HAL open science

Performances and quality of service of PLC networks for MV and LV distribution systemsS

Liping Lu

▶ To cite this version:

Liping Lu. Performances and quality of service of PLC networks for MV and LV distribution systemsS. Networking and Internet Architecture [cs.NI]. Institut National Polytechnique de Lorraine - INPL, 2006. English. NNT : 2006INPL080N . tel-01752773v2

HAL Id: tel-01752773
https://theses.hal.science/tel-01752773v2

Submitted on 18 Dec 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Wuhan University of Technology
Institut National Polytechnique de Lorraine

Département de formation doctorale en informatique **École doctorale IAEM Lorraine**

Performances and Quality of Service of PLC Networks for MV and LV distribution systems

Thesis

22 November 2006

pour l'obtention du

Doctorat de l'Institut National Polytechnique de Lorraine
(spécialité informatique)

par

LU LiPing

Composition du jury

Christian FRABOUL	Prof. INPT/ENSEEIHT (Président et Rapporteur)
Ken CHEN	Prof. Université Paris 13 (Rapporteur)
Françoise Simonot-Lion	Prof. INPL (Directeur)
YeQiong SONG	Prof. INPL (Co-directeur)
GangYan LI	Prof. Wuhan University of Technology (Directeur)
ChungShue CHEN	PhD The Chinese University of Hong Kong (Examinateur)
ChunNian ZENG	Prof. Wuhan University of Technology (Examinateur)
Zhi WANG	Associate Prof. Zhejiang University (Examinateur)

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Acknowledgements

I want to express my sincere acknowledgment to all those who have encouraged me, with their support and suggestions, all the long way of my PhD study.
First of all, I would like to express my gratitude to my advisors, Professor YeQiong Song, Professor Françoise Simonot-Lion and Professor GangYan Li, for their guidance and supports over the past four years. It is my fortune and honor to have them as my advisors. They are always so nice and patient, discussing technical details with me, encouraging me while I was confused or frustrated.
All my gratitude to Professor Ken Chen and Professor Christian Fraboul for the time they spent reading and commenting my thesis, and for the report they wrote. I am also thankful to the members of the dissertation committee, Professor Zhunnian Zeng, Zhi Wang and Dr. ChungShue Chen for kindly consenting to serve on the defense committee and for their constructive comments on my dissertation.
I would also like to thank all those who helped, advised, and gave fruitful discussions during my PhD work. Many thanks are to all my colleagues in TRIO equip and Wuhan University of Technology and the members of the REMPLI project. In particular, I also wish to express appreciation to Raul Brito for his support and assistance with the technologies required for implementing this study. It was my pleasure to have worked with them.
Finally, my sincere appreciation and gratitude to my parents for their support and encouragement during the entire period of this thesis. They supported me in every possible way and in every single moment during these years and they were always beside me although they were so many thousand miles away. They gave me strength and courage to continue my effort in difficult times and they always embraced me with love, support and encouragement. The least I can do in recognition to their love is to dedicate this thesis to them.

To my loving family who always encouraged me to pursue my dreams

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Why do we need a new PLC

Efficient energy management (e.g. electric power) above all requires a communication infrastructure for the monitoring and control of energy distribution and consumption. Solutions for remote meter reading and monitoring and control (mainly based on SCADA) using Power Line Communication (PLC) systems have already existed for several years [1]. Nevertheless, the existing PLC systems' data rates are too low to achieve reasonable time resolution for high performance monitoring and control purposes or to enable advanced applications like transporting load profiles (big files of several Kbytes) or ever increasingly demand online trading of energy since the opening of the energy market. In addition, to further achieve the management efficiency, conventionally separated applications like metering, monitoring and control (using SCADA) and substation automation should be merged on a same communication infrastructure to run in tighter interaction mode [2][3].

1.2 Perimeter of our study and its context: REMPLI

The main purpose of this PhD thesis is the design of a new communication system for achieving the above objectives. Most of our work has been carried out as part of REMPLI (Remote Energy Management over Power Lines and Internet) European project. Recall that REMPLI project globally aims at creating a distributed network infrastructure suitable for real-time data collection based on improved PLC systems. Apart from a better control of the energy flow, this infrastructure also lays the foundation for more add-on services, such as pre-payment of energy, opening thus a large cost saving potential for energy suppliers.

More precisely, our work is focused on the PLC protocol design. The REMPLI PLC covers two power distribution segments: Medium Voltage (MV) and Low Voltage (LV). As most applications are based on master-slave communication paradigm (e.g. metering, monitoring, etc.), REMPLI PLC is naturally organized as shown in Fig. 1.1

In REMPLI PLC network, Bridges at the MV/LV transformer stations provide transparent data connection between the MV and LV segments. Each node connects with one or more Remote Terminal Units (RTU), such as switchgears and meters. For each of

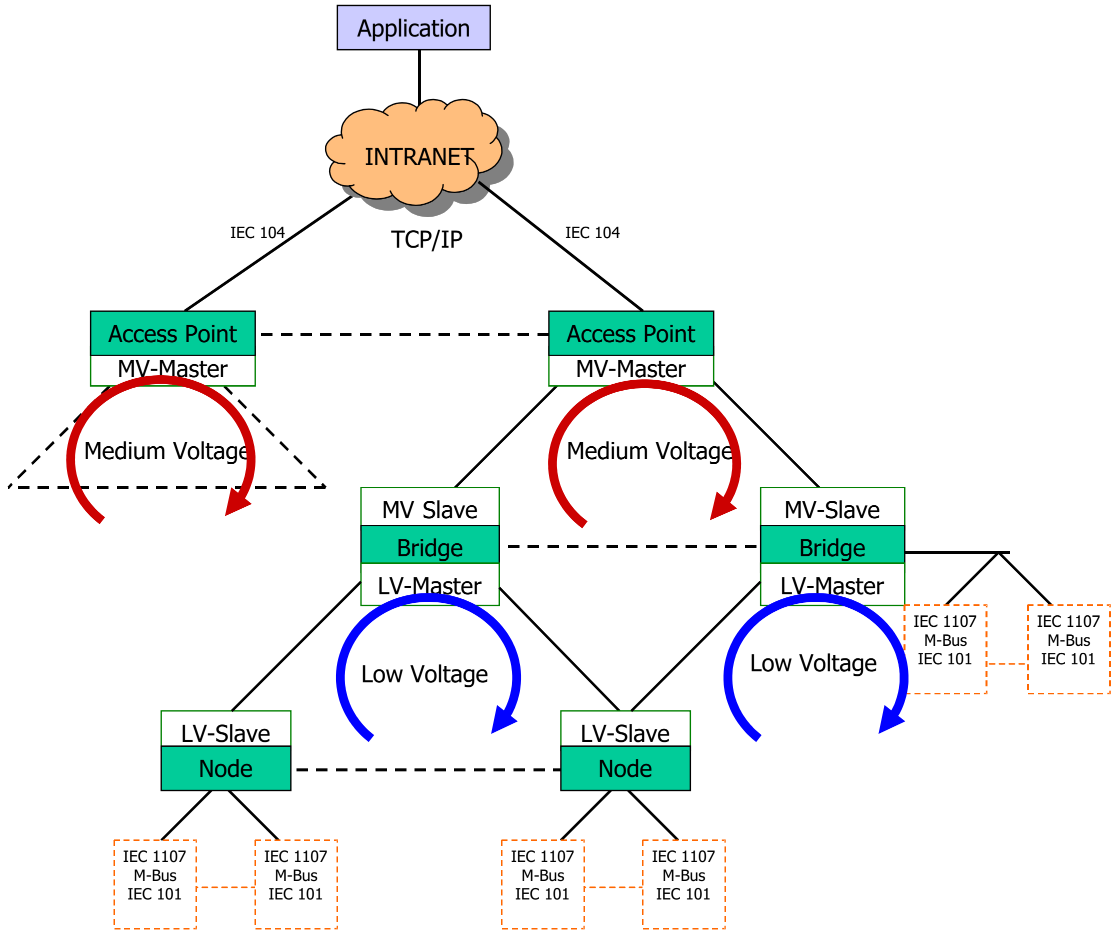


Figure 1.1: REMPLI PLC hierarchy

the voltage segments, the medium access is controlled using the Time Division Multiple Access (TDMA) Master/Slave communication model. The Access Points and the Bridges are Masters for the MV and LV segments, respectively. The Bridges and Nodes are Slaves for the MV and LV segments, respectively. In master/slave model, only master can initiate packet transfer. A slave can only send a packet back to the master upon its master polls it. So in the whole REMPLI network there are as many independent polls as the masters, as shown in Fig. 1.1.

1.3 Research problems

As REMPLI PLC must cover wide area communication need, existing broadband technologies recently developed for in-home applications cannot be used for their shot coverage[4][5]. Protocols such as PLAN (IEC 61334-5-1) [70] has been designed for Wide Area Network (WAN), but they suffer from low bit rates and cannot meet the new application requirements as explained above. So our main research problem is to design a high-speed wide

area PLC network. In our work, we assume that a reasonable high-speed physical layer protocol is available (which uses Orthogonal Frequency Division Multiplexing (OFDM) and efficient Forward Error Correction (FEC), and is provided by iAd (http://www.iad-de.com/) of REMPLI consortium [6]). This protocol provides TDMA channels (slots) as well as the synchronization of the nodes. At the MAC layer, we also dispose a master-slave medium access control protocol in which master assigns the time slots to slaves.
The **first difficulty** is related to the wide area (typically a district of a city or event the whole city) that REMPL PLC must cover. As the master cannot reach directly all its slaves (limited by the signal propagation region due to the emission power limitation imposed by Europe via European Committee for Electrotechnical Standardization (CENELEC) standard on Electro-Magnetic Interference (EMI) pollution [7]), relay is needed for repeating the message. In our study, we assume that in a power distribution grid, a master can always reach any node via intermediate nodes (otherwise, additional intermediate nodes can be added at the network configuration phase). However, due to the fact that electricity grid does not provide a stable topology (dynamic switch on/off of certain circuits, etc.) and the signal suffers strong EMI, the path from the master to a slave is not stable and presents strong time varying behavior and high transmission error rate. With retransmission after transmission errors, it is hard to predict the transmission time from the master to a slave. This arises the *difficulty to efficiently estimate the time needed to poll a slave.*
The **second problem** is related to the random delay the slave node has for getting a response from the attached equipments (e.g. meters). In fact, in a request-response communication scheme based on master-slave model, the master hopes to get slave's response immediately upon a poll. However, considering the system described in Fig. 1.1, the response data are not always available at the slave node. In fact, a slave node may need to get the master request data from devices via the field communication systems (could be M-BUS, IEC1107, IEC101) before to respond to the master request. This delay in general is not constant. Compared to the first problem, this is an application related constraint but not the one on the PLC. However, if a master request cannot immediately get its response, it must re-poll the slave node later on. So this problem must be considered whenever the mater's polling table is concerned.
The **third problem** is related to the willing to integrate the previous separate applications (metering, SCADA, ...) into one communication network. Different applications require different quality of service (QoS), REMPLI PLC must be able to provide differentiated QoS. To our best knowledge, there does not exist wide area PLC networks dealing with this problem. So efficient solution must be proposed, which is not trivial with the presence of the first problem (i.e., time-varying transmission behavior).
In a master-slave model, as only the master can initiate a communication, it may arise the problem of network reactivity when a slave node has urgent messages to send (e.g. alarms). In fact, polling more frequently a slave node to ensure the random alarm transmission may waste too bandwidth. It can be unacceptable when a master has to support an important slave nodes since most of the bandwidth is used to poll slaves for transmitting randomly occurred alarms while in most of time, there is no alarm. The **fourth problem** we address is to find a better solution for efficient alarm reporting.

1.4 Our Approaches, contributions and the organization of the thesis

To address the above-mentioned problems, we develop this document as following.
Chapter 2 gives more details on the applications that our REMPLI network aims to support. We point out, from the analysis of those applications, what are the performance requirements on the network.
Chapter 3 investigates the existing PLC technologies. The focus is put on the technologies applicable to wide area PLC networks. As conclusion, we can see that existing wide area PLC cannot totally satisfy the application needs described in Chapter 2. So we need to develop new PLC for wide area networks.
Chapter 4 addresses the **first problem**: design of an efficient routing protocol. Firstly an outline of the SFN (Single Frequency Network) is given. Then the source routing protocol implemented in DLC1000 of iAd has been analyzed. As the PLC presents a quite limited bandwidth (and a small frame size), appending too many path addresses into the frame header reduces seriously the protocol efficiency. We propose a new dynamic routing protocol based on flooding principle. To increase the protocol efficiency, frame header has been carefully chosen. Two hop counters have been proposed. Probabilistic analysis and simulations showed its advantages comparing with DLC1000 routing protocol. This protocol has been implemented in REMPLI PLC.
Chapter 5 deals with the performance evaluation of the whole REMPLI PLC network. At first, a REMPLI PLC network composed of one master and several slaves has been evaluated. Then a whole REMPLI network composed of MV and LV powerline segments, connected by bridges has been investigated (Fig. 1.1). Those evaluations showed the performance bottleneck caused by the **second problem**, i.e. the obligation to poll two times a slave node to get its response and the difficulty to estimate the right moment to poll the second time the slave (neither too early nor too late). An alternative solution is to store the field level device data with their timing stamps inside the slave node (e.g. a cache) in order that a request gets immediately its response, even if the cached datum is not always the latest one. Its performance evaluation has shown its high efficiency. Unfortunately it has not been adopted by REMPLI project because that the device data are not allowed to be stored outside the device itself in some countries. However, it is a much more efficient solution and most of countries have not such kind of limitation.
Chapter 6 proposes a solution to the **third problem**: a traffic scheduling policy providing differentiated QoS. Its implementation in forms of a network layer dispatcher is detailed. This dispatcher has been implemented into the REMPLI final product.
Chapter 7 investigates the **fourth problem**. In fact, for a given alarm occurrence probability per slave node and knowing the total slave number of a master, a trade-off point between time-triggered (i.e. master polling) and event-triggered (i.e. slave initiated) communications should exist. For enabling a slave to initiate a communication upon alarm, ALOHA protocol as well as its variants has been evaluated. Our study showed the interest to implement an ALOHA based protocol for alarm reporting under a certain alarm occurrence rate.
Chapter 8 concludes our work and points out the future research directions.

In *appendix*, we describe some software tools we developed for the performance evaluation of REMPLI PLC. Those tools are mainly based on MATLAB (for calculating the probabilities of chapter 4) and OPNET (for the simulations of chapters 4 and 5). Using those tools one can easily evaluate its own REMPLI PLC configuration.

Chapter 2

Electric Power Distribution and Consumption

In this chapter, we present briefly the structure of power electric distribution systems and the major existing applications used for their control and handling. We analyze the set of requirements of these applications and, in particular those that are taken into account by the communication network. As the Power Line Communication (PLC) technology appears as promising in this context, we introduce it and define the scope of our work.

The process of restructuring, privatization, and deregulation has created a competitive global marketplace for energy. Electric utilities, all over the world, have restructured the vertically integrated electric power systems and moved towards unbundled model of companies. In the past, all electric power distribution-related functions could be transparently coordinated along the whole supply chain. In the future, many distribution companies will manage third-party contacts by delivering bulk power from generation or transmission companies to meters owned by energy service companies [8][9]. In today's deregulated marketplace, simply delivering power to customers is no longer sufficient; utilities must provide a certain level of quality of service and must deliver the commodity safely, reliably and cost-effectively [10] [11]. Moreover, in addition to planning and operating difficulties, retail wheeling (small generators connected to the distribution system selling electricity directly to customers) asks distribution systems to perform the functions for which they were not designed. Restructuring of the power industry, changing expectation of the customers of the digital age, and advancements in technology will gradually impact distribution systems. The technologies that would have the biggest impact are distribution automation, power electronics, distributed energy resources, and Distribution Management Systems (DMS), and Distribution Control Centers (DCC)[8][12][13].
In view of the above, on-line information, remote control and efficient management systems are required for electric power distribution utilities. However, they relies heavily on efficient and economic communication networks [14].

2.1 Power electric distribution system

A power system is conventionally divided into transmission and distribution systems, which are distinguished by their voltage levels and network structures. The transmission system is generally composed of power plants and transmission lines ranging above 110 kV. The distribution system controls the distribution of power from the transformers in substations to transformers at customer sides, with voltages ranging below 35 kV. Fig.2.1 shows the typical parts of the electrical power system in a schematic single-line diagram [15].

Our focus is on the services that support the communication of data for the distribution system for medium and low voltages (in grey in Fig.2.1).

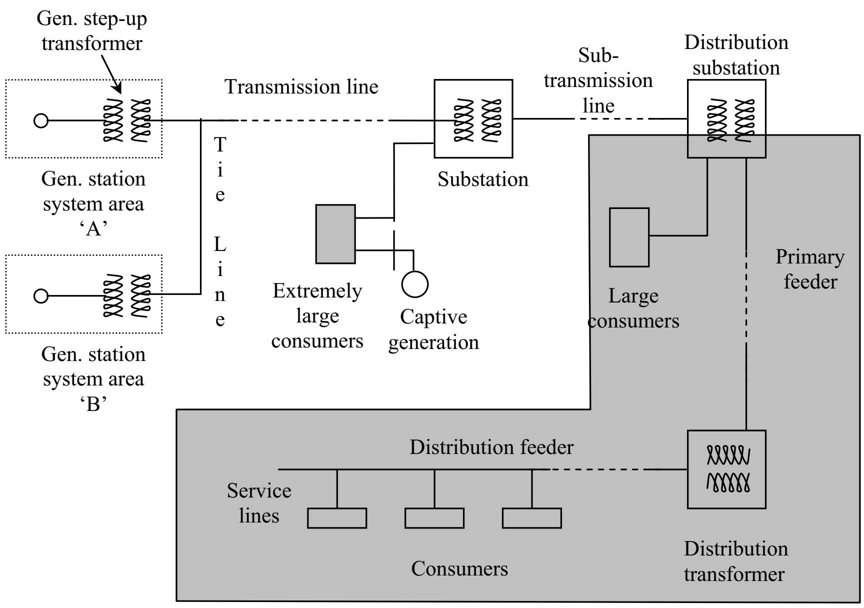


Figure 2.1: Schematic diagram of electric power distribution system

All low voltage distribution systems are operated radially, while two base topologies, radial (tree structure) and open loop, may exist in middle voltage ones. A single supply line is used in a radial layout in order to enable easy and low-cost supply to low load density consumer units with a wide geographical dispersion. Two lines of supply are used in an open loop layout. This is also called open ring layout. Any consumer unit on this structure can be supplied via two possible electrical paths. Only one of them is activated at any one time, while the other one serves as a back-up. In such a layout, there is always an open point in the loop, which leads to a similar operation to two radial layouts. An example integrating both solutions is presented in Fig. 2.2.

In this thesis, the main problem that we have taken into account at the data communication system level is the dynamic topology of the electric power distribution system.

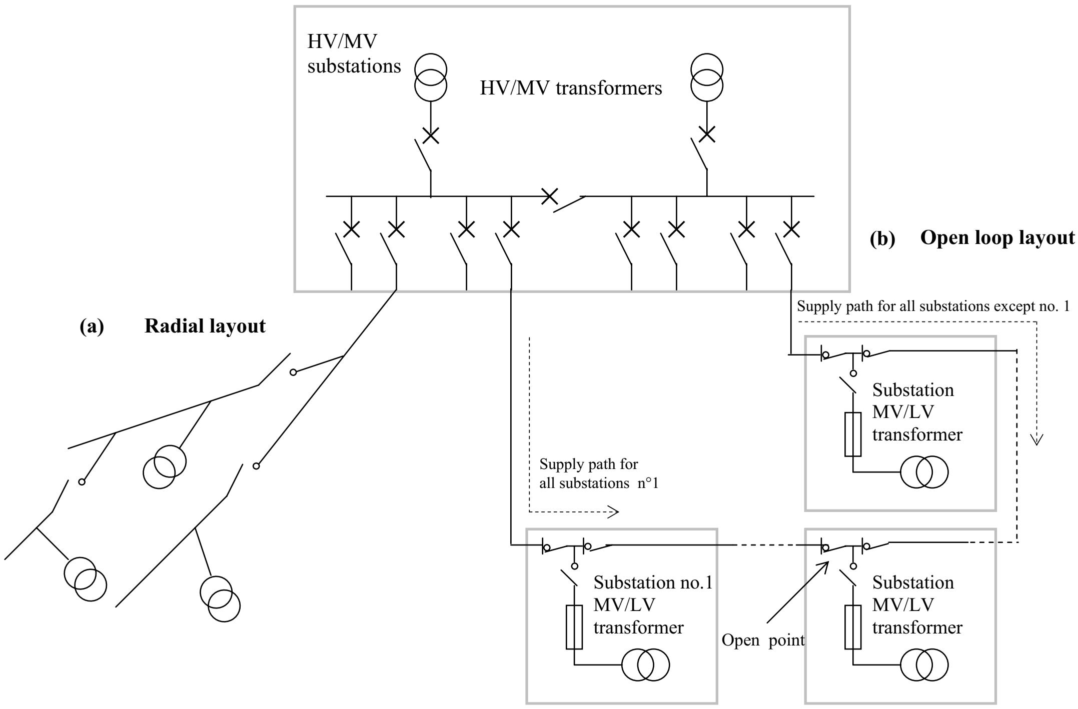


Figure 2.2: Two basic layouts of a middle voltage distribution system, (a)radial and (b)open loop (HV:High Voltage, MV:Medium Voltage, LV:Low Voltage)

2.2 Main applications for the control and handling of electric power distribution

One goal for distribution systems is to accomplish higher reliability and quality while optimizing the resources. Another goal should be improvement in system efficiency by reducing system losses. The evolutionary growth in microprocessor based devices and telecommunication equipments and networks have brought the possibility of integrating protection, control, metering, automation and monitoring systems cost-effectively.
The first kind of application concerns the control and the supervision functionalities. Supervisory Control and Data Acquisition (SCADA) systems and Distribution Automation (DA) systems are representative of this category. A second area for application is directly related to customer-related functions, such as remote load control, AMR(Automated Meter Reading), and remote connect/disconnect. Finally, a common trend aims to integrate

several platforms across many utility departments (standard mapping systems, customer information systems, work management systems, etc.).
The future trends are to shift from a very limited control of load and demand response to utilities that should provide a certain level of Quality of Service (QoS) and should deliver the electricity safely, reliably and cost-effectively. In order to supply a high-quality electricity product, the utility must reduce the number and the length of interruptions in supply to customers, minimize the consequences of them and avoid disruptions such as voltage and frequency fluctuations. Customers will have to participate in the energy and reliability services market. In the following, we present some functions that should be implemented in the future power electric distribution system [16] [18][17][19] [20]:

Building self-healing distribution system

In the development of tomorrow's electric infrastructure, a self-healing transmission and distribution system that is capable of automatically anticipating and responding to disturbances while continually optimizing its own performance, will be critical for meeting the future electricity needs of an increasingly transactional society.

Some functions for the open electricity market

For instance, real-time pricing (RTP) system provides customers with dynamic pricing information of electric power. The objective of this dynamic pricing scheme is to encourage customers to perform load management strategies to lower their electric demands during high price periods.

Integrated Distributed Energy Resources (DER)

Distributed generation (DG) is able to encompass any small-scale electricity generation technologies that provide electric power at a site close to consumers. Proponents of DG further suggest that it will enable utilities to expand into new markets, minimize investment in existing infrastructure, increase flexibility, increase reliability and power quality, optimize asset utilization, and reduce the overall cost of providing power to end users. These trends will increase the number of Intelligent Electric Devices (IEDs) and Remote Terminal Units (RTUs) associated with the operation and control of DG units and their interconnection with the distribution system.

Value-added electricity services

Electric utilities will be able to expand the portfolio of their business services to include communication, Internet access, real-time online monitoring, and other associated service. Customers will gain real-time access to energy markets and thus control energy cost and energy utilization better. All of these are based on the integration of electricity with communication system.
The implementation of these applications has to ensure reliability and availability of the electric power distribution system. These properties that are expressed at a functional level must be guaranteed by the implementation of these applications. The reliability and dependability purpose has to be translated, on the one hand, by equivalent reliability and dependability properties imposed on each devices and the architecture and, on the other hand, by timing properties on data exchanges and function execution. These properties can therefore be expressed as schedulability properties on each CPU, end-to-end response time, data freshness, etc. Some of these properties required by applications are given in Section 2.3. More specifically, if we focus on the data exchanged within one application or between several applications, we have to take into account the communication archi-

tecture between computers supporting these applications and to evaluate the network technology for ensuring these exchanges in a reliable way.

> Our work aims to provide a method that contributes to ensure the timing properties of a control system for electric power distribution by providing some services at the communication level and the way to evaluate their performances and by taking into account the evolution at middle and long terms of such systems.

2.3 System requirements

The incorporation of high speed Ethernet (100 Mbps -1 Gbps) as a future communication backbone both inside a substation and between substations and control centers appears to be an accepted fact [15]. Moreover, the modern distribution system demands a high-speed, real-time communication for the satisfaction of the requirements of employing a wide variety of different types of equipments which are located along the feeder, often on the pole tops. For example, for power quality of service, fault passage indicators (FPIs) are installed on existing pole mounted or substation switchgear. The extension of computer monitoring and control to customer level has become a new trend. And the future utility communication network will include not only the system automation, but also the business information, billing, customer service, and voice applications.

2.3.1 Functions

There are a certain number of applications that use the communication network to support the application services. We divide those application services into following functions:
Control function
The control functions are related to switching operations, such as switching a capacitor, or reconfiguring feeders. There must be the maximum guarantee on the certainty concerning remote control commands. This is obtained by using a powerful communications network that allows access to information in real time. Thus a switching command for a MV device is transmitted and confirmed by the feedback of a confirmation signal. Remote control processes also integrate requests for validation and confirmation before execution of a switching command.
Monitoring function
Monitoring functions are those needed to record meter readings and the system status at different locations in the system, and events of abnormal conditions. Remote measurements can be obtained by several different ways, such as cyclic polling (periodic), polling on demand, or threshold trespassing (event-driven). There are some differences between the type of measures sent to Distribution Control Central(DCC) and the frequency of the polling.
Metering function

AMR application has evolved significantly as a separate area. So we distinguish AMR function from metering reading in monitoring function. Metering data take from various meters which could be watt-hour meters in home, or in transformer, etc. Metering data with time stamp are useful for utilities electricity distribution, consumption, and developing revenue billing, load management, energy loss detection, tariff management, etc. The customer meter reading collection is done by the application servers with periodic collection intervals range from about one second to one or more hours, even monthly. And the customer meter reading requested is an event-driven or human-driven demand to get certain customer meter data.

File function

This service supports the transmission of a large amount of data (in the order of megabytes) through the network. The RTU/IED stores data to provide information about the reaction of the power system over time or to an event [22]. These collections include system profiles, event reports, Sequential Event Recorder (SER) reports, power quality reports, and protection quality reports. Report generation is triggered automatically by system disturbances, other events, RTU program or an operator. Some system values are captured and archived periodically to enable trending analysis. Furthermore, this service permits, for example, the upload of new versions of software for RTU/IED. For this type of service, the transmission correction is more important than the transferring time.

2.3.2 Timing constraints

In fact, the information of timing constraints of applications are difficult to obtain, because the productions of different companies have different timing performances and confidential limits. So, the following time constraints are mainly coming from [21] [23]. We notice that current control and monitoring functions are applicable in the MV grids. In [21], it mainly presents the timing constraints of message delivery times between IEDs, for applications of substation automation. We are only interested in the outsider substation communication requirement which is a part of distribution automation application. [23] gives out the requirements for hybrid DA/DSM communication system.

Timing constraints are specified for response time. Response time is specified in terms of the time when the message leaves the sending application to the time when the receiving application gets the message. Fig. 2.3 shows time components that define the time requirement. Application-to-application time is defined as the sum of the times required for the sending IED communication processor to accept the data from the sending application "f1", and exit the output queue of the sender "a", plus time over the communication network "b" (includes the processing time required by routers, bridges, gateways, etc), plus the time "c" required for the receiving IED communication processor to extract the message content and present it to the receiving application "f2". For giving the performance constraints of the communication network, we assume that the processor time is a very short duration and negligible which is not necessary to involve into the value "a", "b", "c".

Control function

Table 2.1 shows an example of communication performance requirements needed to support the distribution automation application [21]. Values in the column titled "Maximum

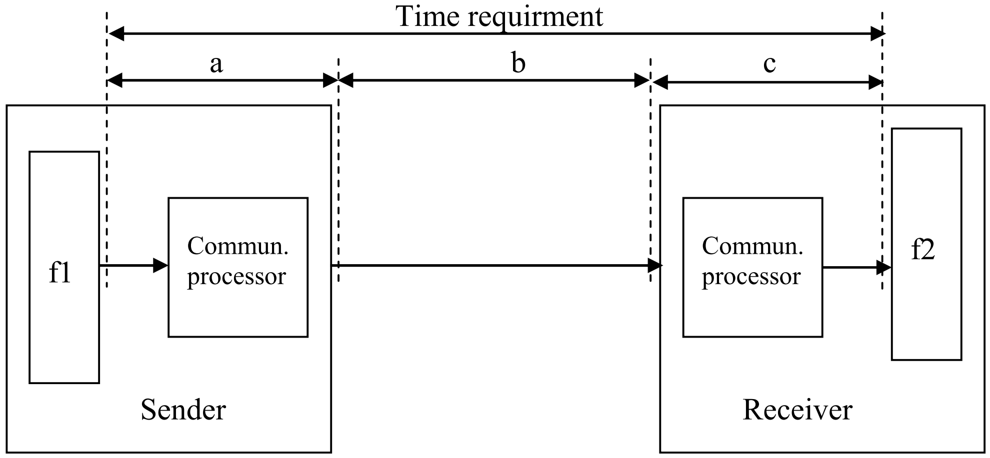


Figure 2.3: Application-to-Application communication times

Delivery Time" define a range, or the qualifier "up to", which means that the maximum delivery time is determined by specific implementations and operating constraints for the application. We conclude that the most application communication time constraints are greater than 1 second; only two of them have very small transmission time. From [23], it simply limits the response time to command to less than approximately 10 seconds. Obviously, control function has hard real-time requirement.

Monitoring function

Required periodic update intervals range from about one second to one or more hours with required data delivery times that are less than the update intervals. Occasional data errors and late or missing data may be acceptable when the data user can recover by the receipt of the next routine update. Generally, the timing constraints are that the periodic polling can be done within the polling cycle.

The monitoring function can also get the aperiodic information, such as alarm from the field devices to the control centre. In [23], it limits the alarm delay to less than approximately 30 seconds, where alarm delay is the time from the alarm generated by feeder equipments to the time when this alarm information gets back to control centre.

Metering function

It is mainly concerned on the LV customer. Most significantly, the real-time requirements are much less rigorous. On the other hand, the amount of data that needs to be sent to a central point may be more than an order of magnitude higher than the data required for other applications, such as DA. It is not necessary to have the data available in real-time, but the metering reading collection should be finished within the cycle. Although the applications do not have strict timing constraints, the customer meter reading requested should be finished within a reasonable time (a few seconds). In [23], it limits the response time to an individual read to less than approximately 30 seconds.

Data/Application	Maximum Delivery Time	Note
a. Line Sectionalizing	5 s	
b. Load control and load shedding	10 s	Non-underfrequency condition
c. Load shedding for underfrequency	10 ms 10 ms	Triggered by underfrequency relay
d. Fault identification, isolation and service Restoration	10 s	reporting function
e. Fault isolation service restoration	Several minutes	
f. Transfer switching	24 ms	
g. VAR dispatch	1 s	
h. Voltage dispatch	1 s	
i. SCADA–stand-alone or distributed	1 s	

Table 2.1: Distribution automation communication performance requirements

2.4 Communication networks

The reliable and economic operation of power electrical distribution systems relies heavily on its efficient communications system. There are many communication methods available. As aforementioned, the high-speed Ethernet is applied widely. Herein, we focus on Wide Area Network (WAN) between DCC and RTUs/IEDs distributed widely along the feeder. The fundamental requirements for communication infrastructure [21] [24] [25] are given below.

- The requisite amount of data and multitasking can be handled.
- Data throughput and system response times should meet various application requirements.
- A reliable (i.e. acknowledged) transmission within a bounded time should be provided.
- There should be support for priorities to allow distinction between urgent and non-urgent message;
- Both periodic and aperiodic (asynchronous) traffic types should be supported.
- Regular topology changes on the distribution system should be adapted.

- A long distance communication should be supported.
- It should allow for network growth and added applications.

The communication media can either be wired (cable, fiber, telephone) or wireless (wireless local area network(802.11), radio etc.) [26] [27].

Radio

In radio frequency communication, the radio frequencies in the range of VHF (Very High Frequency), UHF (UltraHigh Frequency), EHF (Extremely high frequency) were used for remote telemetering and SCADA applications on a shared basis with voice traffic. Point to point and point to multipoint analogue and digital radio links are evolved with Network Management System (NMS) and diagnostic features. The disadvantages of such a solution include spectrum congestion, high installation cost, limited channel capacity per licensed transmitter, requirment of system design to allow path reflection, refraction and absorption, etc.

Satellite

Satellite communication networks that are capable of providing economical and reliable transmission of information including voice, video and data can virtually interconnect the whole world from the busiest urban centre to the most remote islands. However, for the present, there exists no small and low cost equipments. The deployment and use of this technology is expensive. Moreover, it involves an inherent intolerable channel delay.

Telephone line

Leased telephone lines have the merit of involving simple technology with one pair for simplex or semi-duplex, voice or low speed data transmission. Each communication terminals have modems so there is no need for. Nevertheless, the response time is too long. There is no possibility of control on the communication media and no flexibility of the network configuration.

Private metallic cables

Cable installations for applications such as SCADA involving canal structure and pipeline control for water systems typically are buried along the canal right-of-way. Cables can be tied to the conductors on the transmission line structures or on telephone poles. Cables must be protected against induced voltage surges or lightning strikes by protective devices or from huge differences in ground mat potential during fault current conditions in high voltage substations. Such installation is costly. Furthermore, the network configuration is also not flexible.

Fiber optic cable

A significant advantage of fiber optic cable systems is the availability of a large amount of bandwidth. Nevertheless, its installation would be costly for the targeted applications and in this case, the network configuration is also inflexible.

Power line

Power Line was one of the first reliable communications media available to electrical communication channels that could not be subjected to intolerance and unreliability of leased telephone circuits. It locates where the circuits are required and equipments are installed in utility owned land or structures[28].

Analogue PLC (Power Line Communication) uses the power transmission lines to transmit radio frequency signals in the range of 30 kHz to 500 kHz. It is used to provide voice,

telemetry, SCADA, and relaying communications on some portions of the 220/230 kV, 110/115 kV, or 66 kV interconnected power transmission networks [29] [30].
Digital PLC technology is a relatively new technology. The implementation of complex digital modulation schemes makes it possible to use the frequency range more efficiently and to recover the signal at the receiver and provides a high transmission speed. New digital PLC systems offer the possibility for reliable bi-directional data transfer form a HV/MV substation to any customer [31].

> Powerline can be a candidate for the new requirements of communication system for distribution system (MV and LV). So the purpose of our studies is to propose new services for ensuring timing properties in a PLC-based architecture.

2.5 Conclusion

In the deregulated market environment it is becoming more important to bring information from the field devices for faster processing and making a quick decision.
Because the investments of fiber optic cable and satellite are very high, it is impossible to extend such communication networks to the wide area distributed feeder equipments and enormous end customers. Moreover, the telephone line with the limited bandwidth is not suitable too. So the existing communication system that many utilities have for distribution automation is based on an 800 or 900 MHz wireless system, but the number of channels available for future communication needs is limited and there are some difficulties to access to certain electrical equipments using wireless technology since they are often located in closed environments with metallic obstacles such as reinforced concrete walls and tubes.
With the development of the digital PLC technology, it is ideal to use the power line as a communication medium to construct an economic, secure and reliable communication system for communicating from the distribution substation authority to a number of field components and to end users. It has at least the following advantages: no new wires are required, resulting in low cabling cost; reliable and high transmission speed PLC chips are now available to provide an efficient communication; power line is owned by the distribution utility for ensuring a certain level of security.

Chapter 3

Power Line Communications - Technologies and Problems: Requirements for their Control and Open Issues

This chapter introduces PLC technologies including historical development, current PLC technologies and some existing applications. Then the advantages of applying PLC technologies for utilities applications are analyzed and the problems of PLC networks in utilities applications are pointed out. Finally, the architecture and main objective of REMPLI project is presented.

The abbreviation PLC derives from the original term **P**ower **L**ine **C**arrier. In recent years, it has also come to stand for **P**ower **L**ine **C**ommunication. The basic concept of PLC is to transmit information and electricity simultaneously along electricity lines as an alternative to constructing dedicated communications infrastructure. The electrical power distribution system represents a perfect local area network, present everywhere and immediately usable. Recently, the technology has been enhanced to be capable of supporting high-speed transmission of data, voice, and video services. PLC network has encouraged the deployment of many trial and commercial systems around the world.

3.1 Historical development of data communication over power line

Power utilities began to be used additionally for data transmission for operations management and optimum energy distribution soon after full-coverage electrification. Carrier Transmission over power lines (CTP) is used on the High Voltage (HV) power line, and provides a long distance and bi-directional transmission with low transmit power. In contrast, Ripple Carrier Signalling (RCS) on Medium Voltage (MV) and Low Voltage (LV) power line allows only a very low bit rate and unidirectional transmission with enormous transmit powers[4].

These PLC systems aimed mainly for the distributors' own requirements and they were not publicly available. PLC systems on HV power line are used for protection signaling and voice and data communications. A typical PLC communication network consists of one or multiple point-to-point links that can cover a distance of hundreds of kilometers without repeaters. To avoid interference between different links, they are typically separated by line traps. The voice services are still in use today, despite that reliable radio transmitters are now available.
Also the utilities have used RCS systems for many years, mainly to address load distribution, i.e. the avoidance of extreme load peaks and the smoothing of the load curve. RCS works at low frequencies near the power frequency. Low frequencies used as carriers for the packets allow information to flow over the transformers between the MV and LV segments without particular and generally costly coupling measures. Despite that data rate is low and less than 120 bps, it is sufficient for the tasks involved in load distribution, because often only enabling or disabling commands have to be issued.
In the past, only the supply utilities could make use of PLC as the communication method for their own purposes. Recently, the situation has changed fundamentally.

3.2 Regulatory of PLC communication

The PLC technology as telecom services over the electric grid makes regulation both in the energy and the telecommunication sectors relevant for PLC development. In addition, PLC should comply with EMC (Electro Magnetic Compatibility) regulations for sharing the same frequency bands with wireless services.
In Europe the allowed bandwidth is regulated by the CENELEC standard (EN50065-1, Signaling on low-voltage electrical installations in the frequency range 3 kHz to 148.5 kHz). The standard only allows frequencies between 3 kHz and 148.5 kHz. Fig. 3.1 shows the bandwidth specified by the CENELEC standard. The frequency range is divided into five sub-bands.

- 3-9 kHz for use by energy providers
- 9-95 kHz (A-band) for use of energy providers
- 95-125 kHz (B-band) for use of consumers
- 125-140 kHz (C-band) for use of consumers, CSMA protocol defined
- 140-148.5 kHz (D-band) for use of consumers

The EN 50065 limits the signal amplitude. This puts a hard restriction on power-line communications, and the maximum data rate is up to 144 kb/s over distances around 500m. It might not be enough to support high speed rate applications, such as real-time video.
In addition, EN 50065 differs considerably from other regulations, e.g., those applicable in the United States or Japan, where a frequency spectrum of approximately 500 kHz

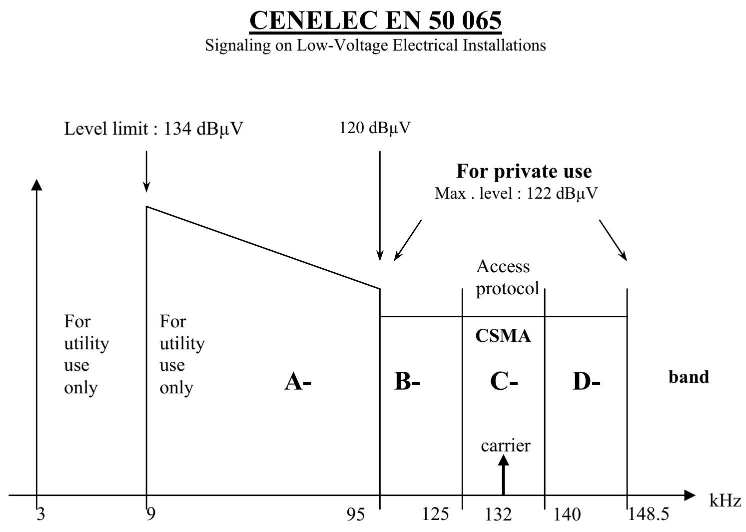


Figure 3.1: Frequency ranges and PLC signal level limits specified in EN 50065

is available. In the United States the use of PLC is regulated by FCC Part 15 (Code of Federal Regulations Title 47 Telecommunication: Chapter 1 FCC Part 15–Radio Frequency Devices), which distinguishes between low-speed applications for signalling and switching purposes and high-speed data transmissions. Low-speed systems are allowed to operate at frequencies below 490 kHz. For high-speed PLC, FCC Part 15 can be regarded as highly generous and no obstructing, since it allows power line communication outside 533 to 1705 kHz[32]. Signal injection between neutral conductor and protective ground is admissible in these countries[33]. Due to the differences in standards, it is not possible to buy PLC systems in the United States or Japan and use them in Europe.
Efforts are going on in the United States through the Electronics Industry Association (EIA), Institute of Electrical and Electronics Engineers (IEEE), and Automatic Meter Reading Association (AMRA) Committee SCC31, and in Europe via European Committee for Electrotechnical Standardization (CENELEC), to develop new EMC standards for PLC systems from 2 MHz up to 30 MHz. [5]

3.3 Present PLC technologies and applications

Deregulation of the telecommunications and energy markets was initiated in 1998. Power utilities have to adapt the future competition in the electric power market, and they want to open up new business fields with growth potential in the deregulated telecommunications market. Those new applications focus on the MV and LV grids. However, in contrast to overhead lines on the HV power line, MV and LV power lines are hostile communication media. The principal property of the channel is that the noise and attenuation has time and frequency varying [34] [35]. However, modern modulation and coding techniques allow using PLC as a high speed, robust and reliable transmission media. With the specification of well-suited and efficient services, in particular a suitable medium access control (MAC) algorithm, a PLC network can provide data, voice and video services.

