Resource Allocation for Wireless Distributed Computing Networks

Xuetao Chen

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Tamal Bose, Chair
Jeffrey H. Reed, Co-Chair
William H. Tranter
Yaling Yang
Christopher W. Zobel

April 27, 2012
Blacksburg, Virginia

Keywords: Wireless Communications, Distributed Computing, Power Efficiency
Copyright 2012, Xuetao Chen

Resource Allocation for Wireless Distributed Computing Networks

Xuetao Chen

(ABSTRACT)

Wireless distributed computing networks (WDCNs) will become the next frontier of the wireless industry as the performance of wireless platforms is being increased every year and wireless industries are looking for "killer" applications for increased channel capacity. However, WDCNs have several unique problems compared with currently well-investigated methods for wireless sensor networks and wired distributed computing. For example, it is difficult for WDCNs to be power/energy efficient considering the uncertainty and heterogeneity of the wireless environment. In addition, the service model has to take account of the interference-limited feature of wireless channels to reduce the service delay.

Our research proposes a two-phase model for WDCNs including the service provision phase and the service access phase according to different traffic patterns and performance requirements. For the service provision phase, we investigate the impact of communication channel conditions on the average execution time of the computing tasks within WDCNs. We then discusse how to increase the robustness and power efficiency for WDCNs subject to the impact of channel variance and spatial heterogeneity. A resource allocation solution for computation oriented WDCNs is then introduced in detail which mitigates the effects of channel variations with a stochastic programming solution.

Stochastic geometry and queue theory are combined to analyze the average performance of service response time and to design optimal access strategies during the service access phase. This access model provides a framework to analyze the service access performance and evaluate whether the channel heterogeneity should be considered. Based on this analysis, optimal strategies to access the service nodes can be determined in order to reduce the service response time. In addition, network initialization and synchronization are investigated in order to build a multiple channel WDCN in dynamic spectrum access (DSA) environments. Further, an efficient primary user detection method is proposed to reduce the channel vacation latency for WDCNs in DSA environments.

Finally, this dissertation presents the complete design and implementation of a WDCN on COgnitive Radio Network (CORNET). Based on SDR technologies, software dedicated to WDCNs is designed and implemented across the PHY layer, MAC layer, and application layer. System experiments are carried out to demonstrate the performance issues and solutions presented in this dissertation.

This work is dedicated to Amy and Hong.

Acknowledgments

I am heartily thankful to my advisors, Dr. Tamal Bose and Dr. Jeffrey H. Reed, for their encouragement, guidance and support throughout my study here at Virginia Tech. They encouraged me and led me through both my research and my personal goals, and the completion of this dissertation is in large part because of their vision and dedication. I also would like to thank Dr. Willian H. Tranter, Dr. Yaling Yang, and Dr. Christopher Zobel for serving on my committee and their invaluable suggestions about my dissertation. I extend great gratitude to all my colleagues in Mobile and Portable Radio Research Group (MPRG) with whom I have experienced such a wonderful journey for the last five years.

I would like to particularly thank my parents and my wife for believing in me, and for their encouragement and support.

This work was supported in part by NSF Grant No. 0917973, ONR Grant No. N00014-07-1-0536, NSF PFI Grant No. 23708-0229-S02, and ICTAS research funding.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless distributed computing is an emerging area combining both wireless communications and distributed computing technologies. It is the means by which processing is shared among wireless devices in order to improve performance and exchange needed resources. The progress of wireless devices, the innovations of new applications in the wireless industry, and the increasing demands to support very complex applications with simple wireless devices provide the motivation for this research.

The evolution of computing and communication technologies makes distributed computing possible with wireless devices. Wireless industries have witnessed a dramatic increase of the computer processing capability of wireless platforms for the last ten years. Currently, main stream smart-phones have a CPU frequency around 1 GHz and 512M RAM [1]. The computing power of these mobile devices is comparable to the desktop computer ten years ago. In addition, wireless broadband technologies provide a high speed freeway for data distribution and collection. Further, energy efficient communication technologies reduce the overhead of communication power consumption while maintaining the required QoS, which is critical for wireless distributed computing.

Innovations in the wireless industry create a lot of opportunities and demands for future wireless networks. Wireless technologies, especially wireless personal access networks (WPAN), are used to collect utility data, distribute billing information, and transfer the status of the grid, as well as controlling information in the demand response and smart grid (DRSG) community. In cellular systems, Amazon's e-reader Kindle is a good example of a content distribution network operating through wireless technologies. The future success of these distributed applications will be largely supported by wireless machine-to-machine (M2M) technologies which provide wireless connections for distributed nodes. Verizon and Qualcomm are working together to develop a solution named nPhase in order to provide a platform for M2M applications [2]. All of the other carriers, such as Sprint, AT&T, and T-mobile, have their own strategies to create new M2M applications and increase the revenue of their networks. Some researchers even propose integrating device-to-device (D2D) technologies into

LTE systems to boost capacity and support high data rate applications such as video [3]. These M2M and D2D applications are expected to be the next frontier in wireless [2]. Newly introduced innovations in the wireless industry involve the cooperation between computing functions and communication functions. Moreover, the Global Information Grid (GIG) provides an information access and sharing platform utilizing both commercial communication technologies and a military information infrastructure [4].

1.1 Wireless Distributed Computing Network

A wireless network has advantages as an access technology instead of a transmission technology. Compared to optical fiber systems, wireless networks do not perform well for long-range large-volume data transmissions, such as low energy efficiency, high error ratio, low capacity, and higher latency. Therefore most successful large area wireless systems adopt wireless links for their radio access networks and optical fibers for their core networks.

However, current commercialized hierarchical wireless systems are not efficient for localized traffic. For example, a cellular phone call needs to go a long distance and through multiple network elements to reach someone in another room within the same building. Moreover, interests of local information (I) can be represented as a decreasing gradient function of the range (d) from the location generating the information. For example, a severe weather event, such as a thunderstorm, normally has a scale of tens of miles and lasts several hours. People living within a certain range close to the event have the most effects from it. A better way to share local information is to keep the traffic local and reduce the distance between the information source and potential users.

In the author's opinion, transferring a huge amount of data over a long distance with multiple hops may be not a good solution for these types of applications. The future of wireless communications is to improve local information collection, processing, storage, and sharing/access.

WDCNs will have the capability to provide an information service utility available everywhere like power and water today. In the future, It can group heterogeneous devices with wireless links. For example, the information is first collected by sensors which measure physical parameters and translate them into electrical signals. These electrical signals may be accessed, processed, and stored by computing devices in a distributed way. Anyone within a WDCN can access this information. Nodes within WDCNs can access and share not only information but also hardware and software with a unified file system. For example, a unique ID can be assigned to each resource using hash functions in order to identify the resource. Each node maintains a portion of the whole index table about the resource attached to itself and its neighbor nodes. Obviously, storage and process are both optional for hardware sharing.

The author's concept of WDCNs is shown in Figure 1.1. There are four possible roles for

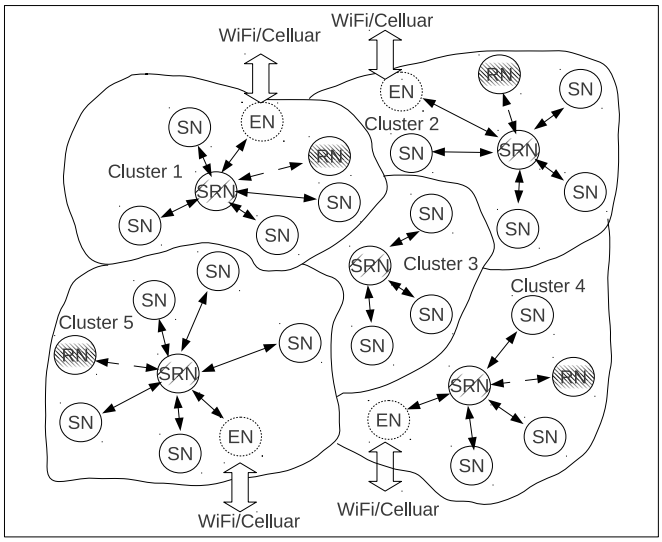

Figure 1.1: The author's concept of WDCNs.

each node within a WDCN. Each node may play different roles in different service sessions in this cluster based flat network. Desired resources are attached to service nodes (SNs). Those requesting resource sharing are request nodes (RNs). Service request nodes (SRNs) are the agents between RNs and SNs within each cluster. SRNs also maintain a status table of all SNs within one cluster and are in charge of resource allocation and access control. Edge nodes (EN) relay the information flow from inside the cluster to the outside. In this way, the traffic is localized within a certain range. When needed, the local information may be exchanged over a long distance with existing hierarchical systems, such as WiFi or cellular systems.

A WDCN is a heterogeneous network. The heterogeneities of wireless links and devices combined with the new definition of the network utility impose strict performance requirements on data latency and reliability. Moreover, energy efficiency and power efficiency need to be investigated before WDCNs become practical. To support the network operation, research and development of resource allocation methods are required, which we will discuss briefly in the following sections.

1.2 Problem Statement

WDCNs have unique problems compared to traditional distributed computing and wireless communications, among which power management with joint communication and computing optimization, latency reduction, and system architecture are three important issues.

For each node, power consumption consists of two parts, computing power consumption and communication power consumption. If either the computing process or communication process dominates the power consumption, traditional power management for distributed computing or communication is used to design an efficient power management solution. Recent measurements of video conferencing on a mobile station show that the power consumption of the computing process and communication process can be at the same level for WDCNs [5]. Thus, a cross-layer energy efficient protocol needs to be designed considering two power consumptions at the same time. The communication power consumption is the price paid for distributing computation and should be reduced to the minimum. On the other hand, the communication rate should be fast enough to support the computing rate. This means the reduction of communication power consumption is constrained by the computing rate requirement. Thus a distributed cross-layer power-rate allocation scheme is an essential component of power management in WDCNs.

The concept of latency in WDCNs is different from wireless communications. The execution latency of a task in WDCNs is determined by the slowest node. The performance indicator of latency is the maximum latency among WDCN nodes instead of the average latency among nodes. In addition, the latency reduction in WDCNs is challenging with military radios, especially software defined radios [6]. Unpredictable latency may destroy the timing among components and compromise protocol designs.

The third problem is to create an effective system architecture, which includes the formulation, initialization, and resource discovery, as well as how to design fault-tolerance and security mechanisms. Although some lessons can be learned from traditional wired distributed computing, new approaches need to be created to handle a wireless channel with more uncertainty and less reliability. Recent progress of dynamic spectrum access (DSA) and cognitive radio (CR) offer an opportunity for the implementation of WDCNs. However, this requires some special consideration about the DSA environment.

As multi-core processors become prevalent, multi-threading programming is another new research challenge for the software of WDCNs. There are at least two processes for wireless distributed computing, the communication process and the computing process. The software implementation has to monitor, control, and maintain these two processes independently. In addition, because of the link uncertainty caused by the wireless channel, the CPU can be under utilized and still satisfy the delay requirement. The lower utilization leads to lower power consumption if CPUs run at a lower supply voltage and lower clock frequency [7]. Lower CPU utilization can be implemented by scheduling among multi-cores. Depending on different multi-core technologies, scheduling strategies may differ. It remains an open research question of how channel conditions influence scheduling multi-threads on multi-core CPUs in order to reduce power consumption.

The solutions for these problems may vary for different types of WDCNs since wireless distributed computing applications have different purposes and specifications. WDCNs can be grouped into three categories based on the purpose of different types of wireless distributed

computing applications.

- Class 1 WDCNs are utility oriented WDCNs, which include data collection networks such as wireless sensor networks (WSN) and smart grid networks.
- Class 2 WDCNs are computation oriented WDCNs, which include data processing computing such as distributed radio systems.
- Class 3 WDCNs are data oriented WDCNs, which include distributed data storage and sharing systems.

These three categories have different QoS requirements, power consumption models, and system assumptions. There is no one-size-fits-all solution for different kinds of WDCNs. For example, traditional dynamic voltage scaling may achieve more energy saving with Class 2 WDCNs compared to a Class 1 WDCN since Class 2 has a higher duty cycle and a larger portion of computing power consumption.

In addition, the service session can be divided into two phases including the service provision phase and the service access phase according to their different traffic patterns and performance requirements. A response from one service node (SN) is sufficient for a successful service request in the service access phase, while multiple responses and data transmissions from several SNs are required in the service provision phase. In addition, the duty cycle in service provision is higher than in the service access phase. Therefore, different strategies and actions are needed for these two phases in order to improve energy/power consumption and computing performance.

