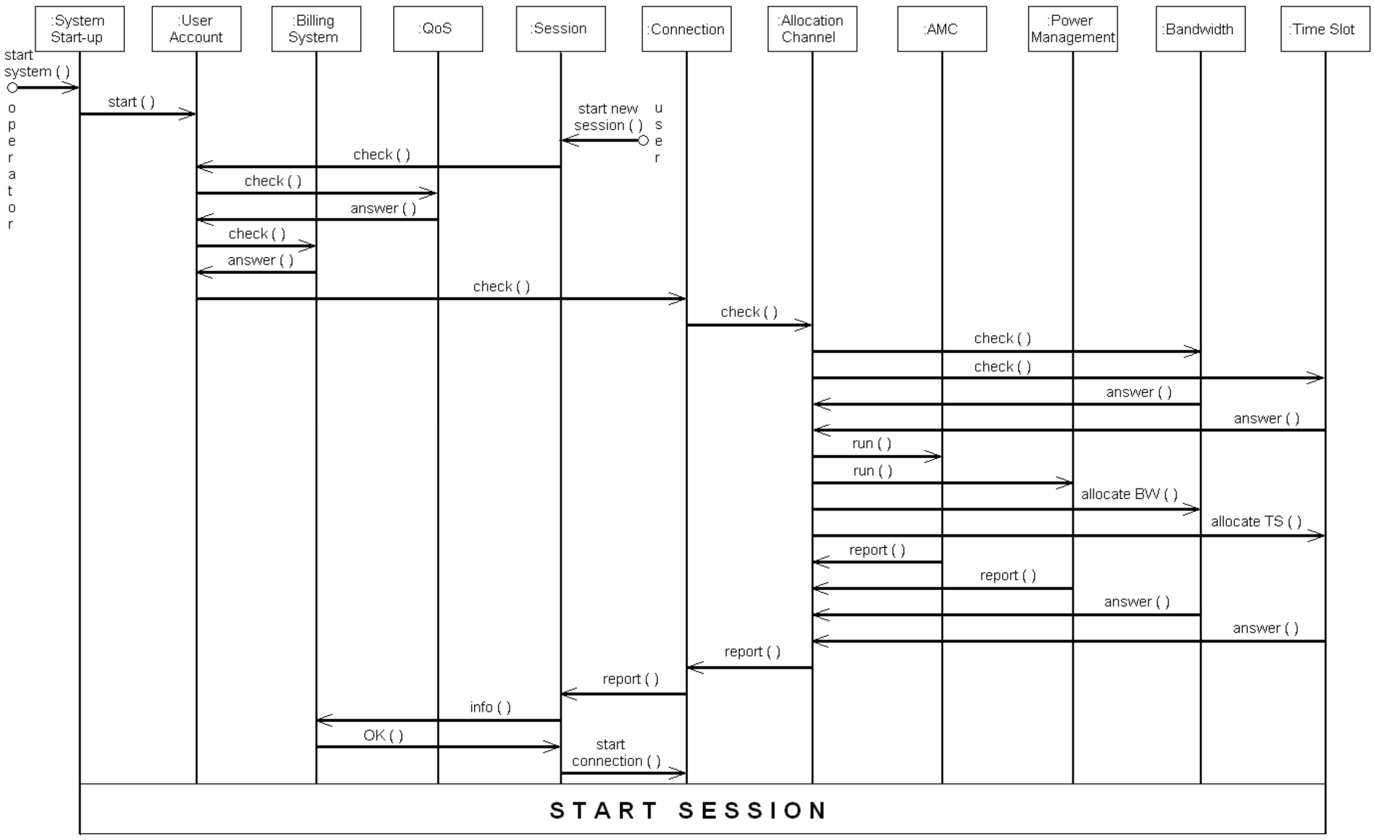

Fig.3 Sequence diagram, which depicts the system star-up and the start of a new session initiate by a customer. The diagram ends when session start is successful.

time slot estimation and time slot validation. The second class is BW. It has three attributes: frequency, free bandwidth, signal to noise ratio (SNR) and two methods, which are: frequency estimation and SNR estimation.