3.3.1 Physical layer

Power line is originally devised for distribution of power at 50-60 Hz. The use of this medium for communications at higher frequencies presents some technically challenging problems. Power line networks are usually made of a variety of conductor types, joined almost at random, and terminating into loads of varying impedance. Such a network has an amplitude and phase response that varies widely with frequency. At some frequencies, the signal may arrive at the receiver with relatively very small loss, while other frequencies may be driven below the noise floor. Furthermore, the channel characteristics can also vary with time when the load on the network changes [36][37].
Power line networks are also affected by interference. Electric appliances with brush motors, switching power supplies and halogen lamps produce impulse noise that can reduce the reliability of communication signals. Due to high attenuation over the power line, the noise is also location-dependent. Apart from these, ingress sources such as amateur radio transmission can render certain frequencies unfit for communications [38].
Due to the above mentioned power line channel characteristics, it is necessary to carefully select modulation schemes and sophisticated error correction and detection technologies for forming up reliable physical layers as a basis for robust power line communications.
In [4], a comparison of the major modulation schemes in PLC is reported in Table 3.1 and they include:

- Spread-spectrum modulation, particularly Direct Sequence Spread Spectrum (DSSS)
- Broadband single-carrier modulation without equalizing
- Broadband single-carrier modulation with broadband equalizing
- Broadband multicarrier modulation with adaptive decision feedback equalizing
- Multicarrier modulation in the form of "Orthogonal Frequency Division Multiplexing" (OFDM)

From Table 3.1, it is clear that most of PLC systems use OFDM-like modulation scheme. OFDM, also known as Multi-carrier Modulation (MCM) or Discrete Multitone (DMT), is a modulation technique which combines excellent bandwidth efficiency (high data rates) with the possibility of a very flexible bandwidth allocation. In a combination with error correction coding, OFDM is very robust in the presence of narrowband interferers, impulsive noise, and frequency selective attenuation, as typically seen on power lines. The main advantage of OFDM is obtained by the fact that the channel bandwidth is divided into a number of sub-channels. In each sub-channel, a carrier is modulated at a much lower data-rate. A multicarrier modulation scheme can be viewed as consisting

	Spectral efficiency	Max. data rate in Mbits/s	Robustness against channel distortions	Robustness against impulsive noise	Flexibility and adaptive features	System costs(incl. equalizers) and repeaters)	EMC aspects regulation
Spread spectrum techniques	<0.1 bits/s/Hz	≈ 0.5	-	0	–	–	++
Single-carrier broadband, no equalizer	1-2 bits/s/Hz	<1	–	+	–	++	–
Single-carrier broadband with equalizer	1-2 bits/s/Hz						
Multicarrier broadband with equalizer	1 - 4 bits/s/Hz	≈ 3	+	0	0	-	0
OFDM	$\gg 1$ bits/s/Hz	>10	++	0	++	-	+

Table 3.1: Comparison of PLC modulation schemes (++ excellent; + good; 0 fair; - bad; – very bad)

of N independently modulated carriers with different carrier frequencies. If the carrier frequencies are selected appropriately, they are orthogonal,and thus they do not interfere with each other [39].
OFDM allows an extremely flexible allocation and use of a given channel bandwidth. For example, the lower and upper limit of the used frequency band can easily be configured. In addition, certain frequencies inside this frequency band can be suppressed, e.g. to prevent interference with other systems. It is also possible to use two or more non-contiguous sub-bands for the transmission of a single data stream.
Each of the carriers can be modulated individually with different modulation schemes, if appropriate. Typical examples of carrier modulation schemes are Frequency Shift Keying (FSK), Phase Shift Keying (PSK), and Quadrature Amplitude Modulation (QAM), with a different number of bits per carrier. With this flexible choice, the available signal-to-noise ratio can be used optimally for each carrier.
OFDM is considerably more robust against InterSymbol Interference (ISI) or group delay distortion caused by the transmission channel than narrowband systems. This is mainly due to the fact that the parallel transmissions on several carriers leads to longer symbol duration. Furthermore, ISI can be eliminated by inserting guard intervals or a cyclic prefix between the symbols.

OFDM is robust in the presence of narrowband interferers, because such jammers typically destroy only a single carrier. With proper Forward Error Correction (FEC) coding, the destroyed bits can be reconstructed.
Trails in a number of power utilities have been in progress for more than a year, and excellent results have been observed, with regard to both the bit error rate and the availability of all modems used in a complete network supplied by a transformer station (up to 400 households). Not a single total connection failure was observed, even in extreme situations.
So we conclude that a combination of OFDM and sophisticated FEC, interleaving, error detection, and Automatic Repeat Request (ARQ) can ensure that the channel appears completely reliable to the network layer protocols [40] [41] [42]. With those technologies, the high-speed PLC chips have been developed. The main chip vendors and maximum data rate of productions are shown in Table 3.2.

Chip vendor	Data rate(max)
HomePlug Power line Alliance http://www.homeplug.org	14 Mb/s (HomePlug 1.0) (100-200Mb/s in development)
DS2 http://www.de2.es	27 Mb/s downstream 18 Mb/s upstream (will converge with HomePlug) new products with 200 Mb/s
ITRAN Communications Ltd. http://www.itrancomm.com	2.5 Mb/s (ITM1) 24 Mb/s (ITM10)
nSine Communications Limited http://www.nsine.com	2.5 Mb/s (10 Mb/s in development)
Easyplug http://www.easyplug.com	4 Mb/s

Table 3.2: Main high-speed power line technologies-chip vendors

3.3.2 MAC layer

A MAC protocol specifies a resource sharing strategy applied to a multiple access scheme. It accommodates multiple users in the sharing of the network transmission capacity. Various channel access algorithms have been demonstrated for dedicated wiring. The algorithms are generally based on either a carrier sense technique or token passing mechanism. However, when those algorithms are not transferable to PLC network, the more attention should be paid to design of MAC protocol for the PLC technology because of the following reasons [43].

- "Carrier Sensing problem": on the power line there is insufficient communications reliability to distinguish between noise and signal. This makes carrier sense difficult.

- "Hidden-node problem": since the power line characteristics can be remarkably different for each node, there is a high probability that a node will not necessarily listen to all the transmissions on the power line. In carrier sense, a node may thus incorrectly sense that the channel is quiet and start transmitting in the middle of another transmission.

The above two problems make carrier sense multiple access with collision detection (CSMA/CD) hard to be implemented for the power line environment. So, other techniques have been explored. Among these techniques, we identified three main classes:

- Because in-home PLC environments are similar to wireless environments, the ***Carrier Sense Multiple Access with Collision Avoidance*** (CSMA/CA), which is IEEE 802.11 wireless standard protocol, is used as the MAC protocol in many PLC home networks. Therefore, a high gross data rate on the medium is necessary to ensure a sufficient QoS and to make PLC systems competitive to the other technologies.

- CSMA based protocols cannot guarantee any Quality of Service (QoS) for time-critical services. And 100% network utilization cannot be reached because of the potential collision. ***Collision-free dynamic protocols*** such as ***token passing and polling*** make possible guarantees on some QoS parameters [43]. These two techniques do not require collision detection. Only the master or the node which holds the token can transmit data over the medium. Furthermore, their implementation in the power line environment is much easier and reliable. However, with an increasing number of network stations, the time between two sending rights for a stations (round-trip time of tokens or polling messages) becomes longer. This makes both protocols not suitable for time-critical services [44].

- At present, a third category of protocols, ***reservation protocols***, is proposed to satisfy the QoS requirements of time-critical applications. In that case, a kind of pre-reservation of the transmission capacity for a particular user is done [45]. A transmission request is submitted by user to a central network unit (e.g. transformer station in PLC network) using either a fixed or a dynamic access scheme. Transmission systems with the reservation access scheme are suitable to carry hybrid traffic (mix of traffic types caused by various services) with variable transmission rates. Satisfaction of various QoS requirements is also possible and good network utilization can be reached. Some kinds of reservation protocol is combined with a TDMA protocol and CSMA/CA protocol. In [46], it proposes that the total time is divided into beacon cycles in which it contains a TDMA period and a CSMA/CA period. In the TDMA period, centralized bandwidth control is executed by a master device which notifies each device of a bandwidth schedule via a beacon frame. In the CSMA/CA period, control is distributed and bandwidth is not guaranteed. The

> MAC Layer of HomePlug AV is designed to combine Time Division Multiple Access (TDMA) with CSMA based access with AC line cycle synchronization for providing high-quality, multi-stream and entertainment oriented networking over existing AC wiring within the home [47].

We notice that current MAC protocols with QoS are limited to the in-home applications [48][49][50][51].

3.3.3 Current PLC applications

Although PLC systems for HV grids have been in use for more than 75 years, LV and MV applications have only recently come into focus. However, the purposes of LV and MV PLC are different: LV applications were primarily designed as technologies for end users, either for transferring high-speed data (e.g. Internet access) or for home automation, or AMR [52][53]. MV PLC can function as a means of communication through which power utilities can automate their distribution systems [54][55][41].

Last-mile solution

Data communication over low voltage power distribution system provides an alternative and cost-effective last mile access technology. PLC access networks typically cover both public areas from transformer substations to customer premises (outdoor) and private areas within customer buildings (indoor). The frequency spectrum available for communications is between 1 and 30 MHz. Most of the systems available provide a maximum network data rate of more than several megabits per second. Currently, many field trials with different broadband PLC systems are running worldwide [56][57].

In addition to broadband Internet connections, these broadband PLC systems are intended to also provide voice (IP telephony), video (VHS video quality), and surveillance systems services [58].

In [59], it presents a field trial of PLC communication on MV and LV grids carried out in the United States to demonstrate that the PLC network could be a general-purpose communication network, which utilizes IP as the network protocol. Throughput rates from 10 to 16 Mb/s have been achieved over the MV grids. End-to-end throughput rates on the network, which refers to the rates that could be provided to consumer premises, range from 3 to 7 Mb/s.

In-home network

In-home power line technology communicates data exclusively within the consumer's premises and extends to all of the electrical outlets within the home. The same electrical outlets that provide power will also serve as access points for the network devices. In-home power line technologies have two primary uses within the context of a home network, namely, data networking, and home control.

The most important types of home automation applications include controlling lights, ventilators, security systems, sprinklers and temperature levels within the home. The common key requirement of these systems is the need for data communications connectivity within a building, requiring the frequent transfers of short messages at data rates between 10 and 20 kbps, rather than the transfer of large files at a data rate of megabit per second. Home control and automation systems are normally based on one of the three

major power line technologies: CEBus, LonWorks, or X-10 [1][60].
After affordable broadband Internet communication to residential customers is available, there is a growing need for in-home networks to share this single full-time Internet access link, while supporting a wide range of digital data and multimedia communication services. The standard electrical wiring in the building becomes a computer network. The user can gain access to the network by plugging a power line modem into any electrical outlet in the building. The power line modem is a special modem that converts analog power line signal into a digital signal that a computer can use to send or receive data.
In the United States, PLC has taken 10 % of market in home networking. The products for in-home networking haves reached 14 Mbps, according to the HomePlug 1.0 standard [61]. The European Home System (EHS) consortium defines EHS 1.3 specification which covers the power line as a medium to transport control data, power, and information. At the moment, the transmission speed of power line is 2.4 kbps [62].

Automatic Metering Reading (AMR)

AMR power line system is to implement the readings of Electricity, Water, Gas or any other meters in the customer premises to be transmitted to a central base-station for further processing, billing, etc. This would typically use the LV segment below the MV/LV transformer [23].
Telemetry information is a relatively slow analog and does not require a high-speed data rate. As a result, a narrow-band PLC channel can be used for this function. An early form of telemetry uses the pulse duration with the On-Off carrier[63]. The length of the duration corresponds to a given analog level. This form of telemetry takes time and is suitable for slow changing levels. Later forms of telemetry use FSK to transmit information [52]. A frequency within a narrow bandwidth is shifted to convey the analog level to the remote. Analog level changes can be conveyed much faster than in the pulse duration method. This result may be accomplished by either using a narrow-band FSK tone, which is then used to modulate a single side-band (SSB) PLC, or using a dedicated narrow-band set[63].

Distribution Automation (DA)

PLC has an advantage of using the power system owned by the utility. It can reach any point in the system and extend automatically to newly added network elements. In the past, low speed PLC is useful for consumer load management and AMR, in most cases less than 100 bps [64][65]. At present, the high speed PLC technology has appeared and it is possible to meet the real-time requirement of DA/DSM application. In [66], it presents a PLC network with FSK modulation of up to 19,2 kbps over both LV and MV distribution system for AMR, DSM, DA and non-utility applications. In [23], it shows a MV PLC using OFDM for hybrid DA/DSM. In [41], it describes a LV and MV PLC system using DCSK (Differential Code Shift Keying) modulation with 9.6 kbps for AMR and DA applications.

3.4 Standards

X10 home automation

X10 technology is a communications standard for sending control signals to home automation devices via the power line (120 V or 220 V, at 50 Hz or 60 Hz). X-10 power line technology employees Amplitude Modulation (AM) to transmit binary data [67]. It is now trying to innovate into higher speeds with regard to establishing the communication between home PCs and controlled home appliances.

Intellon CEBus

The CEBus (Consumer Electronics Bus) standard is an open standard that provides separate physical layer specifications for communication on power lines and other media [68]. Data packets are transmitted by the transceiver at about 10 kbps, using spread spectrum technology. The CEBus protocol uses a peer-to-peer communications model so that a network node can access the media only when the network is available. It uses a Carrier Sense Multiple Access/Collision Detection and Resolution (CSMA/CDCR) protocol to avoid data collisions.

Echelon LONWorks

Echelon, like Intellon, provides a peer-to-peer communication protocol, via Carrier Sense Multiple Access (CSMA) techniques [69]. Echelon offers a 10 kbps power line chip based on spread spectrum technology. The LONWorks has just been passed ANSI/EIA standard proecess and now can be known as ANSI/EIA 701.9-A-1999.

HomePlug 1.0 standard

HomePlug Power line alliance has worked towards a common standard in the United States for high-speed home power line network. It is a non-profit corporation formed to provide a forum for the creation of an open standard and specification for home power line networking products and services and to accelerate the demand for products based on these standards worldwide through the sponsorship of market and user education programs. In July 2001, the HomePlug Specification 1.0 was ready. Now, it is endorsed by about 30 vendor companies. While HomePlug 1.0 was designed mainly to distributed broadband internet access in the home, the objective of HomePlug AV standard is to distribute Audio/Video content within the house, as well as data [47].

IEC 61334

Power line communication systems for MV and LV, are denoted as distribution line communication (DLC) systems by the International Electrotechnical Commission (IEC). DLC communication systems are being standardized by IEC, Technical Committee No 57 (Power System Control And Associated Communications), Working Group 9 (Distribution Automation Using Distribution-Line-Carrier Systems). IEC 61334 describes the structure of distribution systems for both medium and low-voltage levels and presents the architecture for a distribution automation system using distribution line carrier systems and use frequencies below 150 kHz.

One standard with the spread frequency shift keying [70] on low layer profiles and four technical reports on lower layer profiles have been published. They include Frequency Shift Keying (FSK) [71], spread spectrum adaptive wideband (SS-AW) [72], Multi-carrier modulation (MCM) [73], also known as OFDM, and spread spectrum-fast frequency hopping (SS-FFH)[74].

WG 9 has developed IEC 61334-4 series, a distribution Power Line Carrier (PLC) communication standard called Distribution Line Message Specification (DLMS). DLMS provides two-way communications and can be used on medium and low voltage networks. While DLMS can be used to access a wide variety of devices (switches, meters, lighting control, and other load controlling devices). Currently, it is primarily used for retrieving metering information using the IEC 62056 metering standard.

3.5 Identification of problems in PLC technology for utility applications

The PLC physical layer offers highly variable characteristics due to the time-varying noises injected by electrical devices. The noise can produce high bit error rate, which will require suitable measure for error detection and correction. FEC coding is very effective to correct certain kinds of error bits. The main drawback of the FEC is that it requires redundancy bits to be transmitted, even in totally error-free situation. So the effective network data rate is permanently reduced. Beside, the bit error caused by some kinds of noise cannot be corrected by the FEC. On the other hand, one may employ ARQ. ARQ is to equip the receiver with very good error-detection capabilities and to request the transmitter to retransmit once a fault is found. From the upper layer, the data rate is also decreased. ARQ reaches its limits when there are permanently high bit error rates, because the effective transmission speed can quickly drop to unacceptable values when many packets have to be retransmitted. Those technologies lead to a various transmission rates. It is not easy to guarantee a certain bandwidth and the maximum latency in PLC network. So the QoS requirements of the time-critical application cannot be satisfied without an effective mechanism in the upper layer.

PLC systems for utilities applications such as AMR have been available on the market for several years. Most of these systems achieve limited data rates of up 100 bps. For cost-efficiency, the use of one single communication system for both DA and DSM applications is mandatory. DA applications require support of real-time communication and event-driven operation of end devices. These requirements call for higher data rates and medium access protocol with QoS supporting. Though there are some trials for DA system [23][41][66], they are only applied in the low speed rate of PLC network and do not provide the QoS mechanism.

Although the high speed products have developed in the domain of LV grid of in-home and last mile applications, utilities applications are greatly different from them, because the field equipment is distributed widely and the long distance communication requires repeaters to relay the packet. Since the unpredicted attenuation and noise, it is impossible to determine a fixed configuration of the repeater. Furthermore, topology changes should be considered, which may be a consequence of switching activity in the distribution grid, where lines are disconnected or reconnected physically. They may as well result from different noise sources like large MV drives starting and stopping operations [2]. Both reasons are not controllable by the communication system. Consequently, the PLC system must be able to detect topology changes and adjust its operation accordingly. This means that the repeater to reach certain field equipment may not be fixed. Therefore, the repeater acts as a kind of router. So, in the PLC network, some adaptive mechanisms are used to optimize routing and possibly prepare alternate routes for communication problems due to topology changes.

3.6 REMPLI project and architecture

The REMPLI (Remote Energy Management over Power Lines and Internet) project is funded by the European Community (Program NNE 2001-00825). The objective of REMPLI is the definition and implementation of a communication infrastructure for data acquisition and control operations, which is suitable not only for distributed/remote monitoring and metering, but also opens to support new not yet planned tasks. This network must be aligned with the needs of energy distribution systems and connect all devices and installations from the supplier down to the customer.
The basis of the REMPLI system is a PLC infrastructure that allows accessing meter and remotely controlling equipment. The primary usage of this infrastructure is remote meter reading and remote control. In addition, the communication platform is open to various add-on services.
As shown in Fig. 3.2, the REMPLI communication infrastructure consists of:

- low-voltage segments (blue lines), which cover groups of energy consumers (for example, a segment can span across one staircase of apartments within a living block, or cover a single production branch);
- medium-voltage segments (red lines) between the primary and secondary transformer stations;
- TCP/IP or IEC 60870 based segments (thick purple lines) between the primary transformer stations and the Application server(s);
- TCP/IP communication (green lines) between the Application Servers and their clients. The interfaces provided by the Application Servers could be available only within the Private Network or also by Internet clients (e.g., SCADA server/client communication).

The bottom-level of the communication infrastructure is comprised of **REMPLI Node**s, each coupled with a PLC interface. A Node is usually installed at the consumer site, e.g., inside an apartment, and has a number of metering inputs (such as S0 (DIN 43864), for electrical energy meters). Each node can also be installed digital outputs that would allow switching electrical/heat/gas/water supply for a particular consumer, upon commands from the utility company.
For the high carrier frequencies typically used for high speed data communication, the transformer stations are "natural" obstacles, which cause a perfect separation. Low- and Mid-Voltage segments are coupled by the **REMPLI Bridge** which is installed at the secondary transformer station for communicating two segments.
In some installations, where a utility company needs to collect information from the secondary transformer station itself or to control it, the Bridge can be combined with an external Node.
The transition between PLC and Private Network (e.g., TCP/IP or IEC 60870 based) communication environments is carried out by the **REMPLI Access Point**. Access Points are installed in the primary transformer stations. Apart from the gateway between two different networks, Access Points also decouple communication: they "hide" low-level

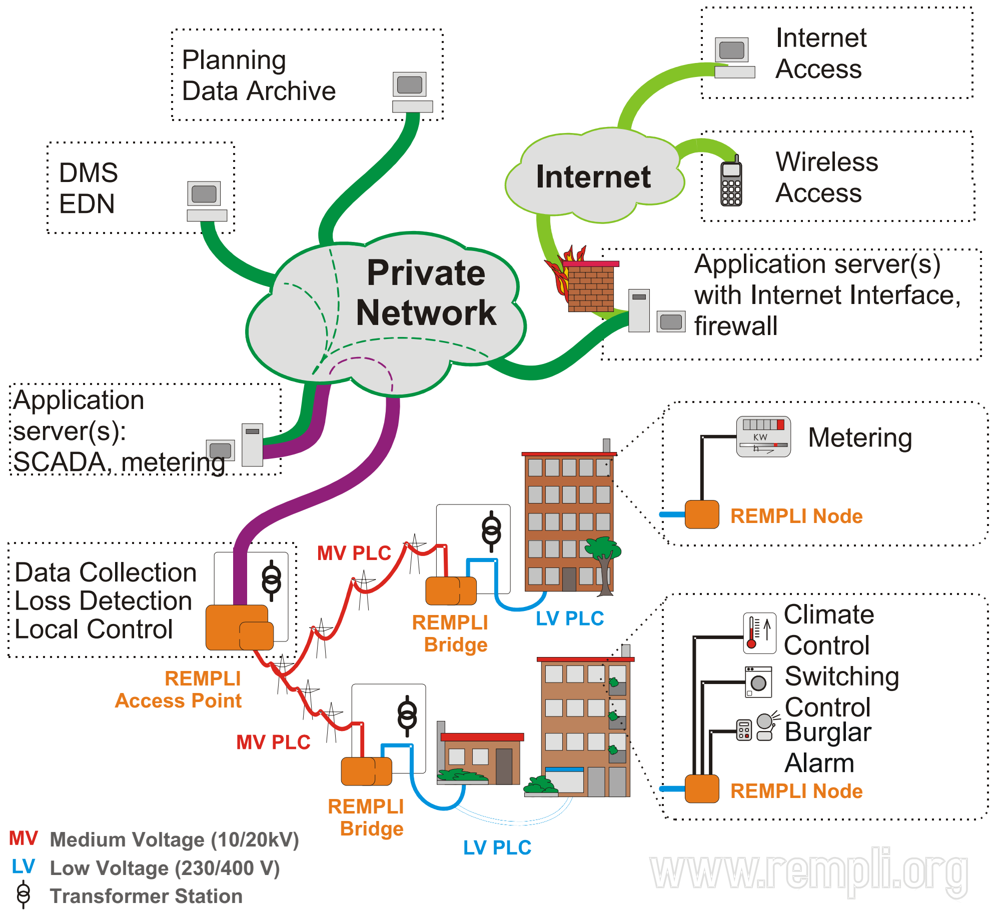


Figure 3.2: REMPLI system architecture, including the Private Network

properties of the underlying power line system from Private Network clients, offering a uniform interface for exchanging data with the Application Nodes.
Another important function of the Access Points is data caching. Process data available from Nodes, is continuously collected by the Access Points into a global "process image". This is done in background, independently of the requests received from the Private Network side. Subsequently, Access Points use the collected process image when responding to requests from their clients (Application Servers). And also, the direct access to the Nodes (bypassing the data cache) is provided, as certain tasks which run on the Application Servers may require the most recent values in stead of those stored in the process image. Requests for transferring data into the Nodes will be always accomplished directly.
Application Servers communicate with one or more Access Points over the private communication lines to obtain data from the Nodes. These Application Servers might be, for example, SCADA systems (performing data collection, logging historical measurement values, implementing control algorithms and energy management functions), additional servers (implementing other functions, not covered by the SCADA systems) or even client applications. Some anticipated Application Servers can be connected to the Private Network, but some of them will be located outside the Private Network segment, thus com-

municating to the respective Application Servers over the open Internet. To fight against denial-of-service attacks, those Application Servers that provide Internet interface shall be protected by a firewall.

The structure of Private Network (Intranet) is not specifically defined by REMPLI. The Access Point is the interface between the Intranet and the REMPLI communication system. Nevertheless, REMPLI offers appropriate interfaces at the Access Point to facilitate common Intranet structures and applications. REMPLI focuses on the definition and implementation of a communication infrastructure of PLC networks.

REMPLI communication system is organized as shown in Chapter 1. Since there are many independent polls, we call hereafter the piece of PLC network formed by one master and all its slaves as an ***autonomous PLC network***. So the whole REMPLI PLC network is composed of many "one master/multiple slaves" autonomous PLC networks.

3.7 Conclusions

In conclusion, PLC could provide a bi-directional, broadband communications platform capable of delivering real-time data, in order to be applied in a wide range of applications for the utility. With the development of high speed PLC network, the power utilities can benefit from PLC technology as it could increase its operating efficiency and infrastructure usage and incorporate into its portfolio value added services in different areas. Nevertheless, further research and development would be required to ensure the QoS of PLC Communication architectures and to confirm PLC technology as the most adequate one for the targeted applications. This is the objective of the REMPLI project. In this context, the aim of this thesis is to identify and evaluate services at the PLC protocol and architecture levels.

Chapter 4

Routing Algorithms

In this chapter, we present the routing protocols that we considered in Section 4.1 and the performance metrics used for their evaluation in Section 4.2. The theoretical analysis of the steady-state behaviors of these protocols is proposed, respectively, in Section 4.3.1 and 4.3.2. In Section 4.4, we apply the analytic results to the five channel models (i.e. five different network topologies). Numerical results show that the routing protocol of SFN always outperforms that of DLC 1000. In Section 4.5, after taking into account the time-varying channel characteristics (e.g. dynamic topology changes), the two routing protocols have also been studied by means of the simulation approach for the five channel models. The simulation results have revealed the need to find a right way to decrease the repeater levels in the routing protocol of SFN in case that the actual transmission uses less repeater levels than that of the master. So, we evaluate the performance of the different ways to decrease the repeater level in Section 4.6. Section 4.7 concludes this chapter.

In a wide area PLC network, transmitting a packet from a source to a not immediately reachable destination node requires the packet relay of the intermediate nodes (repeaters). However, considering the dynamic topology and impossible prediction of the powerline attenuation, repeaters cannot be statically configured. How to design an efficient routing protocol to dynamically adapt the powerline circumstances and shorten the transmission time under stringent bandwidth limitation is a challenging work.
As mentioned above, powerline channel exhibits random transmission characteristics. It is clear that classical routing protocols like static source routing, dynamic distance vector or state link based routing used in Internet are not suitable.
Although Powerline channel has some similarities with the radio transmission channel, routing protocols developed for wireless ad hoc networks [75] [76] [77] [78] are inefficient since on the one hand, the master-slave model is not taken into account, and on the other hand, REMPLI PLC network has a relatively stable topology when compared to the mobile nodes in ad hoc networks.
For REMPLI PLC network, two routing protocols have been successively proposed:

- dynamic source routing of DLC 1000 [79], and
- flooding-based routing over SFN [6] [80].

These two protocols should be compared to conclude which one has a better performance to guarantee successful packet transfer between the master and slaves with the shorter transmission time in REMPLI PLC environment. Two approaches are used in the performance comparison of these two routing protocols. The first approach is a mathematical analysis based on a static matrix of packet error rate (PER) with which we evaluate the steady-state behaviors of the system. The other is based on the simulation which allows us to study both the steady-state and time-varying behaviors.
For this analysis, 5 channel models are used, Ring_10, Ring_100, RandArea Np_20, RandArea Np_100, and RandArea Np_200. The first and second channel models have the topology of ring and the other three have the topology of a tree with the master as the root and the randomly distributed slaves as leaves. The number in the channel model name indicates the number of the nodes.

4.1 Routing protocols

4.1.1 DLC 1000

In the DLC 1000 protocol, the master maintains the routing table by periodically getting information of the most interesting paths from slaves and thus determines the best routing path for transferring a packet to a slave. Corresponding repeater addresses are indicated in the packet header. A slave sends a packet back to the master via the reverse path. The repeater addresses are included in the address field of the frame as indicated in Fig. 4.1.

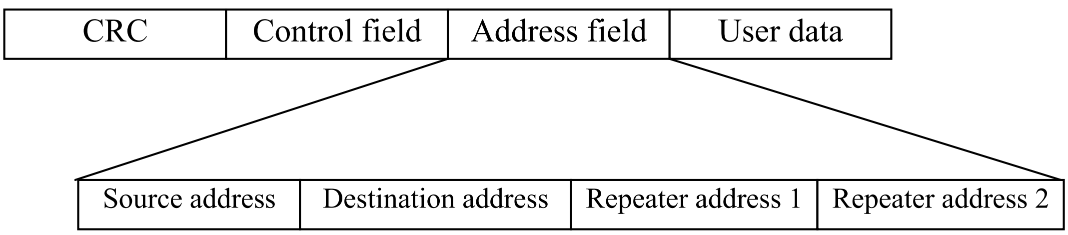


Figure 4.1: Structure of address field [1]

For routing purpose, every slave keeps a table with its favorite repeater slaves and forwards the best five entries to the master in response to status polling periods. The evaluation of the suitability of a slave R as a repeater for the slave Z is done by counting the number of packets sent by R and correctly received by Z. If a packet has been relayed by two repeaters already, the transmitter of the packet is not counted because it will be the third repeater if slave Z is the transmitter. To have prefer short routing paths, weighting factors are applied to the counting depending on the number of repeaters used. If a packet can be transmitted directly by the master, the transmitter is counted twofold in order to prefer short routing paths. Based on the counting of sent and received packets, every slave can

[1]For the confidentiality clause of REMPLI project, the length of the fields is not given here.

compute its own frequency of transmission as well as the frequency of reception by any other slave. By these values the repeater quality of a slave is determined as the reception probability from that slave.
The master updates its network statistics of the routing tables by a cyclic poll of all slaves. Furthermore, the master keeps two response quotas for every slave which refer to the success of the master's last attempt to reach this slave directly and over one repeater, respectively. Therefore, it has an "answer quota (direct)" AQ_0, that is determined by forming an exponential average of successful attempts (weight 1) and unsuccessful attempts (weight 1). There is an analog value AQ_1 that presents the answer quota for accesses over exactly one repeater. The AQ_1, is not taken into account which repeater is used. The advantage of these AQ-values to the values determined by the slaves is that they also consider the reverse direction; symmetrical transmission conditions are not required therefore.
The master has to compare and test eleven different paths including the direct path, five paths using one repeater each and another five paths using two repeaters. For this purpose, every path is judged by an evaluation figure, computed by means of the respective response quota, the reception probability of a possibly applied repeater and an additional weighting factor used to prefer short paths. In the case of two repeater, labeled as R_1 and R_2, the second repeater R_2 is determined by success probability firstly. The first repeater R_1 needs to be determined analogously to find the best repeater for a single-repeater-path to target slave R_2. The "d'Hondt algorithm" is then applied for the sequential determination of the currently best path. This algorithm is stopped in case of success or if the maximum number of attempts is exceeded. Depending on a random number, the direct path is chosen sometimes in the first attempt, even if it is not the best one so as to provide the chance that changes in network conditions can be noticed at all [79].

4.1.2 SFN

In SFN, all slaves can work as repeaters. The single frequency network allows several transmitters at various locations to transmit identical information on the same frequencies of the powerline medium at the same time. The receivers can get the information from the superposition of the signals [6].
The slaves, which received the initially transmitted data packet correctly by checking CRC (cyclic redundancy check), can be repeaters. In the next time slot, the slaves send this packet to the powerline. The same actions should be done until the initial packet reach the destination node. The data propagates automatically through the network without any routing table. This is the SFN routing principle. The procedure is shown in Fig.4.2.
Because the propagation on the powerline has not the direction, for avoiding the collision, the second packet is not sent until the first one has stopped and the network is idle. The effort for managing the network is dependent on the number of repeater levels.
Different repeater levels may be used in the downlink and the uplink, and it depends on the powerline's random channel characteristics. So, the field of downlink and uplink levels are indicated in the packet header shown in Fig. 4.3. The table of the number of repeaters (called hereafter repeater level, represented as $r_{DL}(i)$ and $r_{UL}(s)$ for downlink and uplink) for reaching every slave i is stored in the master.

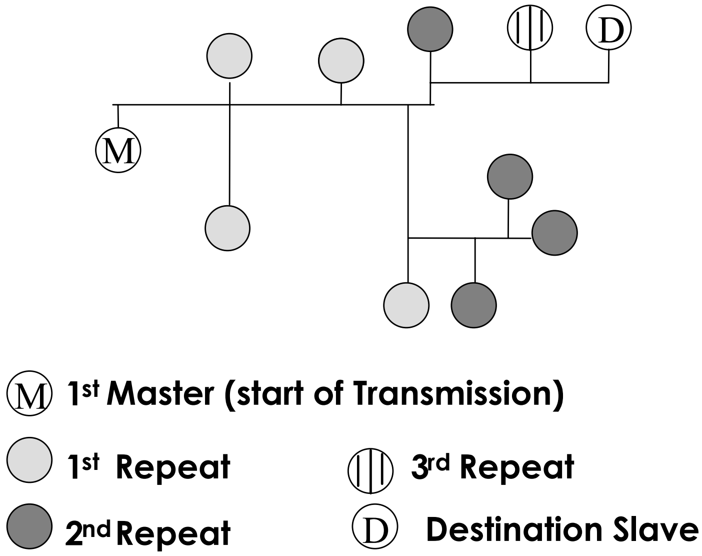


Figure 4.2: SFN routing principle

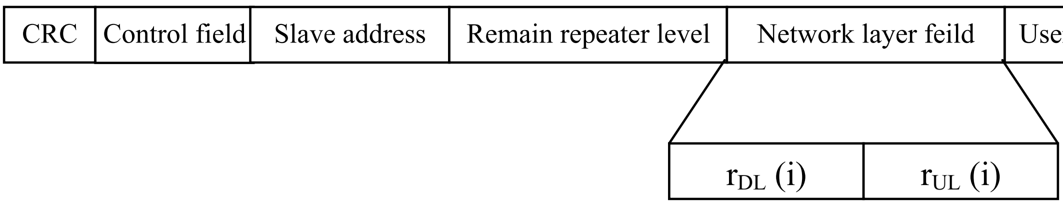


Figure 4.3: Structure of the SFN packet[2]

A slave checks the CRC field in order to decide if a received packet is correct. When the packet is received correctly, the slave checks the packet header, firstly. It will stop transmitting this packet in the following cases:

(i) the destination address is its own address,

(ii) the remaining repeater level is zero, and

(iii) the same packet have been transmitted once.

Otherwise it continues to retransmit the packet after the repeater level has been decreased. When the master does not receive confirmation of the slave i within a bounded transfer time, a retransmission is required. Obviously, the number of repeater levels was not high enough. The master has no information if the transmission failed in the downlink or in the uplink. To have a higher probability of the successful retransmission, the master has

[2] For the confidentiality clause of REMPLI project, the length of the fields is not given here.

to carry out the following:

$$\begin{cases} r_{DL}(i) = r_{DL}(i) + 1 \\ r_{UL}(i) = r_{UL}(i) + 1 \end{cases} \tag{4.1}$$

If the retransmission was not successful with new repeater levels, the master does (4.1) again and retransmits again. This continues until:

- a transmission is successful, or
- the maximum number of retries is reached, or
- the upper bound for downlink repeater level and uplink repeater level is reached.

When the master receives a packet before the predefined time slot, it means that the repeater level of the downlink or/and uplink should be greater than the actual number. In this chapter, we study two cases: one is to decrease the repeater level to the actual level; another one is to take more precaution, i.e., instead of decreasing immediately, we only decrease the repeater level after having observed the phenomenon several times. These two cases will be discussed in the Section 4.6.1.

4.2 Performance metrics

For comparing these two protocols, we use in this chapter an analytic approach to evaluate their performance via a set of common physical PLC topologies called "channel models". Each channel model is represented by a matrix of Packet Error Rate (PER), where $PER(i,j)$ is the packet error rate of the transmission from nodes i to j. In fact, PER is not symmetrical. It means that in general $PER(i,j) \neq PER(j,i)$.
In this chapter, we assume at first that PER is a time-constant value for evaluating the steady state system behaviors, i.e., without considering the topological changes. In fact, according to [82] the interference in the PLC system, such as periodic impulse and noise, is considered as an Additive White Gaussian Noise (AWGN), and consequently the PLC network is considered as a time-invariant system. The values of the PER matrix is calculated by the physical layer emulator [82]. An example of PER matrix of N nodes comprising one master and $(N-1)$ slaves, is shown in Fig.4.4. Node 1 is the master, labeled by M afterward, and the other nodes s ($s \in \mathbb{S} = \{2, \dots N\}$) are slaves.
We further make the following hypotheses in our theoretical analysis.

- Master - Slave system
  - Request is initiate by master and response is given by slave.
  - Retries are initiated by master (when response is not received). So there is no collision.
- Slotted system
  - One message is equivalent to one packet.

$$\begin{bmatrix} 0 & \cdots & \cdots & PER(M,N) \\ PER(2,M) & \ddots & & PER(2,N) \\ \vdots & & \ddots & \vdots \\ PER(N,M) & \cdots & \cdots & 0 \end{bmatrix}$$

Figure 4.4: PER matrix

  - Duration of packet is fixed and normalised to 1 T_s.
  - System is perfectly synchronised.
  - Analysis is based on PER: packet is OK or lost.

- Characteristics of channel (as indicated by PER) is constant over time and the master knows perfectly the characteristics of channel [3]

From the routing point of view, the routing efficiency is evaluated by the number of bits per data packet involving in the routing and the number of the control packets generated by routing protocols. Moreover, additional packets for the reliable delivery should also be included. However, in REMPLI PLC network, the transmission time from master to each slave is a random variable, due to the different repeater levels for reaching each slave, the possible transmission errors which will lead to a random number of retries (although upper bounded), and the possible dynamic topology changes which will lead to the changes of repeater level of a path. Therefore, we focus on the following ***three performance metrics***:

- ***Average duration of a polling cycle***: it is defined as the average time for the master to poll all the slaves once. The duration of a polling cycle can be explicitly expressed by the following way.
  For DLC 1000

$$D = \sum_{s=2}^{N} \sum_{j=0}^{n(s)} 2T_s \cdot (R^j(s) + 1) \tag{4.2}$$

where the parameters are:

s	destination slave ($s \in \mathbb{S} = \{2, ...N\}$; where N is the total number of nodes in the system)
T_s	duration of one time slot for transmitting a packet
$n(s)$	retry number to reach slave s successfully
$R^j(s)$	repeater number of node s for the j^{th} retry

[3]In fact, we can get different PER matrices during a simulation. The simulation can change the PER matrix after a predefined time, during the simulation runs. So, this hypothesis is only applied to the analytic model for evaluating the steady state behavior.

For SFN

$$D = \sum_{s=2}^{N} \sum_{j=0}^{n(s)} \big(2 \cdot (j+1) + r_{DL}(s) + r_{UL}(s)\big) \cdot T_s \tag{4.3}$$

where the parameters are:

s	destination slave ($s \in \mathbb{S} = \{2, ...N\}$; where N is the total number of node in the system)
T_s	duration of one slot for transmitting a packet
$n(s)$	retry number to reach slave s successfully
$r_{DL}(s)$	repeater level of downlink (master to slave) of slave s for the first transmission
$r_{UL}(s)$	repeater level of uplink (master to slave) of slave s for the first transmission

- ***Bandwidth consumed for routing signalling***: it is defined as the total number of routing information bits.

- ***Routing overhead in a data packet***: it is calculated as the number of routing information bits transmitted per data packet delivered.

4.3 Theoretical analysis

4.3.1 Average polling cycle duration of DLC 1000

It is the decision of the master to transmit with or without many repeaters and which slaves work as repeaters. The destination slave will send the packet back to the master in the reversed path. For a fixed repeater level, there may be many paths from the master to the destination slave if there are many slaves between them.
$P_{n(s)}(s, R^{n(s)}(s))$ is the probability of successful transmission from master M to destination slave s after $n(s)$-th retry with repeater path (R_1, R_2,. . . ,R_n) where R_i can be any slave except the destination slave s and master M and $n = R^{n(s)}(s)$. In fact, the master can increase the repeater number after a certain number of transmission failure. However, in our calculation, we only consider the case of changing the path except increasing repeater number. For simplicity, we use $P_n(s, r)$ to replace $P_{n(s)}(s, R^{n(s)}(s))$.
For transmitting a packet from the master to a slave and receive the confirmation packet in backward without any repeater, two time slots are used. The probability of a successful transmission from the master M to slave s is given by :

$$P_0(s, 0) = \big(1 - PER(M, s)\big) \cdot \big(1 - PER(s, M)\big) \tag{4.4}$$

If the master decides to transmit to slave s with one repeater R_i, it will use four time slots and the probability of successful transmission from master M to slave s is given by:

$$P_0(s, 1) = (1 - PER(M, R_i)) \cdot (1 - PER(R_i, s)) \cdot (1 - PER(s, R_i)) \cdot (1 - PER(R_i, M)) \tag{4.5}$$

where $R_i \in \mathbb{S} | R_i \neq s$.
DLC 1000 will decide a best path which has the highest successful transmission probability. We consider a set of r repeaters, $R_1, \ldots, R_r$, and $\mathbb{E}$ the set of all the paths $\{ch_k\}$ including r repeaters. The number of paths is given by the total analogously of r on r: $A = n!$. $\mathbb{E} = \{ch_k, k = 1, \ldots, r!\}$ where $ch_k = R_{k_1}, \ldots, R_{k_j}$, $k_i, k_j \in \{1, \ldots, r\}$ $k_i \neq k_j$. The probability to get a successful transmission through a path ch_k of r repeaters without retry is expressible as (4.6).

$$\begin{aligned} P_0^{ch_k}(s,r) &= (1 - PER(M, R_{k_1})) \cdot (1 - PER(R_{k_1}, R_{k_2})) \ldots (1 - PER(R_{k_r}, s)) \\ &\cdot (1 - PER(s, R_{k_r})) \ldots (1 - PER(R_{k_2}, R_{k_1})) \cdot (1 - PER(R_{k_1}, M)) \end{aligned} \quad (4.6)$$

And the probability to get a successful transmission without retry, through a path of r repeaters is given by

$$P_0(s,r) = \max_{ch_k \in \mathbb{E}} (P_0^{ch_k}(s,r)) \quad (4.7)$$

The probability to get a successful transmission through a path of r repeaters after n^{th} retry is given by:

$$P_n(s,r) = P_0(s,r) \cdot \big(1 - P_0(s,r)\big)^n. \quad (4.8)$$

The average time slots for a success transmission with r repeater is given by:

$$\bar{D}_n(s) = 2(n+1) \cdot T_s \sum_{n=0}^{\infty} (n+1) \cdot P_n(s,r). \quad (4.9)$$

With some transformations, the average duration $\bar{D}_n(s)$ can be written as:

$$\bar{D}_n(s) = \begin{cases} \dfrac{2T_s \cdot (n+1)}{P_0(s,r)} & \text{for } 0 < P_0(s,r) \leq 1, \\ \infty & \text{for } P_0(s,r) = 0 \ . \end{cases} \quad (4.10)$$

The minimum average duration of a polling cycle results in the maximum throughput of the network. The master is able to do this optimization and decides the number of repeaters. So, the average duration of a polling cycle $\bar{D}_{DLC,\sum}$ is the sum of the minimum average durations of all slaves. Thus,

$$\bar{D}_{DLC,\sum} = \sum_{s \in \{2 \ldots N\}} \min_{0 \leq n \leq n_{max}} \bar{D}_n(s) \quad (4.11)$$

4.3.2 Theoretical analysis of SFN

Probability for successful polling slave s

We recall that in SFN, the transmission of several nodes is independent and the same packet is only transmitted once per node.
Firstly, we analyze the case of downlink for transmitting a packet from the master M to the

destination slave s. $PRx_i(s,r)$ is defined as the probability of node i ($i \in \mathbb{S} = \{2,...N\}$) for the first correct reception in the time slot $t = r+1$ during the transmission process from Master M to the destination slave s. $PTx_j(r)$ is the probability that a node j ($j \in \{M, \mathbb{S}\} | j \neq s \;$ s.t. $j \in \{1,\ldots,s-1,s+1,\ldots,N\}$) sends a packet during the slot $t = r+1$.
Time slot $t = 1$ (or $r = 0$) is defined as the time when the master sends a request. Of course, at $t = 1$ time slot ($r = 0$), only the master can send. The probability of sending a packet $PTx_j(0)$ is given by:

$$PTx_j(0) = \begin{cases} 1 & \text{for j = M,} \\ 0 & \text{for j} \in \{2,\ldots,\text{s}-1,\text{s}+1,\text{N}\}. \end{cases} \tag{4.12}$$

The probability $PRx_i(s,0)$ of the first correct reception of node i in the time slot $t = 1$, will be:

$$PRx_i(s,0) = 1 - PER(M,i). \tag{4.13}$$

If destination slave s can be reached by the master directly, the repeater level $r = 0$. The probability $PRx_i(s,0)$is given by:

$$PRx_s(s,0) = 1 - PER(M,s). \tag{4.14}$$

If the repeater level $r = 1$, SFN routing protocol should be used for reaching the destination slave s. In the transmission procedure from Master M to destination slave s, every node j, except the master M, can act as a repeater and has the probability $PTx_j(1)$ to send the packet at time slot $t = 2$.
If a node does not receive a packet, it also does not transmit. If a node j has a probability of sending a packet in time slot $t = 2$, it means that the node j has not sent the packet in time slot $t = 1$ ($(1 - PTx_j(0))$ and has correctly received the packet in the time slot $t = 1$ ($PRx_j(s,0)$). Obviously, the master cannot be a repeater, so its sending probability is zero in time slot $t = 2$. As a result,

$$PTx_j(1) = \begin{cases} 0 & \text{for j = M} \\ (1 - PTx_j(0)) \cdot PRx_j(s,0) = 1 - PER(M,j) & \text{for j} \in \{2,...\text{s}-1,\text{s}+1,\text{N}\} \end{cases} \tag{4.15}$$

For t =2 ($r = 1$), more than one slaves may transmit during the same time slot. We assume that the transmission of several senders is independent, and a receiver gets the packet from several transmitters. The case that a receiver does not receive a packet correctly happens only when all senders j, which cannot be the destination s and the receiver node i, fail to send the packet to the receiver node i. Otherwise, the receiver node i should have the probability to receive the packet correctly from the sender j. So, the first correct reception probability of node i, $PRx_i(s,1)$, in time slot $t = 2$ is :

$$PRx_i(s,1) = (1 - PRx_i(s,0)) \cdot \left(1 - \prod_{j \in \{M,\mathbb{S}\} | j \neq s, j \neq i} \Big(1 - PTx_j(1) \cdot \big(1 - PER(j,i)\big)\Big)\right). \tag{4.16}$$

In (4.16), $(1 - PRx_i(s,0))$ ensures that node i has not received this packet before time slot $t = 2$ $(r = 1)$, and $\prod_{j \in \{M,\mathbb{S}\}|j \neq s, j \neq i} \Big(1 - PTx_j(1) \cdot \big(1 - PER(j,i)\big)\Big)$ is the probability of all senders j failing to send the packet to the receiver node i.
The probability $PRx_i(s,1)$ of first correct reception of the destination s in time slot $t = 2$ can be expressed as:

$$PRx_s(s,1) = (1 - PRx_s(s,0)) \cdot \left(1 - \prod_{j \in \{M,\mathbb{S}\}|j \neq s} \Big(1 - PTx_j(1) \cdot \big(1 - PER(j,s)\big)\Big)\right). \quad (4.17)$$

By a substitution of (4.14) and (4.15) in (4.17),

$$PRx_s(s,1) = (1 - PRx_s(s,0)) \cdot \left(1 - \prod_{j \in \mathbb{S}|j \neq s} \Big(1 - (1 - PER(M,j)) \cdot \big(1 - PER(j,s)\big)\Big)\right).$$

The rest may be deduced analogously. So, we get the following general formulas.
The first sending probability $PTx_j(r)$ of node j in the $(r+1)$-th time slot is:

$$PTx_j(r) = \begin{cases} 1 & \text{for r = 0 and j = M} \\ \Big(1 - \prod_{v=0}^{r-2} PTx_j(v)\Big) \cdot PRx_j(s, r-1) & \text{for r > 1 and j} \in \mathbb{S}|\text{j} \neq \text{s} \\ 0 & \text{for otherwise} \end{cases} \quad (4.18)$$

where $\left(1 - \sum_{v=0}^{r-2} PTx_j(v)\right)$ is the probability that node j has not already transmitted this packet before time slot $t(= r+1)$, and $PRx_j(s, r-1)$ is the probability that node j has correctly received the packet in the last time slot.
$PRx_j(s,r)$ is defined as the probability of the first correct reception of a packet for the node i in the $(r+1)$-th time slot. It can be written as:

$$PRx_i(s,r) = \begin{cases} 1 - PER(M,i) & \text{for } r = 0 \\ \left(1 - \sum_{v=0}^{r-1} PRx_i(s,v)\right) \cdot \left(1 - \prod_{j \in \{M,\mathbb{S}\}|j \neq s, j \neq i} (1 - PTx_j(r) \cdot (1 - PER(j,i)))\right) & \text{for } r > 0 \end{cases} \quad (4.19)$$

where $\left(1 - \sum_{v=0}^{r-1} PRx_i(s,v)\right)$ is the probability that the node i does not receive the packet before the r-th time slot, and $PTx_j(r) \cdot (1 - PER(j,i))$ is the correct reception probability from node j to *node* i. So, $\left(1 - \prod_{j \in \{M,\mathbb{S}\}|j \neq s, j \neq i} (1 - PTx_j(r) \cdot (1 - PER(j,i)))\right)$ is the probability that a receiver does not receive a packet correctly. This happens only when all senders $j(j \in \{M,\mathbb{S}\}|j \neq s, j \neq i)$ fail to transmit the packet to the receiver node i.