The different approaches for WDCNs compared to the traditional wired distributed computing networks are driven by the wireless channel. The random nature of the wireless channel is the challenging problem for WDCNs. The broadcasting nature of the wireless channel as well as the random location of nodes make interference mitigation a challenging problem. Moreover, the tradeoff between robustness and power efficiency becomes even more important for wireless nodes in WDCNs.

1.3 Background

1.3.1 Energy Efficiency

Significant research has gone into energy efficiency in both the wireless communication and distributed computing areas.

In the communications area, sub-threshold circuit design, high efficiency power amplifier, and new power supply technologies have been introduced to reduce energy consumption and

improve power efficiency [8][9][10]. In addition, adaptive modulation and coding, intelligent MAC, and energy-aware routing protocols help wireless networks improve energy efficiency [11][12]. Opportunistic transmission responding with power and rate adaptation saves energy and increases the robustness of communications for fading channels. Recently, dynamic programming and Markov models have been introduced for opportunistic transmission in order to improve the transmission power efficiency by adaptively changing the status among active, idle, and sleep modes [13][14][15]. Moreover, there is also some progress concerning multi-hop wireless networks in terms of energy efficiency and delay reduction [16]. The tradeoff between delay and capacity for wireless networks is currently an active research topic [17][18][19].

In the distributed computing area, server clusters exploit dynamic voltage scaling (DVS) combined with CPU scheduling for better energy efficiency [20][21][22][23]. Spatial and temporal heterogeneities are considered in the workload distribution to improve energy efficiency and robustness [24]. Advanced numerical algorithm design can reduce the energy consumption for large data set analysis, which reduces the computation load from a cubic function of the data size to a quadratic function when computing the inverse covariance matrices [25].

Increasing the system operation temperature is another promising energy saving technology [26]. A higher operation temperature was supposed to lead to a higher device failure ratio. The system operating temperature is normally controlled within a temperature range by a cooling system which consumes a large amount of power. However, recent experiments show there is no significant correlation between the temperature and the disk failure ratio [27]. The system operating temperature may be increased to reduce the power consumption of the cooling system, which will potentially benefit server farms.

There are two parts of power consumption for distributed computing: computation power consumption and communication power consumption (see Section 1.2). The ratio of these two modes of power consumption depends on the application as well as the platform design. In addition, computation power consumption can be proportional with respect to the computation load by DVS technologies, while communication power consumption is more related to the inter-connections links. Recently, flattened butterfly and dragonfly inter-connection topologies for energy-proportional inter-connection networks have been introduced to reduce communication power consumption by 85% for a wired data center. The communication power consumption in this system is more proportional to the amount of traffic it is moving, [28][29]. These techniques form virtual routers by grouping several routers to increase the radix, which can reduce the global diameter to one. Within the virtual router, a cheap electrical channel is used for local communication while optical fibers are used for the global channels among these virtual routers. These hierarchical topologies also avoid the problems of the conventional tree structure, such as bandwidth limits and increasing hop counts as packets travel up the hierarchy.

1.3.2 Performance Related Methods

In addition to energy efficiency improvement, system performance related technologies, such as the wireless network capacity scaling law, queue and scheduling theory, multi-core technologies, and implementation software, are important tools to evaluate and design WDCNs.

When modeling wireless network capacity, researchers noticed that interference, the propagation index, and node locations cause different choices of network topologies and cooperation strategies to maximize the network capacity [30][31][32][33]. The development of interference alignment removes the requirement of spatial reuse and approaches a linear scaling of network capacity for the high SNR region [34]. It is still arguable whether network capacity has a linear scaling law or a square root scaling law [35]. One of the other emerging research trends is to use non-equilibrium theory to model the dynamic performance of wireless networks. Instead of considering the average performance from zero to infinity, researchers evaluate the performance, such as convergence rate, complexity, etc., within a limited time and amplitude scale [36]. Recently, random graph and stochastic geometry have been used to model the interference of wireless networks to find the optimal solution to enhance the performance by integrating the location, fading, and protocols [35].

For computing theory, queue theory is used to model the service delay performance for network systems [37]. Randomized workload allocation frameworks, such as the ball and bin model, large deviation theory, fluid model, etc., can be used to find both the limiting bound and dynamic characteristics of the service process [38]. It is not surprising to see that the service process can also be modeled with differential equations and Markov chains [39].

One of the major advantages of WDCNs is the execution time speedup for computing tasks with cooperation among multiple nodes. Depending on different multi-core and multi-thread technologies, the speedup performances are different. Speedup is defined in term of the number of cores (n) used in parallel implementation as

$$Speedup = \frac{Time_{best\_in\_sequential}}{Time_{parallel\_implementation}(n)} \tag{1.1}$$

where the numerator is the execution time for the best sequential implementation and the denominator is the execution time of the paralleled implementation. Amdahl claimed this function can be represented as [40]

$$Speedup = \frac{1}{S + (1 - S)/n} \tag{1.2}$$

where S is the time spent on executing the serial portion of parallel implementation and n is the number of cores. This equation assumes that the program takes one unit of time to execute the sequential algorithm. When $n \to \infty$, the speedup has the upper bound of $1/S$, which indicates that reducing the portion of the serial part of the implementation is much more important than adding more cores/threads to speed up the program.

When including the overhead of adding threads and the communication between threads $H(n)$, the speedup equation turns out to be

$$Speedup = \frac{1}{S + (1 - S)/n + H(n)} \tag{1.3}$$

which indicates a poor parallelization can potentially slow the performance. Amdahl's law is a good approximation of speedup performance for hyper-threading technologies (HT) of CPUs since HT has a shared cache among different cores. It is also a good tool to describe the speedup performance for WDCNs when the wireless nodes share the channels.

However, a scaled linear speedup equation, Gustafson's law, developed in the late 1980s shows the real potential of parallel computing as

$$Scaled\ Speedup = n + (1 - n) * S \tag{1.4}$$

where this equation assumes the computing problem size increases as the number of cores increases [41]. Moreover, multi-thread technologies can be combined with a virtual machine to combat the network churn [42].

For implementation tools, SDR, dynamic spectrum access (DSA), and cognitive radio (CR) are the enabling tools for WDCNs. By moving physical layer functions from the hardware to the software domain, SDR offers an unprecedented level of flexibility in radio development and operation, which can facilitate the research and development of WDCNs. The radio RF parameters and even the protocols are reconfigurable for SDR. In addition, DSA measures the spectrum usage and dynamically accesses radio channels when a white space of spectrum is recognized. Further, CRs have the cognition to learn from their experience and make the optimal decisions about radio and network configurations based on its observation. These capabilities are extremely useful for WDCNs considering the heterogeneous nature of wireless links and devices.

1.3.3 Tradeoffs

There are two important tradeoffs for WDCNs, one is the tradeoff between energy and delay performance, the other is the tradeoff in the system architecture. The most important tradeoffs for WDCNs are among energy/power consumption, energy/power efficiency, and delay performance. Without careful design, the reduction of power consumption does not necessarily lead to a reduction of energy consumption or a increase of energy efficiency. Energy efficiency is the normalized energy consumption, which is equal to the product of the power consumption and execution time, normalized by the traffic load.

$$Energy\ Efficiency = \frac{(Power\ Consumption) \times (Execution\ Time)}{Traffic\ Load} \tag{1.5}$$

Inverse of this energy efficiency gives another definition used in some literatures. For DVS technologies, power consumption has a polynomial relation with the clock frequency of the

CPUs. The scaling down of voltage or frequency increases the execution delay and reduces the power consumption.

Power efficiency is the ratio between power consumption for a certain purpose and the total power consumption.

$$Power\ Efficiency = \frac{Useful\ Power}{Total\ Power} \tag{1.6}$$

For example, power efficiency for a WDCN can be defined as the ratio between the computation power consumption and the total power consumption. An optimal power allocation for WDCN only allocates enough power for communications and puts as much power as possible into the computation, which is called a power efficient solution. Again, the power efficient scheme normally may not lead to the delay optimal solution. However, the delay requirement varies from application to application. An energy efficient and power efficient solution satisfying the QoS requirement may exist if the channel conditions are carefully considered [45].

The Lyapunov method can provide optimal delay performance with guaranteed energy consumption for time varying wireless networks. Paper [46] introduces a virtual queue model for a power constraint in order to transform time average inequality constraints into the queue stability problem. It defines conditional Lyapunov drift as the sum of squares of the queue backlog. Then the bound on the drift-plus-penalty is computed, based on which the policy to minimize this bound can be designed, such as the max-weight algorithm. This drift-plus-penalty method offers a performance bound of $[O(1/V), O(V)]$ where V is a constant multiplier for the penalty term. Unfortunately, the solution is highly complex for wireless networks with interference [47] which makes this method more suitable for the opportunism TDMA protocol by only allowing one transmission, for each slot, from the node with the highest weighted rate.

The second important design choice is the network architecture, especially the tradeoff between the number of hops and the length of a hop. It is interesting to notice that both wireless networks and wired networks have this tradeoff. In the computer science field, researchers have found that it is more efficient to use increasing pin bandwidth by creating high-radix routers with a large number of narrow ports instead of low-radix routers with fewer wide ports [29]. For WDCNs, more hops increase the delay but reduce interference. Long hop length reduces the number of hops but is much more open to interference. This indicates that a hierarchical network structure based on clustering may be energy efficient for WDCN applications if the cluster size and the hierarchical layers are also carefully selected [31][33]. Some researchers show that there is actually a transition point after which multi-hop is a better choice than single hop [48]. In addition, researchers from both areas agree that the link design among nodes can be more energy efficient by choosing optimal data rate and channel equalization technology [28][49].

1.4 Contributions

Research and development in this dissertation intend to provide not only a system/network level design but also analysis for individual underlying technologies. This research makes the following contributions

- It identifies the major research challenges for WDCNs: how to consider the impacts of large scale channel heterogeneity on power efficiency and delay performance. Several methods to solve the problem are proposed. Detailed explanations and quantitative results are provided. The results show that both the mean and variation of the channel condition should be considered for workload allocation.
- It proposes a cross-layer resource allocation method dedicated to WDCNs that considers communication power consumption, computing rate, and computing workload together. A two-time scale framework is proposed in order to improve power efficiency and computing robustness.
- It analyzes the performance of service response time with a novel framework combining node geometry, protocol, and queue theory. It presents the analysis of the impacts of the wireless channel, protocol, and workload arrival density on service access performance, which provides a service model for WDCNs.
- It addresses practical issues for network initialization and synchronization. Contention history is used to build multi-cluster and multi-channel WDCNs.
- It presents a primary users detection method based on support vector machine (SVM) for WDCNs in a dynamic spectrum access environment.
- It designs and implements a software radio based WDCN example which makes an experiment study of the channel impacts on WDCNs feasible. The example illustrates the key performance issues for WDCNs and their solutions with experimental data.

1.5 Dissertation Organization

The main body of this dissertation is organized into seven chapters discussing the service access phase and the service provision phase as shown in Figure 1.2. While Chapter 2 and Chapter 3 present the research challenges and their solutions for the service provision phase, Chapters 4 to 6 are dedicated to the service access phase. An example of WDCN on CORNET is presented in Chapter 7. Chapter 8 concludes the dissertation.

The dissertation starts in Chapter 2 with an investigations into the impacts of channel variation on power efficiency and delay performance for WDCNs. It has been found that the

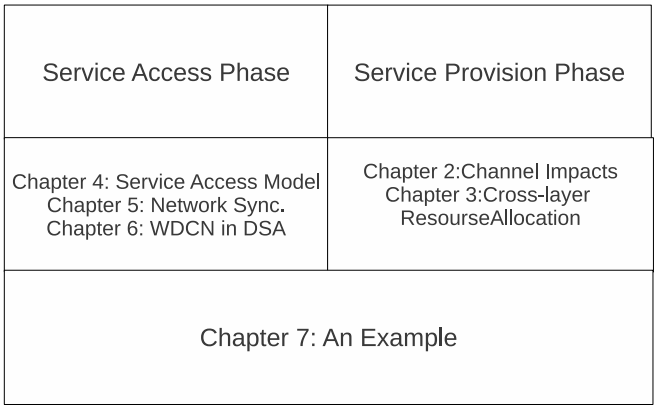

Figure 1.2: The organization of the dissertation.

delay performance of wireless distributed computing is influenced by both the average channel condition and the variation of channels. In addition, the impact of channel heterogeneity is also investigated to show the possibility of the optimal workload distribution for energy saving and robustness. Based on these analyses, a workload distribution approach exploiting the spatial heterogeneity of the channel condition is proposed to balance energy efficiency and robustness.