C. Sequence Diagram

Next, we will describe message flow from the initial moment, when the operator turns on the system and user (costumer) who wants to begin a new session, with parameters that are correspond to his requirements. We will discontinue the message flow when the session starts. Each message has various specific parameters, which are denote with brackets.

In Fig. 3 we show the message sequence. The First message comes from a wireless / mobile network operator. Its purpose is to start the system when reach SS. When SS receives this, it performs the actions that have been programmed in it. Actions are logically connect with attributes and methods in class diagrams. Further, the SS sends a start message to UA to activate its functionality.

From some arbitrary moment on, our customer wants to begin a new session. The message representing this action is call *start new session*, and it is receive from the Session object. This message contains information about customer requirements (video call, video on demand, file transfer etc.). The next step is to send a message *check* from Session to UA to check for illegal usage. If the user has a valid account, the system will send a *check* message to the QoS object to obtain the parameters that characterize the specific session. Further, the QoS sends back an *answer* containing the specific characteristics of this customer session. Then UA sends the *check* to BS to verify whether the customer has enough credit to perform the session. If he has enough credits, the *answer* message between BS and UA is positive and then the UA sends the *check* message to Connection to inspect, whether the wireless / mobile system can perform the requirement. Further, Connection sends again the *check* message to the AC object. The aim is to check-up whether the system has enough capacity to accept one more session. This is perform by sending the *check* message from AC to BW and to the TS objects. They, on their part, send back an *answer* to AC. The answer is either yes or no, and it depends on whether the system has or has not enough free time slot and bandwidth. If the answer is positive, the AC sends a *run* message to AMC and PM; *allocates BW* and *allocates TS* to BW and TS objects. The next step is to send the corresponding reply messages to AC. They are: *report* from PM and *answer* from BW and TS. The aim is AC to verify that AMC and PM are in good condition. Moreover, when BW and TS return the *answer* message, they report that they are allocate in this slot and are ready to use it. Further, AC sends a *report* message to the Connection, and the Connection also sends a *report* to the Session. Then the Session sends an *info* message to BS with the parameters of session such as session start time (to start the bill) and others that relate to the user profile. The answer message from BS to Session is *OK*. The last message is from the Session to the Connection and it is call *start connection*, thus the session is starting. Logically, termination of a connection is doing reversely, but it has many variations depending on the type of session.

IV. The Algorithm for Power Control, Code Rate and Modulation

In this section we propose an algorithm, which deals with power control, convolutional code rate and modulation. It is done as block scheme, which is illustrate in Fig. 4.