The first correct reception probability $PRx_s(s,r)$ of destination slave s in $(r+1)$-th time slot is expressible as:

$$PRx_s\left(s,r\right) = \begin{cases} 1 - PER(M,s) & \text{for } r = 0 \\ \left(1 - \sum\limits_{v=0}^{r-1} PRx_s\left(s,v\right)\right) \\ \cdot \left(1 - \prod\limits_{j \in \{M,\mathbb{S}\}\,|j \neq s} \left(1 - PTx_j\left(r\right) \cdot \left(1 - PER\left(j,s\right)\right)\right)\right) & \text{for } r > 0 \end{cases} \tag{4.20}$$

The probability$P_{DL}(s, r_{DL})$ of successful transmission from master M to slave s after repeater number $r_{DL}(s)$, will be the sum of the probability $PRx_s\left(s,r\right)$ of first correct reception in each time slot, and is given by

$$P_{DL}(s, r_{DL}(s)) = \sum_{r=0}^{r_{DL}(s)} PRx_s\left(s,r\right). \tag{4.21}$$

The uplink transmission procedure is similar to that of the downlink transmission. We only need to redefine the symbols. $PRx_s\left(s,r\right)$ is defined as the probability of the first correct reception of node i ($i \in \{M, \mathbb{S}\} | i \neq s$) in the transmission process from the destination slave s to the master M in the time slot $t = r + 1$. $PRx_j(r)$ is the probability that a node j ($j \in \mathbb{S}$ i.e. $j \in \{2, \ldots N\}$) transmits a packet during the slot $t = r + 1$. We define the beginning time as $t = 1$ (or $r = 0$). For the uplink, receiving a packet correctly after downlink repeater level $r_{DL}(s)$, the destination slave s has the transmission probability $PTx_s(0)$ to transmit the response/confirmation back to the master. The transmission probability $PTx_s(0)$ is equal to the downlink's receiving probability $P_{DL}(s, r_{DL}(s))$ in the beginning time $t = 1$, as given in (4.22).

$$PTx_s(0) = P_{DL}(s, r_{DL}(s)) \tag{4.22}$$

The probability that the master correctly receives the response/confirm in time slot $t = r+1$, denoted by $PRx_M(M,r)$, can be calculated in a similar way as that of the downlink.

$$PRx_M\left(M,r\right) = \left(1 - \sum_{v=0}^{r-1} PRx_M\left(M,v\right)\right) \cdot \left(1 - \prod_{j \in \mathbb{S}} \left(1 - PTx_j\left(r\right) \cdot \left(1 - PER\left(j,M\right)\right)\right)\right) \tag{4.23}$$

where $$PTx_j(r) = \begin{cases} P_{DL}(s, r_{DL}(s)) & \text{for } r = 0 \text{ and } j = s \\ \left(1 - \sum\limits_{v=0}^{r-2} PTx_j(v)\right) \cdot PRx_j(M, r-1) & \text{for } r > 0 \\ 0 & \text{else} \end{cases}$$

and

$$PRx_i(M, r) = \begin{cases} P_{DL}(s, r_{DL}(s)) \cdot (1 - PER(s, i)) & \text{for } r = 0 \\ \left(1 - \sum\limits_{v=0}^{r-1} PRx_i(M, v)\right) \\ \cdot \left(1 - \prod\limits_{j \in S | j \neq i} (1 - PTx_j(r) \cdot (1 - PER(j, i)))\right) & \text{for } r = 0 \end{cases}$$

Analogously, we can get the probability $P_{UL}(s, r_{UL}(s))$ of successful transmission from the destination slave s to master M after repeater number $r_{UL}(s)$. It is given by

$$P_{UL}(s, r_{UL}(s)) = \sum_{r=0}^{r_{UL}(s)} PRx_M(M, r). \tag{4.24}$$

The master knows the number of repeater levels for the downlink $r_{DL}(s)$ and for the uplink $r_{UL}(s)$. The probability of successful polling slave s with repeater levels of downlink $r_{DL}(s)$ and uplink $r_{UL}(s)$ and without retry can be expressed as:

$$\begin{aligned} P(s, r_{DL}(s), r_{UL}(s)) &= P_{DL}(s, r_{DL}(s)) \cdot P_{UL}(s, r_{UL}(s)) \\ &= \left(\sum_{r=0}^{r_{DL}(s)} PRx_s(s, r)\right) \cdot \left(\sum_{r=0}^{r_{UL}(s)} PRx_M(M, r)\right) \end{aligned} \tag{4.25}$$

Average polling cycle duration

When we get the correct reception probability of each slave, the average duration for a downlink transmission per slave can be calculated. In this case, we should consider the retry in the calculation. For a predefined repeater level, if the correct reception probability is not equal to 1, there exits a probability of transmission failure. After each transmission fails, the retry happens and its repeater level should be incremented by one. The retry will stop until the transmission succeeds. Herein, we assume that after each successful transmission, the repeater level should be changed to the actual repeater level used in current transmission. The process of transmission for each slave s presented in SFN protocol can be modeled as a Markov reward process.

Let $\{X(n), n \geq 0\}$ represent a homogeneous discrete-time Markov chain describing the successful transmission behavior of the n^{th} packet, because the n^{th} packet transmission is only related with the state of last packet transmission. The state of the system $\{X(n)\}$ at the time when the n^{th} packet has been transmitted successfully, is defined as the repeater level at that time. Since there are repeater level of downlink r_{DL} and repeater level of uplink r_{UL}, we define a vector (r_{DL}, r_{UL}) as the possible repeater level state. For example, $(0, 0)$ denotes that repeater level of downlink is 0, and the repeater level of uplink is 0. For simplicity, without loss the generality, we further assume that r_{DL} and r_{UL} can be identical for each slave s. Indeed, the following mathematical model can be used for the case of different r_{DL} and r_{UL}. Thus, in a simplification, (r_{DL}, r_{UL}) can be written as RL. Thus, let $\{0, 1, \ldots, N\}$ be the set of repeater level states, where N denotes the finite number of repeater levels of the system, and $RL \in \{0, 1, \ldots, N\}$. Let the probability of transition from state i to state j be $p_{i,j}$, and let $\mathbf{P} = [p_{i,j}]$ denote the $N \times N$ transition probability matrix. Therefore, we construct a Markov chain. Herein, we will introduce a new probability $P'(s, r_{DL}, r_{UL})$ denoting the successful transmission probability with the actual repeater level r_{DL} and r_{UL}. This probability can express that the repeater level should be changed to the actual repeater level used in current transmission after each successful transmission.

$$P'\left(s, r_{DL}(s), r_{UL}(s)\right) = PRx_s\left(s, r_{DL}(s)\right) \cdot PRx_M\left(M, r_{UL}(s)\right) \tag{4.26}$$

So the Markov chain shows as Fig. 4.5.

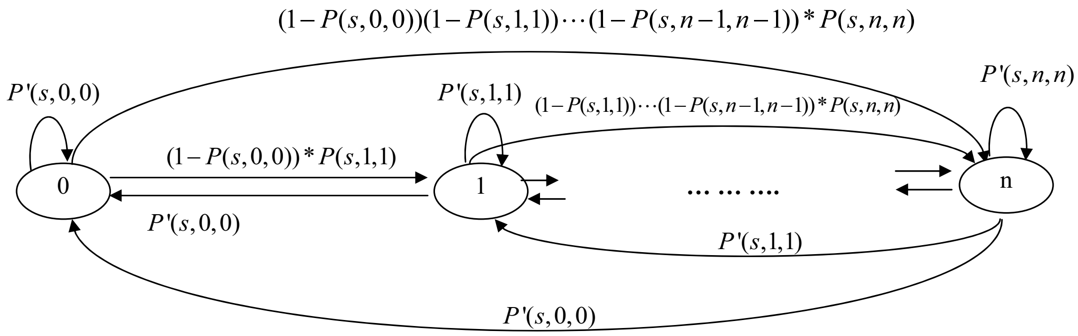


Figure 4.5: Markov chain model of repeater level

If $i < j$, it means that the transmission is successful after $(j - i)$ retries. And if $i \geq j$, it means that the transmission is successful with the actual repeater level value j.

$$p_{i,j} = \begin{cases} \left(\prod_{v=i}^{j-i} \left(1 - P(s, v, v)\right)\right) \cdot P(s, j, j) & \text{for } i < j \\ \\ P'(s, j, j) & \text{for } i \geq j \end{cases} \tag{4.27}$$

where $P(s, r, r)$ can be obtained by (4.25) and $P'(s, r, r)$ can be obtained by (4.26). Notice that

$$P(s, r, r) = \sum_{v=0}^{r} P'(s, v, v)$$

If we denote the state distribution vector at time interval k by $\pi(k)$, we have $\pi(k + 1) = \pi(k)\mathbf{P}$. For an irreducible and aperiodic Markov chain, there exists a stationary distribution vector $\pi(\infty) = \pi$ given by $\pi = \pi\mathbf{P}$, that is, π is the left eigenvector of $\mathbf{P}$

corresponding to the eigenvalue 1.
Let $r_{i,j}$ be the nonnegative reward associated with the repeater level state i of $X(n-1)$ and the repeater level state j of $X(n)$. We define that the reward is the transmission time (unit of time slot) for the n^{th} packet. So

$$r_{i,j} = \begin{cases} \sum_{k=0}^{j-i}(2*(k+1+i)) & \text{for } i < j \\ \\ 2*(i+1) & \text{for } i \geq j \end{cases} \tag{4.28}$$

We can obtain r_i by formula (4.32). Let R_{n+1} be the expected used transmission time (unit of time slot) that originates at X_n and terminates at X_{n+1}. The average transmission time for slave s can be obtained from (4.33):

$$\bar{D}_{SFN}(s) = \sum_j r_j \pi_j \tag{4.29}$$

where $\pi = \pi * \mathbf{P}$, with $\sum_i \pi_i = 1$ and $\mathbf{P} = [p_{i,j}]$.
The average duration of a polling cycle is given by:

$$\bar{D}_{SFN,\sum} = \sum_{s\in\{2,\ldots N\}} \bar{D}_{SFN}(s) \tag{4.30}$$

In what follows we just recall some basic concepts related to Markov reward process.
Theory of the Markov reward process [85]
An irreducible positive-recurrent Markov chain $\{X_j\}$ in which each occupancy of state j generates a reward r_j is called a Markov reward process. Let R_n be the reward associated with the state visited at epoch n, where R_n may depend on both X_n and X_{n-1}. The cumulative reward $C(n)$ overs the first n transitions is

$$C(n) = \sum_{s=1}^{n} R_s(X_{s-1}, X_s)$$

, with expected value $c(n) = E(C(n))$. Given X_0 and any sequence of transitions X_1,...,X_n, we assume the rewards R_s, $s = 1, \ldots, n$ are independent, where for each s, the distribution of $R_s$depends only on X_s and X_{s-1}. For any (i, j) such that

$$(X_{s-1}, X_s) = (i, j),$$

let

$$r_{i,j} = E\{R_s(i,j)\} \tag{4.31}$$

where $r_{i,j}$ is assumed to be finite, and

$$r_i = \sum_j r_{i,j} p_{i,j} \tag{4.32}$$

wherer_i are well defined and finite based on assumptions made below. Thus, $r_{i,j}$ is the expected reward associated with a transition of type $i \to j$ and r_i is the expected reward associated with a departure from state i.

Theorem: Reward Rates for Markov Chains

For a positive irreducible Markov chain, with $E\left\{\sum_{s=1}^{T_{ii}} |R_s(X_{s-1}, X_s)|\right\} < \infty$ for any initial state $X_0 = i$,

$$\lim_{n\to\infty} c(n)/n = \sum_{j=0}^{\infty} r_j \pi_j \tag{4.33}$$

where π_j is the long-run fraction of time the process is in state j.

4.4 Numerical comparison between SFN and DLC 1000

4.4.1 Five channel models

To give a numerical insight of the performances of the two routing protocols, we apply our analytical results to the 5 channels models represented by the fives PER matrices. Because of the huge size of the matrices, here we only show herein that of the Ring_10 channel model in Fig. 4.6.

$$\begin{bmatrix}
0.00 & 0.00 & 0.00 & 0.46 & 1.00 & 1.00 & 1.00 & 0.46 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.46 & 1.00 & 1.00 & 1.00 & 0.46 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.46 & 1.00 & 1.00 & 1.00 & 0.46 \\
0.46 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.46 & 1.00 & 1.00 & 1.00 \\
1.00 & 0.46 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.46 & 1.00 & 1.00 \\
1.00 & 1.00 & 0.46 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.46 & 1.00 \\
1.00 & 1.00 & 1.00 & 0.46 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.46 \\
0.46 & 1.00 & 1.00 & 1.00 & 0.46 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.46 & 1.00 & 1.00 & 1.00 & 0.46 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.46 & 1.00 & 1.00 & 1.00 & 0.46 & 0.00 & 0.00 & 0.00
\end{bmatrix}$$

Figure 4.6: An example PER matrix of Ring_10 channel model

4.4.2 Average duration of a polling cycle

With PER matrices and formulas in Section 4.3, we use MATLAB program given in Appendix A to get the analytical results of average duration of a polling cycle for DLC 1000 and SFN. Table 4.1 shows the results. In fact, DLC 1000 protocol restricts the repeater number to two. However, we assume the maximum repeater number can be seven in our calculation. Even with optimal assumptions for DLC 1000 routing, the performance of SFN is better than that of DLC 1000 in all channel models.

Channel Model	SFN	DLC 1000
Ring_10	29.13	30.0
Ring_100	422.77	427.3
RandArea Np_20	43.19	63.0
RandArea Np_100	397.1	420.6
RandArea Np_200	1014.76	1026.6

Table 4.1: Average duration of a polling cycle of SFN and DLC 1000

4.4.3 Bandwidth consumed for routing signaling

In the initial phase, the master works in a similar way for both routing protocols. The master sends packets to try to connect with all slaves using the serial number and distributes a network address for each connected slave. After the initial phase, the master will periodically send the packet to request slaves (connected or not). There is a difference concerning the slave responses. For SFN, only the confirmation will be sent back. For DLC 1000, besides the confirmation in the response packet, the slave will add five preferable repeaters and the corresponding evaluated channel quality which is the routing information used by the master to decide the best routing path. So obviously, the bandwidth consumed for routing signaling in DLC 1000 is more than in SFN.

4.4.4 Routing overhead

In the same packet length, for example 64 bytes packets, the packet headers of two protocols are different about the routing information. The comparison is shown in Table 4.2.

	Routing information (bits)	Routing bits / total packet
DLC 1000	Two repeater address (2*12bits)	4.7%
SFN	Repeater levels (8 bits) 4 bits for downlink 4 bits for uplink	1.6%

Table 4.2: Routing overhead

4.4.5 Conclusion

From the above comparison, we can draw a conclusion that in general case, the flooding-based routing protocol of SFN outperforms the dynamic source routing protocol of DLC 1000. In fact, for a given receiver, there are more transmitters in SFN than those in DLC

1000 for a repeater level greater than 1. Because the transmission of each transmitter is independent, in SFN, not only the best path is used but also all the other possible paths. The probability of correct reception in SFN is therefore higher than that in DLC 1000 in which only a single transmitter is used.

4.5 Simulation and performance evaluation

Simulation is an important tool used to design and implement advanced communication systems that deliver optimal performance. However, available channel models in current simulator such as OPNET Modeler [84], and ns-2 [83], did not include powerline models. Unfortunately, an accurate channel models for the power line environment [92][91][95][94] has not been standardized yet, and there is no widely accepted channel model similar to those derived for mobile radio or telephone channels [93].
In our simulation program (see Appendix B for some details), we choose the Physical Layer Emulator [82] to simulate the behavior of the channel and the physical layer of the PLC system. It includes channel encoding and synchronization. The powerline channel in field trials is time-variant and cannot be completely analyzed. In the emulator, the interference in the PLC system, such as periodic impulse and noise, can be considered as the Additive White Gaussian Noise, and consequently, the PLC network is considered as a non-time variant system for certain channel models. The characteristics of the transfer medium are extracted into matrices and tables, further called channel models. Different channel models are developed to represent typical structures of powerline networks. Moreover, the time-varying characteristics of power-line channels are considered in the Emulator. In fact, the emulator allows periodic changes of channel models during a simulation in order to analyze the influence of this behavior in the routing protocol.
However, the Physical Layer Emulator does not define a structure for the network protocol. On the top of the Physical Layer Emulator, two simulation models are constructed using C++ and OPNET Modeler respectively for simulation of DLC 1000 and SFN protocols. The main objective of the simulation study is to gain insights into the performance of these two routing protocols. The ***main performance metrics*** is the ***average duration of a polling cycle***.

4.5.1 Physical Layer Emulator

Since DLC 1000 and SFN use Time Division Multiple Access (TDMA) scheme, the Physical Layer Emulator uses a slotted system and calculate system time using the duration of one time slot as the unit. The definition of the channel model includes the characteristics of the channel, the behavior of the synchronization, topology of the network and the number of nodes. Therefore, the influence of the parameters carrier frequency, bandwidth, bandwidth efficiency and synchronization, are handled in the configuration of the channel model which is chosen by the upper layer simulator.

Channel model

As the topology can differ widely, emulations have to be created for different networks and the resulting matrices. However, consideration needs to be given to the fact that it is impossible to emulate all possible networks with all their different characteristics exhaustively. To get an impression of the system behavior in critical cases, certain scenarios were specifically designed.
According to the power distribution network topologies, two kinds of channel models are constructed: ring model and tree network model based on a stochastic area model. However, channel models can also be designed from actual measures of real networks. Five channel models are used that differs by the topology (ring or tree) and the number of nodes (10, 20, 100, 200). And these models are denoted by Ring_10, Ring_100, RandArea Np_20, RandArea Np_100 and RandArea Np_200. Respectively, the number in the name of channel models indicates the number of nodes which include one master and all the slaves.

Ring Model

In power supply networks, rings are a popular structure. They are open or closed. They can be redundant structure in MV distribution network. Currently, LV power networks in residential areas are structured as rings as well, from which branch lines go to houses. A ring can be interrupted in any place and certain parts can be taken off-line while the remainding the ring is still supplied.
As the length of the cable is not the dominant part of signal attenuation, ring topology can be described as a very regular structures if different distances between the nodes are negligible.

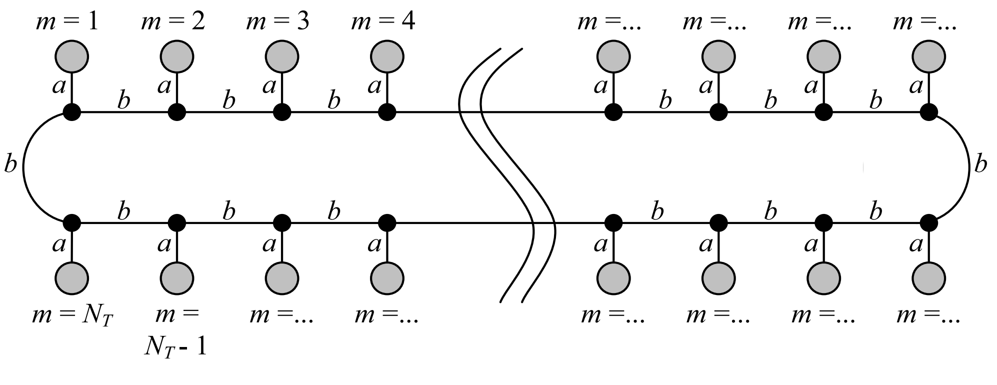


Figure 4.7: Structure of the constructive ring model

The branch line between branch point and node is of length a, while the distance the single branch points is b.
To emulate open rings, it is possible to add an open switch. The position of the switch is determined by the number of nodes in front of this switch on the ring. Even if the switch is open, there is some kind of crosstalk. Therefore, we handle an open switch as additional attenuation between two nodes. The critical situation for the communication is that PER over the switch communication is between 20% and 80 %. If the PER is lower, the transmission signal is able to through the switch and the receivers can get the

packet. On the other side, if the PER is high, it is necessary to decide another direction for the routing.

Random Distributed Tree Model

Most of the LV distribution systems have a tree structure. Even if loops exist, usually there are open ring at the most of time. The distribution of nodes is not regular, but we can assume that the locations of nodes are uniformly distributed inside an area. We assume the master at the center of the area. The position of every slave is given by a complex number (real → x-axis, imaginary → y-axis) and the master is located at the crossing point of the coordinate axis. The cables are parallel to the co-ordinate axis. An area is restricted by a defined maximum cable length between a participant and the master. An example for a distribution network with 100 participants is shown in Fig. 4.8.

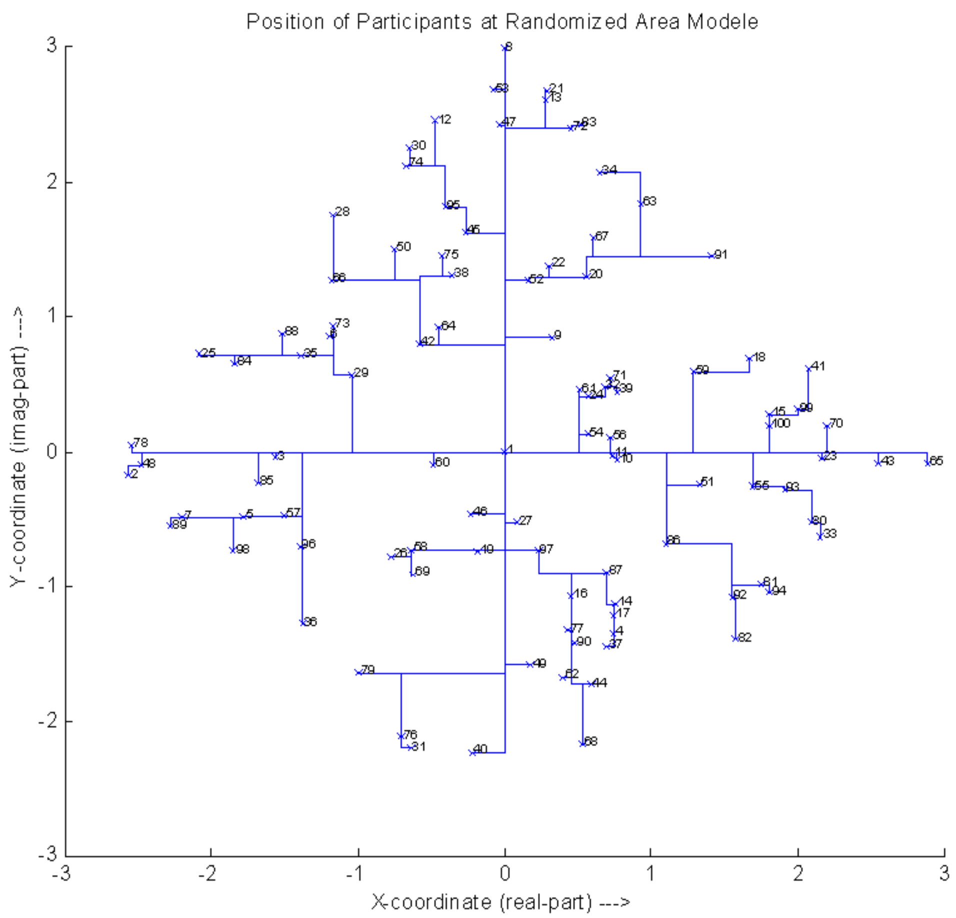


Figure 4.8: Cable plan for random distributed tree model

4.5.2 Simulation parameters for routing protocols

For both routing protocols (DLC 1000 and SFN), the master tries to logon the slaves in the initial phase of the simulation. When this step is achieved for each slave, the master starts polling slaves one by one. The main parameters for a simulation are:

- Logical channel: 1
- Retry number: 5 for DLC 1000 and 2 for SFN
- Data length: 64 bytes
- Channel models: Ring_10, Ring_100, RandArea Np_20, RandArea Np_100, RandArea Np_200
- The duration of the simulation time is fixed at 200 hours

4.5.3 Simulation results of DLC 1000

Problem in simulation of DLC 1000

Because the repeater number is fixed to two in the packet format, some slaves that are far from the master cannot communicate with it. For example, in channel model RandArea Np_100, 4 slaves fail the logon. The simulation results of duration of a polling cycle cannot include those nodes. For comparing with the results of theoretical analysis, we give theoretical results without those nodes which cannot be reached by two repeaters in simulation. The results are shown in Table 4.3.
In the case of high PERs, the consecutive unsuccessful transmission will happen. As a result, the confirmed request cannot be sent back to the master after reaching the maximum retry number. An example is shown in Fig. 4.9. In consequence, the simulation result gives an abnormal value. This is one of the reasons that there is a great difference between the simulation results and theoretic results in the channel mode of RandArea Np_20.

```
Slave Number 9 : Number of unconfirmed requests after all slaves logged in: 42
Slave Number 15: Number of unconfirmed requests after all slaves logged in: 31
Slave Number 17: Number of unconfirmed requests after all slaves logged in: 11
Slave Number 20: Number of unconfirmed requests after all slaves logged in: 35
```

Figure 4.9: Some simulation results in the channel model of RandArea Np_20

Comparing analytic results with the simulation ones

In Table 4.3, we compare the calculated values of average duration of a polling cycle with the simulation results. Due to the reason mentioned above, simulation results for some channel models cannot be given out. So we calculate two analytic results for those channel models: one with all slaves and the other with reachable slaves in the simulation program. From Table 4.3, we see that the most analytic results are close to the simulation results. And we also notice that there is a big difference between the analytic result and the simulation one in Table 4.3 for the case of RandArea_20 channel model. This is due to the particularity of this channel model and the simulation program. In channel model of RandArea Np_20, there is an exceptional interval, in which some slaves cannot be connected by the master as shown in Fig. 4.9. In simulation, if the master cannot reach a slave after the maximum retry number, the master will start to poll the next slave. In the next polling cycle, the repeater number will be the last retries value for the problematic slave, and the duration of a polling cycle will be greater than its normal value, even if this slave can be reached at that moment.

Channel Model	Maximum repeater number	$\bar{D}_{DLC,\Sigma}$	$\bar{D}_{DLC,\Sigma}$ (without some nodes)	Simulation results	Relative difference
Ring_10	2	30.0		30.94	3%
Ring_100	3	427.3	363.48 (91slaves)	399.12 (91slaves)	9.8%
RandArea Np_20	2	63.0		77.20214	22.5%
RandArea Np_100	3	420.6	388.55 (95 slaves)	421.71 (95 slaves)	8.5%
RandArea Np_200	4	1026.6	755.0835 (160 slaves)	806.03 (160 slaves)	6.7%

Table 4.3: Analytic and simulation results of average duration of a polling cycle

A more detailed comparison between transmission time of each slave obtained by simulation and theoretical analysis is given for channel model of RandArea Np_20 and is shown in Table 4.4. The transmission time is defined as the number of time slot used for the master sending a packet to a slave and receiving a packet from a slave. Duration of a polling cycle is the sum of the transmission time of all slaves. The values highlighted in grey correspond to nodes which sometimes have problems to communicate with the master. Since the retry is included in the transmission time, the simulation values are greater than the theoretical values. From Table 4.4, one can see the great difference between simulation and theoretical analysis results for some slaves. So, we conclude that the theoretical analysis of DLC 1000 is not suitable in the channel mode of RandArea Np_20.

Slave	Theor.	Simul.	Relative difference
2	4	4.1265	0.031625
3	2	2.02402	0.01201
4	2	2.01912	0.00956
5	4	2.01912	-0.49522
6	2	4.15188	1.07594
7	4.5335	2.02478	-0.5533738
8	2	2.02466	0.01233
9	2	***7.0038***	2.5019
10	4	2.02898	-0.492755
11	4	2.02142	-0.494645
12	2	2.01814	0.00907
13	2	4.2562	1.1281
14	2	2.02274	0.01137
15	5.6926	***7.97136***	0.40030215
16	4.6875	6.76658	0.44353707
17	2	***6.87912***	2.43956
18	5.0175	4.12986	-0.1769088
19	2	2.02518	0.01259
20	7.0313	***11.68868***	0.66237822
Duration of a polling cycle	62.9624	77.20214	0.2261626

Table 4.4: Comparison of the simulation and theoretical transmission time of each slave in RandArea Np_20

4.5.4 Simulation results of SFN

In SFN, 4 bits are used for each repeater level of downlink and uplink. The maximum repeater level can be 16 and this is sufficient to connect all the slaves. From Table 4.5, we see that the analytic results are close to the simulation results except for the RandArea Np_20 channel model. The reason is similar to that given for DLC 1000 routing protocol. So from now on, we do not use channel model RandArea Np_20 for the simulation, since it is a special case.

We find that the repeater level is not stable because it will be changed to the actual value

Channel Model	Theoretical results	Simulation results	Relative difference
Ring_10	29.13	35.67	18.33%
Ring_100	422.77	448.21	5.68%
RandArea Np_20	43.19	81.44	46.97%
RandArea Np_100	397.1	440.42	9.8%
RandArea Np_200	1014.76	1079.76	6.0%

Table 4.5: Analytic and simulation results of SFN

after each successful transmission. For example, in channel model Ring_10, the repeater level of slave 6 can be 1 or 2, and the successful transmission probability with repeater level 1 is 50% and that with repeater level 2 is 100%. The retry has 50% probability to be happened when the repeater level is 1. It brings in the long transmission time. So the repeater level is an importance value in SFN protocol. In the next section, we will study how to improve SFN protocol by stabilizing repeater level.

4.6 Improvement of SFN protocol

In this section, we will study how to be efficient to set repeater levels in a consideration of two aspects: reliable transmission and short transmission time. Sometimes, these two aspects conflict, because a greater repeater levels have a high successful transmission probability but long transmission time and, on the contrary, a lower repeater levels has small successful transmission probability, but shorter transmission time. Unfortunately, if the retry happens, the transmission time will be greater than that in using a higher repeater level for higher successful transmission probability. We use the SFN simulation to evaluate the network performance of those methods. The results are presented in Section 4.6.1 for determining which one is the most suitable in the PLC environment. The ***main performance metric*** is the ***average duration of a polling cycle***.

4.6.1 Methods to decide the number of repeater levels

Normally, the master does not know the matrix of PERs. Also, the PERs vary due to the changes in the channel. And the changes of the topology, for example due to switch of

line, are abrupt and can require completely new values for the number of repeater levels $r_{DL}(s)$ and $r_{UL}(s)$. It requires the protocol to have a fast reaction time. And it should have a high efficiency for the static case.
The master keeps a list of $r_{DL}(s)$ and $r_{UL}(s)$ for all $s \in \{2, \ldots, N\}$. r_{DL} and r_{UL} are not required to change until a transmission failure happens. Therefore, the decision in the master will be increment, decrement or unchanged.
Now 2 cases have to be analyzed.

Case 1: Master receives no confirmation from the slave In this case, a retry is required. The master increases the downlink and uplink repeater level as the description in Section 4.1.2. In fact, the increment of both sides is not necessary due to the asymmetric channel characteristic of downlink and uplink.
If we add the information of whether the first request of downlink was successful by the slave side in the confirmation packet, the master can decide whether the increment of downlink repeater level in the last retry is necessary. To give this information, one bit has to be reserved in the confirmation packet. The slave sets this bit, when it received a request before $r_{DL}(s) + 1$ time slots. The master looks on this bit only when it sends a retry. If this bit is set, i.e., the failure happened on the uplink in the last transmission, the increment of $r_{DL}(s)$ is not required and $r_{DL}(s)$ can be decreased again.
Analogously, if the master receives the packet from the slave before the $r_{DL}(s)+r_{UL}(s)+2$ time slots after it sends a retry, last transmission failure happened on the downlink and the increment of $r_{DL}(s)$ was necessary. The increment of $r_{UL}(s)$ is not based on a transmission failure, so $r_{UL}(s)$ will be decreased to get the old value again.

Case 2: Transmission task was successful without retry In this case, the effective methods of decrement of $r_{DL}(s)$ and $r_{UL}(s)$ are considered for less transmission time as well as the reliable transmission.
When the slave receives the request at the last repeater value, $r_{DL}(s)$ has not to be changed. If receiving the request before the $r_{DL}(s) + 1$ time slots, $r_{DL}(s)$ may require to be decreased at the next transmission. Similarly, $r_{UL}(s)$ may require to be decreased. However, the time gain of a decrement of $r_{DL}(s)$ and $r_{UL}(s)$ that causes a retry is possibly lower than that in the maintenance of $r_{DL}(s)$ and $r_{UL}(s)$. So, we use two counters $c_{DL}(s)$ and $c_{UL}(s)$ for counting the number of consecutive events respectively, in which the repeater level is greater than the actual repeater level. When the counter crosses a certain level, the decrement is done. Two counters $c_{DL}(s)$ and $c_{UL}(s)$ are cleared after a decrement. A transmission failure or all number of repeater levels were necessary for a transmission task.
We propose a dynamic mechanism to set the upper limit $c(s)$ depending on $r_{DL}(s)$ and $r_{UL}(s)$:

$$c(s) = r_{DL}(s) + r_{UL}(s) + 2 \tag{4.34}$$

It is necessary to evaluate this result by simulation.

4.6.2 Simulation results

Recall those methods and label them in the following:

- *Protocol 1:* to decrement repeater level by one after the counter $c_{DL}(s)$ or $c_{UL}(s)$ is reached the upper limit $c(s)$, when transmission task is successful without retry and the actual repeater levels are less than the current ones.
- *Protocol 2*: to decrease the repeater levels to the actual number when transmission task is successful without retry and the actual repeater levels is less than the predefined ones.

Mechanism for decreasing repeater level in protocol 1

In Section 4.6.1, we propose some mechanisms about setting $c(s)$ for decreasing repeater level. We give the analytical results with some hypothesis which are limited and cannot show a universal conclusion. In this section, we will compare the simulation results of those mechanisms and choose the best one for Protocol 1.
We recall the four methods for $c(s)$ proposed in Section 4.6.1:

- decrease one when $c(s) = 1$
- decrease one when $c(s) = 2$
- decrease one when $c(s) = 8$
- decrease one when $c(s) = r_{DL}(s) + r_{UL}(s) + 2$

The simulation results are shown in Table 4.6. The results of $c(s) = 8$ and $c(s) = r_{DL}(s)+r_{UL}(s)+2$ are approximate and better than the others. Obviously, the mechanism $c(s) = r_{DL}(s) + r_{UL}(s) + 2$ can have a quick reaction time for the abrupt topology change, such as open ring case, which we will discuss in the following section. So we choose $c(s) = r_{DL}(s) + r_{UL}(s) + 2$ for the Protocol 1.

Channel Model	$c(s) = 1$	$c(s) = 2$	$c(s) = 8$	$c(s) = r_{DL}(s) + r_{UL}(s) + 2$
Ring_10	33.05	32.89	30.47	30.59
Ring_100	438.7	439.51	429.67	430.14
RandArea Np_100	425.98	424.81	420.12	420.25
RandArea Np_200	1056.42	1060.81	1034.70	1034.25

Table 4.6: simulation results with different $c(s)$ of protocol 1

We present the stability of repeater level using Protocol 1 with $c(s) = r_{DL}(s) + r_{UL}(s) + 2$ in Table 4.7. It shows the only a small number of repeaters are required to reach a certain node, even in large networks and and the reties percentage is low, which can get a stable transmission time. And the maximum repeater level difference indicates the difference between the low number of repeater level and high number of repeater level to reach the destination slave. Since the maximum value is 2, we reasonably configure the retry time to 2 in our simulation configuration of Section 4.5.2. in Table 4.7.

Channel model	Maximum repeater level	Maximum repeater Level difference	Average of packet retries per polling cycle
Ring 10	1	1	1.1%
Ring 100	3	2	0.4%
RandArea Np_100	4	2	0.5%
RandArea Np_200	4	2	1.1%

Table 4.7: Repeater level stability characteristics of protocol 1

Performance comparison of protocol 1 and protocol 2

When the system starts, the repeater level is equal to the broadcast repeater level r_{broad}. Note that, as the actual repeater level may be greatly less than r_{broad}, for a better performance, the repeater level needs to decrease to the actual value rapidly. Another case is the open ring phase. Open ring is a special type of power distribution networks where the ring topology will change to the bus topology by the switch opening on the line. Due to the topology change, some slaves are required to change the repeater levels to communicate with the same master.

Initial phase

Fig. 4.10 and Fig. 4.11 [89] correspond to the average duration of a polling cycle in the start-up phase. This is the most interesting period, since afterwards the system tends to stabilize.

In all scenarios, Protocol 1 works better. Nevertheless, Protocol 2 provides a faster adaptation to the optimal repeater level in start-up phase, while Protocol 1 needs more time to stabilize.

After the start-up phase, it is also possible to notice a slight advantage of Protocol 1, despite it possesses less complete information than Protocol 2. This difference is given by the fact that, in Protocol 2, there is a higher number of retries when the repeater level increases. Table 4.8 presents the retries percentage per polling cycle. In fact, as the repeater level updates are only based on the last transmission task, a frequent change in repeater level forces the master to send a retry packet in case of an underestimation of the actual repeater level. It consequently increases the polling cycle duration.