Chapter 3 introduces a power-rate-workload allocation scheme for the service provision phase. A detailed cross-layer resource allocation solution for computation oriented WDCNs is introduced. By maximizing the network computing capability and minimizing the communication power consumption, the majority of the power supply can be used to support the computing process. With less of the workload allocated to nodes with slower computing rates, the robustness of the average execution time could be improved. Next, this chapter introduces a two-time-scale framework to combine these resource allocations together.

Chapters 4 to 6 focus on performance in the service access phase. Chapter 4 explores the performance issues of the service access model by combining the interference model, the protocol model, and the queueing model. This chapter provides a framework to analyze the service access performance of WDCNs, especially the average service response time. It also introduces optimal access strategies for multiple server cases in WDCNs. A special case is studied where the performance of two SNs speeds up when compared to methods that do not consider channel heterogeneity. A numerical method is used to find the optimal policy for the cased with more than three SNs.

The recent progress of software defined radio (SDR) and dynamic spectrum access (DSA)

provides a powerful implementation tool for WDCNs. However, the application of these two in WDCNs is a challenging research problem. Chapters 5 and 6 intend to shed light on this issue. Due to heterogeneous wireless links and different communication protocols among nodes, reconfiguration capability is essential. Although software defined radio may leverage this problem to some extent, how to synchronize the reconfiguration process in a dynamic environment with heterogeneous nodes remains an open problem. Chapter 5 describes a network initialization and synchronization method for WDCNs, especially those based on SDR technologies. It first discusses the reconfiguration requirements of WDCNs. Then, the design challenges for logic clock synchronization are introduced and analyzed in terms of the lower bound and practical design. A method for SDR based WDCNs is proposed to form a multiple-cluster and multiple-channel network without a significant increase of time complexity compared with the reference design.

After a brief review of research progress in signal detection and classification, Chapter 6 describes WDCNs in a DSA environment. It proposes an approach to detect the primary user during the communication of the secondary users using the concept of interference detection in the presence of a desired signal. The detection problem is first formulated as a multi-class classification problem. The pattern with medium bit error rate (BER) and low interference to signal power ratio (ISR) is identified as the most difficult case. A classifier based on support vector machine (SVM) is proposed to solve the problem.

Based on the above enabling technologies, Chapter 7 presents the design and implementation of WDCNs using COgnitive Radio NEtwork Testbed (CORNET). The resource allocation ideas are implemented using SDR technologies. It then presents the performance experiment results which match the analysis results in previous chapters. The combination of SDR and multi-process programming lay a foundation to implement all kinds of WDCNs with different computing applications in the future.

The last chapter concludes the dissertation and looks forward to some future research.

Chapter 2

The Impacts of Channel Variations on WDCN

2.1 Introduction

Wireless distributed computing based on hand-held platforms has several unique problems compared with currently well investigated wireless sensor networks. These problems include the impact of channel variation on power allocation, different traffic pattern with higher utilization, and more restricted delay constraints. This chapter investigates the impact of communication channel conditions on the average execution time of the computing task within WDCNs. It has been found that the delay performance of wireless distributed computing is influenced by both the average channel condition and the variation of channels. In addition, the impact of channel heterogeneity is also investigated to show the possibility of the optimal workload distribution for energy saving and robustness. Finally, a workload distribution approach exploiting the spatial heterogeneity of the channel condition is proposed to balance energy efficiency and robustness.

Significant research has gone into workload balance for power efficient distributed computing. Server clusters with efficient power management can save millions of dollars each year [52]. The computing speed can be adjusted to balance mean response time and mean energy consumption [53]. Platform heterogeneity is exploited to build power efficient data centers [54]. Because of the heterogeneity of the system, the work load needs to be balanced, based on the availability of computing power [55]. To balance the workload, the prediction of execution time is estimated by considering the available power and the channel impact [56]. In addition, a lot of energy efficient designs are proposed for distributed wireless networks, especially for wireless sensor networks and ad-hoc networks [57]. Few papers discuss the combination of power allocation and workload distribution for a WDCN task in a heterogeneous enviroment.

In wireless distributed computing, a wireless communication process serves the distributed

computing process. One of the problems is how much power consumption it takes for communication process to guarantee a robust information exchange amongst nodes in a timely manner. The communication power consumption is actually the "transportation" price paid for distributing computing. The nodes with poor channel conditions have to spend more power to maintain reliable communications. If more power is consumed by the communication process, the less power can be used by the computing process. Therefore, the total computing capability of the cluster is determined by the maximum computing power excluding the communication power from the total power supply. Lower computing power may lead to weaker network computing capability. In addition, the channel variation makes the robustness of the network fragile since the delay performance in wireless distributed computing is determined by the maximum delay instead of the average delay as in traditional wireless communication systems.

The rest of this chapter is organized as follows. Section 2.2 discusses the system assumptions and commonly used techniques in wireless computing networks. Section 2.3 presents the impact of channel variation and a workload distribution algorithm. Simulation results in Section 2.4 demonstrate the improvement of robustness and energy efficiency with the proposed approach. Section 2.5 concludes the chapter.

2.2 System Assumption

In a WDCN, the distributed service nodes, which provide computing resources, are around a client node. The client node distributes its workload to service nodes and wait for the processed data. The channel variation or shadowing loss of each link is assumed as an i.i.d. random variable with log-norm distribution. We only consider single-hop wireless network without random locations of nodes, interference among links, and the buffer/queue process. The capabilities of the communication process and the computing process are constrained by the power supply when the wireless channel is not bandwidth-limited. The execution time of the computing task is determined by the maximum execution time among links, because the computing task is completed only after the client node receives all the processed data from the service nodes in a WDCN.

2.2.1 Power Supply

We consider two types of power supplies: battery and solar power. Most current mobile terminals use batteries as their power supply. For mobile terminals, the computing process power increases according to Moore's law, doubling within 18 months, while the increase of the battery capacity will be only about 80% within next ten years [58]. At some point in the future, the battery capacity may not support a powerful mobile station without frequent charging. Power efficient mobile terminals will leverage this constraint to some extent. The

other solution is to use renewable energy, such as solar power. Some cell phone vendors have started to develop solar-powered handsets [59]. In military applications, the solar soldier concepts have been proposed to satisfy the power requirement for missions longer than 48 hours. The main functionality of solar power is to extend the interval between two adjacent battery chargings. Power efficient designs are still needed to extend the battery life.

Although solar power can generate sustainable power supply, the output power is still too small for each node to support a complex computing task. WDCNs may group the nodes together to fulfill the request task and exploit power as much as possible with an efficient power management scheme.

2.2.2 DVS and APC

Dynamic voltage scaling (DVS) is commonly used to increase the power efficiency of the computing components, especially CPUs [60]. The fundamental units for modern digital circuits are transistors. The power consumption of transistors includes a static power consumption component and a dynamic power consumption component, which can be described as

$$P = P_{static} + P_{dynamic} \tag{2.1}$$

Static power consumption is due to a leakage current when the applied voltage is smaller than the "turn-on" voltage and no current is supposed to flow through the transistors. Dynamic power consumption or switching power consumption can be expressed as in [60]

$$P_{dynamic} = CV^2 f \tag{2.2}$$

where C is the capacitance being switched per clock cycle, V is the applied voltage, and f is the clock frequency. A faster switching frequency or computing clock frequency can be supported with higher applied voltage. When the computing burden is light, the voltage and clock frequency can be switched to a low level to save power. Since the increase of clock frequency is accompanied with an increase of voltage, the dynamic power consumption is actually proportional to a low-order polynomial function of clock frequency. In practice, this relationship can be modeled by a curve fitting using the measured data.

Adaptive power control (APC) is used to match transmission power to the channel condition [63]. When the channel availability is poor, a higher transmission power is used to satisfy the constraint of error performance. When the channel availability is good, the transmission power is reduced to save power consumption.

The goal of power efficiency is to use the optimal amount of power needed for the computing process or the communication process. DVS and APC technologies shown above can achieve this goal separately for the computing process and the communication process. The choice of the optimal power allocation scheme balances the tradeoff between power and delay performance as well as robustness of the network.

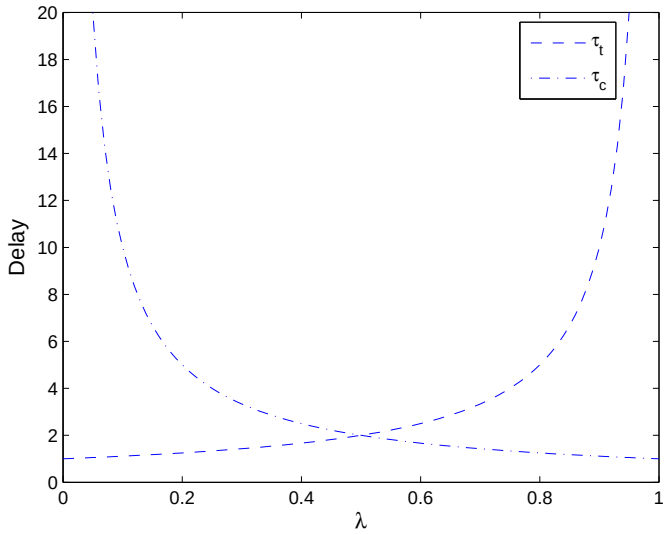

Figure 2.1: The power allocation for delay hiding. $a = 1$, $b = 1$, $P_0 = 1$; $n = 1$;$\lambda_{opt} = 0.5$ in this case.

2.2.3 Delay Hiding

Overlapping of computing and communication processes have been shown to improve the power efficiency of iterative computing jobs [61]. There is also the issue of transmitting while computing or transmitting after computing. To improve delay performance further, delay hiding techniques hide the delay of the fast process behind the delay of the slower processes to reduce the total execution time. This is feasible when the power supply can support the communication process and the computing process simultaneously. In this paper we consider only the transmission delay and the computing delay. The queue delay and propagation delay are ignored for simplicity of analysis. The queue delay can be added to the framework by adding the power consumption model for the memory access bounded computing task [54]. It can be shown that the transmission delay τ_t is inversely proportional to the transmission power P_t, and the computing delay τ_c is inversely proportional to the $(P_c)^{1/n}$, where P_c is the computing power and n is the order of a polynomial equation 2.5. The delay for node i with total power of P_0 using the delay hiding technique can be represented as

$$\begin{aligned} \tau_i &= max(\tau_t, \tau_c) \\ &= \max\left(\alpha_t/(\lambda P_0), (\alpha_c/(1-\lambda)P_0)^{1/n}\right) \end{aligned} \tag{2.3}$$

where $\lambda \in [0\ 1]$ is the power allocation policy between the computing process and the communication process, α_t and α_c are system parameters. Since τ_t is a monotonical increasing function and τ_c is a monotone decreasing function, it can be shown that the optimal power allocation for $\min \tau_i$ satisfies $\tau_c = \tau_t$ as shown in Figure 2.1. In an N node wireless dis-

tributed computing scenario, the execution time of a completely parallel computing task can be represented as

$$T = \max_{i=1,2...,N}(\tau_i) \tag{2.4}$$

The delay performance in wireless distributed computing is determined by the node with the slowest rate. This is different for the communication system whose metrics are average performances, such as average delay or average throughput. As we will see in Section III, the variance of the channel condition will impact the average execution time.

2.2.4 Traffic Pattern

The power saving techniques and workload distribution have been well investigated for wireless sensor networks and ad-hoc networks [57] [62]. One of the differences compared to WDCN is the traffic pattern. Distributed computing has a higher duty cycle and higher CPU utilization, which uses more power consumption than wireless sensor networks. The ratio of transmission and idle status in the communication process for WDCN is much higher than for a sensor network. The static power has less impact on the communication power consumption compared with a sensor network. For the computing process, if we denote the CPU utilization as μ, the average computing power is

$$\begin{aligned} P_{ave} &= \mu P_{active} \\ &= \mu C V^2 f \\ &\approx a f^n. \end{aligned} \tag{2.5}$$

The traffic pattern will influence the relationship between average power consumption P_{ave} and clock frequency f by varying the coefficients n and a, which will also impact delay performance.