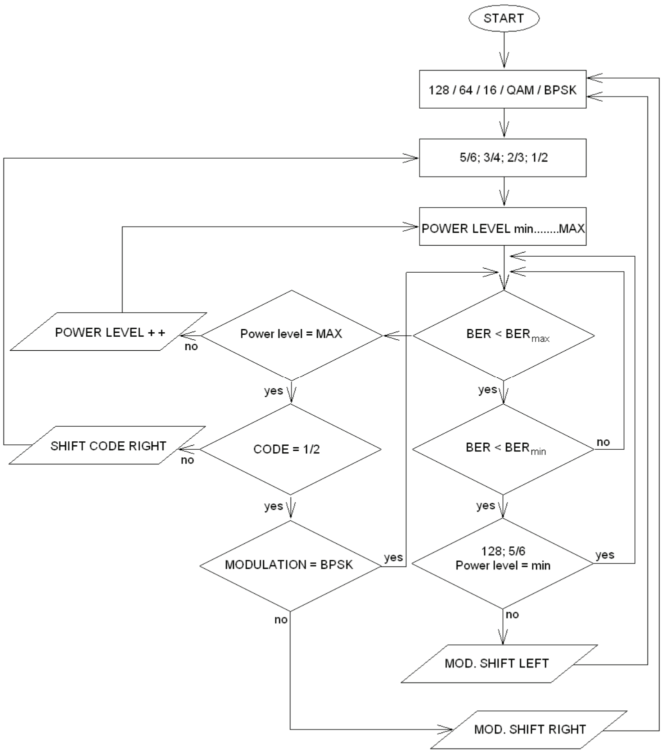

Fig.4 Block scheme of algorithm that controls power level, modulation and convolutional code rate.

The algorithm begins at start block. Next is a block where the system changes the modulation scheme from among five types (128QAM, 64QAM, 16QAM, QAM and BPSK). They are arrange from more effective to most robust. The next block is the convolutional code rate. It has four schemes which are 5/6, 3/4, 2/3 and 1/2. Here, we again arrange it from low information excess to higher. Further, comes the block call power level. If we have to interpret upper logic, from the left side is that of minimum power level, which gradually increase when we move right. The final level, of course, is maximum permit transmitting power. For these three blocks, we have to notice, that if the arrow comes from above, blocks choose the value most on left. If the arrow comes sideways, the block selects the value which is indicate by the algorithm. Next, the algorithm has a combination of decision blocks, where the algorithm is branch depending on the channel condition. Finally, the system has four processing step blocks, which provoke action (move left or right) to change power level, convolutional code rate and modulation scheme. Parameters BER_{min} and BER_{max}, can be choose specifically depend from application and customer necessary. Typical values for BER_{min} and BER_{max} are 10^{-8} and 10^{-4}. If BER is less than BER_{min}, we have to change parameters of system because we waste resource. If BER is bigger than BER_{max}, system is trying to mitigate this.

The logic behind this is as follows: when the channel quality is good, the MAC layer chooses the highest modulation rate, (128 QAM), giving the system the highest throughput. When the channel quality degrades, the MAC layer reduces the modulation rate (64 QAM), reducing the throughput. In practice, adaptive modulation and coding rate are use in conjunction with power control. In point-to-multipoint network deployments with multiple users in a cell service by a base station, when a link degradation arises for a user, the base station first increases the transmit power of the user to provide extra link budget gain, until it reaches the maximum permitted. If the receive signal quality does not improve, then the coding rate is reduced. Extra redundancy is add to provide more coding gain for better error correction performance. If the receive signal quality still does not improve, then the modulation rate is reduce as a last resort (as this significantly affects the throughput then others). Similar (reverse) process is also follow when link quality improves.

V. Conclusion

In this paper we propose three types of UML diagrams for wireless / mobile system. The first (use case) is more general and illustrates the main subsystems and topics in the system. The second one (class) depicts separate systems with their attributes and methods. It also shows the types of dependencies and relations between classes. The third diagram (sequence) depicts the sequence of messages between different objects. Parameters of messages are explain briefly.

Finally, we propose an effective algorithm for power control, code rate adjustment and modulation management. The algorithm is present as a block scheme, and it is so easy to implement in hardware. Further, we consider developing the algorithm and include more important parameters such as buffer control and interaction between customers.

Acknowledgment

This work is supported by the Technical University of Sofia Science Fund – Grant No. 102ni191-7/2010 „Investigation of adaptive methods for QoS in communication networks".

References

[1] G. Booch, I. Jacobson, J. Rumbaugh, "UML Distilled Second Edition", A Brief Guide to the Standard Object Modeling Language, 1999.
[2] Y. Zhang, "WiMAX Network Planning and Optimization - Wireless Networks and Mobile Communications" CRC Press Taylor & Francis Group an Auerbach Book, 2009.
[3] Т. Николов, А. Ценов, „Моделиране на Телекомуникационни Процеси и Системи" София 2007.
[4] http://en.wikipedia.org/wiki/Sequence_diagram