Open Ring phase

Open ring model has ring topology which can be opened or re-closed by a switch. The deployment of this structure guarantees a higher availability of the network. The Physical Layer Emulator provides such a type of topology with ring_100. After a certain number of time slots, a switch is alternatively opened at node 20 or node 70. Results are also available for this special model, with switches performed at intervals of 10000 or 50000

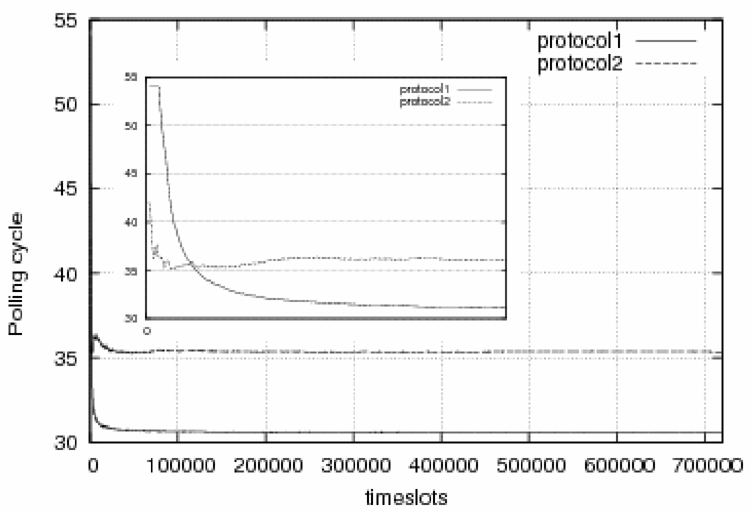


Figure 4.10: Polling cycle average for Ring_10 scenario with start-up phase zoom

Channel Model	Protocol 1	Protocol 2
Ring_10	1.1%	16.6%
Ring_100	0.4%	4.2%
RandArea Np_ 100	0.5%	4.8%
RandArea Np_200	1.1%	4.4%

Table 4.8: Average of packet retries per polling cycle

time slots.
Open ring phase represents a situation with constant start-up phases, since abrupt topology changes bring a high variation on the repeater level to reach a certain node. Fig. 4.14 [89] presents the polling cycle average, for the two simulated protocols and for the two topology change periods. Both protocols adapt well when the period of topology switch is 50000 time slots. in a comparison,Protocol 2 adapts faster in the initial phase. However, when the period of topology switch is 10000 time slot, the protocol 1 has no enough time to decrease the overestimated repeater level, and have a large average time of a polling cycle.
In open ring phase, the repeater level difference is possible to be large than two repeater levels. There are lost packets if the slave cannot be reached after the maximum retry time which be configured to 2. We show the lost packets percentage in the simulation time in Table 4.9.
Notice that in the 10000 period switches scenario, Protocol 1 has no packet loss. Considering the current situation that both protocols overestimate the repeater levels to attain their slaves due to a switch closing, Protocol 2 can recover this shortly and adapts to this situation quickly. However, this time period is not enough for Protocol 1 to adapt, as

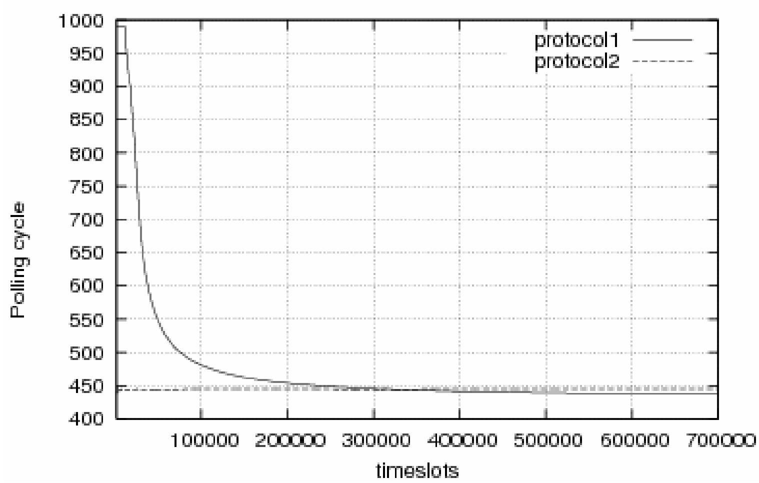


Figure 4.11: Polling cycle average for Ring_100 scenario

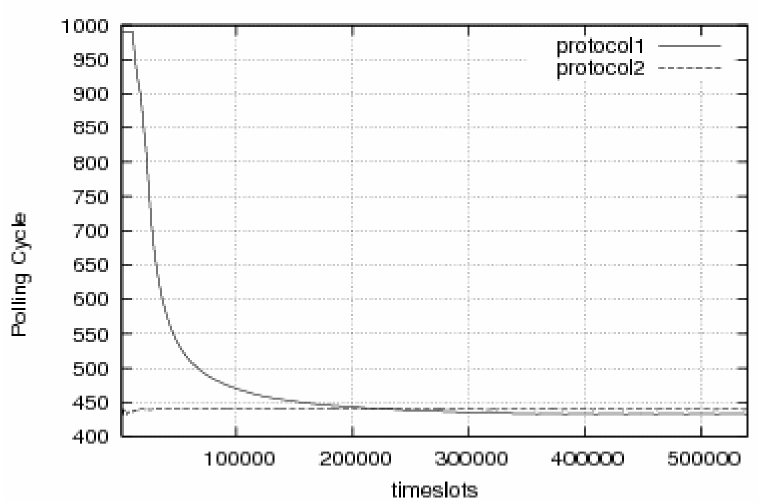


Figure 4.12: Polling cycle average for RandArea Np_100

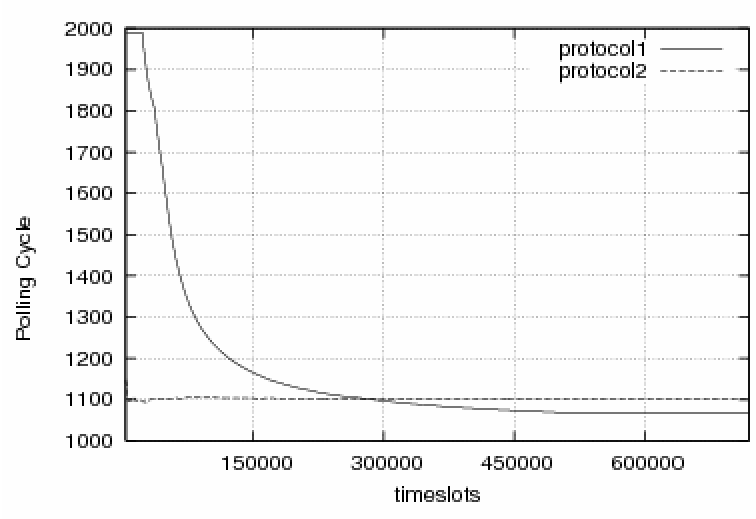


Figure 4.13: Polling cycle average for RandArea Np_200

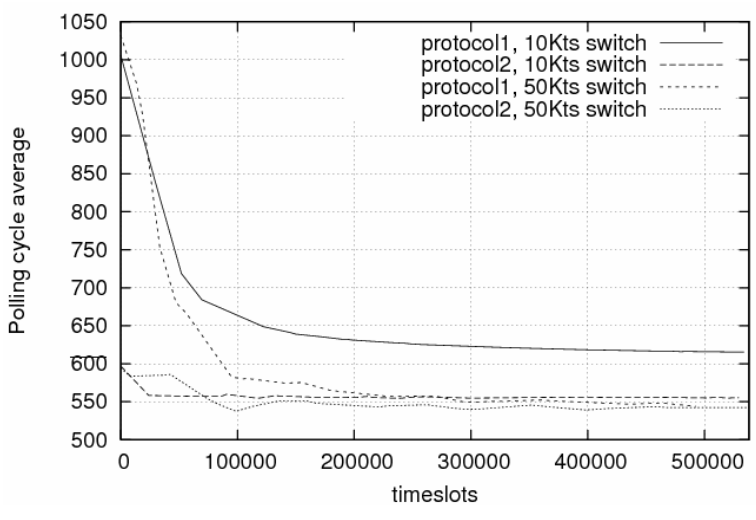


Figure 4.14: Average duration of a polling cycle for open ring 100 network

	Protocol 1		Protocol 2	
	10k ts switch	50k ts switch	10k ts switch	50k ts switch
% of affected nodes	0%	13%	15%	15%
% of lost packets	0%	0,086%	0,339%	0,07%

Table 4.9: Packet losses in open ring 100 scenario (ts stand for time slot)

shown in Fig. 4.14. Due to its slow reaction, this protocol maintains an overestimation of the optimal repeater level, which could be an advantage in terms of packet loss in the next switching period.
From the above comparison, Protocol 1 shows a good adaptation to networks with temporary topology disturbances, since its reaction to the optimal repeater level is slow. And it also shows a good performance in both the start-up phase and stable phase.

4.6.3 Conclusion

From the analytical comparison, the SFN protocol has a better performance than DLC 1000 protocol. In the simulation results, the master cannot connect with all slaves in a large network using DLC 1000 protocol, since the repeater number is limited to 2. Correspondingly, the SFN protocol has no such problem. So, we can draw a conclusion that in general case, the flooding-based routing protocol of SFN outperforms the dynamic source routing protocol of DLC 1000. At the end, we improve the performance of SFN protocol by configuring $c(s) = r_{DL}(s) + r_{UL}(s)$. The simulation results proved our proposition.

Chapter 5

REMPLI Performance Evaluation

In this chapter, we use three main timing metrics, which are the response time to a command, the response time to a request and the average polling duration, to evaluate the network performance of an autonomous PLC network and REMPLI PLC network. REMPLI PLC network consists of MV and LV autonomous PLC networks, which a bridge connects those two autonomous PLC network. We will give out the formulas for calculating the timing metrics and compare the simulation results with the timing constraint remarked in Chapter 2 to show if REMPLI PLC network protocol can be suitable for the application requirements.

5.1 Logical channel

In Time Division Multiple Access (TDMA) scheme, the time is divided into time slots (T_s). The usage of time slots is managed by the master. As OFDM systems transmit data in parallel, demodulation cannot start before the end of a packet. Therefore, the repetition cannot start immediately after the message is received. So using the time required for computation for a transmission of the other packets can result in an efficient usage of resources. If the whole system has been well synchronized, the logical channel division takes place in a chronological way. The time slot assignment follows a predetermined pattern that repeats itself periodically. Each such period is called a cycle, which can be taken for a duration of logical channel, as shown in Fig. 5.1

In Fig. 5.1, we see an example for 3 logical channels A, B and C. Only one packet is sent to the powerline within a time slot of T_s. The spare time between two packets on the powerline depends on the synchronization. The process of repeating which includes the demodulation, then higher network layer procedure and the modulation again, can be carried out by the Microprogrammed Control Unit (MCU) of the physical layer and the network layer, if there are 2 time slots between receiving and transmitting slots. The network bandwidth is divided into the n ($n \geq 3$) channels. Each channel transmission rate is equal to the total rate divided by the number of the logical channels.

The logical channels are used independently in the network layer. The protocol layer

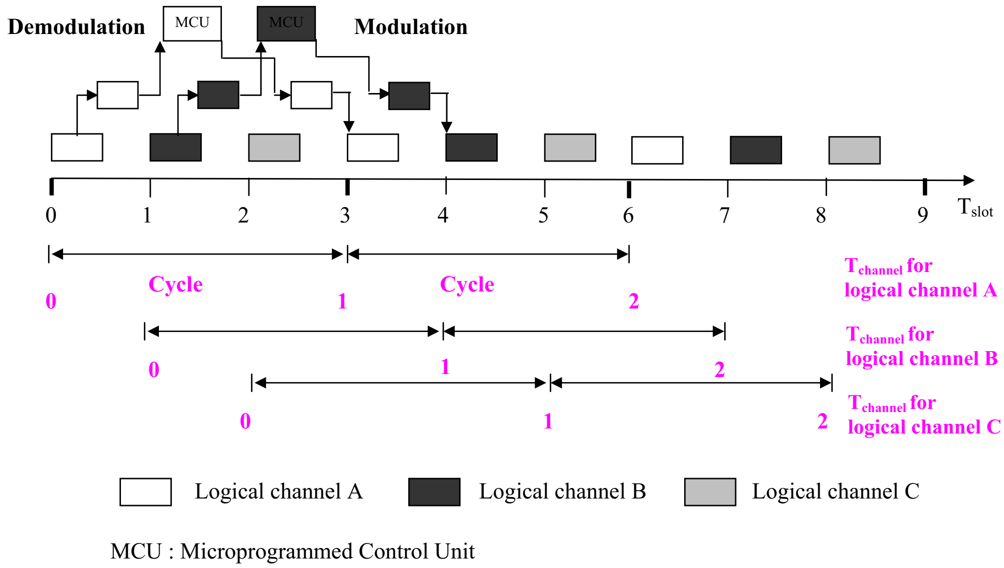


Figure 5.1: Pipeline of logical channels

concept with independent logical channels is shown in Fig. 5.2. Logical Channel Unit (LCU) is responsible for the administration of the logical channel. It receives data from the network layer and constructs the packet according to the communication protocols. It also manages the repeater function of SFN protocol. The Logical Channel Multiplexer checks if a packet is to be sent and forwards it to the physical layer.
The logical channels can be used in many ways. Different logical channels can be allocated to different medium access ways, which are managed by the independent logical channel unit. For example, we use 2 logical channels for a master/slave system with cyclic polling and the third time slot as slotted aloha for fast messages from slave to master. So, different types of communication systems can be integrated and work in parallel. Also, it is possible to use the logical channels for different masters, which have to be synchronized. The slave is able to communicate with them in parallel. So, the system provides a high redundancy for applications. A slave can also communicate with only one of the logical channels, since it is not necessary that a slave is connected to all logical channels.

5.2 Network layer services

Within an autonomous PLC, medium access is controlled by master station in master/slave protocol. The other stations are slaves which are polled and supervised by the master station. They can transmit only when they are polled by the master. Polling uses a priority-based scheme. Priority assignment and the bandwidth allocation are done by the dispatcher in the network layer which will be explained in the chapter 6.
The network layer offers two types of services to the upper layers: poll with acknowledgement (ACK), send without acknowledgement. The former is used for providing a reliable

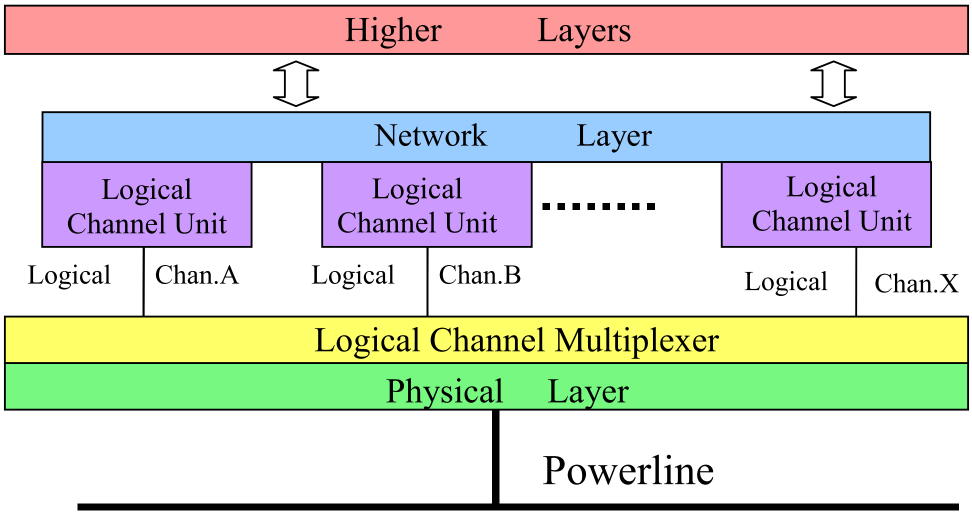


Figure 5.2: Layer concept with logical independent channels

transmission. The destination slave should send an ACK back to the master if it has received a packet correctly. However the ACK can piggy back data in the queue of the slave. The latter is basically an unacknowledged packet service, applicable to unicast- and multicast-/broadcast-addresses. In this chapter, we focus on the service of poll with ACK.

In PLC system, the transmission may be failed as the noise disturbs it. So Automatic Repeat Request (ARQ) is used. After it sends a packet to a slave, the master will configure the timer with a value which is calculated by (5.1), where T_{ch} is the duration of n logical channels. The slave then arranges to send acknowledgement packets (ACKs) back to the sender for each correctly received packet. If there are data in the slave queue, it will be put into the ACKs.

$$Timer(s) = (r_{DL}(s) + 1 + r_{UL}(s) + 1) * T_{ch} \tag{5.1}$$

When an ACK is not received after the timer expires, the master retransmits the relevant packet and sets a new timer. Both the repeater levels $r_{DL}(i)$ and $r_{UL}(i)$ should be increased by one after each retry, according to the SFN protocol. This requires the master to store each packet it is sending until it receives an ACK for that packet. It is possible that a transmission failure happens in the uplink. So, the data in queue should be stored until the new polling packet is received, when the data are sent in ACKs. A sketch of the interaction is shown in Fig. 5.3.

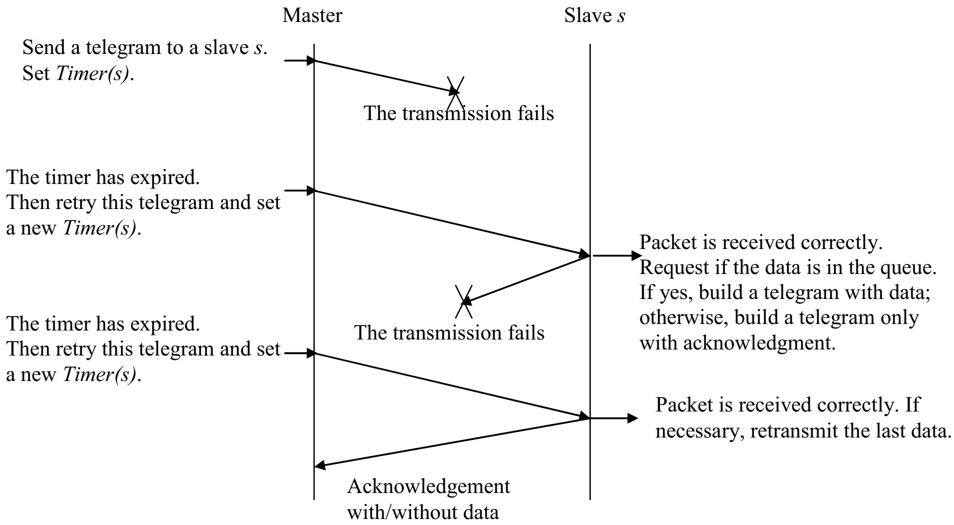


Figure 5.3: Interaction of the service of polling with acknowledgement or data

5.3 Performance Evaluation

In the following performance evaluation, we assume that the system has three channels. The timing performance of REMPLI system only concerns in one of the logical channel. Herein, the unit of time is T_{ch}. In REMPLI production parameters, the channel number can be 3 or 4, and the slot length can be 8.8 ms, 14.1 ms or 28.2 ms. To obtain the best case or worst case performances and show some examples, we use two values of T_{ch} as given in Table 5.1.

Channel number	T_s	T_{ch}
3	8.8 ms	26.4 ms
4	28.2 ms	112.8 ms

Table 5.1: Value of T_{ch}

5.3.1 Performance in an autonomous PLC network

When REMPLI PLC network works as communication medium for the substation automation application, to transfer data between the MV substation control system and MV feeder equipments along MV power grid or the LV distribution substation, a MV autonomous PLC network performance should be evaluated to show if the application requirement can be guaranteed. From the application view, there are two kinds of traffics in the system: one is generated by the master for SCADA and AMR, and the other is generated by the slave for the alarm application. They have different time constraints: response time and message transfer delay. There are two kinds of response time: response time to a command and response time to a request. The transmission direction is from master to slave. On the contrary, the message transfer delay is defined as the duration from the time when the data is generated in the slave to the time when this data is brought back to the master by the master periodical poll. The transmission direction is from the slave to the master.

Response time

The response time to a command (T_c) is the time between the time when the packet is ready for transmission in the master and when it arrives to its destination slave s correctly. The total time is given by:

$$T_c(s) = T_{acc} + T_{M\to s}(s). \tag{5.2}$$

T_{acc} refers to medium access delay. When the packet is ready to be sent, the channel has been occupied by the transmission of other packets or high priority packets. The command transmission cannot begin until the current transmission tasks have been finished by receiving ACK from the destination slave or the transmission failure after the maximum retry number. So, T_{acc} is a random value and cannot be predicted in a multi-tasking environment with aperiodic application functions. In the simulation scenarios, the T_{acc} can be ignored because we are only interested in $T_{M\to s}$ as it is network related. T_{acc} depends on the priority of the command. We simulate the scenario that new command message and request/response message arrive to the master, sequentially after the last message has finished the transmission.

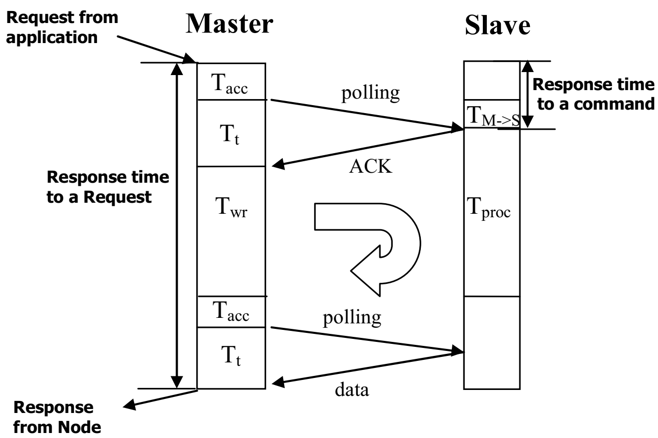


Figure 5.4: Schematic representation of application-to-application delay at the network layer level in an autonomous PLC network

$T_{M\to s}$ represents the transmission duration from sending the command by the master M

to receiving it correctly by the destination slave s. Considering the occurrence of the retry, $T_{M \to s}$ can be calculated by the following formula:

$$T_{M \to s}(s) = \left(r_{DL}(s) + n(s) + 1 + \sum_{j=0}^{n(s)-1} (2 \cdot (j+1) + r_{DL}(s) + r_{UL}(s)) \right) \cdot T_{ch} - (n-1) \cdot T_s. \tag{5.3}$$

Parameters:

n	number of logical channel
T_s	duration time of one slot
T_{ch}	duration time of n logical channels
$n(s)$	retry number to each node s
$r_{DL}(s)$	repeater level of downlink of node s for the fist transmission
$r_{UL}(s)$	repeater level of uplink of node s for the first transmission

In (5.3), the part of $r_{DL}(s)+n(s)+1$ is to calculate the time slots for sending the command to node s when the transmission has been successful in $n(s)$-th retries, while the part of $\sum_{j=0}^{n(s)-1} (2 \cdot (j+1) + r_{DL}(s) + r_{UL}(s))$ is to calculate the time slots for transmission before retries. The part of $(n-1) \cdot T_s$ is the rest of T_{ch} after the command has been received in the slave in its own T_s.

The response time to a request (T_r) is the duration from the time when the request packet is ready for transmission to the time when the response packet is received by the master. The response may not available immediately upon receiving the request in the slave, but only after a certain delay. Unfortunately, this value is random, depending on the type of the field devices and current process tasks, etc. For example, the delay for a meter to provide a response to the attached node depends on the type of the meter, the data volume and the link (RS232, IEC1107 or M-bus). This range value corresponds between 200 ms to 1.5 seconds [81]. For effective utilization of PLC bandwidth, the slave sends an ACK back to the master when it receives a request. The upper layer should waiting for a response duration (T_{wr}) to the network layer of the master. In the process time T_{proc}, the response has been prepared and stored in the slave queue. During T_{wr}, the master may commence other transmissions. After waiting time of T_{wr}, the master will poll the slave again to get the response back. If the master transmits other packets in T_{wr}, the T_{acc} is possible to be required. We assume that one request only generates one packet response from the device. From Fig. 5.4, $T_r(s)$ is the response time to a request for slave s and the formula to calculate can be expressed as

$$T_r(s) = T_{acc} + T_t(s) + T_{wr} + T_{acc} + T_t(s). \tag{5.4}$$

$T_t(s)$ is transmission time from a request packet sent by the master to a response packet received from the destination slave s.

$$T_t(s) = \sum_{j=0}^{n(s)} (2 \cdot (j+1) + r_{DL}(s) + r_{UL}(s)) \cdot T_{ch} - (n-1) \cdot T_s \tag{5.5}$$

In fact, the value of T_{acc} is random and cannot be predefined. So we will ignore it in the calculation. So $T_c(s)$ and $T_r(s)$ are expressible as

$$T_c(s) = T_{M \to s}(s) = \left(r_{DL}(s) + n(s) + 1 + \sum_{j=0}^{n(s)-1} (2 \cdot (j+1) + r_{DL}(s) + r_{UL}(s)) \right) \cdot T_{ch} - (n-1) \cdot T_s \tag{5.6}$$

and

$$\begin{aligned} T_r(s) &= T_t(s) + T_{wr} + T_t(s) \\ &= (\sum_{j=0}^{n(s)} (2 \cdot (j+1) + r_{DL}(s) + r_{UL}(s)) \cdot T_{ch} - (n-1) \cdot T_s) \\ &+T_{wr} + (r'_{DL}(s) + 1 + r'_{UL}(s) + 1) \cdot T_{ch} - (n-1) \cdot T_s \end{aligned} \tag{5.7}$$

where $r'_{DL}(s)$ and $r'_{UL}(s)$ are the stable repeater levels after the last retry.
Table 5.2 shows the simulation results about the slave node which has maximum transmission time in the channel model due to the instable repeater level which can lead to the retry occurrence and the stable repeater level after the retry. According to Chapter 4, only the repeater level of the link that causes the failure transmission should be increased. This is the reason why the repeater level before retry and after retry is the same value. With the parameters in Table 5.2, we can use (5.6) and (5.7) to get Table 5.3 and 5.4.

Channel Model	Slave number	Max. retry number	Before retry		After retry	
			$r_{DL}(s)$	$r_{UL}(s)$	$r'_{DL}(s)$	$r'_{UL}(s)$
Ring_10	6	1	2	1	2	2
Ring_100	47	1	3	2	3	3
RandArea Np_100	65	1	3	3	4	3
RandArea Np_ 200	130	1	4	3	4	4

Table 5.2: Simulation results of the slave node with maximum retry number and maximum repeater level

In Table 5.3, the minimum transmission times correspond to the best case, in which the slave can be reached by the master without repeater and the transmission is successful without retry. The maximum response times correspond to the worst case, in which the transmission needs retry and uses maximum repeater numbers to reach the destination slave. In the cases of $T_{ch} = 0.0264\ s$, the transmission times are smaller than the time constraints for substation automation which is mentioned in Chapter 2. However the transmission time of the case of $T_{ch} = 0.1128\ s$, can satisfy most of the application requirements.
If T_{wr} is configured to 57 time slots (≈1.5/0.0264) or 14 time slots (≈1.5/0.1128) which is equal to the maximum delay time for a response generated by the slave, we can get Table 5.4. In Table 5.4, one can see that the main component of the maximum and minimum response time to a request is the response generation delay, as the transmission time is small. If the network layer can get a precise T_{wr} value, the response time to a request can

Channel Model	T_{ch}=0.0264 s		T_{ch}=0.1128 s	
	$\max_s(T_{M\to s})$	$\min_s(T_{M\to s})$	$\max_s(T_{M\to s})$	$\min_s(T_{M\to s})$
Ring_10	0.22	0.0088	0.9306	0.0282
Ring_100	0.2992	0.0088	1.269	0.0282
RandArea Np_100	0.3256	0.0088	1.3818	0.0282
RandArea Np_ 200	0.3784	0.0088	1.6074	0.0282

Table 5.3: Response time to a command

Channel Model	T_{ch}=0.0264 s		T_{ch}=0.1128 s	
	$\max_s(T_{M\to s})$	$\min_s(T_{M\to s})$	$\max_s(T_{M\to s})$	$\min_s(T_{M\to s})$
Ring_10	1.8128	1.5752	2.8764	1.8612
Ring_100	1.9184	1.5752	2.6508	1.8612
RandArea Np_100	1.9712	1.5752	3.5532	1.8612
RandArea Np_ 200	2.024	1.5752	3.7788	1.8612

Table 5.4: Response time to a request

be smaller. In current case, the response time to a request can satisfy the requirement of individual metering reading time constraint (< 30 seconds).

Message transfer delay

In a Master/Slave model, since only the Mater can initiate a communication, the message transfer delay from a slave to the master will depend on the period with which the slave is polled by the master. Fig. 5.5 gives the model of this communication paradigm. If the message represents time constrained data such as an alarm, the message transfer delay must be guaranteed by correctly configuring the polling period.

Moreover, if the alarm messages are queued at the slave node, we should also configure the polling period according to the message arrival rate λ_i.

We assume that the master should poll every slave once per cycle. The polling cycle duration then gives us an idea about the network capacity for message transfer delay. In Chapter 2, the time constraint of alarm delay is limited to small than 30 seconds. Obviously, it is impossible to do polling cycle every 30 seconds for a large network such as RandArea Np_200 from Table 5.5. Nevertheless, we will give some idea about the transfer message delay based on master/slave mode. We consider that two typical network topologies: ring structure in MV power grid and the tree-like structure in LV power grid, are used in European power supply network [33]. We focus on analyzing the feature of message transfer delay in channel model of Ring_10 and RandArea Np_100.

In a polling cycle, the master will poll all slaves once at first. Then if ACKs bring the

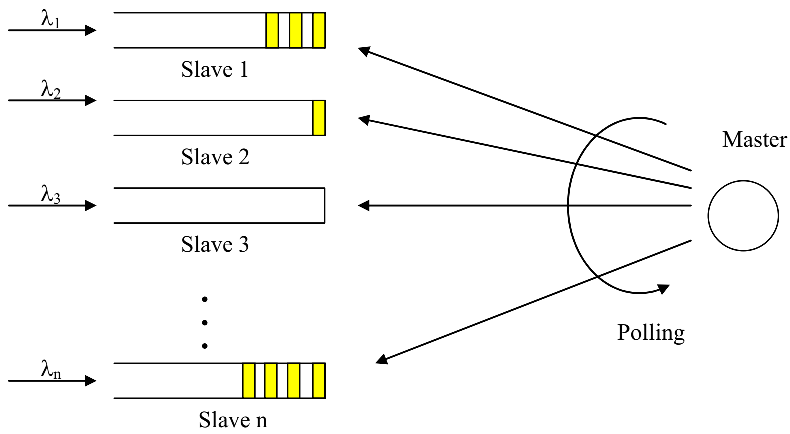


Figure 5.5: Pipeline of logical channels

Channel Model	Simulation results	T_{ch}=0.0264 s	T_{ch}=0.1128 s
		Average duration (s)	Average duration (s)
Ring_10	30.59	0.807576	3.450552
Ring_100	430.14	11.355696	48.519792
RandArea Np_100	420.25	11.0946	47.4042
RandArea Np_200	1034.25	27.3042	116.6634

Table 5.5: Average duration of a polling cycle

information that the slave still has alarm packet in the queue, the master polls those slaves again until alarm queues of slaves are empty. The master constructs a polling table to poll slaves for getting alarm packets as shown in Table 5.6.

We will give the simulation results with the following assumptions:

- T_{ch} is equal to 0.0264 s
- Periodic polling time is 30 s ($\approx$30/0.0264=1136 T_{ch})
- Alarm messages are generated in the slaves according to the same Poisson processes of intensity λ
- High traffic load in Ring_10 channel model is λ=0.02 packets/T_{ch} (i.e. 0.76 alarms/s); light traffic lode is λ=0.001 packets/T_{ch} (i.e, 0.04 alarms/s)
- High traffic load in RandArea Np_100 channel model is λ=0.001 packets/T_{ch} (i.e. 0.04 alarms/s); light traffic lode is λ=0.00001 packets/T_{ch} (i.e, 0.004 alarms/s)

1	Poll + ACK/Data Node 1	Poll Node (1...N) Get the alarm information and current alarm queue length
	Poll + ACK/Data Node 2	
	...	
	Poll + ACK/Data Node N	
2	Poll+Data packet 1 of Node 1	Poll slaves which have the alarm packet in the queue
	Poll+Data packet 2 of Node 1	
	Poll+Data packet M of Node 1	
	Poll+Data packet of Node 2	
	...	
	Poll +Data packet 1 of Node N	
	...	
	Poll + Data packet M of Node N	

Table 5.6: Polling table in a periodic polling cycle

Channel Model	Poisson distribution λ (packets/T_{ch})	Average message transfer delay (s)
Ring_10	0.02	13.82
	0.01	14.31
	0.001	15.04
RandArea Np_100	0.001	16.23
	0.0001	15.05

Table 5.7: Simulation results of average message transfer delay

In Table 5.7, the simulation results show that the application requirement can be guaranteed in certain alarm message generating rates. Unfortunately, if all slaves have very high payload, polling table will be long and cannot be finished in a polling cycle. The message transfer delay cannot satisfy the application requirement. We can roughly consider that the maximum λ in channel mode RandArea Np_100 is equal to 0.0023 packets/T_{ch} ($\approx 30/11.09/1136$). In fact, due to the variable average duration of a polling cycle and the randomicity of message arrival, the actual the maximum λ is smaller than 0.002 packets/T_{ch}. So, we propose to use random access scheme for this type of application in Chapter 7.

5.3.2 End to end performance in REMPLI PLC network

Recall that REMPLI PLC system consists of four main components. An Access Point (AP) is connected to a medium voltage (MV) PLC network. Bridges, which are responsible for connecting MV and LV PLC network segments, communicate with the APs (MV) on the top and with the Nodes (LV) at the bottom. A Node is the link between the field equipment and the low voltage (LV) PLC network.
The key of timing performance in REMPLI PLC network is the Transport Layer Round Trip Time (RTT) which is the duration from the time when a packet is sent by the AP, to the time when the consequent response from the Node is received.
Fig. 5.6 depicts a RTT duration of a request/response procedure, consisted of several different parts. These values are similar to those in the section 5.3.1. The subscripts of MV and LV are used to denote as the values in the MV and LV autonomous network.

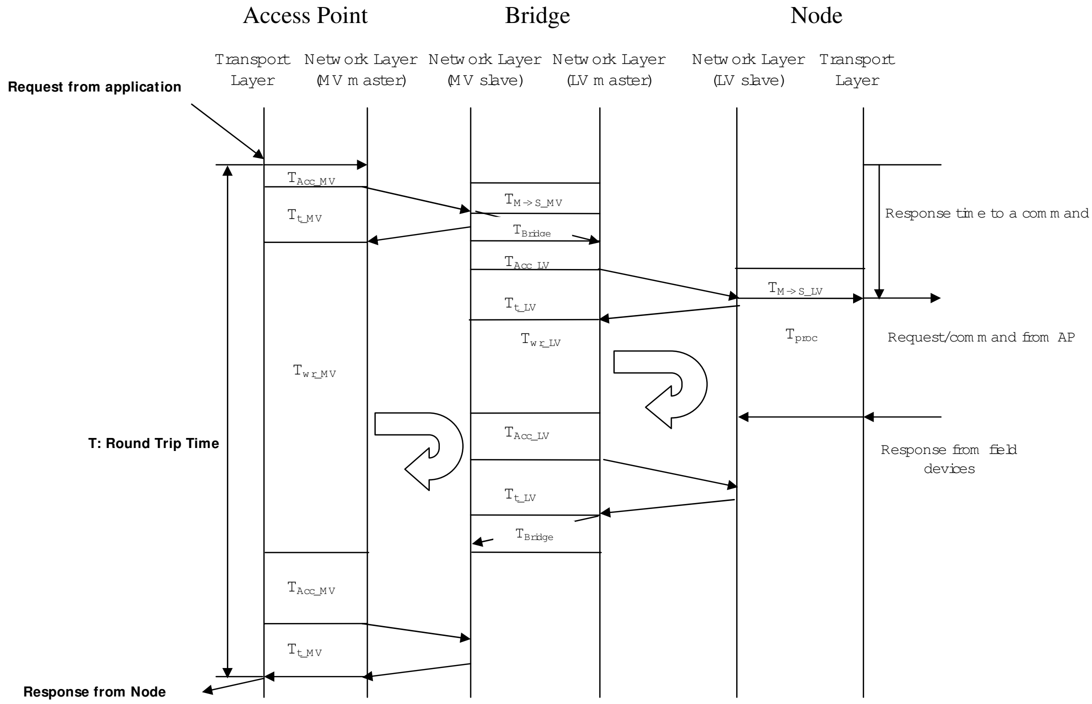


Figure 5.6: Schematic representation of Round Trip Time in REMPLI PLC network

In the following analysis we do not count small finite constant values such as T_{Bridge} (between 10 and 20 ms according to [81]). We assume that one packet request from the AP only generates one packet response from the Node. So, we can get the following formula of calculating RTT from Figure 5.6:

$$RTT = T_{acc\_MV} + T_{M \to s\_MV} + T_{acc\_LV} + T_{M \to s\_LV} + T_{proc} + T_{acc\_LV} + T_{t\_LV} + T_{acc\_MV} + T_{t\_MV} \tag{5.8}$$

In simulation scenarios, MV power grid is Ring_10 channel model and the MV slave part of the Bridge can be node 2 or node 6, while the LV power grid can use all available channel models. From the Physical emulator, we know that node 2 always communicates with master directly and without the retry in channel model Ring_10. In contract, node 6 communicates with master through one or two repeater and sometimes, the retry will be required.

Response time

In this section, we will study the timing performance of transmitting a command or a request in the REMPLI PLC network. The generic communication procedure is shown in Fig. 5.6.
The response time (T_c) to a command is defined as the time from the command packet ready for transmission in AP to that the packet arriving the LV Node s. After the AP sends the command message, the MV slave side of Bridge receives it and checks the destination address in the packet. If the destination address is a LV Node address, which the Bridge connects with, the Bridge transfers it to its LV master part. If the LV master has been transmitting some packets, this new packet must wait T_{acc_LV} for transmitting. The response time to a command can be represented by the following formula.

$$T_c(s) = T_{acc\_MV} + T_{M\to s\_MV}(Bridge) + T_{acc\_LV} + T_{M\to s\_LV}(s) \tag{5.9}$$

We recall (5.3) to calculate $T_{M\to s}$.
The response time T_r to a request is defined as the duration from the request packet ready for transmission in AP to the Response received from Node i. It is expressible as:

$$T_r(s) = T_{acc\_MV} + T_{t\_MV}(Bridge) + T_{wr\_MV}(s) + T_{acc\_MV} + T_{t\_MV}(Bridge) \tag{5.10}$$

and T_{wr_MV} should obey the following inequality.

$$T_{wr\_MV}(s) > T_{acc\_LV} + T_{t\_LV}(s) + T_{wr\_LV} + T_{acc\_LV} + T_{t\_LV}(s) \tag{5.11}$$

T_{acc_MV} and T_{acc_LV} are ignored as they are random and cannot be predefined. To assure that the AP can get the LV slave response from the Bridge, $T_{t\_LV}(s)$ should be the maximum value in (5.11), i.e., including the retry. T_{wr_LV} is configured to 1.5 second (57 T_{ch} (≈1.5/0.0264) or 14 T_{ch} (≈1.5/0.1128)). So, (5.9) and (5.10) can be simplified in the following ways.

$$\begin{aligned}
T_c(s) &= T_{M\to s_MV}(Bridge) + T_{M\to s_LV}(s) \\
&= \left(r_{DL}(Bridge) + n(Bridge) + 1 + \sum_{j=0}^{n(Bridge)-1} (2\cdot(j+1) + r_{DL}(Bridge) + r_{UL}(Bridge)) \right) \cdot T_{ch} \\
&-(n-1)\cdot T_s + \left(r_{DL}(s) + n(s) + 1 + \sum_{j=0}^{n(s)-1} (2\cdot(j+1) + r_{DL}(s) + r_{UL}(s)) \right) \cdot T_{ch} - (n-1)\cdot T_s
\end{aligned} \tag{5.12}$$

$$\begin{aligned}
T_r(s) &= T_{t_MV}(Bridge) + T_{t_LV}(s) + T_{wr_LV} + T_{t_LV}(s) + T_{t_MV}(Bridge) \\
&= (\sum_{j=0}^{n(Bridge)} (2\cdot(j+1) + r_{DL}(Bridge) + r_{UL}(Bridge)) \cdot T_{ch} - (n-1)\cdot T_s) \\
&+(\sum_{j=0}^{n(s)} (2\cdot(j+1) + r_{DL}(s) + r_{UL}(s)) \cdot T_{ch} - (n-1)\cdot T_s) + T_{wr} + (r'_{DL}(s) + 1 + r'_{UL}(s) + 1)\cdot T_{ch} \\
&-(n-1)\cdot T_s + (r'_{DL}(Bridge) + 1 + r'_{UL}(Bridge) + 1)\cdot T_{ch} - (n-1)\cdot T_s
\end{aligned} \tag{5.13}$$

Using (5.12), (5.13) and Table 5.2, we can get Table from 5.8 to 5.10. The minimum transmission times correspond to the best case, in which the slave can be reached by the master without repeater and the transmission is successful without retry in each MV and LV autonomous PLC networks. The maximum response times correspond to the worst case, in which the transmission need retry and use maximum repeater numbers to reach the destination slave in each MV and LV autonomous PLC networks.

LV segment	T_{ch}=0.0264 s		T_{ch}=0.1128 s	
Channel Model	Max.	Min.	Max.	Min.
Ring_10	0.2552	0.0176	1.0716	0.0564
Ring_100	0.3344	0.0176	1.41	0.0564
RandArea Np_100	0.3608	0.0176	1.5228	0.0564
RandArea Np_ 200	0.4136	0.0176	1.7484	0.0564

Table 5.8: Response time to a command (Node 2 as Bridge)

LV segment	T_{ch}=0.0264 s		T_{ch}=0.1128 s	
Channel Model	Max.	Min.	Max.	Min.
Ring_10	0.4664	0.0704	1.974	0.282
Ring_100	0.5456	0.0704	2.3124	0.282
RandArea Np_100	0.572	0.0704	2.4252	0.282
RandArea Np_ 200	0.6248	0.0704	2.6508	0.282

Table 5.9: Response time to a command (Node 6 as Bridge)

Comparing the performance constraints given in Chapter 2, the response time of a command is small than 10 seconds as shown in Table 5.8 and Table 5.9, while the response time of a response is small than 30 seconds as shown in Table 5.10 and Table 5.11. The DA and DSM application requirements can be satisfied.

Average polling time

The other performance metrics is the average polling time for the AP to poll all LV Nodes to get their data through a Bridge. We use this performance metrics to evaluate the timing duration for the periodic metering reading application, considering that this application requires the system to collect a large volume of data with time stamps from millions of customer, instead of real time data. Periodic metering reading application can be considered as the periodic task in the network layer. Assume that a utility has 1,000,000 customers that are on a monthly billing cycle. In the typical manual meter

LV segment	T_{ch}=0.0264 s		T_{ch}=0.1128 s	
Channel Model	Max.	Min.	Max.	Min.
Ring_10	2.0152	1.6456	3.7224	2.1432
Ring_100	2.1736	1.6456	4.3992	2.1432
RandArea Np_100	2.2528	1.6456	4.7376	2.1432
RandArea Np_ 200	2.332	1.6456	5.076	2.1432

Table 5.10: Response time to a request (Node 2 as Bridge)

LV segment	T_{ch}=0.0264 s		T_{ch}=0.1128 s	
Channel Model	Max.	Min.	Max.	Min.
Ring_10	2.3848	1.7512	5.3016	2.82
Ring_100	2.5432	1.7512	5.9784	2.82
RandArea Np_100	2.6224	1.7512	6.3168	2.82
RandArea Np_ 200	2.7016	1.7512	6.6552	2.82

Table 5.11: Response time to a request (Node 6 as Bridge)

reading system, the meters are read on a 20-day cycle. Thus, let us assume that the meters will be read automatically on the same 20-day cycle to minimize the impact on the established billing system. This means that 50,000 meters must be read per day. If the communication system is being or will be used for other applications, then we consider only 50% of the total availability to be allocated to the meter reading application [86].
Average polling time is from AP sending the polling LV Node 1 message to AP receiving the data of the LV Node N (where N is the total number of slaves in an autonomous PLC LV network). The transmission can be concurrent in the autonomous MV PLC network and autonomous LV PLC network. It means that the AP sends the next polling after it receives the ACK of last polling. At the same time, the LV master starts to poll the LV slave after receiving the polling packet from the MV slave.
Due to certification constraints imposed by some utility companies, it is unfortunately not allowed to temporarily maintain, metering data in any intermediary location, even for time stamped metering data. In this way, all requests sent by AMR Applications need to get the response directly from the metering devices. It requires extra polling for taking response data back to the master. In the limited bandwidth, this scheme is obviously extravagant.
We also give the solution of the installation of cache in the Node which permits storing historical data and makes them accessible to the AP via REMPLI PLC network, in considering that other utilities may not have those certification constraints. All samples that are being received from the metering devices are stored at the directly attached Node.

Every data is associated with a current time stamp. Real-time clocks of all nodes in the network and the AP are synchronized.

Without cache in the slave Since the data must be obtained directly from the metering device, each metering reading procedure is a request/response communication procedure. The whole procedure for the average polling time is shown in Fig. 5.7. It starts from AP sending a packet to the bridge for polling LV Node 1, ending with AP receiving the data packet of response from LV Node N.