2.3 Workload Distribution

The communication process takes more power to mitigate the channel variation compared with AWGN channels [63], which results decreased available power for the computing process and increased execution time. Moreover, the workload cannot be changed when the channel is changing and the available computing power is varying once it is assigned to a particular node. Therefore, statistical parameters, such as the variance of the large scale path loss, should be used for workload distribution decisions.

In this section, we first define the models for the communications process and computing process. Then we introduce how to find the relationship between the channel variation and the execution time. Finally, the workload allocation algorithm is proposed.

2.3.1 Communication Process

Transmission power for the communication process can be expressed as shown in [63]

$$\begin{aligned} P_t &= (1+\zeta)P_r\frac{(4\pi)^2 d^\kappa}{G_r G_t (C/f)^2}M_l \\ &= Kd^\kappa P_r \end{aligned} \tag{2.6}$$

where $K = (1+\zeta)\frac{(4\pi)^2}{G_r G_t \lambda^2}M_l$, ζ is the ratio between static power and transmission power and represents the extra static power consumption, $P_r = E_b R_b$ is the recieved power, d is the distance of the wireless link, κ is the propagation index which is related to the enviroment, G_r and G_t are the reciever antenna gain and transmitter antenna gain respectively, f is the carrier frequency, C/f is the wavelength of the system, and M_l is the link margin. To support a reliable communication link, for un-coded M-QAM without pulse shaping, the error performance follows the standard bit error constraints for M-QAM signal as shown in [63]

$$\begin{aligned} P_e &\leq \frac{4}{b}\left(1-\frac{1}{\sqrt{2^b}}\right)Q\left(\sqrt{\frac{3E_b}{(2^b-1)N_0}}\right) \\ &\leq \frac{4}{b}\left(1-\frac{1}{\sqrt{2^b}}\right)e^{-\frac{3E_b}{(2^b-1)N_0}} \end{aligned} \tag{2.7}$$

where $Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}}e^{\frac{-u^2}{2}}du$, $b = log_2(M)$. E_b and N_0 are bit energy and noise spectrum density, respectively. By approximating the bound as an equality, we obtain

$$\frac{E_b}{N_0} = \frac{1}{3}\left(2^b-1\right)log\left(\frac{4(1-1/\sqrt{2^b})}{bP_e}\right) = r(b). \tag{2.8}$$

If the bandwidth is B, the transmission rate is $R_b = bB$, and the required receiving power is

$$P_r = E_b R_b = N_0 bBr(b). \tag{2.9}$$

Therefore, to support a reliable communication link, the transmission power must be

$$\begin{aligned} P_t &= Kd^\kappa P_r \\ &= KN_0 bBd^\kappa r(b) \end{aligned} \tag{2.10}$$

Thus the transmission power is a function of system parameter K, receiver design N_0, modulation scheme $r(b)$, bandwidth B, node location d and wireless environment κ. If we consider the channel variation, the transmission power can be expressed as

$$\begin{aligned} P_t &= KN_0 bBd^\kappa r(b)e^{xln10} \\ &= K^{'}e^{xln10} \end{aligned} \tag{2.11}$$

where $K^{'} = KN_0Bd^{\kappa}r(b)$. The variation of channel path loss $L_v = e^{xln10}$ can be modeled as a lognormal distributed random variable, or $x \sim N(0, \sigma_v)$. This is the large scale path loss variation. Extra power is required to maintain the reliable communication to a certain level, or outage probability. Here, we assume a perfect power control, that is, an exact extra transmission power of $\delta P = 1/L_v$ is added to transmission process. The delay for the communication process with the transmission workload W_t is given by

$$\tau_t = \frac{W_t}{bB} \leq \tau_0. \tag{2.12}$$

where W_t is the amount of transmission data for computing process. τ_0 is the delay requirement.

2.3.2 Computing Process

To guarantee a reliable communication process, the transmission power has to be controlled to follow the channel variation. As a result of this, the computing power is a random variable because the sum of transmission power and computing power is constrained by the total power supply of each node. From (3), (5), (11),the delay of the computing process can be shown as

$$\begin{aligned}
\tau_c &= \frac{W}{(\frac{P_c}{a})^{\frac{1}{n}}} \\
&= \frac{W}{\left(\frac{P_0(1-\lambda e^{xln10})}{a}\right)^{\frac{1}{n}}} \\
&\approx W\left(\frac{a}{P_0}\right)^{1/n}\left(1+\frac{\lambda}{n}e^{xln10}\right) \\
&\approx \beta_0\left(1+\frac{\lambda}{n}e^{xln10}\right) \\
&\approx \beta_0 + \beta_c e^{xln10}
\end{aligned} \tag{2.13}$$

where $\beta_0 = W(\frac{a}{P_0})^{1/n}$, $\beta_c = \frac{\lambda\beta_0}{n}$, W is the computing workload, and λ defined in 2.3. The delay consists of two terms, a static term β_0 and a varying term $\delta\tau_c = \beta_c e^{xln10}$. The probability density function (pdf) and cummulative density function (cdf) of $\delta\tau_c$ can be shown as

$$\begin{aligned}
f(\delta\tau_c) &= \frac{1}{\sqrt{2\pi}\sigma_v ln10\beta_c} e^{-\frac{(log_{10}(\tau_c/\beta_c))^2}{2\sigma_v^2}} \\
F(\delta\tau_c) &= \int_0^{\tau_c} f(x)dx = Q(log_{10}(\tau_c/\beta_c))
\end{aligned} \tag{2.14}$$

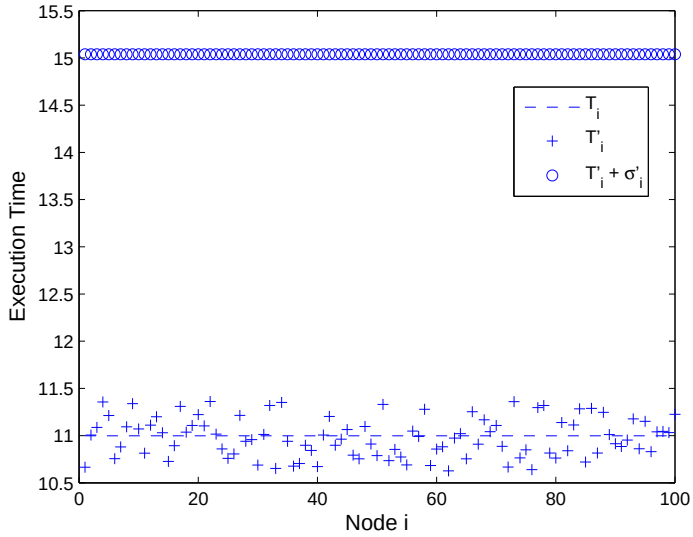

Figure 2.2: The two-step workload allocation considering robustness.$N = 100$,$W = 1000$,$n = 2$,$\lambda = 0.2$,$(P_0/a) = 1$.The first step makes T_i the same for all the nodes; The second step makes $T_i + \sigma_{T_i}$ the same for all the nodes.

The delay of computing process is varying due to the term of $\delta\tau_c$. This will potentially increase the variance of execution time when the channel variances σ_v are different among links.

2.3.3 Prediction of the Average Execution Time

The execution delay of the computing process for node i can be expressed as

$$\tau_i = \beta_{0,i} + \beta_{c,i} e^{xln10} \tag{2.15}$$

where $\beta_{0,i} = W_i(\frac{a}{P_0})^{1/n}$, $\beta_{c,i} = \frac{\lambda\beta_{0,i}}{n}$, W_i is the computing workload for node i, and β_0 is the system constant defined in 2.13. The mean and variance of τ_i can be shown to be

$$\begin{aligned} T_i &= \beta_{0,i} + \beta_{c,i} e^{\frac{\sigma_v^2 ln^2 10}{2}}; \\ \sigma_{T_i}^2 &= \beta_{c,i}^2 e^{(\sigma_v ln10)^2}(e^{(\sigma_v ln10)^2} - 1). \end{aligned} \tag{2.16}$$

Both the mean and variance of execution time are functions of the channel variance σ_v, which may vary from link to link. We assume σ_v is a i.i.d random variable with a uniform distribution to represent the channel heterogeneity.

2.3.4 Workload Distribution Algorithm

The workload allocation can be used to mitigate the impact of channel variation on the delay performance and energy efficiency by allocating more workload to the nodes with better channels. The optimal workload distribution in term of minimum execution time is the solution to the following programming problem

$$\begin{aligned} &\textit{minimize} && (\max(\tau_i)), i = 1, 2, 3...N \\ &\textit{subject to} && \sum_{i=1}^{N} W_i = W \\ & && P_{c,i} + P_{t,i} = P_0 \end{aligned} \tag{2.17}$$

where $\tau_i = \max(\tau_{c,i}, \tau_{t,i}) = \max\left(\frac{W_i}{(P_{c,i}/a)^{1/n}}, \frac{W_i}{bB}\right)$. P_0 is the power supply.

A heuristic solution which strives to have all nodes finish around the same time is to distribute the workload W_i proportional to the term $A_i(t) = (1 + \frac{\lambda}{n} e^{\frac{\sigma_v^2 ln^2(10)}{2}})^{-1}$ and let $T_0 = T_1 = T_2 = ... = T_N$. Then the workload W_i is adjusted by δW_i considering σ_v to make

$$T_1^{'} + \sigma_{T_1}^{'} = T_2^{'} + \sigma_{T_2}^{'} = ... = T_N^{'} + \sigma_{T_N}^{'} \tag{2.18}$$

The results of similar idea in a wired system showed that the adjusted workload δW_i can be calculated by solving linear equations [55]. One result of this workload distribution is shown in Figure 2.2 which shows the two-step workload allocation, $T_i + \sigma_{T_i}$ for each node satisfies the Equation (2.18).

Equation (2.18) improves the delay and robustness performance without considering energy efficiency. From (2.13)-(2.16), the nodes with longer T_i will also have larger σ_{T_i} and drain more transmission power, which leads to less available power for the computing process. In other words, it is not energy efficient for some of the nodes with bad channel conditions to join WDCNs. We can choose the nodes with better channel conditions and pay a smaller price for reliable communications, which will improve the energy efficiency for the WDCNs. The possible solution is to choose the nodes with the smallest T_i satisfying the delay constraint until all the workload can be accommodated. The other possible solution is to form a WDCN first with all the nodes, then eliminate the nodes with higher channel variations, and distribute their workload to others. Compared with the first solution, the second one has lower overhead complexity.

The normalized average energy consumption per bit for the proposed algorithm can be

approximated with Jensen's inequality as

$$\begin{aligned} E &= E\left\{\frac{\sum_{i=1}^{N} P_i\tau_i}{W}\right\} \\ &\approx E\left\{\frac{P_0\sum_{i=1}^{N}\tau_i}{W}\right\} \\ &\geq \frac{P_0\sum_{i=1}^{N} E\{\tau_i\}}{W} \end{aligned} \tag{2.19}$$

where N is the number of nodes, P_0 is the total power for each node, and W is the total workload to be accomplished. The actual power consumption P_i can be approximated by P_0 considering $\tau_c = \tau_t$. Therefore, the minimum amount of energy efficiency is equivalent to the minimum of $\sum T_i$. To improve the energy efficiency by removing the N_{th} node from the network requires

$$\begin{aligned} \sum_{i=1}^{N}(T_i) &\geq \sum_{i=1}^{N-1}(T_i^{'}) \\ \Leftrightarrow \frac{1}{A_N} &\geq \sum_{i=1}^{N-1}\frac{1}{\sum_{i=1}^{N-1} A_i} \end{aligned} \tag{2.20}$$

where T_i' is the new average execution time for node i after removing node N with the largest σ_v from the network. Since the available power for the computing process for each node has the relation of $0 < A_N < A_{N-1} < A_{N-2} < ... < A_1$, this inequality is always satisfied.

The workload for Node N, W_N, will be allocated among $N-1$ nodes according to their average channel condition once the Nth node is removed from the network. The new maximum execution time is

$$\begin{aligned} \max_{i=1...N-1}(T_i^{'}) &= \max_{i=1...N}(T_i) + \delta T \\ &= \max_{i=1...N}(T_i) + \frac{W_N}{\sum_{i=1}^{N-1} A_i} \\ &> \max_{i=1...N}(T_i) \end{aligned} \tag{2.21}$$

Then the $\max(T_i)$ becomes even larger after re-distributing the workload considering σ_{T_i}. Therefore removing nodes with larger channel variation from the wireless distributed computing networks will degrade the mean network execution time, or $\max\{T_i\}$. However, it will reduce the variance of the execution time.