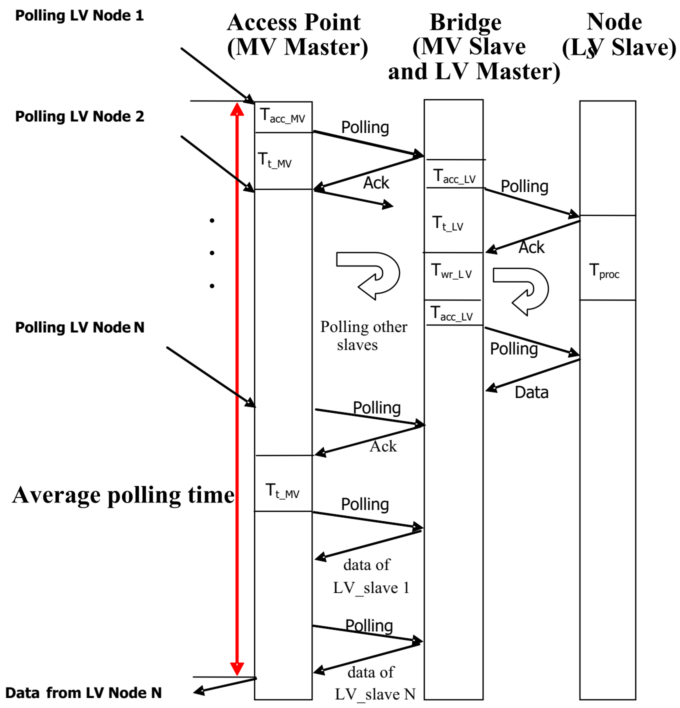


Figure 5.7: Schematic representation of average polling time at the Network Layer level in case of whithout cache in the slave

The problem is to set the time (T_{wr_LV} and T_{wr_MV}) when the master initiates a polling/data service to get response packets back from each slave. Considering that the generating time for a response (T_{proc}) varies between 200 ms and 1.5 s, if precise T_{proc} may be obtained through the destination slave or the upper layer of the bridge, LV master can set T_{wr_LV} to T_{proc}. By contraries, T_{wr_MV} is not possible to estimate by (5.11), since T_{wr_MV} is random while depending on the current tasks in the LV master, and become long when the polling packet containing the request has been continuously received from MV grid.
With a view to the periodic metering reading in AP, the first polling to the Bridge sends request packets for LV slaves and then the second polling to the Bridge is to achieve the possible data back. The second polling cycle should begin after T_{wr_MV}, when the first polling packets of all LV slaves have been sent. In the second polling cycle, the AP will add the additional polling service for getting the LV data back according to the queue

size of Bridge which stores the LV metering data.
So, we propose two possible solutions:

- Case 1: LV master has no knowledge of the value of T_{proc}, but LV slave has its response after a random delay (in a uniform distribution between 200ms and 1.5 s). For taking the response as soon as possible, after the commands have been sent, we use a periodical polling cycle to poll the slaves per 20 or 10 T_{ch} in an autonomous LV PLC network, and use a periodical polling cycle to poll the bridge per average LV polling time in the autonomous MV PLC network until each master (AP and Bridge) gets the data. In this way, the master does some useless polling without data, because the data of response has not yet been generated. Then the master wastes some bandwidths for trying to get the response from the slave.

LV segment	T_{wr_MV}	T_{ch}=0.0264 s T_{wr_LV}=20 T_{ch}		T_{ch}=0.1128 s T_{wr_LV}=10 T_{ch}	
Channel Model		Average duration (node_2 as Bridge)	Average duration (node_6 as Bridge)	Average duration (node_2 as Bridge)	Average duration (node_6 as Bridge)
Ring_10	28	3.53	3.88	10.36	12.39
Ring_100	418	34.86	35.66	132.41	145.9
RandArea Np_100	379	35.74	35.72	126.5	133.79
RandArea Np_ 200	983	84.67	78.31	303.26	309.33

Table 5.12: Average duration of Case 1

- Case 2: Bridge knows the maximum value of T_{proc}(1.5 s), and only send another poll after this delay. T_{ch} is set to the average duration of a polling cycle in LV autonomous PLC network.

From Table 5.12 and Table 5.13, the average duration of Case 2 is smaller than of Case 1 when the T_{ch} is 0.0264 second, whereas the average durations of Case 1 and Case 2 are roughly the same when T_{ch}=0.1128 s. For getting the 199 LV slaves metering data, the time is about 1.4 minutes in the case of T_{ch}=0.0264 s and is about 5.2 minutes in the case T_{ch}=0.1128 s.

Alternative solution with cache in the slave Although some power utilities do not permit data storing in the Node, some other power utilities have not this legality constraints. So, we can install the cache in Nodes in some cases. Cache is a type of the nodes' Random Access Memory (RAM) and on-board flash. The capacity of cache is sufficient to store measurement values for several days (or even up to moths for some data rows), despite it is not too large. In this section, we will evaluate the network performance for this solution.

LV segment Channel Model	T_{wr_MV}	T_{ch}=0.0264 s T_{wr_LV}=20 T_{ch}		T_{ch}=0.1128 s T_{wr_LV}=10 T_{ch}	
		Average duration (node_2 as Bridge)	Average duration (node_6 as Bridge)	Average duration (node_2 as Bridge)	Average duration (node_6 as Bridge)
Ring_10	28	3.25	3.74	10.62	11.83
Ring_100	418	31.72	35.53	132.46	146.27
RandArea Np_100	379	33.25	35.16	119.94	134.44
RandArea Np_ 200	983	71.89	62.73	303.34	309.45

Table 5.13: Average duration of Case 2

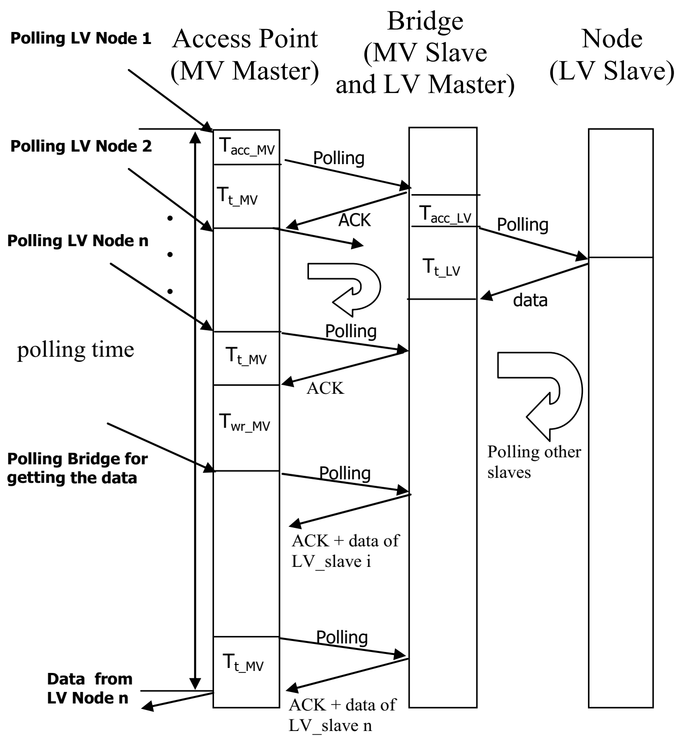


Figure 5.8: Schematic representation of average polling time at the Network Layer level in the case of with cash in the slave

When a LV slave has received a polling packet which contains a request of periodic metering reading from AP, a data packet which contains the metering information storing in the cache of the Node, will be sent back to the LV master. This method has the maximum utilization of bandwidth. But in the MV autonomous network, MV master still needs T_{wr_MV} to give a new polling for getting back the data of LV Node from Bridge. T_{wr_MV} can be

$$T_{wr\_MV}(s) = T_{acc\_LV} + T_{t\_LV}(s). \tag{5.14}$$

Unfortunately, $T_{acc_{LV}}$ cannot be predefined. So, we configure T_{wr_MV} to the average polling time of the LV autonomous PLC network, for assuring that the AP could get the data back from the bridge when it polls the bridge again after the request packet has been send.

LV segment Channel Model	T_{wr_MV}	T_{ch}=0.0264 s		T_{ch}=0.1128 s	
		Average duration (node_2 as Bridge)	Average duration (node_6 as Bridge)	Average duration (node_2 as Bridge)	Average duration (node_6 as Bridge)
Ring_10	28	1.48	2.27	6.32	9.70
Ring_100	418	19.07	16.64	81.32	71.11
RandArea Np_100	379	18.00	20.89	76.92	89.27
RandArea Np_ 200	983	42.77	32.65	182.73	139.48

Table 5.14: Average duration

In Table 5.14, the time is about 0.7 minutes in the case T_{ch}=0.0264 s and is about 3.0 minutes in the case T_{ch}=0.1128 s, for getting the 199 LV slaves metering data. Comparing Table 5.12, Table 5.13 and Table 5.14, the average duration of installation of cache in LV slave is half of that in the case of without an installation of cache in LV slave.

5.4 Conclusion

In this chapter, we analyzed and simulated the REMPLI system to evaluate the communication performance by three metrics: response time to a command, response time to a response, and the average polling time. From those analysis and simulation results, we can draw a conclusion that REMPLI PLC system can meet the demands of DA/DSM application.

Chapter 6

Dispatcher

In Chapter 4 and 5, we have shown the good behaviors of SFN protocol from the network performance point of view (minimized network transfer delay with small protocol overheads) and its feasibility for meeting the requirements of REMPLI applications in an autonomous PLC network. However, the PLC network for REMPLI applications is in a multi-tasking environment, because functions such as AMR, load management, on-request meter reading, and other distribution control and monitoring functions are simultaneously implemented. And those applications require different QoS. The REMPLI PLC must also allow the possibility to serve those applications differently in order to satisfy their different QoS requirements.
A dispatcher within the network layer of the master is proposed to provide a QoS mechanism by permitting an optimal sharing of the network bandwidth among different traffics generated by the applications.
The purpose of this chapter is to describe the traffic policy adopted in the REMPLI PLC under the name of Dispatcher, and to show its benefit in supporting different application traffics with differentiated QoS. In this sense, the Dispatcher thus ensures the QoS mapping between an application and a REMPLI PLC network traffic class.

6.1 Traffic classes and priority levels in network layer

The REMPLI system divides the traffics into 3 generic classes:

- Network Management: this traffic permits to maintain and monitor a network for the autonomous PLC network to function normally. It requires the master and the slaves to exchange information periodically.

- Events: this traffic class gathers data information originated by REMPLI Nodes, mainly critical alarms generated by metering devices or by the REMPLI Node itself.

- Requests: three types of requests from application:
  - Control and Monitoring (critical request done by the AP)
  - Metering (normal request done by the AP)

- File Transfer (update of system software)

At the Network Layer level, the previous traffic classes can generate two types of network traffic: periodic and aperiodic traffics. Requests from master applications are sent from the transport layer to the network layer and stored in the aperiodic request queues, classified by priority levels. The Network Layer provides three priority levels: 0, 1 and 2 for aperiodic task, where a lower number means a higher priority:

- Priority 0 (CRITICAL), at Master side only.
- Priority 1 (EMERGENCY), at Master and Slave sides.
- Priority 2 (NORMAL), at Master and Slave sides.

Note that as critical commands using priority 0 do not generate a critical response, this priority does not exist at the slave side.
The Network management and event traffics are periodic. There are four different types of periodic packets.

- PLST (Physical Layer Service Telegram): this packet is periodically transmitted to all slaves. It contains the physical management information, such as the number of logical channels per logical unit. It also has the purpose of being a sign of life from the Master. This type of packet is a part of the Network Management traffic class.
- Logon: this type of packets should be periodically sent with small period in the start-up phase, then with large period during normal phase to allow the logging of new slaves in the network. After all slaves are connected to the master, the master stops transmitting this packet. Packets of this type belong to the Network Management traffic class.
- Status: sending periodic status packets permits to verify slaves' liveliness. This kind of packets is a part of the Network Management traffic class.
- Application Poll: this type of packets is used to periodically verify the existence of aperiodic packets in the slave queues. Thus, we guarantee that a slave will have a minimum throughput towards the slave. This packet type is a part of the Events traffic class.

Depending on the type of packets, the respect of periodic polling can be more or less strict. Due to this fact, we distinguish two types of periodic polling level: hard periodic and soft periodic. Hard periodic polling refers to periodic polling that has stricter constraints relatively to the time period. For example, delaying an Application Poll can make an important event generated at the slave node out of date. Soft periodic polling adds a certain timing relaxation relatively to hard periodic polling. That is, if one deadline is missed, no major problem arises. According to the requirements, the packet traffic precedence is defined in Fig. 6.1, for every time that the dispatcher mechanism is called.

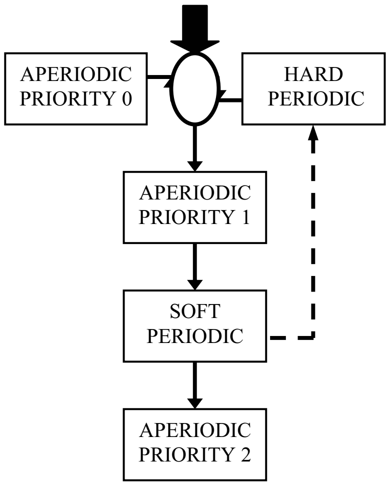


Figure 6.1: Representation of the process of the different types of traffic

6.2 System constraints

In REMPLI PLC networks, a dispatcher is used at each master node to schedule the traffics for guarantee different QoS requirements. The main objectives of the dispatcher are to guaranteeing periodic traffic while minimizing the average response time of aperiodic one in the best-effort way, and to optimize the bandwidth utilization. The functions of dispatcher will be called to decide which is the next packet to be sent, whenever a logical channel has finished its previous communication task.
Since in REMPLI system there may be more than one different Network Layer protocols for different logical channels, the dispatcher should work according to the various queues and tables, which are stored by each Network Layer Unit. Fig. 6.2 illustrates this structure.
Contrary to simpler Master-Slave polling systems, our system must handle several types of constraints.
Constraint1: Each node must be guaranteed to be visited at least every T seconds (measured in time slots in REMPLI PLC). There might be different types of periodic packet and different periods.
The configuration of the periodic polls is done based on the demands of the application. This type of configuration (or dynamic re-configuration) relies on a required schedulability test before it can be applied to the system.
Constraint2: Response time of the aperiodic traffic should be minimized. Also, it should follow the priority hierarchy provided by the three different priorities.
Constraint3: Response packets of slaves are more important than new request packets from the Master, which allows to reduce the delay between a request and its response.

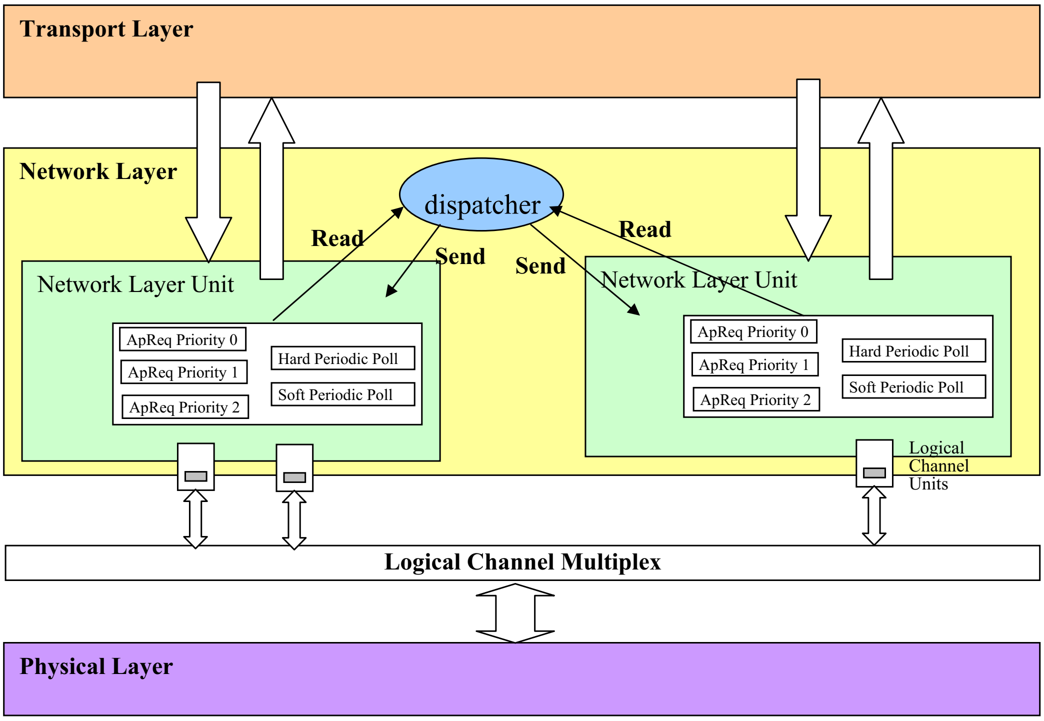
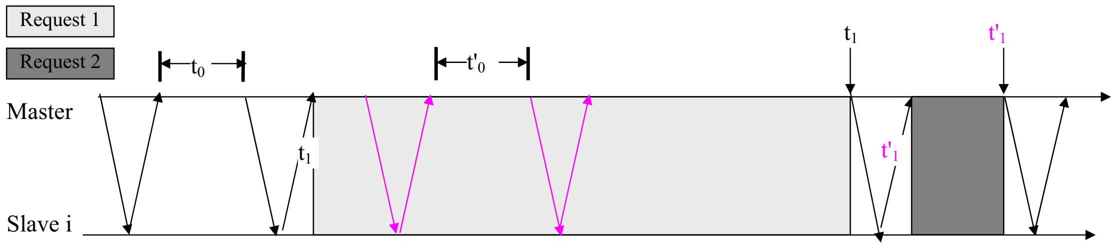


Figure 6.2: Dispatcher location and priority system for each Network Layer Unit

6.3 Aperiodic traffic

The traffic packets from the transport layer differentiate via priorities and store in different buffer queues in the Network Layer. Each buffer queue follows FIFO principle. Although some strategies could optimize the bandwidth efficiency, they are more complex and susceptible to starvation situations. For example, when sending an empty periodic poll to a node, it would be more efficient to withdraw a packet from the queue for the same destination and insert this packet into the periodic poll packet. However, certain packets can wait indefinitely for their turns of transmission.
For request/response network service in a master/slave network, it is important to manage the time for taking the responses back from the slaves, since preparing a response in slaves can take between 200 ms and 1.5 seconds, depending on the type of application running. In Chapter 5, we simply configure T_{wr} to 1.5 seconds and it assures that the master can get the data from the slave by the second poll, although the master may have waited for a unnecessary time. If the time for preparing a response is not known in the system, the master should do a series of polls to try to get the responses from the slaves after sending a request packet. Otherwise, the Master would not do any unnecessary poll before the predicted generation time of the response. If we assume that this time value is provided by the Application Layer of the slave side, it is necessary to transfer it to the Master side.
Fig. 6.3 shows the proposed strategy, based on two values: $t_0(i)$ and $t_1(i)$ which represent the time in waiting for doing the poll to fetch t_1 and the time at which the slave i has

generated the response respectively.



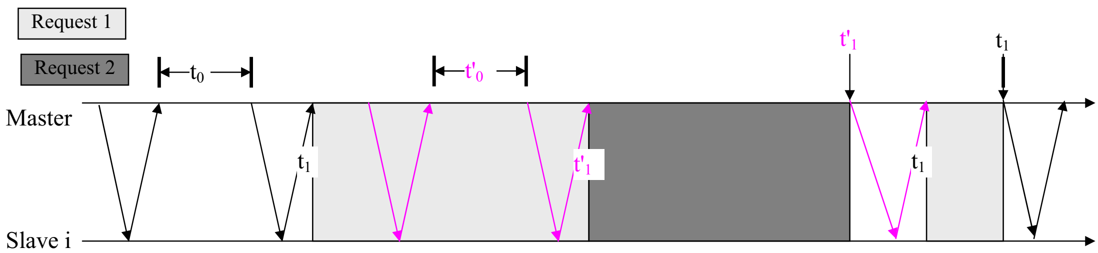


Figure 6.3: Master poll strategy for determination of Slave Application Layer response

The value $t_0(i)$ can be a predefined value that each master has for all slaves. Thus, it is only necessary to add $t_0(i)$ to the current time in order to calculate the precise time when the $t_1(i)$ value is already available in the slave. The value t_1 is an absolute time (in time slots). After t_1, new poll packet for slave i is generated in the master for getting the response back.
Since a Slave can have several responses for several requests, different t_1 can arise. The Slave side provides only the earliest t_1, if it is different from an already changed one. Thus, the slave has to keep track of all t_1 for the node, and only delivering new information to the Master.
As shown in Fig. 6.3 b, the earliest generation of another response implies that the transmission of value t'_1 is sent to the Master. This reduces the waiting time that was previously set to t_1. Only after t'_1, the slave retransmits again the t_1 value which will be inserted into a confirmation packet.
Obviously, this mechanism can shorten T_{wr} in an autonomous PLC network. However, if the request/response is from the Access Point to the LV Node, we find there is a problem that the MV slave in the Bridge is difficult to get the information of t_1 for the MV master, since the data is obtained from LV slave and the transmission time is uncertain. So, in Chapter 5, we propose that the MV master can get the maximum $T_{wr\_MV}(s)$ for the aperiodic traffic by (5.11), which contains the maximum transmission time of LV slave s and the maximum response generation time. Unfortunately, it also cannot guarantee that the LV slave data is ready when the MV master polls the MV slave due to the multitasking environment in the LV master. This mechanism is also useful for setting the third poll.

6.4 Periodic traffic

The network management at the master side periodically generates packets for every slave, which permits to exchange information such as node status, liveliness, etc. And the event traffic needs to periodically poll all slaves to guarantee a minimum bandwidth for every slave. For periodic polling of slave node service, the most important thing is that each slave is regularly polled with a bounded jitter. The periodic packet class i for polling slave i, denoted by $P_i(C_i, T_i, \bar{D}_i, X_i)$, is characterized by the following attributes:

- C_i : execution time of packet i. C_i corresponds to the transmission time for polling slave i which is equal to the number of time slots between transmitting a packet to the slave i and receiving a confirmation packet at the master.

- T_i : period of packet i. It means that, within every T_i interval time slots, an instance of packet class i is transmitted.

- $\bar{D}_i$: the relative deadline of packet i. In our system, we suppose$\bar{D}_i = T_i$, for all i.

- X_i : defines whether packet i has a hard or soft periodic constraint through a boolean variable, i.e, $X_i \in \{0, 1\}$ in which 0 means hard periodic and 1 means soft periodic. Hard periodic means that in each period a packet transmission is necessary, while the missing of a deadline is not problematic in the case of soft periodic packet.

6.4.1 Execution time C_i

When explicitly referring to activation j of packet class i, where $j \in$ N, the adopted index notation is i, j (e.g. the j^{th} activation of packet class i execution time is denoted by $C_{i,j}$). From Chapter 4, it is known that $C_{i,j}$ depends on downlink $r_{DL}(i, j)$ and uplink $r_{UL}(i, j)$ values, and also retry time. Thus, the execution time $C_{i,j}$ is defined by:

$$C_{i,j} = r_{DL}(i,j) + 1 + r_{UL}(i,j) + 1 + \sum_{k=1}^{n(s)} (r_{DL}(i,j) + k + 1 + r_{UL}(i,j) + k + 1) \qquad (6.1)$$

where $n(s)$ represents the retries time used for this polling slave i. In the worst case, if the slave is not alive, polling this slave will last until reaching the maximum retries time configured by the system.

In Chapter 4, we conclude that there are not retries in most cases and only one retries for certain slaves. So, assuming that the worst case always includes the maximum number of retries is a very pessimistic approach, since in a vast majority of the cases the retries are not necessary. A coarser calculation can be provided by eliminating the retries part and is shown below:

$$C_{i,j} = r_{DL}(i,j) + 1 + r_{UL}(i,j) + 1. \qquad (6.2)$$

In fact, $r_{DL}(i, j)$ and $r_{UL}(i, j)$ cannot be forecasted before $(j-1)^{th}$ activation of packet class i executes. After each transmission, new r_{DL} and r_{UL} are stored in the master memory.

When the values are used for calculating (6.2), they can be available directly from the master. It is not necessary to mention j. So, by ignoring j, (6.2) can be simplified to be

$$C_i = r_{DL}(i) + 1 + r_{UL}(i) + 1. \tag{6.3}$$

Although obviously it does not guarantee a strict real-time system, we suppose that our system accepts a certain jitter. Therefore, we adopt the last formula for the calculation of the transmission task execution time.
The periodic polling of a slave can be guaranteed in different ways: static cyclic scheduling or dynamic scheduling (fixed priority, Earliest Deadline First, Least Laxity First, etc).

6.4.2 Static Periodic Polling

Static polling approach constructs a simple static polling table with fixed times for the periodic traffic, since the start of the system, the periodic packets have already set priorities. It permits a lighter calculation of the periodic polling. This provides a simpler and faster calculation of the next periodic packet to be sent.
Since periodic polling is statically configured, aperiodic traffic can only be sent whenever no periodic polling is scheduled to be sent. In case of an aperiodic packet arriving right before the periodic packet is scheduled, the aperiodic packet is transmitted. However, this extra time is always bounded by the transmission task execution time, which is an acceptable delay.

6.4.3 Dynamic polling approach

Dynamic periodic polling provides a more complex approach to periodic polling. The main objectives consist of guaranteeing the periodic traffic constraints and minimizing the average response time of aperiodic traffic in the best-effort way. To achieve this objective, the dual-priority (DP) scheduling strategy [98] is adopted. The idea is to do the periodic polling as late as possible so as to make the possibility to serve the aperiodic traffic as early as possible.

Dual-Priority scheduling

In DP, periodic packets possess two levels of priorities: low and high levels, whilst aperiodic packets are scheduled using a medium priority level. Accordingly, periodic packets can run immediately at a low level while there is no aperiodic traffic. In the presence of aperiodic traffic, a periodic task can only be sent when promoted to the high priority level, as late as possible. To calculate the time instant when a periodic packet is promoted, it suffices to calculate the response time for that packet. This will always guarantee the transmission of the periodic packet by its deadline. So, the promotion time L_i for periodic packet i scheduled according to DP is defined as:

$$L_i = \bar{D}_i - R_i$$

where R_i is the response time and $\bar{D}_i$ is the relative deadline for the transmission of periodic packet i. The calculation of the response time R_i of the periodic packet $R_i =$

$C_i + a_i^q$ is defined by P_i, according to the follow recursion [87]:

$$\begin{cases} a_i^q = 0 \\ a_i^{q+1} = \max\limits_{j \succ i}(C_j) + \sum\limits_{j \prec i} \left(\left\lfloor \frac{a_i^q}{T_j} \right\rfloor + 1 \right) \cdot C_j \end{cases}$$

where the higher the packet index is, the lower its priority is. When $a_i^{q+1} = a_i^q$, response time a for slave i is found.
The set is feasible if the response time of every periodic packet is inferior to the relative deadline. Otherwise, it is unfeasible.
Based on this policy, our system must undergo a different approach for the calculation of the promotion time. It deals with the fact that the execution time C_i is random, and therefore cannot be calculated off-line. Moreover, we deal with non-preemptive tasks, which are the case of the packet transmissions in the network and therefore have to wait until the transmission completion. We introduce the notion of promotion time α, which allows to guarantee that in at least one call of the dispatcher. Every periodic packet can be successfully promoted without surpassing its deadline.
Thus, each time the dispatcher is called, it is necessary to verify on-line, for all periodic packets, if a promotion time is due. Packet class P_i is promoted to the high priority level if the following inequality is true in the current period:

$$\left\lceil \frac{\mathrm{H}}{T_i} \right\rceil \cdot T_i < L_i$$

where H is the current time of the system, in time-slot units since at the start of the system.
Fig. 6.4 shows an example of the method. One can observe that the dispatcher is called at least once during the promotion period of packet i, between L and $L + \alpha$, since α corresponds to the blocking factor due to non preemption (i.e, the maximum execution time of a packet, as considered in the definition of the response time).
In case of the existence of a mixed periodic and aperiodic traffic, the issue deals with knowing if an aperiodic packet can still be served, or if an urgent periodic packet should be treated first.
In order to apply this approach, the Network Layer must associate, for each slave node, the *Current Polling Deadline* (D_i in the case of slave i) of the slave node using the following formula:

$$D_i = H + T_i + C_i.$$

Before the Dispatcher makes a decision, it is necessary to examine if there is any packet class P_i that could take the system infeasible 'soon'. In this case, we have to do a periodic polling on the slave with the closest D_i, in order to guarantee that slave i can still be polled before current time reaches it. This decision is made when the following inequality is true:

$$D_i - 2\alpha < C_i + \sum_{j \prec i} C_j + \sum_{j \prec i} \left\lfloor \frac{t_i - C_i}{T_j} \right\rfloor C_j < D_i - \alpha$$

where α is a security margin for assuring the feasibility of all periodic polls, with a minimum value of $\max\limits_{i \in \mathbb{S}}\{C_i\}$.

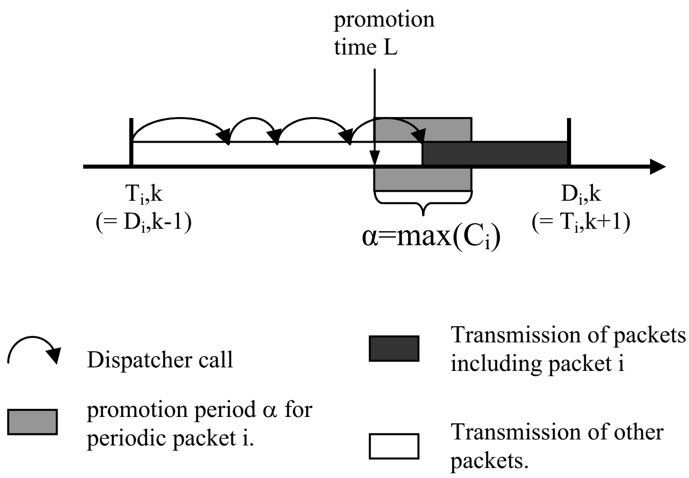


Figure 6.4: Dual-Priority scheduling policy, with successive dispatcher calls until arrival into a packet promotion period

Dual-Priority Scheduling with Deadline Relaxation

The cost of computing the promotion time (and consequent promotion period) for every periodic packet is very high in dual-priority scheduling. This relates to the fact that the periodic packets have a deadline which ought to be considered. Although this cost cannot be quantified at a design stage, we predict that the timing constraints in an implementation would be stringent. Therefore, in this section, we propose an alternative approach.

We now show that by relaxing the deadline constraint, it is possible to build a simpler and faster dispatcher based on the Dual-Priority policy and with respect to the most important constraint in our system: the periodicity. The idea is to consider, for each periodic packet, its deadline as the promotion time L' to have $L' = T$. In this way, the high cost in calculating the promotion instant that was inherent in the previous approach is inexistent since we have a static promotion time.

In Fig. 6.5, an example of both approaches is depicted for execution of the same periodic packet. The upper time line represents the previous approach where the promotion time is calculated based on the deadline of the periodic packet. The lower time line represents the approach where the promotion time is dictated by the arrival of its deadline. In this case, since several periodic packets can have the same promotion time, in the worst case, a periodic packet P_i can have a maximum bounded jitter of $T_i + \alpha$. However, it still accomplishes the needed sense of periodicity.

This approaches also influences on the allowed maximum charge of the periodic traffic. That is, the relaxation of the deadline constraint permits the increase of the periodic packet to a maximum charge of 1. Thus, one only needs to guarantee that:

$$\sum_{i=1}^{M} \frac{C_i}{T_i} < 1$$

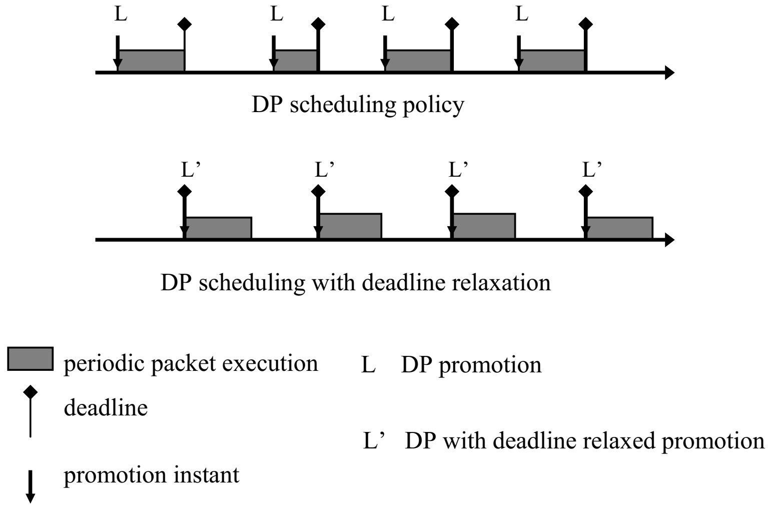


Figure 6.5: Promotion time of DP scheduling and DP scheduling with deadline relaxatio

where M is the number of periodic packet classes.

6.5 Simulation results

We will compare three approaches of dispatcher mechanisms described in the following:

- Static periodic polling
- DP scheduling
- DP scheduling with deadline relaxation

There are two comparison indexes: response time of aperiodic packets and acceptance percentage of the packet in the queue of the network layer. The former one is the time duration that the packet has to wait in the Network Layer queues until being sent to its destination. It permits to evaluate the minimization of the delay of the aperiodic packets, at the master side. Rejection of a packet is caused by a queue buffer overflow.
The comparison will be done in a simple scenario with channel model Ring_10, where the master will send aperiodic packets to all slaves according to a uniform distribution.

6.5.1 Static schedule vs dynamic schedule

We simulate two approaches of the dispatcher mechanism. The first approach will permit the aperiodic traffic to send only when no periodic polling is scheduled to be sent. The second approach is DP scheduling policy.
In the first simulation, we have a single priority for the aperiodic traffic which is always with a lower priority than periodic packets. Afterwards, two levels of aperiodic priorities are available, which permit to analyze the aforementioned promotion strategy, as presented in the dynamic periodic approach of the dispatcher mechanism.

Single aperiodic queue scenario

In this scenario, there is a single aperiodic queue, with a buffer of 40 packets and packets are generated every 5 to 40 time slots. The simulation results are shown in Fig. 6.6 and Fig. 6.7. We can conclude that the dynamic dispatcher approach provides a better result than that of the static approach. Moreover, the percentage of packets accepted is also higher in the dynamic dispatcher.

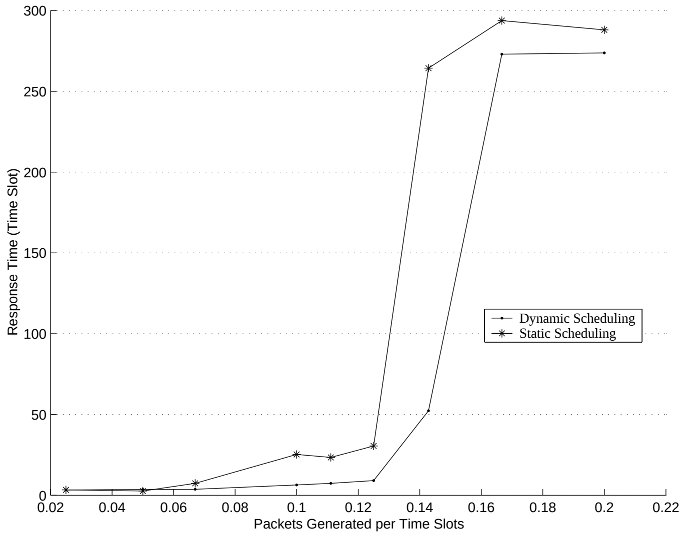


Figure 6.6: Response times of aperiodic packets

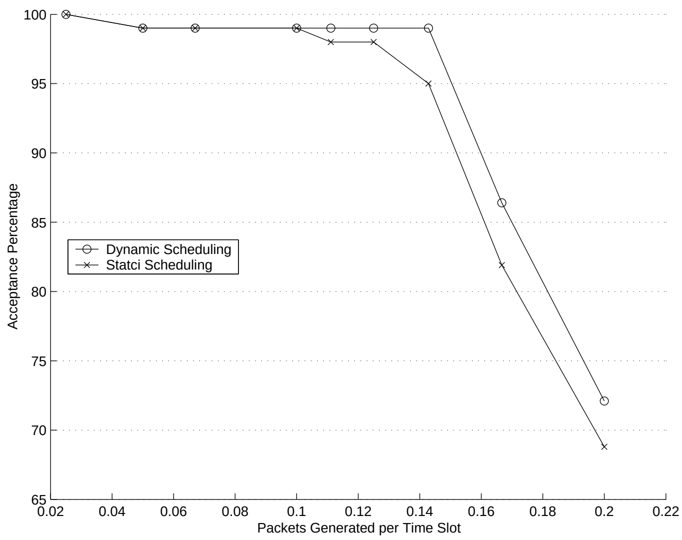


Figure 6.7: Percentage acceptance of aperiodic packets in the queues buffers

Two aperiodic queues scenario

We also simulate the scenario in which there are two aperiodic queues, each with a buffer of 20 elements and aperiodic packets in the queue 1 have a higher priority than soft periodic packets. However, in the case of missing deadlines, these soft periodic packets are promoted to the hard periodic table, with a higher priority than any aperiodic queue. This avoids missing the deadline again.

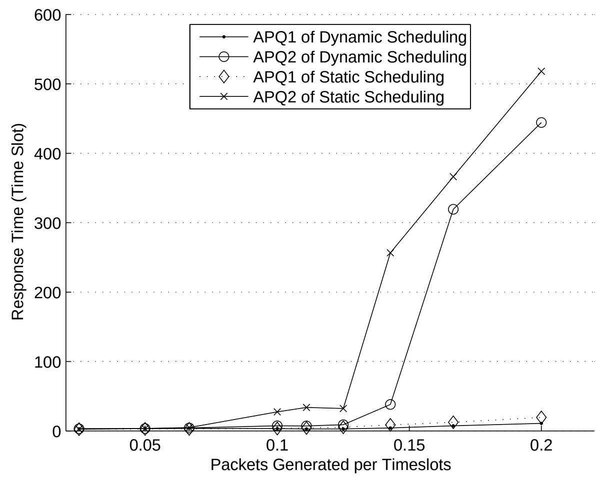


Figure 6.8: Response times of aperiodic packets with two aperiodic queue

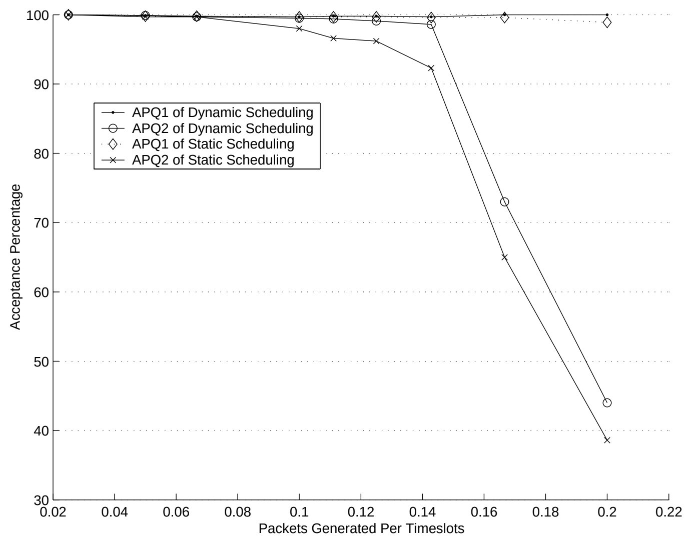


Figure 6.9: Percentage acceptance of aperiodic packets in the queues buffers

According to the simulation results presented in Fig. 6.8 and Fig. 6.9, it is possible to verify that the dynamic dispatcher valorises the highest aperiodic priority (aperiodic queue 1) with regard to the lowest aperiodic priority (aperiodic queue 2). On the other hand, since there is a time fixed periodic traffic with the static dispatcher, the aperiodic traffic increases the response time with higher loads in both queue buffers.

6.5.2 DP vs DP with deadline relaxation

The first approach is based on the dynamic calculation of the promotion time of periodic packets, whereas the second approach is based on a static promotion of the same type of packets.
The comparison will be done in a ring network with one master and nine slaves, where the master sends aperiodic packets to all slaves in a uniform manner, but always respecting the following periodic traffic:

- $P_0\,(C_0, 255, 255, 1)$ for slave 0
- $P_i\,(C_i, 3840, 3840, 0)$, i=0,... 8
- $P_i\,(C_i, 378, 378, 1)$. i=0,... 8

Single Aperiodic Queue scenario

The results are given in Fig. 6.10 and Fig. 6.11. Fig. 6.10 shows the delay time of transmission of the aperiodic packets. The percentage of packet acceptance in the queue buffer is high, while the difference between the two approaches is low.

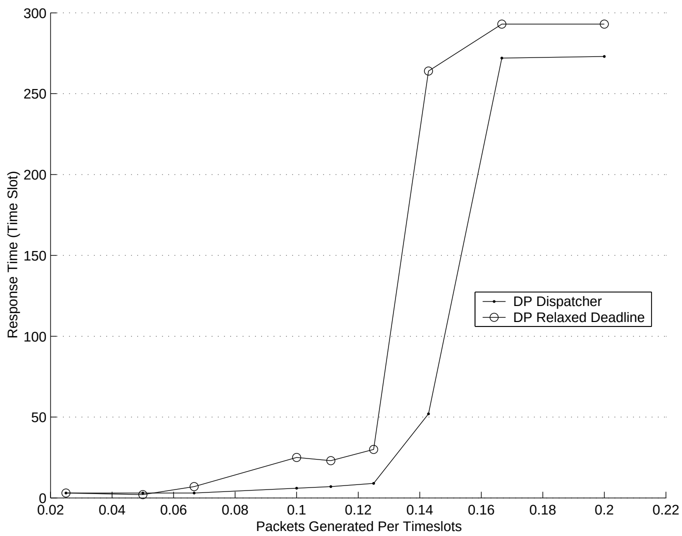


Figure 6.10: Response times of aperiodic packets

Relatively to the packet acceptance in the queue buffers, it can verify its appearance, although the DP dispatcher approach is slightly better.
Fig. 6.12 shows that the DP dispatcher approach meets the periodic packet deadlines,

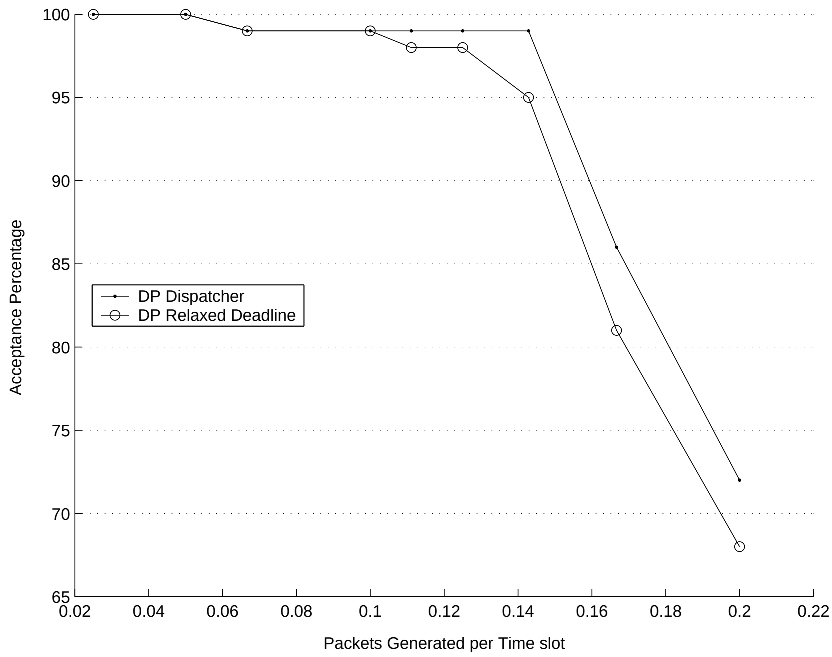


Figure 6.11: Percentage acceptance of aperiodic packets in the queues buffers

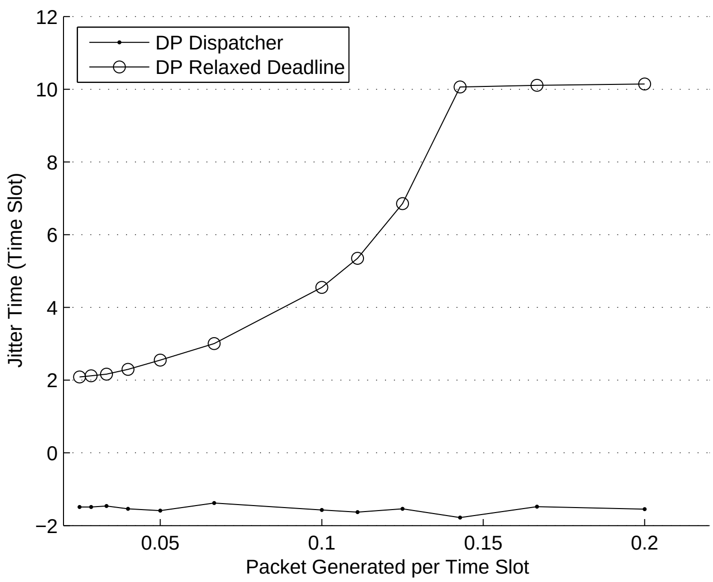


Figure 6.12: Jitter of the periodic packets where jitter is equal to zero, with a single aperiodic queue

while the DP relaxed deadline dispatcher approach does not. However, the jitter is not very important, especially when the aperiodic traffic load decreases.

Double Aperiodic Queues scenario

The simulation scenario with two aperiodic queues (Aperiodic Priority 1 and Aperiodic Priority 2 queue buffers) has the purpose of verification of the promotion strategy of periodic packets. In this way, aperiodic packets in the first queue (Aperiodic Priority 1) have a higher priority than soft periodic packets. However, in the case of missing deadlines, these soft periodic packets are promoted to the hard periodic table, with a higher priority than any aperiodic queue, which avoids to miss the deadline again.

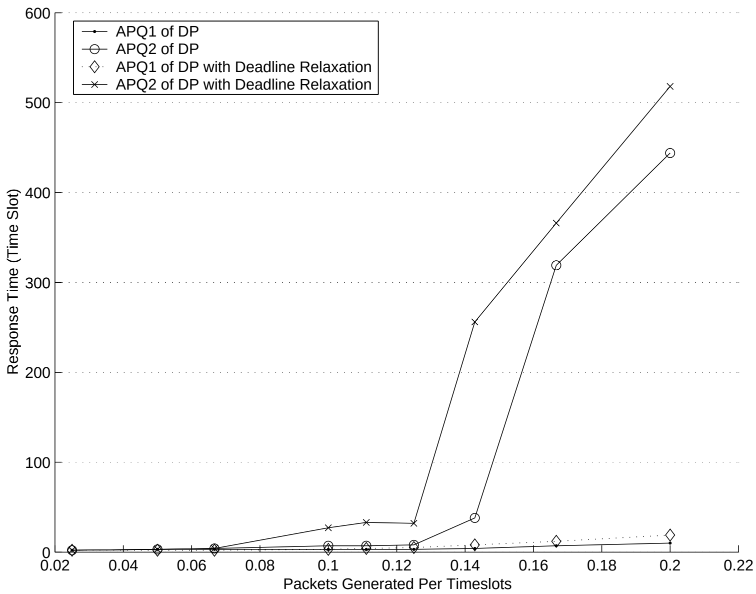


Figure 6.13: Response times of the aperiodic packets

The introduction of the promotion from soft to hard periodic level shows that with a high load (aperiodic packets generated every 5 to 8 time slots), the aperiodic priority 2 suffers greatly in both approaches, while aperiodic priority 1 still maintains a reduced delay of transmission (check in Fig. 6.13). Again DP dispatcher approach performs better than the DP relaxed deadline approach. The same high load provokes a lower acceptance of aperiodic packets (one out of two packets rejected) as observed in Fig. 6.14.
As shown in Fig. 6.15, which shows the jitter of periodic packets, with aperiodic packets generated for aperiodic queues 1 and 2, we find the same behavior as that in the single aperiodic queue scenario. Thus, we can say that what influences the jitter is the load of the aperiodic queues in the system but not its priority for the deadline relaxed dispatcher approach.
These simulations show that the DP dispatcher approach produces better results. However, the DP relaxed deadline dispatcher still allows maintaining a good performance

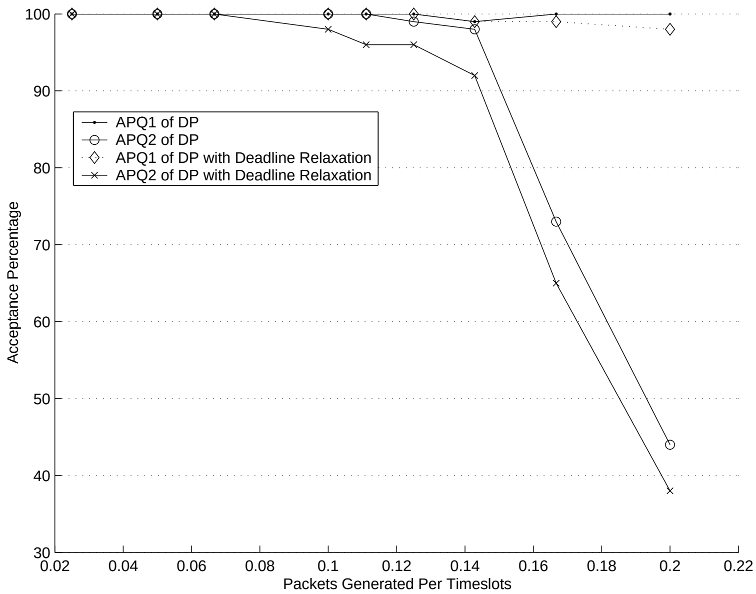


Figure 6.14: Percentage acceptance of aperiodic packets in the two queue buffers

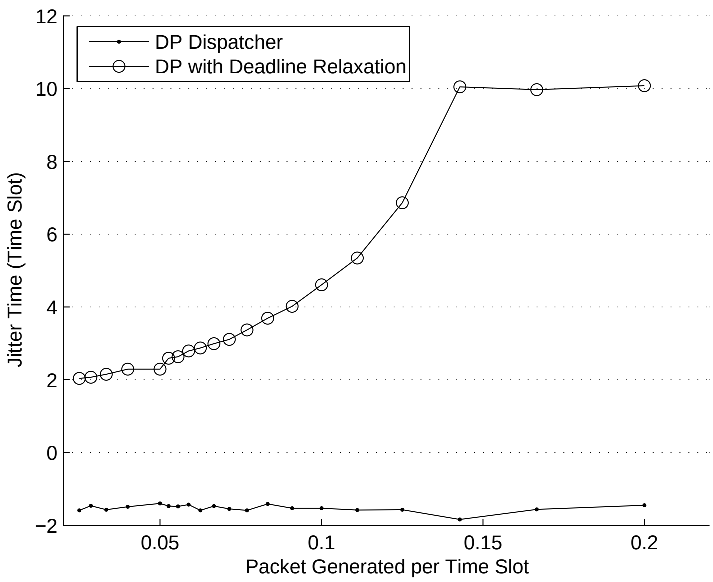


Figure 6.15: Jitter of the periodic packets

relatively to the first approach, while decreasing the computation time for the calculation of the promotion time of the periodic packets.
Nevertheless, when comparing the simulation speeds of computations of both approaches, the relaxed deadline dispatcher is considerably faster.

6.6 Towards dispatcher implementation

Based on the previous considerations, this section integrates all information required for the interaction of the dispatcher with the auxiliary structures, either periodic or aperiodic. On the left hand side of Fig. 6.16, different aperiodic request data is divided into 3 priority levels. On the right hand side, the periodic data is represented by two poll tables. These structures resume and conform to the constraints that were previously presented in the document.
Thus, every time the dispatcher is called by the Network Layer to decide which packet to send to which node, it verifies these structures.

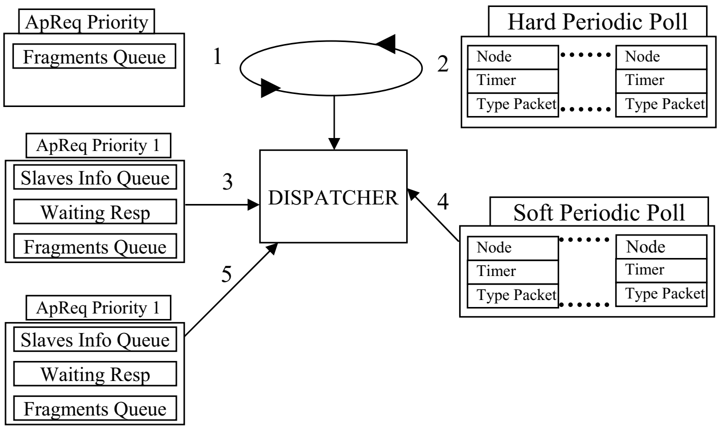


Figure 6.16: Detailed dispatcher structures

The order in which the dispatcher inspects the structure is graphically shown in Fig. 6.16 and noted in Table 6.1:

Priority Level	**Packet Type**
1 - Aperiodic Requests Priority 0	Critical
2 - Hard Periodic Polling	Application Poll
3 - Aperiodic Requests Priority 1	Emergency
4 - Soft Periodic Polling	PLST, Logon Packet, Status Poll
5 - Aperiodic Requests Priority 2	Normal

Table 6.1: Response times of aperiodic packets

Although network management service, e.g. status poll, is given a lower priority - soft

periodic polling class, this service needs to be guaranteed, because in a communication network, it is vital to know linking status and transfer routing information.
We provide an approach to dynamically increase the priority level of the soft periodic polling when these polling tasks are required to meet their deadlines, by promoting them to the hard periodic polling table.
The management of levels 1 and 2 (Aperiodic Requests Priority 0 and Hard Periodic Polling) is based in a Round Robin scheduling strategy. This permits to multiplex both traffic. It guarantees the execution of the critical commands transmitted by the applications, and at the same time permits the normal work of the network to avoid a network failure in case of a higher transmission charge.
When a transmission of an aperiodic request priority 0 tasks is finished, the dispatcher should first inquire the hard periodic polling table. If there are one or more urgent tasks in the Hard Periodic Polling table, the first one is transferred immediately. If there is not task in Hard Periodic Polling table, the dispatcher should go back to execute the task of the Aperiodic Requests Priority 0.
Also, note that in each Aperiodic Request Priority queue, there are three different levels of precedence for the dispatcher. Since we assume that responses from slaves are more important than new requests of the master, sending new aperiodic fragments has a lower priority (referred as “Fragment Queue” in Fig. 6.16). For the other two elements of the aperiodic priority queue, the dispatcher first verifies the responses from slaves where it is guaranteed to exist packets (referred as “Slaves Info Queue” in Fig. 6.16), instead of polling slaves without this guarantee (referred as “Waiting Resp Queue” in Fig. 6.16).
If there is nothing to do, the dispatcher looks for the next periodic poll and does it earlier.

6.7 Conclusion

This chapter proposes the dispatcher as a network layer mechanism that provides a quality of service to the applications in the REMPLI system. Based on the Dual-Priority scheduling policy, two dispatcher approaches are proposed in order to obey periodic traffic constraints while minimizing the end to end transmission time of aperiodic packets.
The comparison between these two approaches shows that the relaxation of the deadline constraint permits to obtain a reasonable approximation to the strict deadline approach while reducing the computation time, which is a crucial factor in the consideration for deploying a system.

Chapter 7

Random Access Protocol Based on SFN

For event driven applications such as alarm application, in nature, the random access protocol is suitable to this kind of application. Considering that logical channels can run independently and in parallel in the system, different media access protocols can be used in the different logical channels. We propose that the master/slave protocol and random access protocol are used in respective logical channels. Carrier sense multiple access (CSMA) and ALOHA are two types of random access protocols [100] [88]. CSMA/CD (Carrier Sensitive Multiple Access/Collision Detection) is not appropriate for PLC as it requires hardware for both carrier sensing and collision detecting, while in powerline networks the wide variation of received signal and noise levels makes collision detection difficult and unreliable. An alternative to collision detection that can be easily employed in PLC is collision avoidance (CSMA/CA) [88], a technique that uses random backoffs to further reduce the collision probability. Beside, ALOHA is a possible medium access scheme for PLC environment because it is simple and does not require carrier sensing and collision detection. However, CSMA/CA and ALOHA protocols should be modified to adopt the characteristics of REMPLI PLC network.
For adapting REMPLI PLC feature, two random access protocols are modified to be capable of running SFN routing algorithm: one is based on slotted ALOHA protocol, while the other one is based on carrier sense protocol. We use message transfer delay as performance metrics to compare those two protocols by simulation. At the end, we compare them with the results using periodic polling in master/slave model.

7.1 Characteristic of random access protocol based on SFN

We notice that the classical random access protocol cannot be used directly in REMPLI system due to the following reasons:

- Long transmission distance requires repeaters to reach the destination node. SFN routing protocol should be used. So, all slaves work as repeaters, according the SFN principle.

- All nodes are transceivers and only can work as either the transmitter or receiver but not both. So, the node cannot detect the collision during the transmission.

Since a transmitted packet is received by only a subset of the nodes in the network, there is a possibility that another slave in a different part of the network may also be successfully transmitting a packet during the same time since it is not disturbed by the first slave's transmission. This important phenomenon is called *spatial-reuse* of the channel [96]. One example of this phenomenon is shown in Fig. 7.1. In the first time slot, slave S9 sends packet to the network. The packet is repeated by slaves (S6, S8, S10, S13), which have correctly received this packet in this time slot. In the second time slot, the packet from S9 when repeated by node S8 is collided with the new packet sent from S7. However, this local collision does not affect the successful transmission of S9, because the packet is received by the master in the third time slot when it is repeated by the slaves S3, S4 and S5.

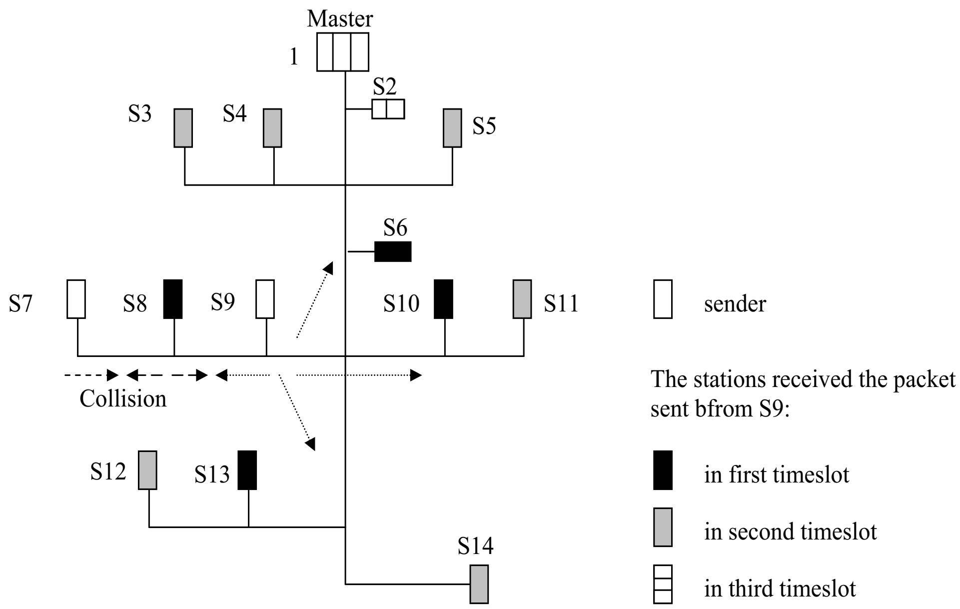


Figure 7.1: Local collision effects

SFN routing principle is that all the slaves can work as repeaters and can only repeat the same packet once. Because SFN is a flood-based routing algorithm and the transmission is not directional in PLC, the packet will be propagated in the whole network and it brings on. Consequently, the payload of each slave increases enormously. For example, in Fig. 7.1, S2 is near the master and will be repeated for all other slaves which require repeaters to reach the master. If we assume that each slave has the same packet arrival rate λ and the number of S is n, the payload of S2 will be $n * \lambda$. The reasonable traffic rate should consider not only the number of nodes which can be heard directly, but also the number of nodes in the network, as it is well known that random access protocols are efficient under light to medium network payloads.
Although the REMPLI PLC network is capable of *spatial-reuse*, there is a high collision probability due to the hidden node problem. For example, in Fig. 7.2, node C cannot

hear node A while node B can hear both node A and C. Node C cannot receive the packet from node B when node A sends a different packet at the same time. This is similar to the classical hidden node problem in a wireless network [99]. IEEE 802.11 protocol uses RTS/CTS (Request to Send / Clear to Send) acknowledgment and handshake packets to partly overcome the hidden node problem [90]. However, SFN protocol requests the packet to be repeated in the next time slot of the same logical channel. RTS/CTS mechanism cannot be used.

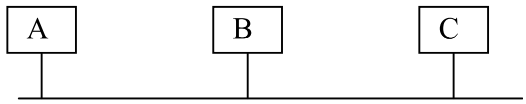


Figure 7.2: Collision caused by hidden node

7.2 Random access protocols design

7.2.1 Time slot partition in random access logical channel

If the synchronization technology is sufficient for dividing the time to smaller time slot, two methods of the time slot partition in random access channel is showed in Fig. 7.3:

- Case 1: A slave uses ALOHA protocol to transmit an alarm packet in random access channel. An example is showed on the top of Fig. 7.3.
- Case 2: A logical channel is divided into several micro time slots. The idea is that a short packet, called as the signal, is sent to the master within a micro time slot by the p-persistent CSMA [97], and it only contains a few bits to indicate that the slave has urgent alarm information. Then after the master gets this signal packet, it will poll this slave for getting the alarm packets in a master/slave logical channel. This way costs much, due to accurate synchronization. An example is showed at the bottom of Fig. 7.3, which logical channel C has been divided into 5 micro time slots.

7.2.2 ARQ mechanism

For providing a reliable transmission in the high packet error environment of PLC network, automatic repeater request (ARQ) protocol is used as error control technology. The destination detects errors by means of built-in error-detection coding. The destination then arranges to send acknowledgment packets (ACKs) back to the sender for each correctly received packet. If an ACK is not received after a suitable delay, the sender retransmits the relevant packet. This requires that the transmitter stores each packet it sends until it receives an ACK for that packet. This process is repeated until the packet

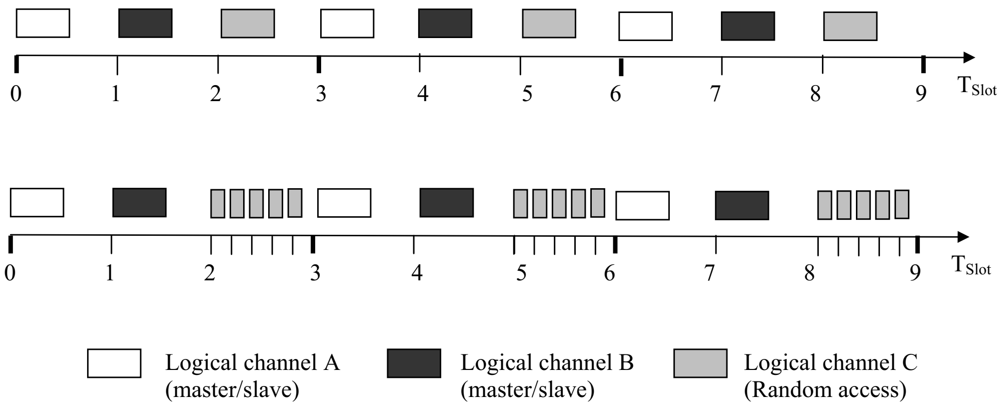


Figure 7.3: Illustration of the random access channel

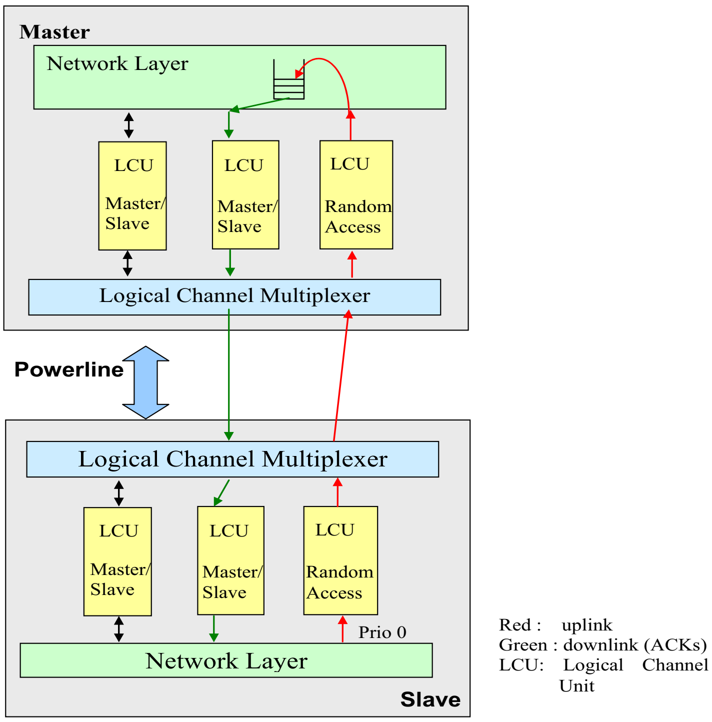


Figure 7.4: Logical channel assignment

is error-free or the error continues beyond a predetermined number of transmissions. ARQ is used in different way in our REMPLI system. Because all slaves send the data packets to the master in one logical channel, obviously there is a high probability of collision if the master sends ACKs via the same logical channel. So we propose that ACKs are sent by the master/slave protocol in another logical channel to avoid the collision. Therefore, the logical channel of random access mode is only used for uplink as shown in Fig. 7.4. In Chapter 6, the slave does not have priority 0. So we add a queue with

priority 0 in the slave for the urgent informations such as alarm application, which are sent via the random access channel.
Usually, it assumes that once a packet is successfully received, an ACK is sent immediately to the transmitter. Lacking of an ACK packet in a certain time, retransmission should be trigged by the expiration of a timer. However, in our random access protocol, the ACK is possible to arrive at the transmitter after a possibly higher delay and it does not confirm the sending packet according to the consecutive sequence. For example, case 1 in Fig. 7.4, the master receives two packets from slave 1 (with repeater level=1) and slave 2 (with repeater level=0) consecutively while the ACK for slave 1 requires 2 time slot, so slave 2 receives the ACK after a delay time.

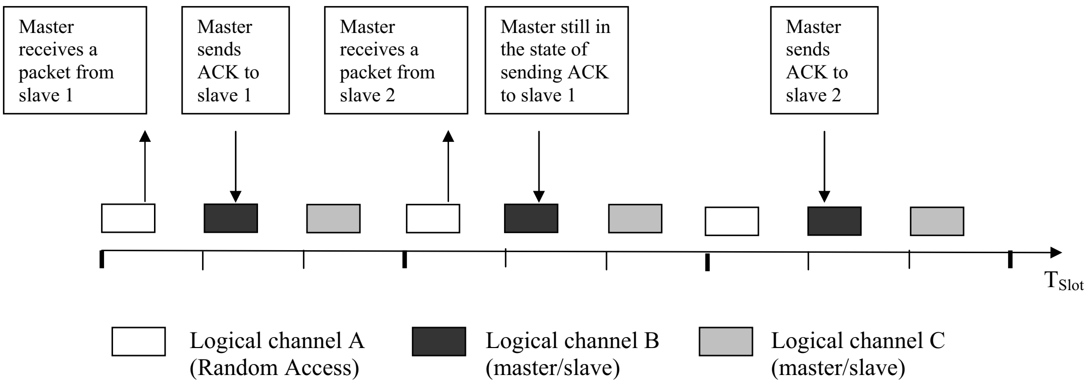


Figure 7.5: ACK delay in Master/slave channel

As the master/slave channel is not reserved for sending the ACKs for random access protocol, it can be used for sending the other packets from application layer. So the delay time will be random and unpredictable due to the waiting for the channel to be free in the multi-task environment. So, the timer for initiating a retransmission at the slave side should be set to a reasonable value.
Moreover, the packet transmission failure can be caused by the noise or the collision, but the transmitter does not distinguish them and simply considers it as the signal of collision. Backoff algorithms are favorably employed in random access communication systems in order to facilitate resolutions of packet collisions.

7.2.3 Repeating in SFN protocol

Each slave has two transmission functions: repeating other slaves' packets and sending its own packets. The former function has a higher priority than the latter. The slave puts the newly generated packets into the queue, if it is at a non-idle slave (in the repeater state or transmission state).
The slave uses the slave address field and sequence number field to determine if it works as repeater and ensures that the same packet is repeated once. The procedure is shown in Fig. 7.6.
Furthermore, we propose that the slave uses the uplink repeater level value that is em-

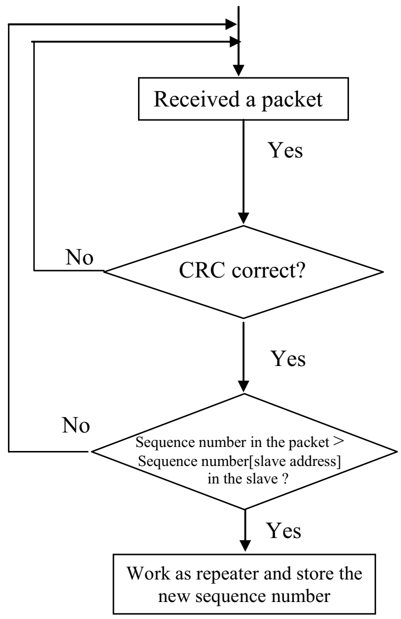


Figure 7.6: Block diagram of the repeating algorithm of SFN

ployed in the master/slave channel for SFN. Since the transmission failure is considered to be caused by the collision, the slave will still use the same value of uplink repeater level instead of increasing the repeater level for the retransmission. In Section 7.4.1, the simulation results show that this mechanism is suitable in REMPLI PLC system.

7.2.4 Protocol description

ALOHA-SFN protocol

The ALOHA transmission principle is that whenever a node has a data, it transmits. In our REMPLI PLC network, the slave sends its own packet at the beginning of the ALOHA logical channel, only when it does not work as a repeater. The procedure of transmitting a new packet is shown in Fig. 7.7.
Whether the transmission is successful is determined by the slave if it can receive the ACK before the retransmission timer expires. Considering that ACKs have the delay feature and are transmitted in the other logical channel, it is not necessary that a slave stops transmitting the next packet for waiting ACK. It will cause a new problem that consecutive ACKs received by the slave may acknowledge inconsecutive packets. So, we put Request_ID field in the packet, which the Master takes over for its ACK. Therewith, the Slave can recognize the received ACKs to acknowledge which packets have been sent. The failed packet should be retransmitted with the same Request_ID. The structure of ALOHA-SFN packet is shown in Fig. 7.8.
The retransmission will begin after a random delay by using the backoff algorithm. We choose binary exponential backoff which is an algorithm being widely used in the MAC-

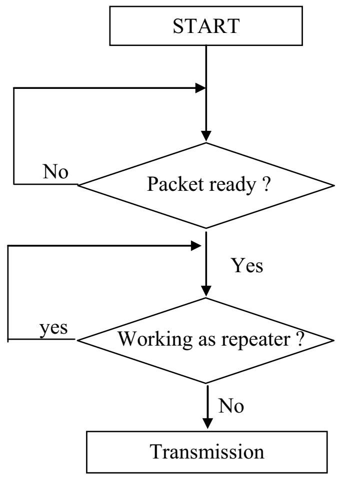


Figure 7.7: Block diagram of the ALOHA-SFN algorithm

CRC	Control feild	Sequence number	Source address	Repeater level number	Request_ID	User data

Figure 7.8: Block diagram of the ALOHA-SFN algorithm

layer protocols. Each node controls its packet transmissions by means of its backoff counter. Generally, the number of backoff counter is increased by one after a retransmission and decreased to zero after a successful transmission. However, in ALOHA-SFN protocol, the number of backoff counter records an accumulated retransmission time for several packets because of the consecutive sending. We design that the number of backoff counter increases by one after each retransmission and decreases by the number of retransmissions of this packet.

Carrier sense-SFN protocol

In random access channel, the slave just retries to send a signal to the master. The packet should be small and can be transmitted within a micro time slot, as shown in Fig. 7.9.

CRC	Control Feild	Source Address	Repeater level number	Sequence number

Figure 7.9: Structure of carrier sense-SFN packet

Although the slave cannot detect the collision in PLC environment, it can know whether the channel is idle or not in the current micro time slot. There is a kind of carrier sense. So we consider the channel as idle when the slave does not receive any packets in the micro time slot. We employ the p-persistent CSMA as a collision avoidance technique.

If a slave has its own packet to send, it generates a random delay from the interval (*0...(n(s)*Wbase)*. $n(s)$ is the retry time of slave s. *Wbase* is the basic number of randomizing slots, and is equal to a constant. The transmission procedure is shown in Fig. 7.10.
To decrease the collision probability in the random access channel, we propose to use the piggybacking method. After the master polls the slave, the slave can use the ACK packet to request a new polling, if there are packets in its queue 0. In this case, the signaling packet is not sent for requesting the master polling, and the collision probability decrease. If the slave does not receive the master polling after pre-configured timer is expired, the slave should retry the signal packet. The procedure is shown in Fig. 7.11.
When the master receives a signal packet from slave, it will put the source slave address into the slave signal queue 0 and wait until the channel is free and then poll this slave to get the alarm information. With the piggyback technology, the master can know if the slave still has the alarm information in its queue 0. If yes, the master can send the second poll for this slave after checking whether it is true that there are not packets in the slave signal queue 0 and master polling queue 1. Otherwise, the master puts the slave address into the master polling queue 1 and then sends the polling to the slave in the slave signal queue 0 or in the master polling queue 1. Fig. 7.12 illustrates this procedure.

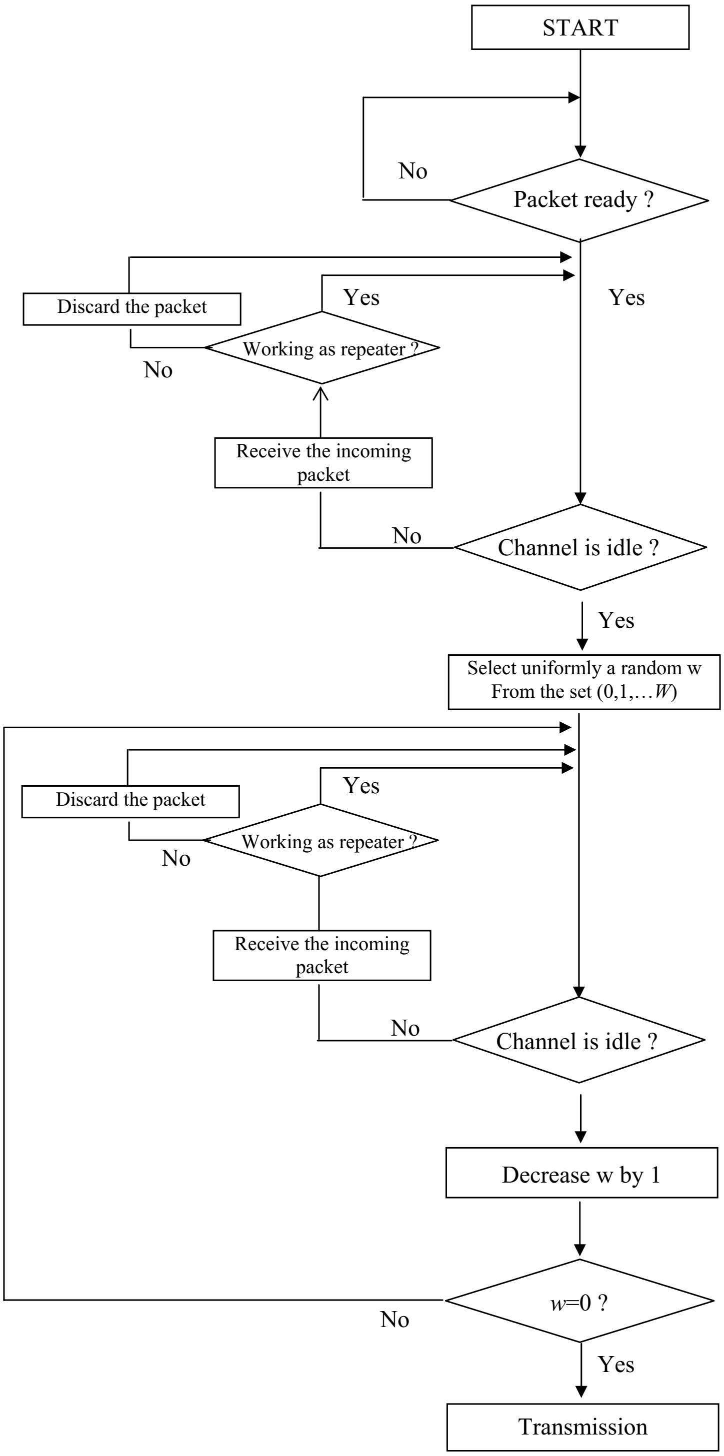


Figure 7.10: Block diagram of the predictive p-CSMA algorithm

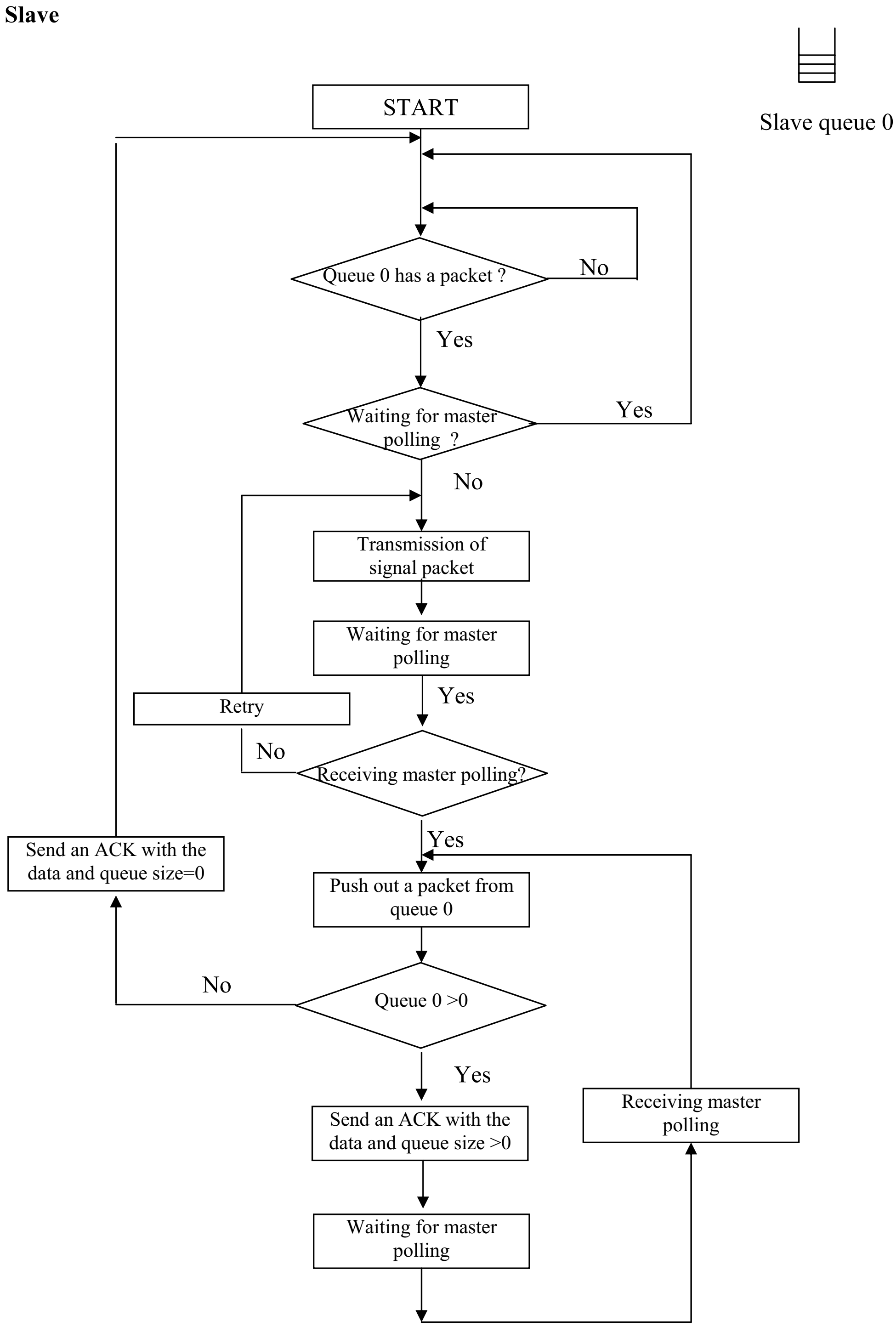


Figure 7.11: Block diagram of the slave transmission

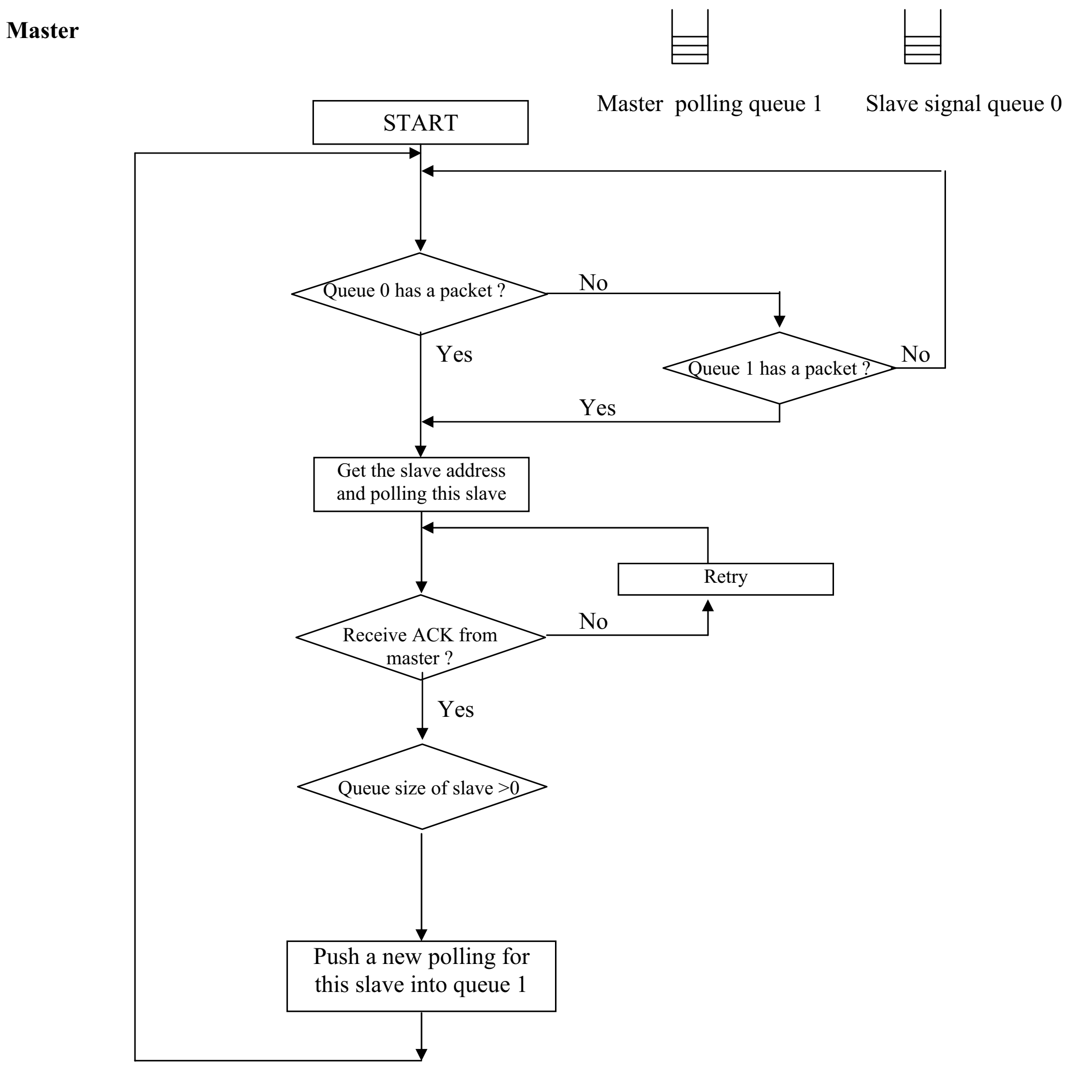


Figure 7.12: Block diagram of the master transmission

7.3 Simulation results

7.3.1 Repeater level

The repeater level can be configured to the broadcast repeater level, which is used to reach all the slaves, and the ones which be used in master/slave model. We use the percentage of successful transmission as a comparison metrics, which is the radio of the number of packets received successfully by the master to the number of packets sending by the slaves. The simulation is done in the channel model of Ring_10 with one ALOHA channel, and the uplink repeater level is configured as Table 7.1 or 2 (the broadcast repeater level in channel model Ring_10). In the simulation scenario, each slave generates new packets at λ packets/slot according to a Poisson distribution.

node_number	repeater level
1	0
2	0
3	1
4	1
5	2
6	1
7	1
8	0
9	0

Table 7.1: Uplink repeater level in master/slave channel

From Table 7.2 and Table 7.3, we can conclude that the usage of repeater levels of master/slave mode can improve the performance of network in both light (λ=0.01) and heavy payloads (λ=0.1). We uses this mechanism in ALOHA-SFN and carrier sense-SFN protocols.

Slave number	Repeater level=2	Repeater level in Master/slave channel
1	88.50%	91.83%
2	83.91%	90.01%
3	82.85%	90.18%
4	79.58%	86.81%
5	84.41%	95.31%
6	80.31%	85.75%
7	81.53%	90.00%
8	84.25%	91.71%
9	85.95%	91.10%
total	83.48%	90.31%

Table 7.2: Percentage of successful transmission with Poisson distribution λ=0.01

Slave number	Repeater level=2	Repeater level in Master/slave channel
1	39.92%	47.83%
2	29.67%	33.49%
3	22.79%	30.29%
4	17.27%	20.11%
5	15.83%	30.65%
6	18.18%	20.74%
7	20.63%	31.28%
8	30.56%	35.08%
9	40.47%	48.08%
total	26.16%	33.06%

Table 7.3: Percentage of successful transmission with Poisson distribution λ=0.1

7.3.2 Two protocols comparison

For the application, we concern the time when the master gets the slave alarm information. We use the average message transfer delay as the performance metrics, which is the same as that in Section 5.3.1. In ALOHA-SFN protocol, the message is transmitted from the slave to the master via random access channel. By contraries, the message is attained

through the master polling in the carrier sense-SFN protocol.
Recall that for master/slave periodic poll, it cannot guarantee the polling cycle, when the λ is equal to 0.002 packets/T_{ch}. So we consider that this value of λ is the high traffic load in the channel model RandArea Np_100.
Simulation parameters:

- channel model: Ring_10 and RandArea Np_100
- message arrival rate λ (packets/T_{ch}) according to Poisson distribution
- simulation time: 24 hours
- delay time for retransmission: 100 time slots for channel model of Ring_10; 400 time slots for channel model of RandArea Np_100
- maximum backoff time slot: 31 for ALOHA-SFN protocol; 39 for carrier sense-SFN protocol
- T_{ch}: 0.0264 s.