The proposed power and workload allocation algorithm is as follows:
1). $\lambda_i^* = minimize(max(\tau_{t,i}, \tau_{c,i})$; $k = 1$
2). $W_i = \frac{WA_i}{\sum_{i=1}^{N} A_i}$;

Table 2.1: System Parameters for Workload Allocation Methods

P_0	6.7 (W)	f	900 MHz
κ	2.8	B	15 kHz
G_t, G_r	3 (dB)	σ_v	[0 1]
d	100 (m)	P_e	10^{-6}
$Mod.$	16 QAM	Ml	3 dB
T_0	300K	a	$2.5e^{-14}$

3). $W_i^{'} = W_i^{'} + \delta_{W_i}$,$\delta_{W_i}$ is calculated with (2.18) ;
4). Remove the node with the largest channel variation;
5). If $\sum_{N-k-1}(T_i^{'}) < \sum_{N-k}(T_i)$, then $k = k + 1$, return to step 2). Otherwise, end the algorithm.

The proposed approach balances the delay performance, energy efficiency and network robustness. It is noted that some extra power may need to collect the information of channel condition from each node, which is a protocol design problem and beyond the scope of this paper.

2.4 Simulation

In this section, we simulate a wireless distributed computing network, and compare the power efficiency and delay performance for three different workload distribution algorithms. The system parameters are listed in Table (2.1). The total computing workload of 240M units, is allocated to $N = 200$ nodes. The nodes have been ordered according to channel variance in an ascending order, $\sigma_{v,1} < \sigma_{v,2} < ... < \sigma_{v,N}$. The baseline workload distribution approach allocates the workload evenly to the nodes regardless of the channel condition. The second workload distribution allocates the workload according to the average channel condition. This corresponds to the first step of proposed workload allocation. The third approach allocates the workload considering both average execution time and its variance, which corresponds to the two-step workload allocation algorithm.

Figure 2.3 shows the impact of the network size and channel variance on the average execution time. When the network size and channel variance is small, the average execution time can be reduced by increasing the number of nodes. As more nodes with larger channel variance join in the network, the dynamic term of Equation (2.13) and (2.15) for those nodes dominate the delay performance and increase the average execution time. Figure 2.4 shows the variances of average execution time among three approaches. Removal of the nodes with larger channel variance will decrease the variance of the average execution time. The advantage of the proposed approach diminishes for smaller network sizes with smaller channel variations. Average execution time and its variance with two-step algorithm are consistently lower than

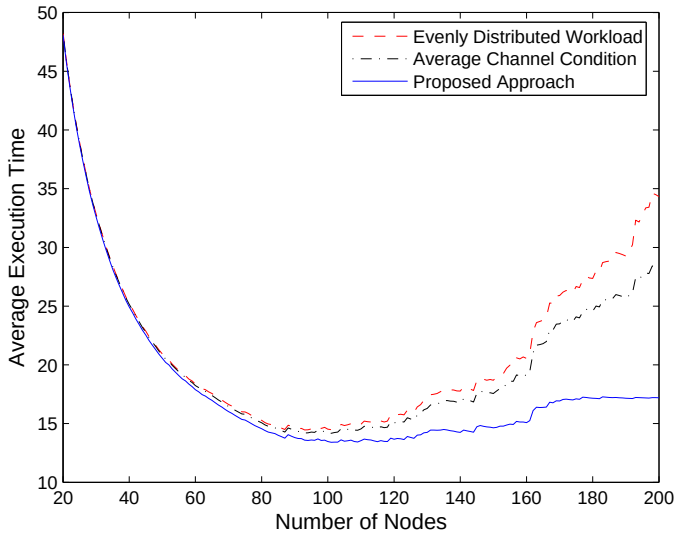

Figure 2.3: The impact of the network size on the average execution time. The unit is generalized. The limited power supply leads to the increase at the right end.

the other algorithms.

The expansion of network size will increase the energy consumption shown in Figures 2.5 and 2.6. Figure 2.5 shows that energy saving with the proposed approach is 10.2% by reducing the network size from 200 to 100 in this case. When the nodes with larger channel variance are removed from the network, the advantage of proposed approach in terms of normalized energy efficiency also diminishes compared with evenly distributed workload shown in Figure 2.6. The proposed approach can improve both the robustness performance and the energy efficiency as shown in Figure 2.4 and Figure 2.5.

2.5 Summary

Workload distribution considering the channel variance can help APC and DVS improve energy efficiency as well as robustness in terms of delay performance by reducing the workload for the channel with larger variance. The removal of the nodes with the worst channels from the network can potentially increase energy efficiency even more. In this paper, we examined a simple system without considering the random location of nodes, propagation environments, and interference. These parameters as well as the buffer between the communication process and the computing process need to be considered in the future work.

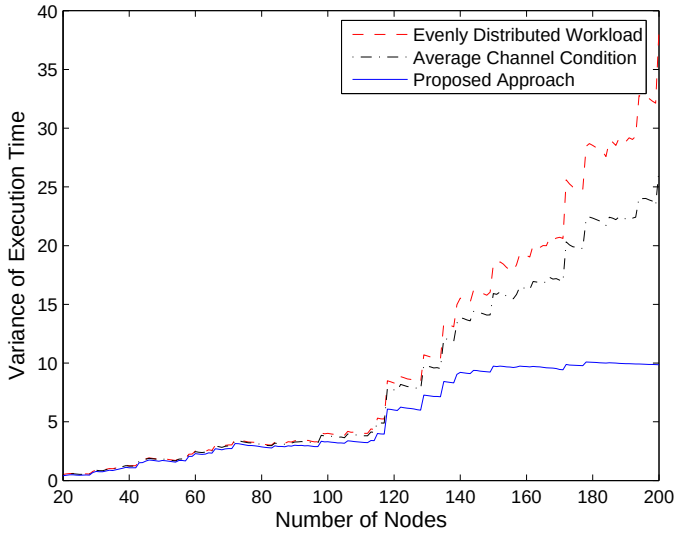

Figure 2.4: The variance of the average execution time with different workload distribution approaches.The unit is generalized.

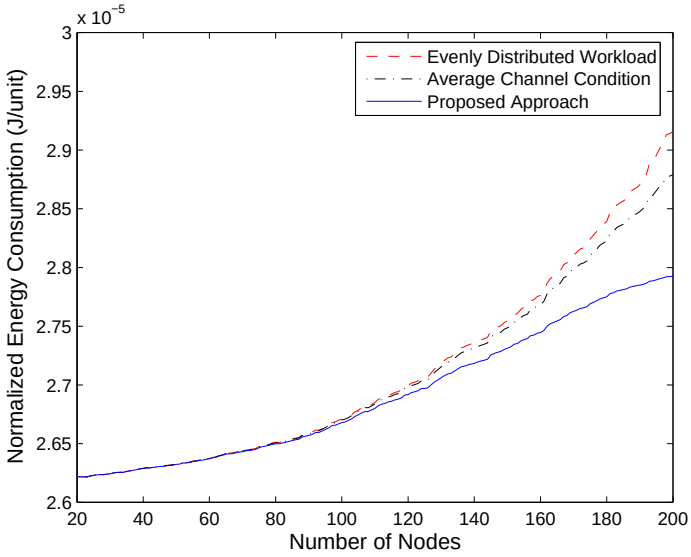

Figure 2.5: The impact of the network size on the normalized average energy consumption.

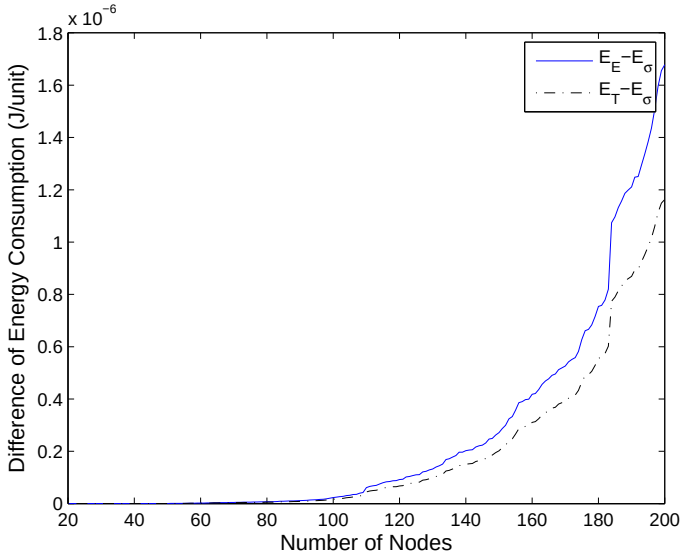

Figure 2.6: The energy gain of proposed workload distributions over the other approaches.E_E, E_T, and E_σ are the normalized energy consumption for evenly distributed workload, the second approach, and the proposed approach respectively.

Chapter 3

Cross-layer Resource Allocation in WDCNs

This chapter presents a resource allocation scheme for the nodes in a wireless distributed computing network (WDCN) during service provision phase. The chapter first analyzes the resource allocation requirements for WDCNs for service provision. A solution for computation rate, communication power and computing workload allocation is then introduced in detail. By maximizing the network computing capability and minimizing the communication power consumption and the computing execution delay, the majority of the available power can be used to support the computing process. The wireless channel conditions are considered to improve both energy and power efficiency. It also presents the impact of components, such as buffer length, and CPU type, on the system performance.

3.1 Introduction

Wireless distributed computing is an emerging area combining both wireless communications and distributed computing technologies. It is the means by which processing is shared among wireless devices in order to improve the performance and exchange needed resources. The motivation behind this new concept is the increasing use of wireless devices, the innovations of new applications in wireless industry, and increasing demands to support complex applications with simple wireless devices.

The evolution of computing and communication technologies makes distributed computing possible with wireless devices. The wireless industry has witnessed a dramatic increase of computing processing capability of wireless platforms over the last ten years. The computing power of current mobile devices is comparable to desktops available ten years ago [1]. Wireless broadband technologies provide a high speed and energy efficient channel for data distribution and collection. In addition, innovations in the wireless industry create a

lot of opportunities and demands for future wireless networks. Currently, wireless personal access networks (WPAN), wireless sensor networks (WSN), and machine-to-machine (M2M) applications in cellular systems have been introduced in order to support new applications by attaching small nodes with larger nodes [2, 3]. As the computing capability of wireless devices increases and wireless broadband technologies evolve, more new applications will be emerging.

Lessons learned from wired distributed computing also indicate that wireless distributed computing has the potential to improve both energy efficiency and delay performance and will be the trend in the future. The nonlinear relationship between clock frequency and power consumption for computing units causes a thermal wall which makes an increase gap between the increase of computing capability from what Moore's Law predicts. Multi-core CPUs and parallel computing have the similar effects of breaking this thermal wall by assigning a task to a group of small processing units instead of one high-performance unit. However, considering the uncertainty conditions of wireless channels, both communication components and computing units in WDCNs need to be designed and tuned together carefully in order to achieve both energy efficiency and computing performance improvement. In other words, newly introduced innovations in the wireless industry involve cooperation between computing functions and communication functions and this has set the stage for wireless distributed computing.

WDCNs have unique problems compared to traditional distributed computing and wireless communications, among which power management, latency reduction, and software architecture are important issues. For each node in WDCNs, power consumption consists of two parts, computing power consumption and communication power consumption. If either the computing process or communication process dominates the power consumption, the traditional power management for distributed computing or communication systems is good enough to design an efficient power management solution [70]. However recent test of video conferencing on a mobile station show that the power consumptions of the computing process and communication process can be at the same level for WDCNs [5]. The way that nodes should be scheduled, if at all, depends on the trade-offs in communication power consumption, computing power consumption, and latency. Thus, a cross-layer energy efficient protocol needs to be designed that considers these two power consumptions at the same time. The communication rate should be fast enough to support the computing rate. But the communication power consumption is the price paid for distributing the computation and should be reduced to a minimum. This means the reduction of communication power consumption is constrained by the computing rate requirement. Therefore, a distributed cross-layer power-rate allocation scheme is an essential component of power management in WDCNs.

The execution latency of a task in WDCNs is determined by the slowest node. The performance indicator of latency is the maximum latency among WDCN nodes instead of the average latency among nodes. In addition, the latency reduction in WDCNs is especially challenging for software defined radios [6]. Unpredictable latency may destroy the timing

among components and compromise the protocol design in WDCNs. The third problem is to create an effective software architecture, which includes how to define resource and service discovery, as well as how to design fault-tolerant and security mechanisms. Although some lessons can be learned from traditional wired distributed computing, new approaches need to be created to handle the uncertain and unreliable wireless channel.

Significant research has gone into energy efficiency and delay reduction in both wireless communication and distributed computing areas. Adaptive protocols, such as adaptive modulation, intelligent MAC, and energy-aware routing protocols, help wireless sensor network improve energy efficiency [11]. Another widely used approach to save energy is to turn off radio when no transmission is needed. The opportunistic transmission combining power and rate adaptation saves energy of communications for fading channels. Dynamic programming and Markov models can be used for this purpose by adaptively changing the status among active, idle, and sleep modes [13, 14, 15]. However, varying radio status will increase the complexity of protocols. In addition, there is also some progress concerning the impact of network topology on energy efficiency and delay reduction, especially multi-hop wireless networks [16]. Researchers have realized that the tradeoff between delay and capacity is a fundamental issue for wireless networks and currently an active research topic [17, 18, 19]. Recently, stochastic programming methods have been introduced to solve resource allocation in wireless networks [69, 71, 72], which has a better approximation of wireless channels compared with current widely used optimization methods in wireless networking. In distributed computing, server clusters exploit dynamic voltage scaling (DVS) combined with CPU scheduling for better energy efficiency [20, 21, 22, 23]. Spatial and temporary heterogeneities are considered in workload distribution to improve energy efficiency and robustness [24].

To the best of our knowledge, there is no literature discussing resource allocation for computation oriented WDCNs in a systematic way. The goal of this chapter is to provide an analysis framework for computation oriented WDCNs in terms of utility, such as computing rate and workload, that can be supported as a function of system parameters, especially power consumption. The impacts of limited buffer length, CPU type, and workload are also analyzed as examples. A dedicated algorithm for computation oriented WDCNs is then proposed considering channel heterogeneities and power consumption models. Simulation results show that the proposed approach supports energy-efficient WDCNs.

The chapter is organized as follows. Section II introduces the basic assumptions and the system model. Section III discusses how to formulate the problem and a relaxed form of the problem is solved using stochastic programming. Section IV presents the proposed algorithm and solutions. Section V presents some computer simulation results. Section VI concludes the chapter.

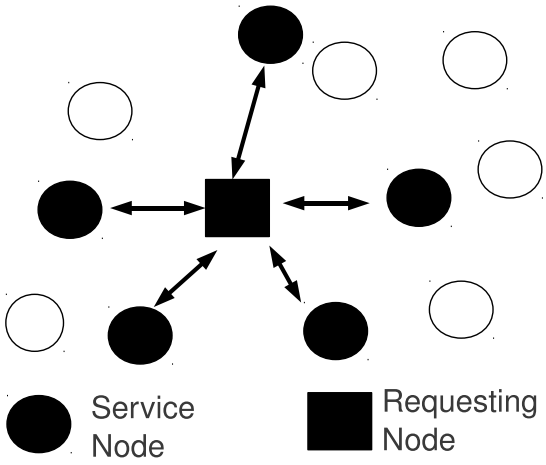

Figure 3.1: A topology example. A requesting node makes a request of the hardware/software resource to service-nodes through a wireless link. Service-nodes respond with different service models. Each service-node i corresponds to a link i

3.2 System Model

An example WDCN topology is shown in Figure 3.1. A service requesting node, Node 0, may not have sufficient computing resources, such as CPU and memory, to finish a task within a required time. It initiates a computation task and distributes the raw data to the service nodes, $S_i, i \in \boldsymbol{S}$. This requesting node is also the data fusion point, where the service nodes will send back the processed data. There is one wireless link i between a service node i and the requesting Node 0. The number of service nodes is N. The goal of these links is to support computing tasks initialized by service request node, Node 0, instead of providing communications between them. Service nodes do not compete with each other for access to Node 0. Neither, node 0 does not need to access every service node.

The definition of fairness in WDCN is different from that in wireless communication networks. Service nodes can consume their resource in an unfair and greedy manner, which makes heterogeneity an important factor to consider. Depending on the location of wireless nodes, statistical characteristics of wireless channels and service nodes may vary from link to link. The distance of wireless links (d_i), propagation index (α_i), and the variance of log-norm distribution (σ_i) are used in radio link budget in an average sense for radio network coverage planning. The variance of log-norm distribution represents large scale channel fading, which is the randomness effect discussed in this chapter. The combination of these three factors represents the channel heterogeneity, $\boldsymbol{H} = \{d_i, \alpha_i, \sigma_i; \forall i \in \boldsymbol{S}\}$. $\boldsymbol{H}$ is assumed to be time invariant for each link but varying from link to link.

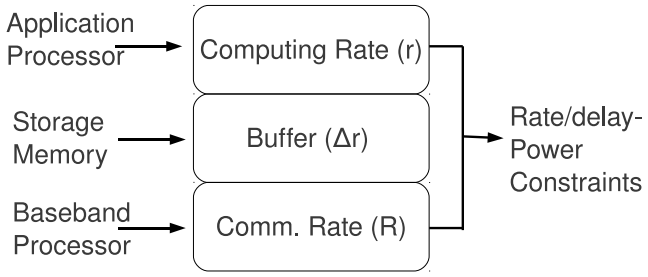

Figure 3.2: Node model for service nodes. The communication process and computing process are coupled with buffer, QoS, and power supply. Rate and power refer to computing rate and communication power, respectively.

In addition to channel heterogeneity, wireless devices may also have some heterogeneity for the RF chains, buffer size, and computing capabilities which need investigation for resource allocation in WDCN. The major goal of this chapter is to integrate channel heterogeneity and device heterogeneity into the problem formulation for resource allocation and figure out the most significant factors for the performance and energy efficiency of WDCN. A node model is shown in 3.2. There are at least three components in a service node, application processor (AP), storage memory(SM), and baseband processor (BP). A computing process runs in AP and a communication process runs in BP. These two processors have capabilities to change their processing rates with DVS and AMC, respectively. SM provides additional storage space in addition to the two buffers in BP and AP, which is normally much larger than the queue size in BP and AP. The communication rate has to be sufficient to support the computing process without overflowing the two buffers in AP and BP. In addition, SM can accommodate an extra rate difference between computing rate and communication rate between these two processes as shown in Figure 3.2. For example, when the average computing rate is higher than the average communication rate, the output data from AP can be stored in SM first and then transferred to BP. The maximum average rate difference allowed between BP and AP, Δr_i, can be derived with the allocated workload size and required delay deadline. The allocated workload size is required to be smaller than the SM size in order to avoid overflowing, which is feasible for WDCN since Node 0 has the control of workload allocation. This is not feasible for wireless communication networks when the workload size is unknown, or infinite, and task completion time cannot be controlled or predicted with workload allocation.

In this section, we will define system parameters and give the system assumptions for computation oriented WDCNs. Variable definitions are listed in Table 3.1.

Table 3.1: System Parameters For Service Provision

P_i	Total used power for node i
P_i, avg	Power supply for node i
$P_{i,comp}$	Computing power consumption
$P_{i,comm}$	Communication power consumption
a, b, θ	Coeff. for computing power-rate equation 3.2
f	Clock frequency
r_i	Computing rate for node i
R_i	Communication rate for node i
G_i	Channel gain for link i
η_p	Power efficiency
η_E	Energy efficiency
α_i	Propagation index for link i
d_i	Distance for link i
x_i	Log-normal fading for link i
σ_i	Variance of log-normal fading
k_i	Weight for commun. power consumption
τ	Time scale for (power, rate) allocation
t	Time scale for workload allocation
$P_{i,max}$	Max. transmission power for node i
W_i	Workload allocated to node i
W_0	Total workload
$T_{i,exe}$	Execution time for node i

3.2.1 Communication and Computation

The computing power consumption $P_{i,comp}(\tau)$ and communication power consumption $P_{i,comm}(\tau)$ for node i are coupled with the total power consumption $P_i(\tau)$ as

$$P_i(\tau) = P_{i,comp}(\tau) + P_{i,comm}(\tau), \quad \forall i \in \boldsymbol{S} \tag{3.1}$$

There are two power-rate relations for each node i. The power consumption of the computing process, especially CPUs, is a polynomial function of the clock frequency ($f(\tau)$) and is given by

$$P_{i,comp}(\tau) = af(\tau)^n + b, \quad \forall i \in \boldsymbol{S} \tag{3.2}$$

This P-f relation for the computing process can be approximated by a curve fitting data from a CPU data sheet to find parameters a, b, and n. For example, ARM10 with a power consumption of 250 mW at a maximum clock frequency of 564 MHz has $a = 0.35\ mW/MHz$ when $n = 1$ and $b = 50\ mW$ [64]. If the CPU has dynamic frequency scaling capability instead of dynamic voltage scaling (DVS), the computing rate r_i will have a linear relation with the computing power consumption. In addition, the clock frequency has a linear relation with the computing rate $r_i(t)$, that enters into the buffer as

$$r_i(\tau) = \theta f_i(\tau), \quad \forall i \in \boldsymbol{S} \tag{3.3}$$

where θ represents the mapping relation between clock frequency and computing rate, which is determined by the application and hardware platform [51]. The communication rate (R_i) as a function of the transmission power ($P_{i,comm}$) can be represented as

$$R_i(\tau) = log(1 + G_i(\tau)P_{i,comm}(\tau)/N_r), \quad \forall i \in \boldsymbol{S} \tag{3.4}$$

where $G_i(\tau)$ is the channel gain which is a random variable, and N_r is the received noise power. For each service node with a distance d_i to the requesting node and a log-normal fading with a variance of σ_i, the channel gain of this link can be represented as

$$G_i(\tau) = \frac{10^{x_i(\tau)} g_i(\tau)}{d_i^{\alpha_i}}, \quad \forall i \in \boldsymbol{S} \tag{3.5}$$

where α_i is the propagation index for node i, typically with a value from 2 to 4. $x_i \sim N(0, \sigma_i^2)$, and g_i follows a Rayleigh distribution with a variance of $\sigma_g, \forall i \in \boldsymbol{S}$. $H \triangleq \{d_i, \alpha_i, \sigma_i\}$ represent the channel heterogeneity and vary from link to link. We exclude Rayleigh fading from the channel heterogeneity discussion because diversity improvement methods and transmitter/receiver technologies, such as FEC coding and MRC, are better solutions for Rayleigh fading than power control.

3.2.2 Metrics

Power efficiency is defined as the portion of computing power out of the total used power. That is

$$\eta_p \triangleq \frac{\overline{P_{comp}}}{\overline{P_{comp}} + \overline{P_{comm}}} \tag{3.6}$$

where

$$\begin{aligned} \overline{P_{comp}} &\triangleq \limsup_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{i=1}^{N} E\{P_{i,comp}(\tau)\} \\ \overline{P_{comm}} &\triangleq \limsup_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{i=1}^{N} E\{P_{i,comm}(\tau)\} \end{aligned} \tag{3.7}$$

where $E\{P_{i,proc}(\tau)\}$, $proc \in \{comp, comm\}$, is the expected power consumption w.r.t randomness for node i. The goal of WDCNs is to fulfill a computing task with less execution time by distributing it among service nodes. Communication power consumption is the price to pay for the message exchange, as well as data distribution and collection. Ideally, power is invested in the computing process as much as possible. A higher power efficiency indicates a resource utilization that supports a particular computing capability with a relative smaller communication power consumption.

Energy efficiency represents the number of bits that each unit of energy can process, and is given by

$$\eta_E \triangleq \frac{\overline{r_{net}}}{\overline{P_{used}}} \tag{3.8}$$

where

$$\begin{aligned} \overline{P_{used}} &\triangleq \overline{P_{comp}} + \overline{P_{comm}} \\ \overline{r_{net}} &\triangleq \limsup_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{i=1}^{N} E\{r_i(\tau)\} \end{aligned} \tag{3.9}$$

The sum of the computing rates is defined as the average network computing rate, which identifies the computing capability of $\boldsymbol{S}$. A larger value of η_E, or a higher energy efficiency, indicates a longer battery lifetime for a particular computing task. The available network power is a constant defined as

$$P_{net} \triangleq \sum_{i=1}^{N} P_{i,avg} \quad \forall i \in \boldsymbol{S} \tag{3.10}$$

where $P_{i,avg}$ is the power supply for each node i.