Channel Model	λ (packets/T_{ch})	Master/slave	ALOHA-SFN protocol	Carrier sense-SFN protocol
Ring_10	0.001	15.04	0.155	0.105
	0.01	14.31	0.49	0.15
	0.02	13.82	1.6	0.39
RandArea Np_100	0.0001	15.05	3.51	1.87
	0.001	16.23	4.025	2.16
	0.002		5.14	9.78

Table 7.4: Simulation results of average message transfer delay (s)

From Table 7.4, we can see that the random access protocol outperforms the master/slave protocol.
We find that the value of delay time for retransmission is a very important factor of the network performance. In Table 7.5, an example in the channel model RandArea Np_100 is shown. In fact, the actual ACK transmission delay in ALOHA-SFN protocol is small, even in the case of λ=0.002. From Table 7.5, we can see that the performance of retransmission delay time equal to 100 T_{slot} is evidently better than that of retransmission delay time equal to 400 T_{slot}.
By contraries, in CSMA-SFN protocol, the retransmission delay time needs be augmented with the increment of message arrival rate. With the high message arrival rate, the master will poll more slaves in the master/slave channel. This leads to a longer queue waiting time. If the retransmission delay time is not suitable, the slave will send more signal for a single alarm message to request the master polling. It worsens the situation of queue

Channel Model	λ (packets/T_{ch})	Delay time for retransmission	
		100 T_{slot}	400 T_{slot}
RandArea Np_100	0.0001	1.72 s	3.51 s
	0.001	1.70 s	4.025 s
	0.002	1.74 s	5.14 s

Table 7.5: Simulation results of average message transfer delay (s) with different retransmission delay time for the ALOHA-SFN protocol

size in the master and make master increase the unnecessary polling for certain slaves. At last, the unsuitable retransmission delay time will result in great message transfer delay. In Table 7.6, we show an example of the message arrival rate λ equal to 0.002. Obviously, the performance with the bigger delay time for retransmission is better than that with the small delay time for retransmission.

Channel Model	λ (packets/T_{ch})	Delay time for retransmission	
		400 T_{slot}	800 T_{slot}
RandArea Np_100	0.002	9.78 s	2.40 s

Table 7.6: Simulation results of average message transfer delay (s) with different retransmission delay time for the carrier sense-SFN protocol

So we propose that the value of delay time for retransmission should be changed with the network traffic.

7.4 Conclusion

We investigate random access protocol to allow slaves to initiate a transmission to reach a master through the SFN routing protocol. From simulation results, the random access protocols outperform the periodic polling of master/slave protocol, in particular, for the cases which the master/slave cannot guarantee its periodic polling cycle, such as λ=0.002. The carrier sense-SFN protocol has better network performance than the ALOHA-SFN protocol, and is suitable for the continuous/bursty traffic. Unfortunately, both random access protocols require two logical channels, in which one is reserved for slaves sending information to the master, and the other is used to send the ACKs. For the limited PLC bandwidth, these methods will use more bandwidth for the alarm application, and increase the queue waiting time for other application packets.
For high alarm occurrence probability per slave node, and knowing the total slave number of a master, there is a trade-off between time-triggered (i.e. master polling) and event-triggered (i.e. slave initiated) communications. Our study showed the possibil-

ity to implement an ALOHA based protocol for alarm reporting under a certain alarm occurrence rate.

Chapter 8

Conclusions and Future Works

This thesis contributes to the ongoing research on the wide area PLC (Power Line Communication), focusing on the design and performance evaluation of QoS mechanisms basing on master/slave communication protocol and the possible usage of random access protocol in such communication architecture.

Our research work follows the following design approach. Starting from the application requirement analysis (both data collection and real-time control), we derived the basic performance needs on the communication networks and found the PLC is a suitable communication technology especially for energy distribution utility. After an overview of the existing PLC technologies, it appears that none of them can be directly used because of their insufficient performance and lack of the QoS support. So a new PLC network has been designed as part of the REMPLI European project.

The design and implementation of such a PLC network represents several research challenges. The first one consists in designing an efficient dynamic routing protocol to cope with the changing power line topology and the unreliable transmission medium. The second one is to find an efficient way to get slave's response in a master-slave model. This is a specific issue of the REMPLI project since upon a master poll, a slave can only give its response after a random delay, making difficult to determine the next polling time.

The third one is to develop a mechanism for providing to divers applications with the differentiated QoS since PLC, although using the state of the art transmission technology (such as OFDM), has always a very limited bandwidth. This limited bandwidth has to be judiciously shared among applications with different performance requirements. The fourth problem is to find the suitable protocol allowing to report slave side alarms.

Concerning the first problem, together with other REMPLI partners, a dynamic routing protocol based on flooding principle, called as SFN (Single Frequency Network) has been designed and implemented. To increase the protocol efficiency, uplink and downlink repeater levels' counters are added in the frame header. Our contribution resides in the proposition of an efficient method to manage the repeater levels in the master in order to shorten the transmission time and decrease the retry probability caused by the lack of repeater levels. Probabilistic analysis of the steady-state performance has been analysed showing its advantage over the old DLC1000 source routing protocol. Extensive simulations have also been conducted revealing the problem of the repeater level oscillation and leading to a suitable repeater level adaptation mechanism. We also contributed to the

protocol implementation in the final REMPLI PLC product.
To deal with the second problem, we analyzed and simulated the REMPLI system to evaluate the communication performance by three metrics: response time to a command, response time to a response, and the average polling time. At first, a REMPLI PLC network composed of one master and several slaves has been evaluated. Then, a whole REMPLI network composed of MV and LV powerline segments, connected by bridges has been investigated. From those analysis and simulation results, we can draw a conclusion that REMPLI PLC system can meet the requirements of most applications which are presented in Chapter 2. Because in REMPLI project, the device data are not allowed to be stored outside the device itself in some countries, the slave node must get a response from the attached equipments (e.g. meters) directly after it receives a request from the master. The master must re-poll the slave node to get its response. In the first part of chapter 6, we proposed a cross-layer mechanism which consists in getting firstly the slave response generating time (t1) and then to re-poll the slave after t1. An alternative solution has also been proposed which stores the field level device data inside the slave node (e.g. a cache with time stamped data) such that a request can get its response immediately, even if the cached datum is not always the latest one. The performance evaluation has shown its high efficiency.
To solve the third problem, in chapter 6, we proposed a dispatcher as a network layer mechanism that provides differentiated quality of service to the applications in the REMPLI system. The application classes can generate two types of network layer traffics: periodic and aperiodic traffics. The aperiodic traffics are implemented according their priorities. The round-robin scheduling is used between the highest priorities of the aperiodic and periodic traffics. The dual-priority scheduling policy with the relaxation of deadline is proposed in order to obey periodic traffic constraints while minimizing the end to end transmission time of aperiodic packets. It permits to obtain a reasonable approximation to the strict deadline approach while reducing the computation time, which is a crucial factor in the consideration for deploying a system. In fact, most of the periodic traffic in REMPLI only require the loose periodicity with rather large jitters. This dispatcher has been firstly simulated and then implemented.
The fourth problem has been addressed in Chapter 7. We investigated random access protocols in our REMPLI PLC network. Two kinds of random access protocol are modified for having capability to run SFN routing protocol. The principle of ALOHA-SFN protocol is that a slave uses ALOHA protocol to transmit an alarm packet in random access channel, and the master sends ACKs through master/slave channel. The principle of carrier sense-SFN protocol is that the slave sends signal packets within a micro time slot by the p-persistent CSMA for requesting master polling, and after the master gets this signal packet, it will poll this slave for getting the alarm packets in a master/slave logical channel. The simulation results show that they outperform master/slave periodic polling. However, those protocols require more bandwidth. If the alarm occurrence probability per slave node, and the total slave number of a master, can be known, there is a trade-off between time-triggered (i.e. master polling) and event-triggered (i.e. slave initiated) communications.

At the moment of finishing this PhD thesis redaction, we can already point out the two following points as the future work:

- The proposed random access protocols have not been implemented in REMPLI PLC. However our analysis has shown their interest over the polling scheme for a large range of the network load (not heavily loaded). Before their implementation, two problems remain to be investigated. One is how to find a way to on-line change the running protocol between the random access and polling in order to get always the best performance for alarm reporting. Another is a more profound analysis of the impact of the flooding-based routing on the collision probability when random access protocol is running. In fact, as the random access is always from slaves to the master, comparing to a classic routing protocol which only use one path from the source to the destination, flooding-based routing will drastically increase the collision probability, especially for the nodes (repeaters) which are near to the master. So the flooding-based SFN routing protocol merit to be revisited.

- The traffic dispatcher we proposed consists in an engineering contribution as it gives a concrete solution to the implementation of the dual-priority scheduling policy. In our schedulability test, a constant value of the transaction time (C_i) is used. However, this value is variable in practice due to the random retransmissions and repeater levels. So it will be interesting to carry out a schedulability study considering a variable Ci in order to give a probabilistic guarantee on the response time. Of course, this requires a long-term theoretic investigation, as there is not much related work.

Appendix A

Calculation Program

A.1 DLC 1000

We use this program in MATLAB for calculating the average polling time of DLC 1000 in a channel model which can be selected by the variable *path*. The result is variable MS_D.

```
path= ‘p:\rempli\vf\matlab\phylayemu\chamod\randarea\np_20\’
load([path ‘PER.dat’])

Np = length(PER)            % number of participants

% add min PER
minPER=1e-6;
PERv=reshape(PER,Np^2,1);
pos=find(PERv<minPER);
PERv(pos)=minPER.*ones(length(pos),1);
PERv((1:Np)+Np*(0:Np-1))=zeros(Np,1);
PER=reshape(PERv,Np,Np);

MS_D=inf.*ones(Np,6);
% calculate average duration without repeater
MS_D(:,2)=2./((1-PER(1,:)’).*(1-PER(:,1)));

% calculate average duration with 1 repeater
s=find(MS_D(:,2)>4);
for k= 1:length(s)
   MS_D(s(k),3)=min(4./((1-PER(1,:)’).*(1-PER(:,1)).*(1-PER(s(k),:)’)
                .*(1-PER(:,s(k)))));
end
MS_D(:,1)=min(MS_D’)’;

% calculate average duration with 2 repeater
```

```
s=find(MS_D(:,1)>6);
for k= 1:length(s)
   s2=find(PER(:,s(k))<0.5);
   d=inf.*ones(length(s2),1);
   for l=1:length(s2)
      d(l)=min(6./((1-PER(1,:)').*(1-PER(:,1)).*(1-PER(s2(l),:)')
      .*(1-PER(:,s2(l))).*(1-PER(s2(l),s(k))').*(1-PER(s(k),s2(l)))));
   end
   MS_D(s(k),4)=min(d);
end
MS_D(:,1)=min(MS_D')';

% calculate average duration with 3 repeater
s=find(MS_D(:,1)>8);
for k= 1:length(s)
   s2=find(PER(:,s(k))<0.5);
   d=inf.*ones(length(s2),1);
   for l=1:length(s2)
      s3=find(PER(:,s2(l))<0.5);
      d2=inf.*ones(length(s3),1);
      for l2=1:length(s3)
         d2(l2)=min(8./((1-PER(1,:)').*(1-PER(:,1)).*(1-PER(s3(l2),:)').
         *(1-PER(:,s3(l2))).*(1-PER(s2(l),s3(l2))').*(1-PER(s3(l2),s2(l)))
         .*(1-PER(s2(l),s(k))').*(1-PER(s(k),s2(l)))));
      end
      d(l)=min(d2);
   end
   MS_D(s(k),5)=min(d);
end
MS_D(:,1)=min(MS_D')';

% calculate average duration with 4 repeater
s=find(MS_D(:,1)>10);
for k= 1:length(s)
   s2=find(PER(:,s(k))<0.5);
   d=inf.*ones(length(s2),1);
   for l=1:length(s2)
      s3=find(PER(:,s2(l))<0.5);
      d2=inf.*ones(length(s3),1);
      for l2=1:length(s3)
         s4=find(PER(:,s3(l2))<0.5);
         d3=inf.*ones(length(s4),1);
         for l3=1:length(s4)
            d3(l3)=min(10./((1-PER(1,:)').*(1-PER(:,1)).*(1-PER(s4(l3),:)')
            .*(1-PER(:,s4(l3))).*(1-PER(s4(l3),s3(l2))').*(1-PER(s3(l2),s4(l3)))
```

```
            .*(1-PER(s2(l),s3(l2))').*(1-PER(s3(l2),s2(l))).*(1-PER(s2(l),s(k))')
            .*(1-PER(s(k),s2(l)))));
         end
         d2(l2)=min(d3);
      end
      d(l)=min(d2);
   end
   MS_D(s(k),6)=min(d);
end
MS_D(:,1)=min(MS_D')';


%calculate average duration of polling cycle
MS_D(1,1)=sum(MS_D(2:end,1));

MSD=MS_D(:,1)
```

Although a general algorithm for n repeaters could be easily given, we have not used it since the complexity is too high $O(N!)$ to make practical sense. In practice, for all physical layer channel models we studied, the repeater number is always lower than 4. In DLC 1000 implementation the repeater number is limited to 2 and the exploited paths for each route update is limited to 11.

A.2 SFN

We use this program in MATLAB for calculating the average polling time of SFN in a channel model which can be selected by the variable *path*. The result is variable D_SFN.

```
path='C:\ChaMod\RandArea\Np_200\'
load([path 'PER.dat'])
Np=length(PER)  % number of participants

% add min PER
   minPER=1e-6;
   PERv=reshape(PER,Np^2,1);
   pos=find(PERv<minPER);
   PERv(pos)=minPER.*ones(length(pos),1);
   % ste diag to 0
   PERv((1:Np)+Np*(0:Np-1))=zeros(Np,1);
   PER=reshape(PERv,Np,Np);

Nr= 7;    %maximum number of repeater (Nr-1)

%-------------------------------------------------------------
% downlink
```

```
%----------------------------------------------------------------

Pr=zeros(Np,Nr);

% Master transmit.
Pr(:,1)=1-PER(:,1);  % Probability of successful recept

% repetitions
for r=2:Nr
   for k=2:Np;
      s=[(2:k-1) (k+1:Np)]; % all slaves exept own
      Pr(k,r)=(1-sum(Pr(k,1:r-1))).*(1-prod(1-Pr(s,r-1).*(1-PER(s,k))));
   end

   if(sum(Pr(:,r))==0)
      r_max=r-1
      TER=(Np-sum(sum(Pr)))
      break
   end
end

%----------------------------------------------------------------
% uplink
%----------------------------------------------------------------

p_ul=zeros(Np,Nr);
for s2=2:Np
  Pru=zeros(Np,Nr);
  % slave s2 transmit.
  Pru(:,1)=1-PER(s2,:).';  % Probability of successful recept
  % all other participant
  ns2=[(1:s2-1) (s2+1:Np)];
  % repetitions
  for r=2:Nr
     for k=1:Np-1;
        s=[ns2(1:k-1) ns2(k+1:end)]; % all slaves exept own
        Pru(ns2(k),r)=(1-sum(Pru(ns2(k),1:r-1)))...
           .*(1-prod(1-Pru(s,r-1).*(1-PER(ns2(k),s).')));
     end
  end
  p_ul(s2,:)=Pru(1,:);      %Probability density
end
p_ul(1,:)=sum(p_ul(2:end,:))./(Np-1);

ta=zeros(1,Np);
```

```
for k=2:Np
  %k is the slave
   for i=1:Nr
       P(i)=sum(Pr(k,1:i)).*sum(p_ul(k,1:i));
       P'(i)=Pr(k,i)*p_ul(k,i);
   end

   %P
   pa=[P'(1)    (1-P(1))*P(2)      (1-P(1))*(1-P(2))*P(3)
   (1-P(1))*(1-P(2))*(1-P(3))*P(4) (1-P(1))*(1-P(2))*(1-P(3))*(1-P(4))*P(5)
   (1-P(1))*(1-P(2))*(1-P(3))*(1-P(4))*(1-P(5))*P(6)
      (1-P(1))*(1-P(2))*(1-P(3))*(1-P(4))*(1-P(5))*(1-P(6))*P(7)

    P'(1)    P'(2) (1-P(2))*P(3)  (1-P(2))*(1-P(3))*P(4)
    (1-P(2))*(1-P(3))*(1-P(4))*P(5) (1-P(2))*(1-P(3))*(1-P(4))*(1-P(5))*P(6)
    (1-P(2))*(1-P(3))*(1-P(4))*(1-P(5))*(1-P(6))*P(7)

    P'(1)    P'(2)   P'(3)   (1-P(3))*P(4)
    (1-P(3))*(1-P(4))*P(5)      (1-P(3))*(1-P(4))*(1-P(5))*P(6)
    (1-P(3))*(1-P(4))*(1-P(5))*(1-P(6))*P(7)

    P'(1)  P'(2)      P'(3)    P'(4)
    (1-P(4))*P(5)  (1-P(4))*(1-P(5))*P(6)   (1-P(4))*(1-P(5))*(1-P(6))*P(7)

    P(1)     P(2)-P(1)   P(3)-P(2)-P(1)    P(4)-P(3)-P(2)-P(1)
    P(5)-P(4)-P(3)-P(2)-P(1)   (1-P(5))*P(6)    (1-P(5))*(1-P(6))*P(7)

    P'(1)     P'(2)  P'(3)        P'(4)        P'(5)
    P(6)-sum(P(1:5))  (1-P(6))*P(7)

    P'(1)     P'(2)         P'(3)     P'(4)
    P'(5)      P'(6)          1-P(6)
      ];


    % add min PER
    minP=1e-6;
    maxP=0.999999;
    for i=1:7
       for j=1:7
         if pa(i,j)<minP
            pa(i,j)=0;
         end
         if pa(i,j)>maxP
             pa(i,j)=1;
```

```
         end
      end
   end

   %solve pa=pa*P
   pai_a(k,:)=[0 1 0 0 0 0 0 ];

   for i=1:20
       pai_a(k,:)=pai_a(k,:)*pa;
   end
%k
   pai_a(k,:);

   r=[2 6 12 20 30 42 56
      4 4 10 18 28 40 54
      6 6  6 14 24 36 50
      8 8  8  8 18 30 44
     10 10 10 10 10 22 36
     12 12 12 12 12 12 28
     14 14 14 14 14 14 14];

   Rr=zeros(1,Nr);
   for i=1:Nr
      for j=1:Nr
         Rr(i)=r(i,j)*pa(i,j)+Rr(i);
      end
   end
%Rr
%pai_a
   for i=1:Nr
   ta(k)=Rr(i)*pai_a(k,i)+ta(k);
   end

end

D_SFN=sum(ta(2:Np))
```

Appendix B

Simulation Model

The simulations are constructed with the Physical Layer Emulator and OPNET, a well know network simulation tool.

B.1 Interface Visual C++ Physical Layer Emulator and OPNET Simulator

OPNET is a discrete event simulator that possesses vast possibilities and support in terms of simulation and analysis of protocols. In our case, since the power-line is not natively supported, we can use OPNET to have a large array of statistical analysis offered by this tool. The C++ Physical Layer Emulator made available by iAd in the context of the REMPLI EU-project permits to emulate the power-line environment.
Usually, OPNET simulations use predefined OPNET Links for the communication between OPNET Nodes. However, since the powerline physical layer is not readily available in OPNET, it is not possible to use this standard communication mode.
Thus, the adopted solution is the creation of a centralised node that acts as the medium for communication between all network nodes. In the OPNET environment, it requires an extra OPNET node module which encapsulates the Physical Layer Emulator, through a shared linked library. Some modifications were made to the PhyLayEmu library. The changes were made in the two files described next in Fig. B.1
These two modifications allow to declare the function send_rx_opnet(), which will be defined inside the OPNET environment. Concretely, this function allows receiving the information of the reception of a packet by each node after the emulation of one time slot by the Physical Layer Emulator.

B.2 REMPLI simulation in OPNET

OPNET provides various types of links to simulate communication mediums. Since in the REMPLI system, the communication medium (powerline) is defined by the Physical Layer Emulator (due to its complex modeling), it is not necessary to define an OPNET link to model the powerline. For the modeling in OPNET, the alternative solution is to

```
File "emuinterf.cpp"
(. . . )
void EmuinterfRC_Phy_Rx(t_PhyRx tele_prx)
{
send_rx_opnet(tele_prx);
}
(. . . )
File "emuinterf.h"
(. . . )
extern void send_rx_opnet(PhyLayEmut_PhyRx tele_prx);
```

Figure B.1: PhyLayEmu modification portion

define the communication medium (represented by the PhyLayEmu) as a node (called as the medium node). All other nodes (represented as node_i, where i is the number of the node) will therefore have a direct connection to this medium node. Internally, the direct connection between node_i and medium node is provided by an OPNET communication function without time delay.
Fig. B.3 shows a scenario of 10 nodes. In OPNET, the number of the nodes in the network needs to match the actual network topology which is selected in the PhyLayEmu. PhyLayEmu has provided five topologies: Ring_10, Ring_100, RandArea Np_20, RandArea Np_100 and RandArea Np_200.
Fig. B.2 and Fig. B.3 show OPNET simulation scenarios of an autonomous network and REMPLI PLC network, respectively.

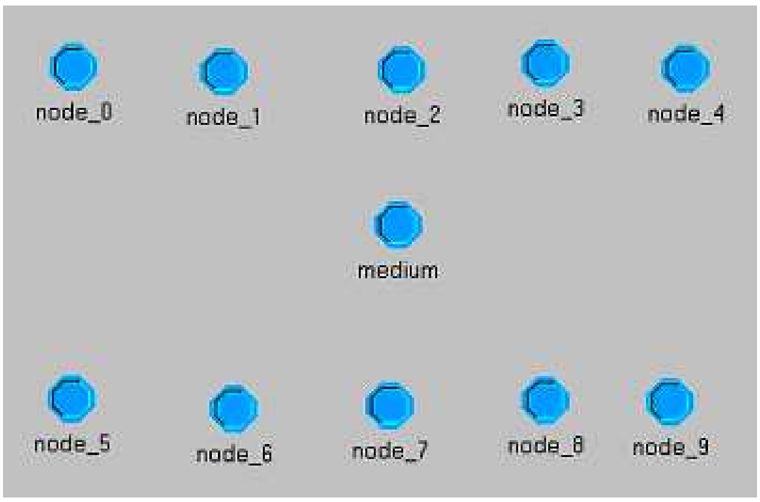


Figure B.2: Scenario of Ring_10 in OPNET

We construct eight types of OPNET Nodes:

- Medium node: is visually located in the center, and contains the PhyLayEmu library, which permits to emulate the powerline medium. This node receives data from all nodes (node_i), emulates the powerline medium, by the PhyLayEmu library, and sends the results to nodes (node_i).
- Master node: implements master's functions of master/slave protocol and SFN protocol, marked as node_0.
- Slave node: implements slave's functions of master/slave protocol and SFN protocol.
- Bridge node: consistes two node: MV_slave node and LV_master node, and implements communicating between MV and LV network, and master's functions of the master/slave protocol and SFN protocol.
- ALOHA-master node: master's functions of the ALOHA-SFN protocol
- ALOHA-slave node: slave's functions of the ALOHA-SFN protocol
- CSMA-master node: master's functions of the CSMA-SFN protocol
- CSMA-slave node: slave's functions of the CSMA-SFN protocol

In the following, we will show those nodes models in two levels of OPNET: node model and process model. Each OPNET Processor Model is defined by a finite state machine, which represents the module's logic, and encapsulates the module's behaviour through OPNET Proto C code. Moreover, we consider the OPNET simulation time unit as a time slot.

Figure B.3: Scenario of MV Ring_10 and LV Ring_10 in OPENT

B.2.1 Medium node

Fig. B.4 shows the node model of medium.

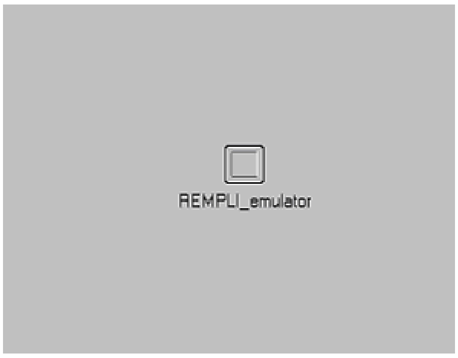


Figure B.4: Node model of medium node

Fig. B.5 shows the process model of REMPLI_emulator.

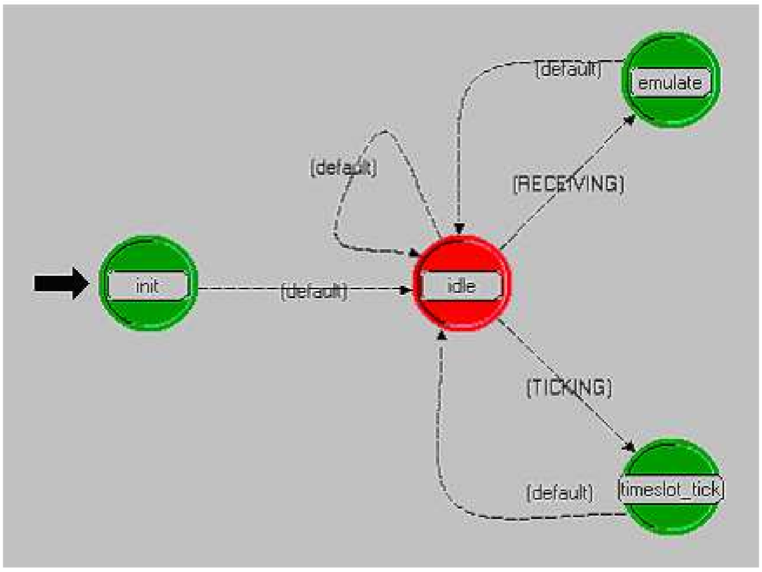


Figure B.5: Process model of REMPLI_emulator

B.2.2 Master node

Fig. B.6 shows the node model of master node. There are two layers: lcm_layer and network_layer.The functions of lcm_layer as shown in Fig. B.7 are:

- Receiving data from the network_layer and sending them to the medium node,

- Receiving the data from the medium node and sending them to the network_layer.

And the network_layer implements SFN protocol and master/slave protocol, are shown in Fig. B.8.

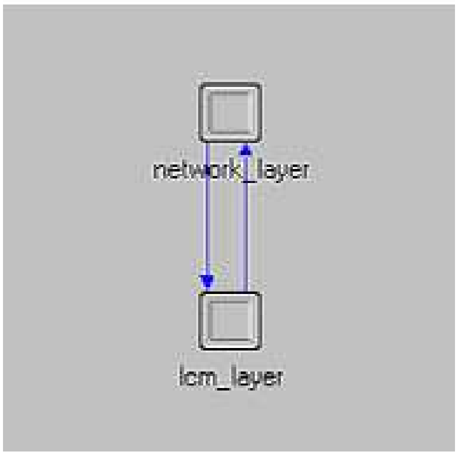


Figure B.6: Node model of master node

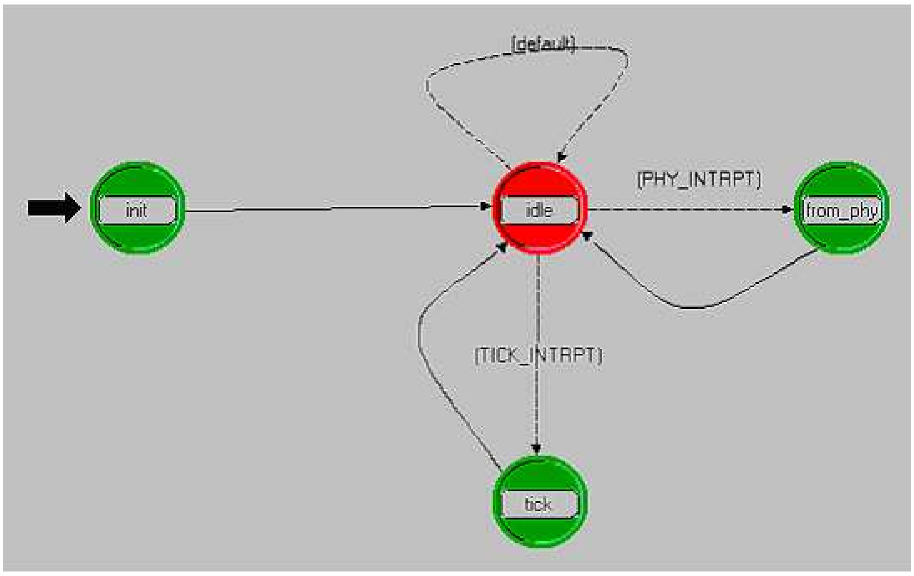


Figure B.7: Process model of lcm_layer in master node

B.2.3 Slave node

Fig. B.9 shows the node model of slave node. Process model of scr generates the data according to the distribution probability function. And process model of lcm_layer implements the receiving and sending the data to the medium node and the functions of SFN protocol and slave in the master/slave protocol, as shown in Fig. B.10.

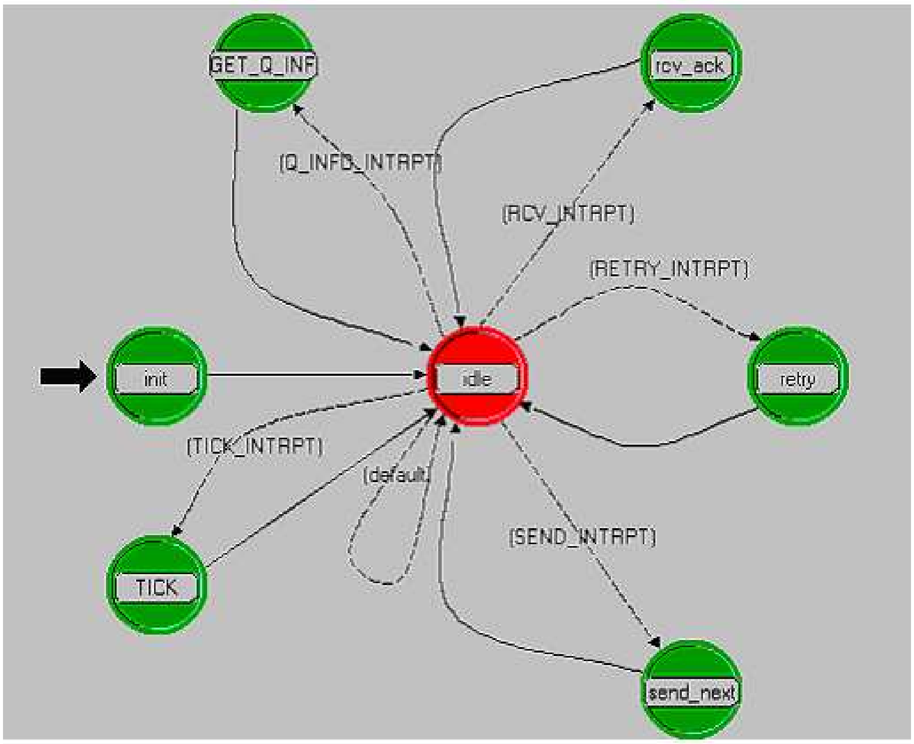


Figure B.8: Process model of network_layer in master node

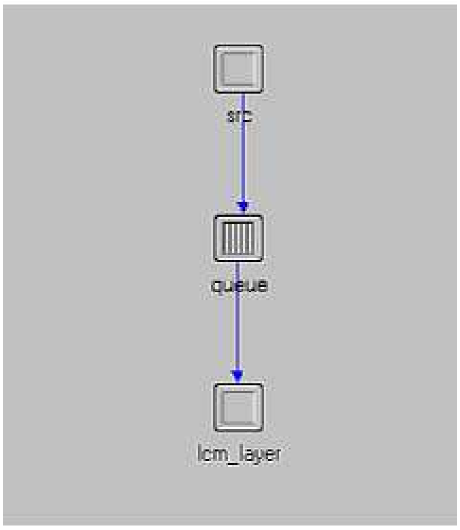


Figure B.9: Node model of slave node

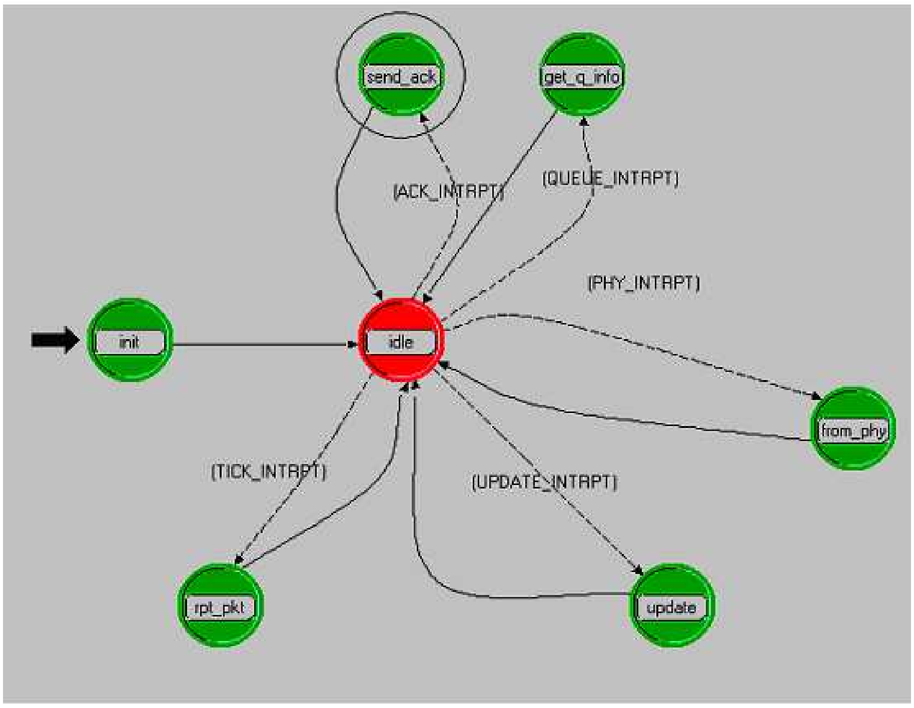


Figure B.10: Process model of lcm_layer in slave node

B.2.4 Bridge node

Bridge is composed of a MV slave and a LV master. Fig. B.11 shows an example, in which node_5 is used the node model of MV_slave and lv_0 is used of the node model of LV_master.

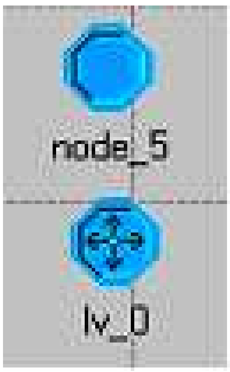


Figure B.11: An example of bridge

MV_slave

Fig. B.12 shows the node model of the MV_slave. The only difference between the slave node and MV_slave, is that the latter has the function to send/receive the data to the LV_master. Fig. B.13 shows the process model of from_lv, which has the functions of receiving the data from the LV_master and sending them to the queue. And queue can store the data.

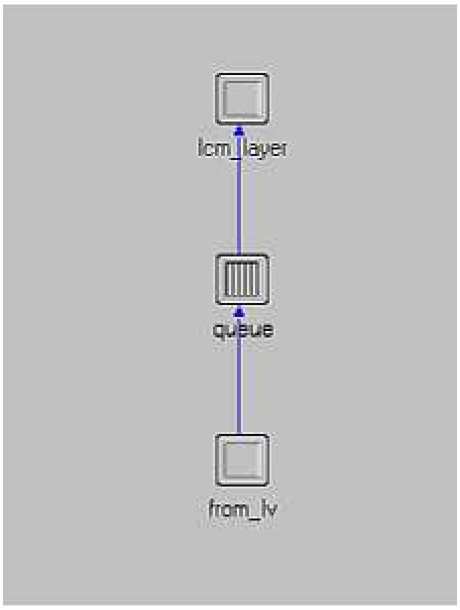


Figure B.12: Node model of MV_slave

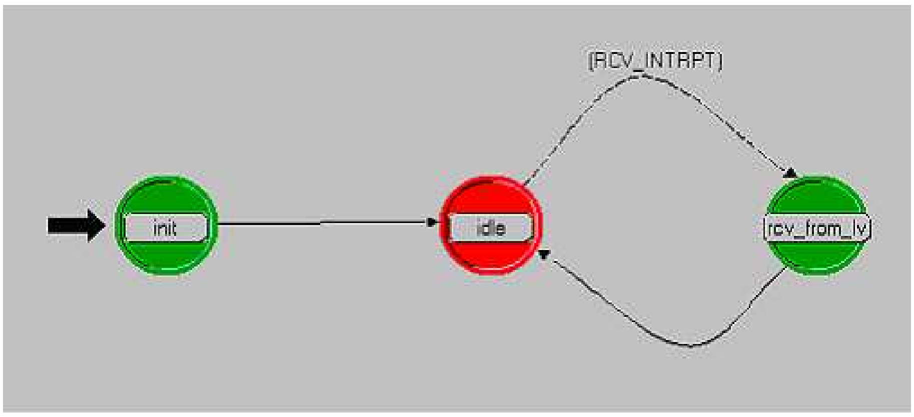


Figure B.13: Process model of from_lv

LV_master

Fig. B.14 shows the node model of the LV_master. The difference between the master node and LV_master, is that the latter has the function to send/receive the data to the MV_slave. The process model of from_mv is used to receive packets from MV_slave and send them to the queue for storing them in the queue, as shown in Fig. B.15. The LV_master gets the data from the queue and send them to their destination slave, and the other functions are the same as those in the master node. The process model of lv_master is shown in Fig. B.16. And the process model of lcm_layer is the same as that in the master node.

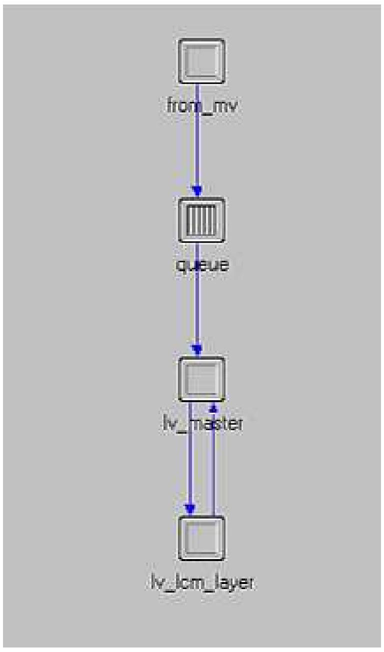


Figure B.14: Node model of LV_master

B.2.5 ALOHA-master node

Fig. B.17 shows the node model of ALOHA-master node. In this node, there are two logical channels, one for the ALOHA protocol and the other for master/slave protocol. In the ALOHA channel, the master is a receiver. In the master/slave channel, the master sends ACKs to slaves.
Process model of lcm_layer is shown in Fig. B.18. The functions of lcm_layer is to receive and send packets to the medium node. In Fig. B.19, the process model of network_layer is shown.

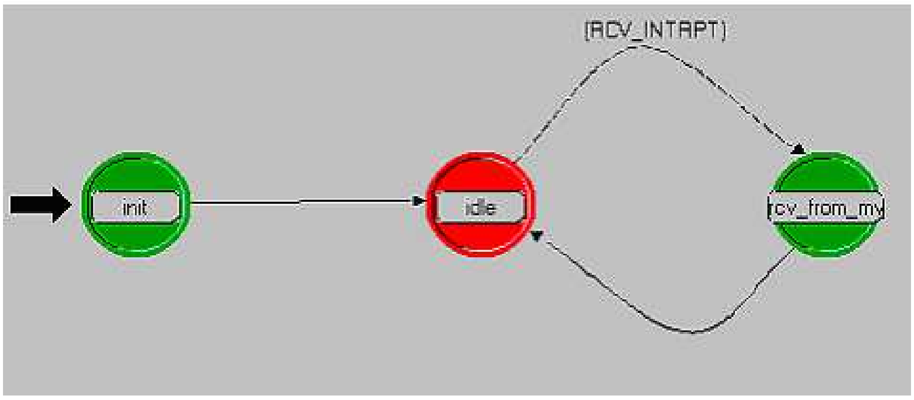


Figure B.15: Process model of from_mv

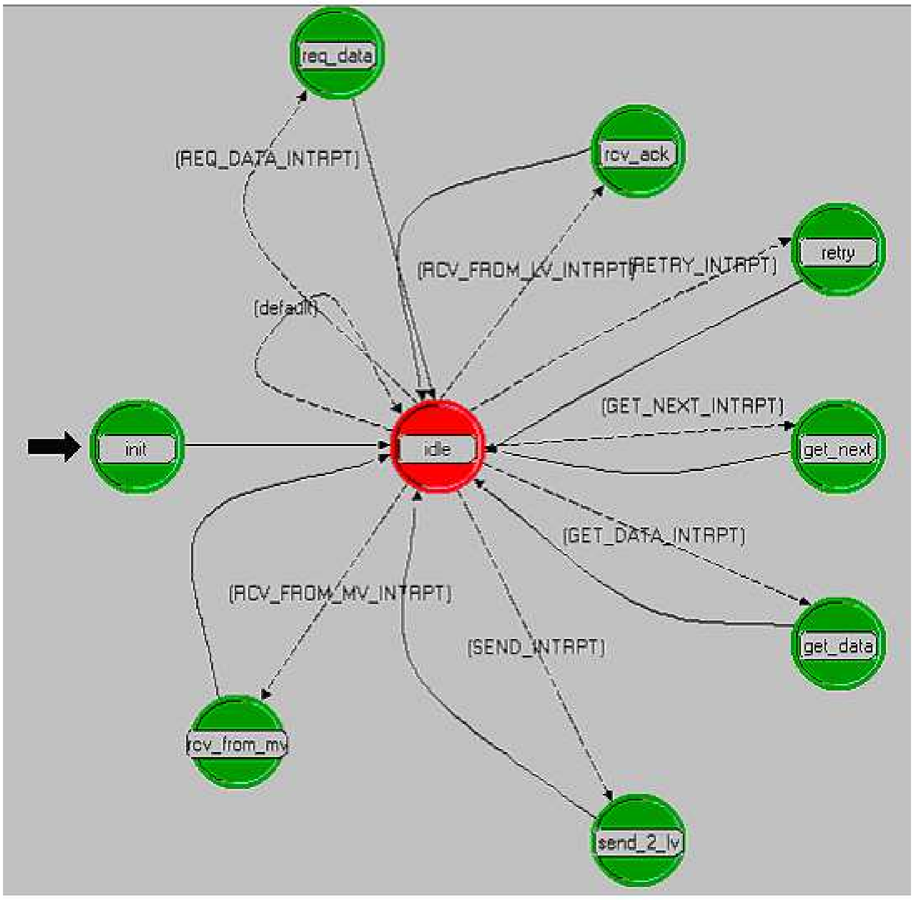


Figure B.16: Process model of lv_master

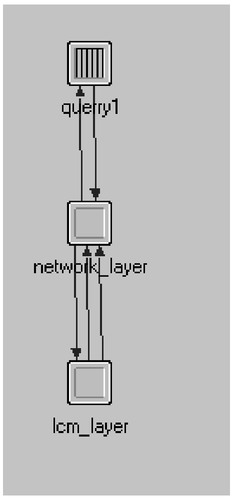


Figure B.17: Node model of ALOHA-master node

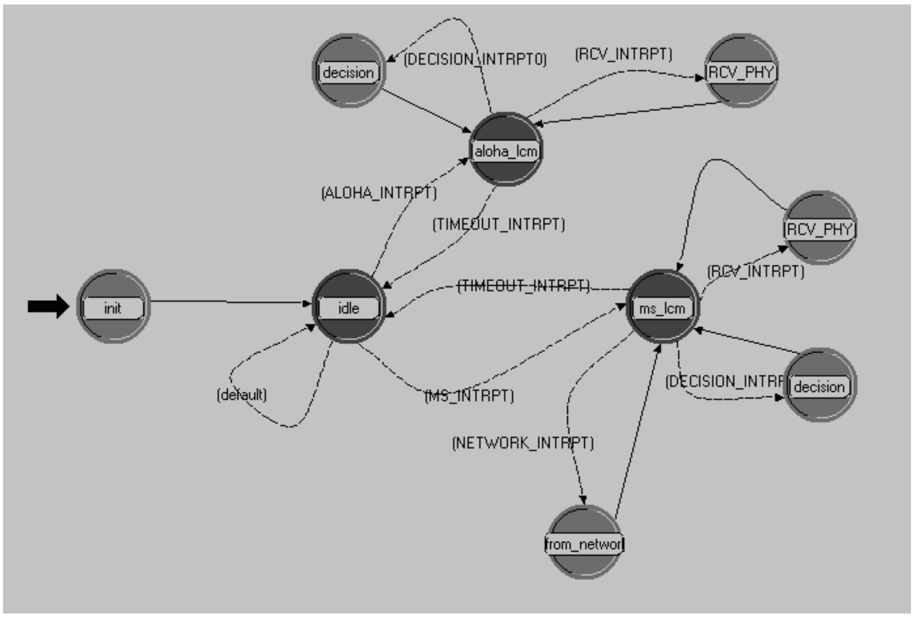


Figure B.18: Process model of lcm_layer

B.2.6 ALOHA-slave node

Fig. B.20 shows the node model of ALOHA-slave node. In this node, there are two logical channels, one for the ALOHA protocol and the other for master/slave protocol. In the ALOHA channel, the slave is a sender. And the scr node generates the data and sends it to the queue. Then, the queue stores the data. And the ALOHA-slave node sends the data according the ALOHA protocol which we design in Chapter 7. In the master/slave channel, the slave receives the ACK from the master.
Process model of lcm_layer is shown in Fig. B.21. The functions of lcm_layer is to receive and send packets to the medium node. In Fig. B.22, the process model of network_layer is shown. And it implements the SFN protocol and ALOHA protocol.