In addition, the percentages of power allocation are defined as

$$\gamma_{used} \triangleq \frac{\overline{P_{used}}}{P_{net}},\ \gamma_{comp} \triangleq \frac{\overline{P_{comp}}}{P_{net}},\ \gamma_{comm} \triangleq \frac{\overline{P_{comm}}}{P_{net}} \tag{3.11}$$

where γ_{used}, γ_{comp}, and γ_{comm} indicate the value of the total utilized power, computing power, and communication power as the percentage of the total available power respectively. Combining (3.6), (3.8), (3.9), (3.11), and (3.10), we have

$$\eta_p = \frac{\gamma_{comp}}{\gamma_{used}},\ \eta_E = \frac{\overline{r_{net}}}{\gamma_{used} P_{net}} \tag{3.12}$$

which indicates increasing r_{net} and decreasing γ_{used} will improve both power efficiency and energy efficiency. It can be achieved by minimizing communication power consumption $\overline{P_{comm}}$ and maximizing network computing rate since $\overline{P_{comp}}$ is a non-decreasing function of computing rate r_{net}. This is a power and rate allocation problem.

If Node 0 distributes a workload of a constant size, W_0, to service nodes $\boldsymbol{S}$ for one session. Each service node, Node i, can processes its allocated workload $W_i(t)$ within a delay of $T_{i,exe}(t)$.

$$\begin{aligned} W_i(t) &= \sum_{\tau=0}^{T_{i,exe}(t)-1} r_i(\tau) = \overline{r_i}(T_{i,exe}(t)) T_{i,exe}(t), \\ s.t.\, W_0 &= \sum_{i=1}^{N} W_i(t),\ \forall i \in \boldsymbol{S} \end{aligned} \tag{3.13}$$

where the time interval for the power-rate adaptation scale (τ) is 1. $T_{i,exe}(t)$ is the execution time of node i for one workload allocation. The average computing rate for node i to finish one session with a workload of $W_i(t)$

$$\bar{r}_i(t) \triangleq \frac{1}{T_{i,exe}(t)} \sum_{\tau=0}^{T_{i,exe}(t)-1} r_i(\tau),\ \forall i \in \boldsymbol{S} \tag{3.14}$$

The expected network execution latency for workload W_0 is

$$\overline{T_{exe}} \triangleq \limsup_{t\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} T_{exe}(t) \tag{3.15}$$

where

$$T_{exe}(t) \triangleq E\{\max_{i\in\boldsymbol{S}}\{T_{i,exe}(t)\}\} \tag{3.16}$$

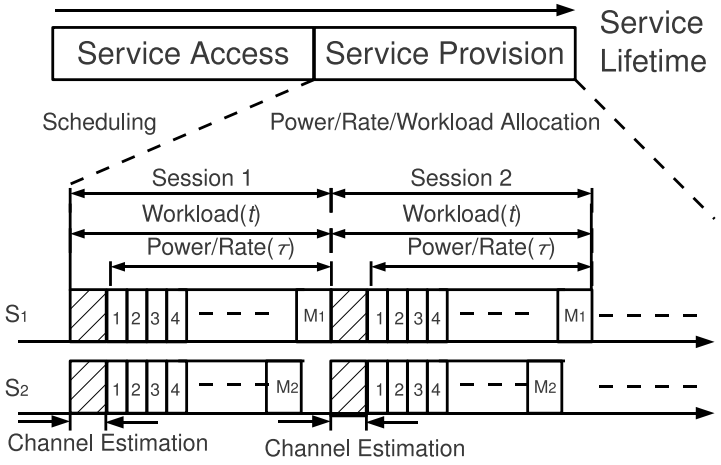

Figure 3.3: Service model for WDCN. Service access determines nodes access. Resource allocation in service provision includes workload and communication power and computing rate allocation. Workload is distributed according to estimated average computing rate. Power/Rate allocation changes according to channel gain.

which is the computing performance metric. The workload is allocated at a session time scale, t with a time interval $T_{exe}(t)$. Both the average and variance of execution latency should be minimized as much as possible in order to implement a faster and stable distributed computing, which is the goal of workload allocation. For example, from (3.13) and (3.16), $\overline{T_{exe}}$ can be optimized with wise choices of $W_i(t)$ if $\bar{r}_i(t)$ is avaliable, which may be approximated with the rate estimation at t and rate information in the previous frame, $\bar{r}_i(t-1)$.

3.2.3 System Assumption

The service lifetime for WDCNs can be divided into two phases: service access and service provision as shown in Figure 3.3. Service access determines how to formulate WDCN, such as configuring network topology and scheduling nodes. For example, a service requesting node decides whether a computing task should be processed by itself or be distributed to service nodes during this phase [73]. Service provision is traffic oriented. During each traffic session t, service requesting node distributes computing workload W_0 to service nodes, $\boldsymbol{S}$, based on their average computing rate ($\overline{r}_i$), that service nodes send to it. This average computing rate for each node is generated from previous traffic sessions and updated with the estimation at the beginning of each session. After workload allocation, service nodes will adaptively change their computing rate and communication power consumption, for each slot (τ), in order to process the assigned workload.

Since the goal of WDCNs is not to provide communication service to each node in $\boldsymbol{S}$, the concurrent lifetimes of batteries are unnecessary. The lifetime of the battery can differ from node to node. Any node with a dead battery is excluded from the WDCN network. However, proportional fairness is still needed for network topology stability and to avoid too much churn. In addition, a larger network size is not necessarily more power efficient for computation oriented WDCNs due to channel conditions for some nodes may require excessive communication power. Computing robustness may be degraded by spatial heterogeneity among nodes. A WDCN formed within one office area is probably more power efficient and robust than the one formed within a building or a city area. Thus we only consider a limited network size of less than 100 nodes for a computation oriented network.

In this chapter, we consider a single hop interference-free WDCN. Concurrent battery lifetime is not required. The traffic pattern follows a uniform distribution, which means W_0 is fixed for each session. Channel fading includes log-normal and Rayleigh fading with spatial heterogeneous links. The channel heterogeneity ($\boldsymbol{H}$) is time invariant. Each service node has two power consumption units, BP and AP, competing with each other for power supply and cooperating with each other for data processing. Under these assumptions, we will define the resource allocation problem in the following section. The following questions will be answered:

- How do we combine (power, rate) allocation problem and workload allocation?
- What are the impacts of device heterogeneity and channel heterogeneity on power/energy efficiency and execution time?

3.3 Problem Formulation

In this section, we investigate how to consider heterogeneous channels and devices for the resource allocation problem in the service provision phase.

3.3.1 Objective Function

The objective function for resource allocation can be formulated as

$$\begin{aligned} Maximize \quad & \overline{r} - \overline{P} - \overline{T_{exe}} \\ s.t. \quad & Constraints \end{aligned} \tag{3.17}$$

The utility term of computing rate $\bar{r}$ is defined as

$$\overline{r} \triangleq \limsup_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{i=1}^{N} E\{log(r_i(\tau))\} \tag{3.18}$$

The penalty term of communication power consumption $\bar{P}$ is defined as

$$\overline{P} \triangleq \limsup_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{i=1}^{N} E\{k_i P_{i,comm}(\tau)\}$$

Both $r_i(\tau)$ and $P_{i,comm}(\tau)$ are normalized to the range of [0 1] in order to get rid of the impacts of different units. The maximum value used to normalize $r_i(\tau)$ only impacts the relative value of objective function as indicated in (3.18). If different weights other than [1, 1, 1] are assigned to these three terms, other solutions in Pareto front can be achieved.

Here, we only consider three cases of the weight for communication power consumption

$$k_i \triangleq \begin{cases} d_i^{\alpha_i} 10^{\sigma_i} & Channel\ Consideration \\ 1 & No\ Channel\ Consideration \\ 0 & No\ Power\ Consideration \end{cases} \tag{3.19}$$

Case 1 gives preference to the nodes with shorter distance and smaller channel variation in order to consider the influence of the heterogeneities in communication channels on the overall resource allocation. Case 2 considers the communication power consumption without differentiating the channel heterogeneity. Case 3 does not consider the power consumption at all.

From (3.4) and (3.5), extra transmission power is needed to compensate for the channel randomness to maintain a reliable communication rate. Assuming d_i and α_i are the same for all the nodes in $\boldsymbol{S}$, the average power consumption of a transmission rate is a non-decreasing function of σ_i for a fixed communication rate. For example, given a fixed communication rate R_i, the average transmission power with perfect power control for log normal shadowing is

$$\begin{aligned} \overline{P_i} &= \int_0^\infty \frac{d^{\alpha_i}(2^{R_i}-1)N}{e^{xln10}} f(x)dx \\ &= d^{\alpha_i}(2^{R_i}-1)Ne^{\frac{(\sigma_i ln10)^2}{2}} \end{aligned} \tag{3.20}$$

The definition of k_i in case 1 represents this effect: more communication power consumption is needed for poor channels. It forces the nodes with poor channels to reduce their computing rate.

3.3.2 Constraints

AP and BP cooperate to support computing task and compete for power supply. The sum of the average communication power consumption and the average computing power consumption is smaller than the average total power supply $P_{i,avg}$ for each node as represented by

$$E\{P_{i,comp}(\tau)\} + E\{P_{i,comm}(\tau)\} \leq P_{i,avg} \tag{3.21}$$

The instantaneous power constraint is also important to power efficiency and energy efficiency since the peak to average power ratio (PAPR) influences the power efficiency of the power amplifier in the RF front-end. To analyze this effect, an instantaneous communication power constraint is introduced as

$$P_{i,comm}(\tau) \leq P_{i,\max} \tag{3.22}$$

where $P_{i,comm}$ is the instantaneous communication power and $P_{i,\max}$ is the maximum instantaneous power constraint for the RF front-end for each node.

Rate choices for AP and BP insure that their buffers inside those two processors do not overflow during a session interval. In addition, we assume the size of SM is large enough compared with the workload within a session interval, W_0. SM can store the data and transmit it in next session, instead of using admission control or dropping data, when computing rate is larger than the communication rate. An average rate difference Δr_i is introduced to represent the capability of SM as

$$E\{r_i(\tau)\} \leq E\{R_i(\tau)\} + \Delta r_i,\ i \in \boldsymbol{S} \tag{3.23}$$

where $\overline{r}_i = E\{r_i(\tau)\}$ and $\overline{R}_i = E\{R_i(\tau)\}$ are the average computing rate and communication rate, respectively. Since total computing workload W_0 is fixed for each session, and $r_i(\tau)$ can be regulated in this problem, the rate stability can be satisfied with (3.23). A larger buffer size can accommodate a larger rate difference between the communication process and computing process. Different values of Δr_i represent SM heterogeneity of devices.

The constraint of the workload allocation among service nodes can be represented as

$$\sum_{i=1}^{N} W_i(t) = W_0,\ i \in \boldsymbol{S} \tag{3.24}$$

where W_i is the workload allocation among service nodes. W_0 is the total computing workload with normalized units, which could be bits or other processing units.

Abstractions are introduced to generalize real world resource constraints in this sub-section. More detailed system constraints can be formulated if hardware measurements and specifications of devices are available.