B.2.7 CSMA-master node

Fig. B.23 shows the node model of CSMA-master node. In this node, there are two logical channel, one for the CSMA protocol and the other for master/slave protocol. In the CSMA channel, the master is a receiver. In the master/slave channel, the master sends poll to slaves.
Process model of lcm_layer is the same as that in the ALOHA-master node. In Fig. B.24, the process model of network_layer is shown.

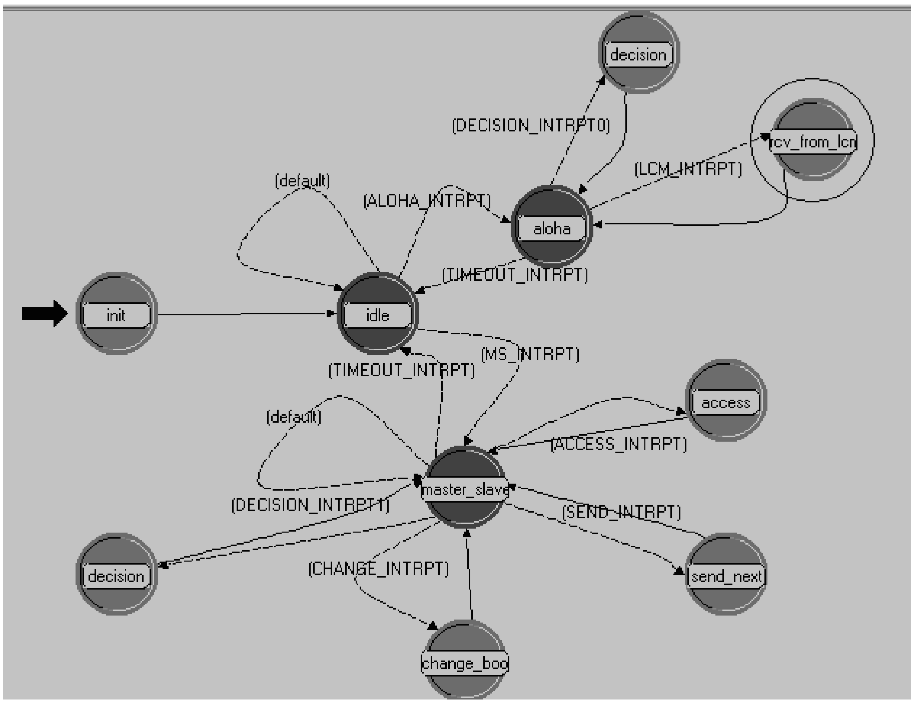


Figure B.19: Process model of network_layer

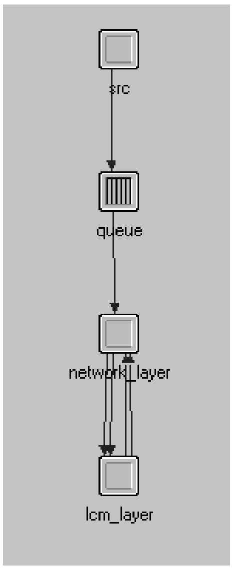


Figure B.20: Node model of ALOHA-slave node

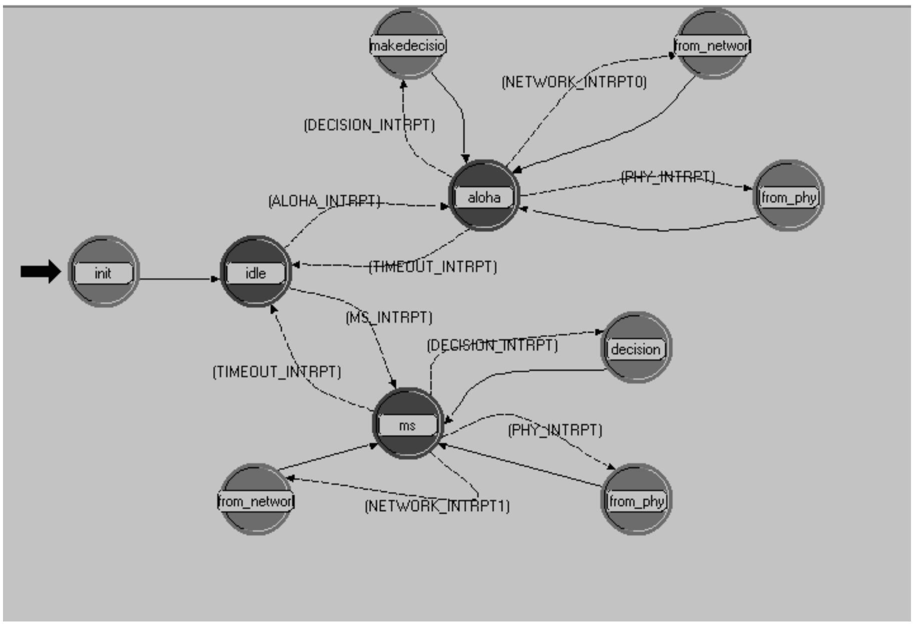


Figure B.21: Process model of lcm_layer

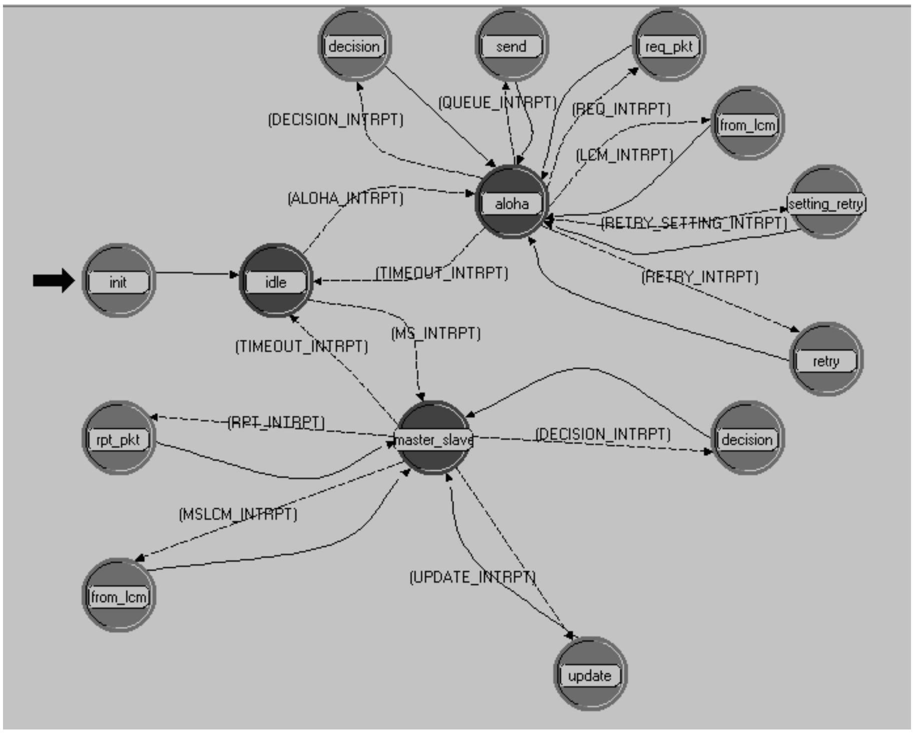


Figure B.22: Process model of network_layer

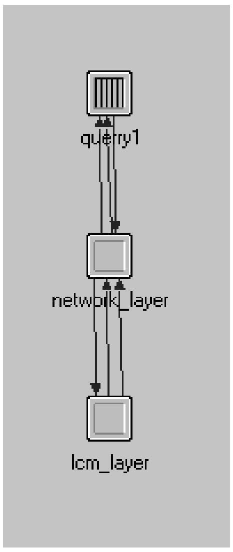


Figure B.23: Node model of CSMA-master node

B.2.8 CSMA-slave node

Fig. B.25 shows the node model of CSMA-slave node. In this node, there are two logical channel, one for the CSMA protocol and the other for master/slave protocol. In the CSMA channel, the slave is a sender. And the scr node generates the data and send it to the queue. Then, the queue stores the data. And the CSMA-slave node send the data according the CSMA protocol which we design in Chapter 7. In the master/slave channel, the slave receives the ACK from the master.
Process model of lcm_layer is the same as that in the ALOHA-slave node. In Fig. B.26, the process model of network_layer is shown. And it implements the SFN protocol and CSMA protocol.

B.3 Configuration

Parts of the necessary configuration parameters are integrated in the OPNET environment, while the remaining is defined by the environment files mechanism which also exists in OPNET. An example of this environment is shown in Fig. B.27.

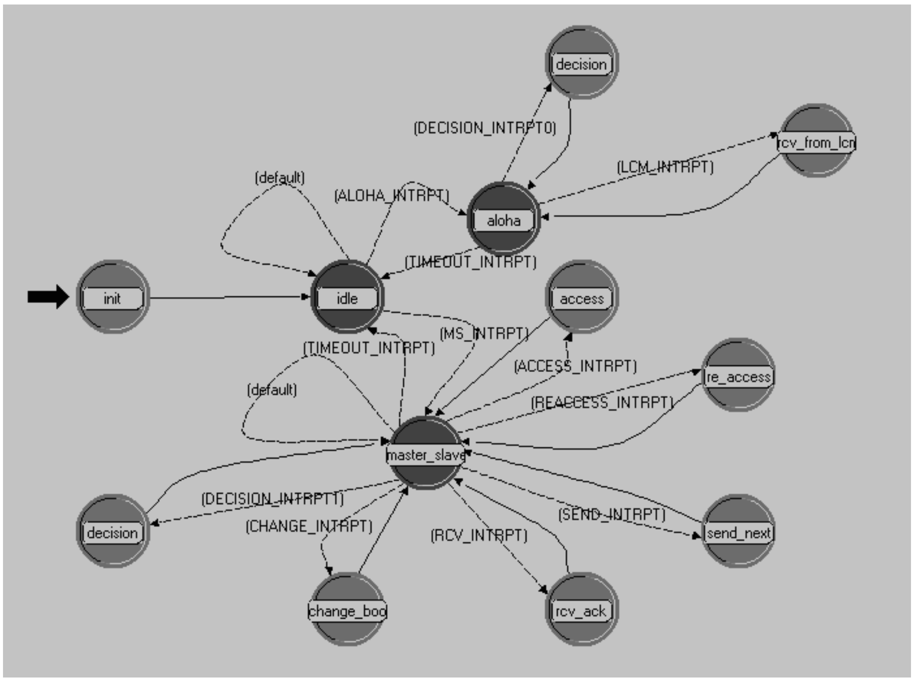


Figure B.24: Process model of network_layer

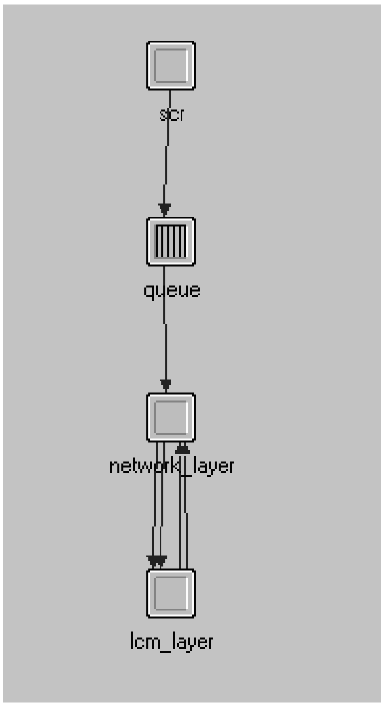


Figure B.25: Node model of CSMA-slave node

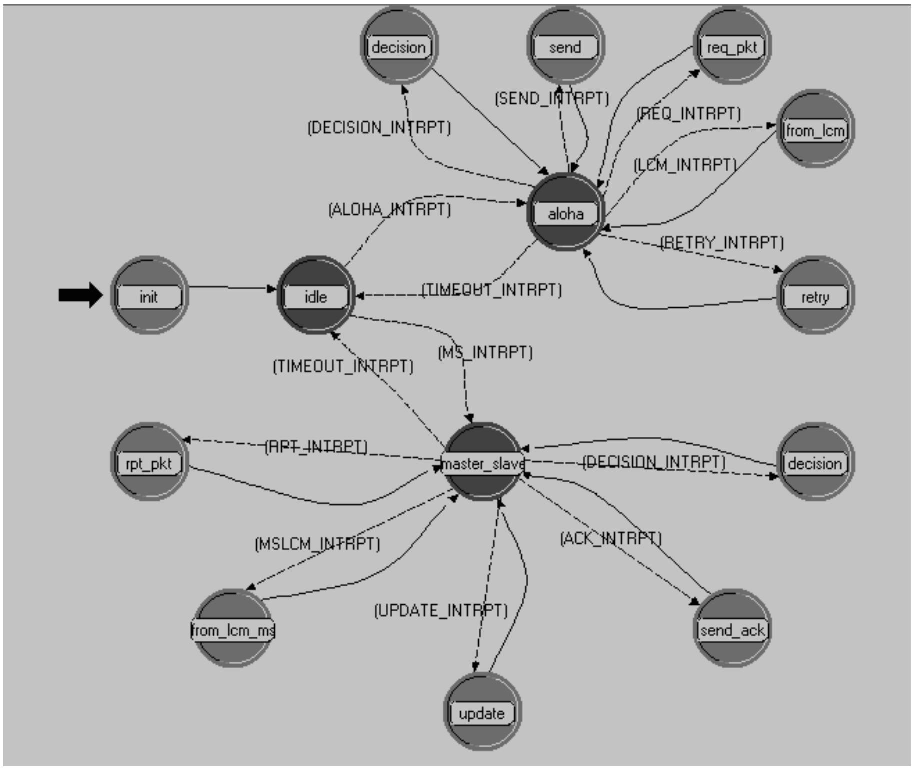


Figure B.26: Process model of network_layer

```
#number of logical channels
log_channels : 1

#tells what type of protocol exists in each logical channel
log_channel_0 : remp_sfn
#log_channel_1 : remp_alt
#log_channel_2 : remp_alt

#tells the number of slave nodes
slave_nodes : 9

#initial repeater level
repeat_downlink : 3
repeat_uplink : 3

#number of retries
number_retries : 2

#timeslot for start of collecting statistics
start_timeslot : 0

#B value parameter
b_value : 0
```

Figure B.27: OPNET configuration file

B.4 Statistics tools

Another aspect that shows the potentialities of the OPNET network simulation tool is its statistics module. An example is provided in Fig. B.28. Through the statistics module, it is possible to analyse the simulation results faster, by the help of generated graphics which are based in mathematical functions of the raw values. It is also possible to create animations for a better comprehension of the evolution of the simulations, but for the REMPLI project this option was not envisaged.
Note that all the statistics can be easily exported to the Excel format for further study and investigation.

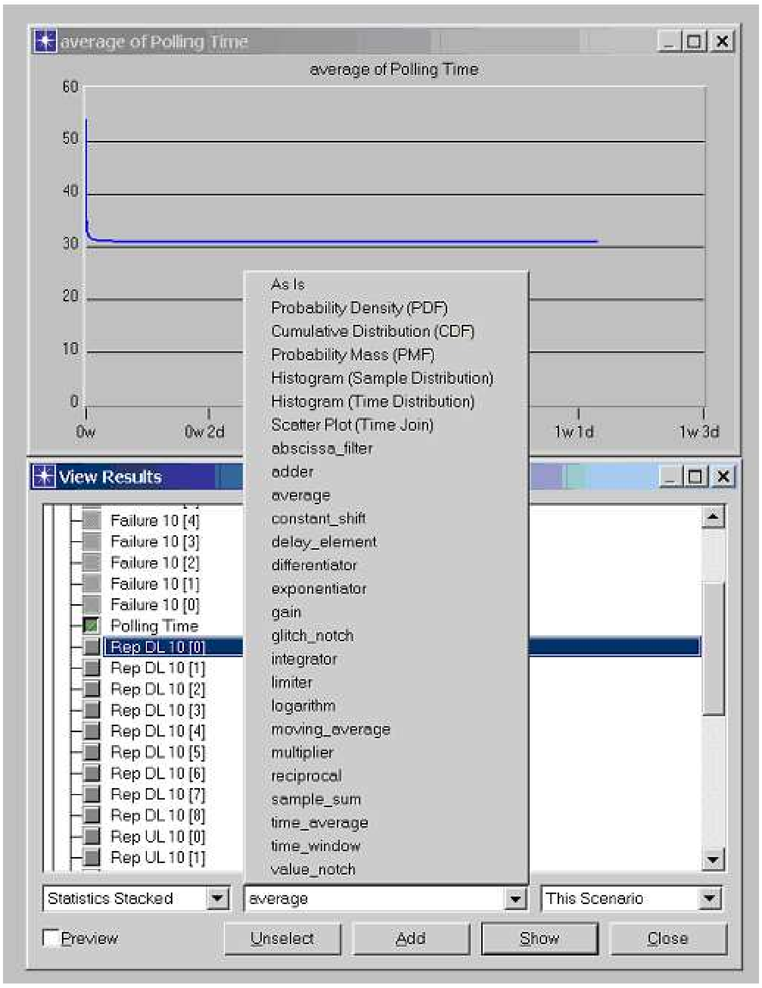


Figure B.28: Example of OPNET Statistics Module

Acronyms

ACK	Acknowledgement
AMR	Automated Meter Reading
ARQ	Automatic Repeat Request
AWGN	Additive White Gaussian Noise
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	carrier sense multiple access with collision detection
DA	Distribution Automation
DCC	Distribution Control Center
DG	Distributed generation
DLC	Distribution Line Communication
DLMS	Distribution Line Message Specification
DMS	Distribution Management Systems
DP	Dual-Priority
DSM	Distributed System Management
EHS	European Home System
EIA	Electronics Industry Association
FEC	Forward Error Correction
FSK	Frequency Shift Keying
QoS	Quality of Service
HV	High Voltage
IEC	International Electrotechnical Commission
IED	Intelligent Electric Device
ISI	InterSymbol Interference
MAC	Medium Access Control
MCM	Multi-carrier modulation
MV	Medium Voltage
LCU	Logical Channel Unit
LV	Low Voltage
OFDM	Orthogonal Frequency Division Multiplexing
PER	Packet Error Rate
PLC	Power Line Communication
PSK	Phase Shift Keying
RAM	Random Access Memory

RCS	Ripple Carrier Signalling
REMPLI	Remote Energy Management over Power Lines and Internet
RTU	Remote Terminal Unit
SCADA	Supervisory Control and Data Acquisition
SFN	Simple Frequency Network
TDMA	Time Division Multiple Access
WAN	Wide Area Network

List of my Publications

1. LiPing Lu, YeQiong Song, GangYan Li; Quality of Service Support in PowerLine Communication networks; In 5th IFAC International Symposium on Intelligent Components and Instruments For Control Applications 2003 - SICICA'2003. Aveiro, Portugal, July 2003

2. G. Bumiller, LiPing Lu, YeQiong Song; Analytic performance comparison of routing protocols in master-slave PLC networks; In 9th International Symposium on Power-Line Communications and its applications, Vancouver, Canada, on 6-8 April, 2005

3. Raul Brito, LiPing Lu, YeQiong Song; QoS requirements analysis; REMPLI Deliverable WP1.6, April 2004

4. LiPing Lu, Raul Brito, YeQiong Song, PLC system model & simulation; REMPLI project Deliverable 5.1, April 2005

5. LiPing Lu, Raul Brito, YeQiong Song; QoS and performance of REMPLI PLC network; In1st NeCST Workshop, Ajaccio, France, on 6-7 October 2005

6. LiPing Lu, J.H. Zhou, J. Hu, G.Y. Li, YeQiong Song; Embedded Powerline Communication in large Scale Distribution Automation and Demand Side Management System; In 2nd IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, Beijing, China on 13-16 August, 2006

7. LiPing Lu, GangYan Li, YeQiong Song; Powerline Communication System for Monitoring and Supervision of Feeder Equipments for MV Substation Automation; In IEEE Symposium on Industrial Embedded Systems, Antibes Juan-Les-Pins, France, on 18-20 October, 2006

Bibliography

[1] C.N. Krishnan, P.V. Ramakrishna, T.V. Prasad, "Power-line as access medium-a survey," *International Millennium Conference on Affordable Telecom and IT Solutions for Developing Countries*, March 2000.

[2] Pao-Hsiang Hsi, Shi-Lin Chen, "Distribution automation communication infrastructure," *IEEE Transactions on Power Delivery*, vol. 13, July 1998.

[3] K. Ghoshal, "Distribution automation: SCADA integration is key," *IEEE Computer Applications in Power*, pp. 31 – 35, Jan. 1997.

[4] D. Klaus, *Powerline communications.* Prentice Hall PTR Prentice-Hall, Inc, 2001.

[5] Pavlidou, N., Han Vinck, A.J., Yazdani, J., Honary, B., "Power line communications: state of the art and future trends," *Communications Magazine*, vol. 41, April 2003.

[6] G. Bumiller, "Single frequency network technology for medium access and network management," *6th International Symposium on Power-Line Communications and its application*, March 2002.

[7] *EN 50065-1: Signaling on Low Voltage Electrical Installations in the Frequency Range 3 kHz to 148.5 kHz.* CENELEC Brussels, 1997.

[8] Venkata, S. Pahwa, A. Brown, R. Christie, and R.D., "What future distribution engineers need to learn," *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 17– 23, Feb. 2004.

[9] S. Rong-Liang Chen; Sabir, "The benefits of implementing distribution automation and system monitoring in the open electricity market," *Canadian Conference on Electrical and Computer Engineering*, pp. 825–830, 2001.

[10] David G. Hart et al., "Managing MV networks for quality of service," *IEEE Computer Applications in Power*, vol. 15, no. 3, pp. 44–49, July 2002.

[11] T. Amakasu, "Power technology development schemes for the deregulated electrical power industry," *International Conference on Electric Utility Deregulation and Restructuring and Power Technologies*, pp. XVIII – XXXII, April 2000.

[12] Bin Qiu, Hoay Beng Gooi, Yilu Liu, Eng Kiat Chan, "Internet-based SCADA display system," *IEEE Computer Applications in Power*, Jan. 2002.

[13] Grasberg L., Osterlund L.A., "SCADA EMS DMS-a part of the corporate it system," *IEEE Power Engineering Society International Conference on Power Industry Computer Applications*, pp. 141 – 147, May 2001.

[14] Adamiak M., Premerlani W., "The role of utility communications in a deregulated environment," *Proceedings of the 32nd Annual Hawaii International Conference on System Sciences*, Jan. 1999.

[15] A. Pabla, *Electric Power Distribution.* McGraw-Hill Professional, 2004.

[16] C. Nietsch,D. Povh, "Optimizing power distribution systems," *IEEE Computer Applications in Power*, pp. 44–49, April 2001.

[17] John D. Kueck, Brendan J. Kirby, "The distribution system of the future," *The Electricity Journal*, June 2003.

[18] C. Gellings, "Power delivery system of the future," *IEEE Power Engineering Review*, vol. 22, no. 12, pp. 7– 12, Dec 2002.

[19] T. Ise, "Functions and configurations of quality control center on FRIENDS," *Transmission and Distribution Conference and Exhibition 2002: Asia Pacific*, vol. 1, pp. 590 – 595, Oct. 2002.

[20] Sato K., Katsumoto M., Miki T., "Future vision distribution system and network," *IEEE International Symposium on Communications and Information Technology*, vol. 1, pp. 274 – 279, Oct. 2004.

[21] "Draft IEEE SA technical report on substation integrated protection, control and data acquisition communication requirements," March 2003, technical Report IEEETR 1525-2003, Substation Committee of the IEEE Power Engineering Society. [Online]. Available: http://grouper.ieee.org/groups/1525/Technical%20Report/

[22] David J. Dolezilek, Dean A. Klas, "Using information from relays to improve the power system." [Online]. Available: http://www.selinc.com/techpprs/6080.pdf

[23] C. Ostertag, M.; Imboden, "High data rate, medium voltage powerline communications for hybrid DA/DSM," *Transmission and Distribution Conference*, vol. 1, pp. 240 – 245, April 1999.

[24] M. Shahidehpour and Y. Wang, *Communication and Control in Electric Power Systems.* John Wiley Sons, Inc., 2003.

[25] P. Roth, "Communications architecture in modern distribution systems," *16th International Conference and Exhibition on Electricity Distribution*, vol. 3, June 2001.

[26] J. V. Sambare, "Supervisory control and data acquisition (SCADA) and telecommunications - an overview," *TCE World*, October 2003.

[27] "NCS technical information bulletin 04-1: Supervisory control and data acquisition systems," October 2004, USA. [Online]. Available: www.ncs.gov/library/tech_bulletins/2004/tib_04-1.pdf

[28] D. Marihart, "Real time monitoring of the distribution network without the need for pooling," *Second International Symposium on Distribution Automation and Demand Side Management*, Jan. 1992.

[29] Marihart D.J., "Communications technology guidelines for EMS/SCADA systems," *IEEE Transactions on Power Delivery*, vol. 16, pp. 181–188, April 2001.

[30] D. G. Fink, H. W. Beaty, *Standard Handbook for Electrical Engineers,sec. 10, SCADA Communications, pp 142, para. 319, Power-Line Carrier;*. McGraw-Hill, Inc.

[31] A. H. Stefan Ramseier, Michael Arzberger, "MV and LV powerline communications: New proposed IEC standards," *Transmission and Distribution Conference*, April 1999.

[32] M. Propp, "The use of existing electrical power lines for high-speed communications to the home," pp. 92–100, 1999.

[33] Gebhardt M., Weinmann F., Dostert K., "Physical and regulatory constraints for communication over the power supply grid," *IEEE Communications Magazine*, vol. 41, pp. 84– 90, 2003.

[34] Meng, H., Guan, Y.L., Chen, S., "Modeling and analysis of noise effects on broadband power-line communications," *IEEE Transactions on Power Delivery*, vol. 20, no. 2, pp. 630 – 637, April 2005.

[35] M. Zimmermann and K. Dostert, "Analysis and modeling of impulsive noise in broad-band powerline communications," *IEEE Trans. Electromagn. Compat.*, vol. 44, no. 1, p. 249¨C258, Feb. 2002.

[36] H. Philipps, "Performance measurements of power line channels at high frequencies," *International Symposium on Power-line Communications and its Applications*, 1998.

[37] Brown, Paul A., "Some key factors influencing data transmission rates in the power line environment when utilizing carrier frequencies above 1 MHz," *International Symposium on Power-line Communications and its Applications*, 1998.

[38] Gotz, M., Rapp, M., Dostert, K."Power line channel characteristics and their effect on communication system design," *IEEE Communications Magazine*, vol. 42, pp. 78 – 86, April 2004.

[39] C. Hensen, "Data transmission applications via low voltage power lines using ofdm technique," *IEEE 5th International Symposium on Spread Spectrum Techniques and Applications Volume*, Sept. 1998.

[40] Onunga J.O., Donaldson R.W., "A simple adaptive data link protocol for performance enhancement on networks with variable-quality links," *IEEE Pacific Rim Conference on Communications, Computers and Signal Process*, vol. 2, pp. 554 – 560, May 1991.

[41] Younghyun Kim, Sungho Joo, Taeyoung Lee, Byungseok Park, Dukhwa Hyun, "PLC technology of KEPCO," *International Symposium on Power Line Communications and Its Applications*, pp. 390 – 393, April 2005.

[42] Ma Y.H., So P.L., Gunawan E., "Performance analysis of OFDM systems for broadband power line communications under impulsive noise and multipath effects," *IEEE Transactions on Power Delivery*, vol. 20, pp. 674 – 682, April 2005.

[43] Ron Gershon, David Propp, Michael Propp, "A token passing network for powerline communications," *IEEE Transactions on Consumer Electronics*, vol. 37, no. 2, pp. 129 – 134, May 1991.

[44] Abdelfatteh Haidine, Halid Hrasnica, Ralf Lehnert, "AC protocols for powerline communications: Aloha, polling and their derivatives," *Neue Kommunikationsanwendungen in modernen Netzen*, February 2002.

[45] Hrasnica H., Lehnert R., "Reservation domains in MAC protocols for broadband PLC networks," *International Symposium on Power Line Communications and Its Applications*, pp. 336 – 340, April 2005.

[46] Ohmi S., Yoshida S., Yamaguchi T., Kuroda G. "A media access control method for high-speed power line communication system modems," *IEEE Consumer Communications and Networking Conference*, pp. 295 – 300, Jan. 2004.

[47] Afkhamie K.H., Katar S., Yonge L., Newman R., "An overview of the upcoming HomePlug AV standard," *International Symposium on Power Line Communications and Its Applications*, pp. 400 – 404, April 2005.

[48] Pujante J.J., Martinez D., Gomez A., Marin I., "PLC QoS management and integration for IPv6 applications and services," *International Symposium on Power Line Communications and Its Applications*, pp. 131 – 135, April 2005.

[49] Yu-Ju Lin, Latchman H.A., Liu J.C.L., Newman R., "Periodic contention-free multiple access for broadband multimedia powerline communication networks," *International Symposium on Power Line Communications and Its Applications*, pp. 121 – 125, April 2005.

[50] Yu-Ju Lin, Latchman H.A., Minkyu Lee, Katar S., "A power line communication network infrastructure for the smart home," *IEEE Wireless Communications*, pp. 104 – 111, Dec. 2002.

[51] Abad J., Badenes A., Blasco J., Carreras J., Dominguez V., Gomez C., Iranzo S., Riveiro J.C., Ruiz D., Torres L.M., Comabella J., "Extending the power line LAN

up to the neighborhood transformer," *IEEE Communications Magazine*, pp. 64 – 70, April 2003.

[52] Saso M., Ouchida H., Iso M.; Naka T., Shimizu Y., "LV power line carrier network system for AMR," *Eleventh International Symposium on High Voltage Engineering*, pp. 23–27, Aug. 1999.

[53] Goffart B., Desneux P., Verbeure T., Gonthier P., Cermeno R, "An S-FSK power line carrier modem for utility AMR applications," *Ninth International Conference on Metering and Tariffs for Energy Supply*, 1999.

[54] Xiao Hui, Zeng Xiangjun,Fan Shaosheng, Wu Xuebin, Kuang Lang "Medium-voltage power line carrier communication system," *International Conference on Power System Technology*, vol. 2, pp. 1536 – 1539, Nov. 2004.

[55] Hui Xiao, Xuebin Wu, Xiaojiao Tong, Rong Lian, Zeng Xiangjun,Sheng Su, "Application of spread spectrum communication in medium voltage power line carrier," *Transmission and Distribution Conference and Exhibition: Asia and Pacific*, pp. 1 – 4, Aug. 2005.

[56] Weilin Liu, Widmer H., Raffin P., "Broadband PLC access systems and field deployment in european power line networks," *IEEE Communications Magazine*, vol. 41, no. 5, pp. 114 – 118, May 2003.

[57] Matsuo T., Maekawa S., "Field test of the world first 200 Mbps PLC modems," *IEEE International Symposium on Circuits and Systems*, vol. 5, pp. 5330 – 5332, May 2005.

[58] "White paper on power line communications (PLC)2004," 2004, PLC Utilities Alliance. [Online]. Available: www.fortress.net.au/Downloads/docs/PUA_White_Paper_Report.pdf

[59] Jee, G., Edison, C., Das Rao, R., Cern, Y., "Demonstration of the technical viability of PLC systems on medium- and low-voltage lines in the united states," *IEEE Communications Magazine*, vol. 41, no. 5, pp. 108 – 112, May 2003.

[60] Cole A.M., Tran B.Q., "Home automation to promote independent living in elderly populations," *24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society*, vol. 3, pp. 2422 – 2423, Oct. 2002.

[61] "HomePlug powerline alliance." [Online]. Available: www.homeplug.org

[62] "European home system (EHS) consortium." [Online]. Available: www.ehsa.com

[63] "IEEE guide for power-line carrier applications," *IEEE Std 643-2004 (Revision of IEEE Std 643-1980)*, 2005.

[64] W. Hagmann, "A spread spectrum communication system for load management and distribution automation," *IEEE Transactions on Power Delivery*, vol. 4, pp. 75 – 81, Jan. 1989.

[65] Craig A.D., Moore P.M., Long C.D., "Experience with a high speed MV/LV power line communications system for remote metering, load control and network automation," *Eighth International Conference on Metering and Tariffs for Energy Supply*, pp. 165 – 169, July 1996.

[66] K. Nunn, C.; Sullivan, "From theory to practice: the development and implementation of powernet -an advanced power line carrier system for AMR, DSM, DA and non-utility applications," *14th International Conference and Exhibition on Electricity Distribution*, vol. 5, pp. 8/1 – 8/5, June 1997.

[67] "X10." [Online]. Available: www.X10.com

[68] The CEBus standard user's guide.

[69] Echelon, "Control network protocol specification," ANWEIA-709.1, April 1999.

[70] "IEC 61334-5-1 (2001-05), distribution automation using distribution line carrier systems - pan 5: Lower layer profiles - section 1 : Spread frequency shift keying (S-FSK) profile, IEC 61334-5-1 ed. 1.0."

[71] "IEC/TS 61334-5-2 (1998-05), distribution automation using distribution line carrier systems -part 5-2: Lower layer profiles - frequency shift keying (FSK) profile."

[72] "IEC/TS 61334-5-3 (2001-01), distribution automation using distribution line carrier systems - part 5-3: Lower layer profiles - section 3: Spread spectrum adaptive wideband (SS-AW) profile."

[73] "IEC/TS 61334-5-4 (2001-06), distribution automation using distribution line carrier systems - part 5-4: Lower layer profiles - section 4: Multicarrier modulation (MCM) profile."

[74] "IEC/TS 61334-5-5 (2001-09), distribution automation using distribution line carrier systems - part 5-5 : Lower layer profiles - section 5: Spread spectrum fast frequency hopping (SS-FFH) profile."

[75] D. B. Johnson, D. A. Maltz, *Dynamic Source Routing in Ad Hoc Wireless Networks*, T. Imielinski and H. Korth, Eds. Kluwer Academic Publishers, 1996.

[76] Royer E.M., Chai-Keong Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Wireless Communications*, vol. 6, pp. 46 – 55, April 1999.

[77] M. Ulema, "Wireless sensor networks: architectures, protocols, and management," *Network Operations and Management Symposium*, vol. 1, pp. 19–23, April 2004.

[78] Ying Chun, Lv Qin, Liu Yong, Shi MeiLin, "Routing protocols overview and design issues for self-organized network," *International Conference on Communication Technology*, vol. 2, pp. 1298 – 1303, Aug. 2000.

[79] Arkus Sebeck, Gerd Bumiller, “A network management system for power-line communications and its verification by simulation,” *4th International Symposium on Power-Line Communications and its applications*, March 2000.

[80] REMPLI deliverable 1.2, "PLC system specification",http://www.rempli.org

[81] Deliverable 1.6 “QoS requiremetns analysis”, http://www.rempli.org.

[82] Gerd Bumiller, “Power-line physical layer emulator for protocol development,” *8th International Symposium on Power-Line Communications and its applications*, March 2004.

[83] “ns2” [Online]. Available: http://www.isi.edu/nsnam/ns/

[84] “OPNET” [Online]. Available: http://www.opnet.com/

[85] Ronald W.Wolff, *Stochastic modeling and the theory of queues.* Prentice Hall,Inc., 1989.

[86] Mak S., Radford D., “Communication system requirements for implementation of a large scale demand side management and distribution automation,” *IEEE Transactions on Power Delivery*, vol. 11, no. 2, pp. 683 – 689, April 1996.

[87] A. Koubaa, Y.Q. Song, “Evaluation and improvement of response time bounds for real-time applications under non-preemptive fixed priority scheduling,” *International Journal of Production Research*, July 2004.

[88] Glass A.M., Brewster R.L., Abdulaziz N.K., “Modelling of CSMA/CA protocol by simulation,” *Electronics Letters*, pp. 692 – 694, May 1988.

[89] Brito, R., Bumiller, G., Yeqiong Song, “Modelling and simulation of a SFN based PLC network", power line communications and its applications,” *9th International Symposium on Power-Line Communications and its applications*, April 2005.

[90] Cali, F., Conti, M., Gregori, E., “IEEE 802.11 wireless LAN: capacity analysis and protocol enhancement,” *INFOCOM ’98. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 142 – 149, March 1998.

[91] Canete F.J., Cortes J.A., Diez L., Entrambasaguas, J.T., “Modeling and evaluation of the indoor power line transmission medium,” *IEEE Communications Magazine*, pp. 41 – 47, April 2003.

[92] Luo W.Q., Tan S.Y., Boon Tiong Tan, “Modeling and measurements for power line communication systems,” *APMC 2005*, Dec. 2005.

[93] Dostert K., Galli S., “Keynote II: modeling of electrical power supply systems as communication channels,” *Power Line Communications and Its Applications*, p. 137, April 2005.

[94] Soo-Young Jung, Tae-Hyun Kim, Myung-Un Lee, Wook-Hyun Kwon, "Modeling and simulation of power line channel," *IEEE Power Engineering Society General Meeting*, pp. 793 – 797, June 2004.

[95] Meng H., Chen S., Guan Y.L., Law C.L., So P.L., Gunawan E., Lie T.T., "Modeling of transfer characteristics for the broadband power line communication channel," *IEEE Transactions on Power Delivery*, pp. 1057 – 1064, July 2004.

[96] Le Phu Do, Hrasnica H., Bumiller G., "Investigation of MAC protocols for single frequency network technique applied in powerline communications," *International Symposium on Power Line Communications and Its Applications*, pp. 22 – 26, April 2005.

[97] Mianyu Wang, Lin, E., Woertz E., Moshe Kam, "Collision resolution simulation for distributed control architectures using LonWorks," *IEEE International Conference on Automation Science and Engineering*, pp. 319 – 326, Aug. 2005.

[98] R. Davis, A. Wellings, "Dual-priority scheduling," *RTSS95*, May 1995.

[99] Yihong Zhou, Nettles S.M., "Balancing the hidden and exposed node problems with power control in CSMA/CA-based wireless networks," *Wireless Communications and Networking Conference*, pp. 683 – 688, March 2005.

[100] Kubbar O., Mouftah H.T., "An aloha-based channel access scheme investigation for broadband wireless networks," *Proceedings. IEEE International Symposium on Computers and Communications*, pp. 203 – 208, July 1999.

INSTITUT NATIONAL
POLYTECHNIQUE
DE LORRAINE

AUTORISATION DE SOUTENANCE DE THESE
DU DOCTORAT DE L'INSTITUT NATIONAL
POLYTECHNIQUE DE LORRAINE

o0o

VU LES RAPPORTS ETABLIS PAR :
Monsieur Christian FRABOUL, Professeur, INPT/ENSEEIHT, Toulouse
Monsieur Ken CHEN, Professeur, Université Paris 13, Villetaneuse

Le Président de l'Institut National Polytechnique de Lorraine, autorise :

Madame LU Liping

à soutenir devant un jury de l'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE, une thèse intitulée :

"Performances and quality of service of PLC networks for MV and LV distribution systems"

en vue de l'obtention du titre de :

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE

Spécialité : « **Informatique** »

NANCY BRABOIS
2, AVENUE DE LA
FORET-DE-HAYE
BOITE POSTALE 3
F - 54501
VANDŒUVRE CEDEX

Fait à Vandoeuvre, le 09 novembre 2006
Le Président de l'I.N.P.L.,
L. SCHUFFENECKER

INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE – NANCY

TEL. 33/03.83.59.59.59
FAX. 33/03.83.59.59.55

Résumé

Dans cette thèse, nous nous intéressons à la problématique liée à la communication numérique sur courant porteur dans l'objectif de fournir une infrastructure de communication qui permet la surveillance et le contrôle de la distribution et de la consommation de l'énergie. Ce travail a été effectué dans le cadre du projet européen REMPLI (Real-time Energy Management via Power Line and Internet) et a produit la spécification REMPLI PLC (Power Line Communication). Pour fournir une meilleure performance de la communication et effectuer le transfert de données en temps réel, en utilisant les réseaux électriques de moyenne tension et de basse tension, les problèmes principaux qui sont le routage dynamique de paquets de données, la gestion de la qualité de service et la notification des événement en temps réel, ont été traités. Nous avons développé un protocole de routage efficace pour s'adapter au changement dynamique de topologie du réseau électrique. Les performances de REMPLI PLC sont évaluées en utilisant des approches de simulation couplée avec des approches analytiques. Il est prouvé que la majorité des besoins applicatifs peuvent être satisfaite par REMPLI PLC. Nous avons proposé et implanté un nouvel "ordonnançeur de traffic" fournissant différents niveaux de qualité de service pour les applications. Des variantes de protocole ALOHA ont été proposées et évaluées afin de vérifier que les propriétés temps réel requises sur les notifications d'événements sont respectées.

Mots-clés: Réseau Longue Distance, Communication sur Courant Porteur, Routage, Qualité de Service, Evaluation de Performances

Abstract

In this thesis, we are interested in a wide-area PLC (Power Line Communication) network to provide a communication infrastructure for monitoring and control of energy distribution and consumption. This work has been carried out as a part of REMPLI (Real-time Energy Management via Power Line and Internet) European project and resulted in the definition of REMPLI PLC. For achieving high network performance and real-time data transfer using medium voltage and low voltage electricity grids, major problems such as dynamic packet routing, quality of service management and real-time event reporting are addressed. We designed an efficient routing protocol to cope with dynamic electricity grid topology changes and to relay packets to reach the destination. REMPLI PLC performance is evaluated using simulations and complemented by analytic studies. It is proved that most of the application requirements can be satisfied by REMPLI PLC. We proposed and implemented a new traffic dispatcher providing differentiated quality of service for applications. Based on ALOHA protocol, some variants have been designed and evaluated for enabling efficient real-time event notification.

Keywords: Wide-Area Network, Power Line Communication, Routing, Quality of Service, Performance Evaluation

摘 要

本文主要研究利用中、低压电力线构建广域网络平台来监测和控制能源的分配和消耗．本文的工作属于欧洲REMPLI "通过电力线和因特网进行实时能源管理"(Real-time Energy Management via Power Line and Internet)项目的一部分，利用中、低压电力线构建了REMPLI PLC网络．为了得到高的网络性能和实时的数据传输，需要解决以下主要问题：动态路由、网络服务质量的管理、和实时事件报告等问题．本文设计了一个基于洪泛机制的的路由协议来解决电力网网络结构动态变化的问题和远距离传输中数据包中继的问题；并使用仿真和概率计算两种方法分别对REMPLI PLC网络性能进行了评价，结果证明REMPLI PLC网络能够满足大部分的应用要求；其次，本文提出并实现了一个新的任务调度算法，为上层应用提供区分服务，保证网络的服务质量；最后，为了满足实时事件的获取，我们对现有的的随机接入协议(ALOHA和CSMA协议)进行了分析及改进，最终利用仿真做出了性能评价．

关键词: 广域网, 电力线通讯, 路由, 服务质量, 性能评价