3.3.3 Resource Allocation Problem

Resource allocation for computation oriented WDCNs must balance the computing rate, communication power consumption and workload for each node in order to maximize power and energy efficiencies, reduce the network delay performance, and improve the computing robustness with channel and device heterogeneity. This resource allocation problem can be

formulated as the following programming problem.

$$\begin{aligned} Maximize \quad & \overline{r} - \overline{P} - \overline{T_{exe}} \\ Constraints: \quad & (3.21), (3.22), (3.23), (3.24) \\ Variables: \quad & r_i(\tau) \geq 0, P_{i,comm}(\tau) \geq 0, W_i(t) \geq 0. \end{aligned} \tag{3.25}$$

As shown in Figure 3.3, workload allocation and power-rate allocation have different time scales. The workload allocation updates over a larger time interval t than that of power-rate allocation, τ. In addition, the workload can be allocated only once for each session. Since the objective functions are monotonically non-decreasing and the update rates for power/rate and workload are different, it can be shown that both separability and monotonicity are satisfied. The primary problem may be decomposed into two sub-problems [66]. The solution of power/rate allocation (sub-problem 1) can be used as the input for workload allocation problem (sub-problem 2). Sub-problem 1 is a rate-power allocation problem at t time scale,

$$\begin{aligned} Maximize \quad & \overline{r} - \overline{P} \\ Constraints: \quad & (3.21), (3.22), (3.23) \\ Variables: \quad & r_i(\tau) \geq 0, P_{i,comm}(\tau) \geq 0. \end{aligned} \tag{3.26}$$

which can be solved with a stochastic programming approach discussed in the next section. Sub-problem 2 is a workload allocation problem over the larger time scale t,

$$\begin{aligned} Maximize \quad & \overline{T_{exe}} \\ Constraints: \quad & (3.24) \\ Variables: \quad & W_i(t) \geq 0, \overline{r}_i(t) \geq 0. \end{aligned} \tag{3.27}$$

where the $\overline{r}_i(t)$ is defined in (3.14). It can be proved that the optimal solution of this minimax problem for one session is

$$T_{1,exe}(t) = T_{2,exe}(t) = ... = T_{i,exe}(t) = \frac{W_0}{\sum_i \overline{r}_i(t)} = T_{exe}(t) \tag{3.28}$$

The proof is given in Appendix A. However, allocating workload with average computing information for session t is non-causal. $\overline{r}_i(t)$ can be approximated with $\overline{r}_i(t-1)$ and its estimation, $\overline{r}_{i,e}(t)$, during channel estimation phase for each session, that is

$$\overline{r}_i(t) = \beta \overline{r}_i(t-1) + (1-\beta)\overline{r}_{i,e}(t), \forall i \in \boldsymbol{S} \tag{3.29}$$

where $\overline{r}_i(t-1)$ is the average computing rate for last session $t-1$ and $\overline{r}_{i,est}(t)$ is the estimated average computing rate for current session. $\beta \in [0\ 1]$ is a constant.

3.4 Algorithm and Solution

In the above discussion, the primary problem formulation (3.25) can be solved with sequential programming for workload allocation in sub-problem (3.27), within which the power rate

allocation can be solved with a sub-gradient method for sub-problem (3.26) which leads to a distributed solution.

3.4.1 Power and Rate Allocation

The Lagrange for each service node in problem (3.26) is

$$\begin{aligned} D_i(\tau) = log(r_i(\tau)) - P_{i,comm}(\tau) - \lambda_1(r_i(\tau) - R_i(\tau) - \Delta r_i) \\ -\lambda_2(P_{i,comm} + P_{i,comp} - P_{avg}), \forall i \in \boldsymbol{S} \end{aligned} \tag{3.30}$$

where λ_1 and λ_2 are the Lagrange multipliers. Let $\boldsymbol{\lambda} = [\lambda_1; \lambda_2]$; the dual function of (3.26) is

$$g_i(\boldsymbol{\lambda}) = max\{D_i(r_i(\tau), P_{i,comm}(\tau), \boldsymbol{\lambda})\}, \forall i \in \boldsymbol{S} \tag{3.31}$$

where $r_i(\tau) > 0$ and $P_{i,comm}(\tau) > 0$ are the solution of the computing rate and communication power consumption. Thus, the dual problem of (3.26) can be represented as

$$\min_{\boldsymbol{\lambda} \geq 0, i \in \boldsymbol{S}} g_i(\boldsymbol{\lambda}) \tag{3.32}$$

This dual problem can be solved with a sub-gradient approach. At time τ, the channel gain is estimated and used to find the optimal computing rate and power consumption allocation as

$$\begin{aligned} \{r_i(\tau), P_{i,comm}(\tau)\} = argmax\{D_i\}, \\ r_i(\tau) \geq 0, P_{i,comm}(\tau) \geq 0, \lambda \geq 0, \forall i \in \boldsymbol{S} \end{aligned} \tag{3.33}$$

The sub-gradient of dual funcion can be generated as

$$\boldsymbol{\delta} = \begin{bmatrix} r_i(\tau) - R_i(\tau) - \Delta r \\ P_{i,comp}(\tau) + P_{i,comm}(\tau) - P_{avg} \end{bmatrix} \tag{3.34}$$

To approach the optimal $\boldsymbol{\lambda}^*$, $\boldsymbol{\lambda}$ is updated as

$$\boldsymbol{\lambda}(\tau + 1) = max\{\boldsymbol{0}, [\boldsymbol{\lambda}(\tau) - \boldsymbol{M}\boldsymbol{\delta}(\tau)]\} \tag{3.35}$$

where $\boldsymbol{M}$ is step sizes. The proof for the convergence and convergence rate can be found in [67, 68, 69]. It has been shown that, for a generalize form of problem (3.26), the duality gap between the primary problem and dual problem vanishes in the presence of fading [71].

In this algorithm, each node makes the choice of computing rate and power consumption allocation based on their channel condition by updating the Lagrange multiplier to approach the optimal solution. It should be noted that there is no message exchange amongst service nodes at this stage.

3.4.2 Workload Allocation

The choice of supported computing rate is determined by each service node without information from the other nodes and the total workload. The channel estimation can be done with a pilot signal during the link setup phase before there is a computing task request. When a requesting node requests a computing task, it will first collect the computing capability of each node, $\overline{r}_i(t)$, according to 3.29. Then the requesting node will determine whether the expected execution time T_{exe} is larger than the required delay performance T_0 according to (3.28). If the network computing capability cannot satisfy the delay requirement, the requesting node may invite more nodes to join the network or issue a resource access deny command. If the execution time is much smaller than the required delay, the requesting node may remove those service nodes with larger channel variance.

Once the requesting node gets all the computing capability of service node $\overline{r}_i(t)$ and has determined the number of WDCN nodes, it will allocate its workload among service nodes according to

$$W_i(t) = \frac{W_0 \overline{r}_i(t)}{\sum_{i=1}^{N} \overline{r}_i(t)} \tag{3.36}$$

The proposed resource and workload allocation approach for computation oriented WDCNs is summarized in Table 6.1 . Each service node makes a choice of its supported computing rate based on its own channel condition. There is no message exchange among service nodes except for the sending of information on the average computing rate to the requesting node.

3.5 Simulation

In this section, a computation oriented WDCN with 20 nodes is simulated with the proposed resource allocation method for each node according to Table I. The service nodes are distributed randomly within a unit area. The requesting node is located in the center of this area. The channel variance σ_i is different from node to node with a uniform distribution to simulate the spatial heterogeneity. A Rayleigh distribution with a mode of 1 is used to simulate fast fading. Each node has a power supply with $P_{i,avg}$=1 Watt output power. The RF front-end can support a maximum transmission power of $P_{i,max}$, and a rate difference term Δr_i corresponding to the buffer size. Device heterogeneity is simulated when the nodes within the network have different values of $P_{i,avg}$, $P_{i,max}$, and Δr_i. The most important factors for Class-2 WDCNs will be identified to provide insight of protocol implementation.

Table 3.2: Proposed Approach

At Service node i, for each session

1: $\tau = 0$; $\boldsymbol{\lambda} = 0.1$; $M = \frac{C}{\tau}$;
2: while $(\tau < C)$ and $(|\lambda(\tau) - \lambda(\tau - 1)| > \epsilon)$
3: Channel estimation at time t for service node i
4: $\{r_i(\tau), P_{i,comm}\tau\}$ according to (3.33)
5: generate sub-gradient according to (3.34)
6: update $\boldsymbol{\lambda}$ according to (3.35)
7: $\tau = \tau + 1$
8: end

At Requesting node 0

1: Requesting node request for computing information
2: Service nodes send $\overline{r}_i$ to requesting node
3: if (first session)
4: $W_i(t) = W_0/N$
5: else
3: Requesting node estimates T_{exe} according to (3.28)
4: If $(T_{exe} > T_0)$
5: if (there are more nodes)
6: Invite more nodes joining
7: else
8: deny the service and return insufficient resource information.
9: end
10: elseif $(T_{exe} << T_0)$
11: remove the service nodes with larger channel variances
12: re-calculate T_{exe}
13: end
14: Allocate the workload according to (3.36), (3.29) to service nodes.
15:end

Start computing service.

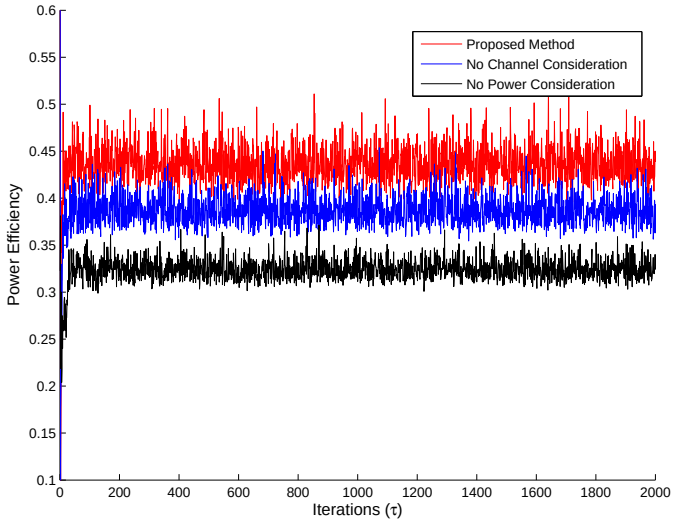

Figure 3.4: The convergence of power efficiency. $a = 0.15$,Δr=0. More power can be put into computing process when considering channel heterogeneity compared with no channel heterogeneity consideration and no power consideration.

3.5.1 Channel Impacts

When the channel variation is considered, higher rates and more workload are assigned to the nodes with smaller σ_i to achieve better energy efficiency as well as power efficiency. Figure 3.4 shows that more power can be put into the computing process when the channel condition is considered. The communication process without channel consideration consumes a larger portion of power supply compared with the scheme considering σ_i as shown in Figure 3.4. Energy efficiency is also improved when considering channel variations, as shown in Figure 3.5, where a has a smaller value of 0.15 due to highly efficient CPU units or a low duty cycle computing task defined in equation (2). Although these power efficiency and energy efficiency improvements by considering σ_i come with a loss of the supported computing rate and a slower expected execution time as in Figure 3.6, the variance of execution time is reduced with proposed method compared with no channel consideration case as in Figure 3.7. The loss of expected execution time may be leveraged with more service node in practice. A smaller execution variance leads to a more robust computing process.

3.5.2 P-f Relation

Previous work has shown that the parameter a in Equation (2) is one of the important factors in the power-rate relation for the computing process as well as system performance [49]. The value of a may be influenced by the power efficiency of computing units, traffic pattern, and

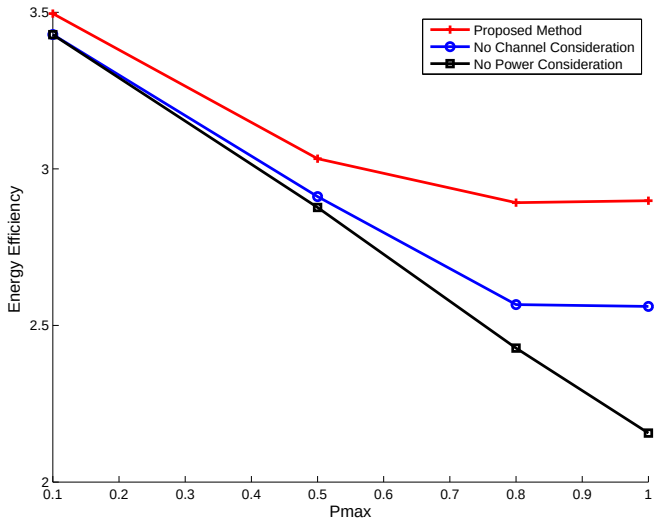

Figure 3.5: Energy efficiency as a function of P_{max}. Δr=0. Considering channel difference improve energy efficiency.

hardware. When the computing unit is power efficient or the duty cycle is low, the value of a is small. When the computing unit is designed with little consideration of power efficiency or the duty cycle is high, the value of a is large.

The impact of the value a on energy efficiency for three cases is shown in Figures 3.8. As the value of a increases, the computing process dominates the system performance, the power efficiency and energy efficiency tend to be similar regardless considering σ_i or not. The channel variance only needs to be considered in the small a region.

3.5.3 Storage Size

When computing rate is higher than communication rate, communication process makes an admission control in order to satisfy the rate stability for the queue inside BP. Instead of dropping packets, these data is stored in SM and retransmitted in the following session. The storage size can determine the rate difference between the computing process and communication process. The impact of the buffer size on the network energy efficiency is shown in Figures 3.9. A larger buffer size increases the network energy efficiency.

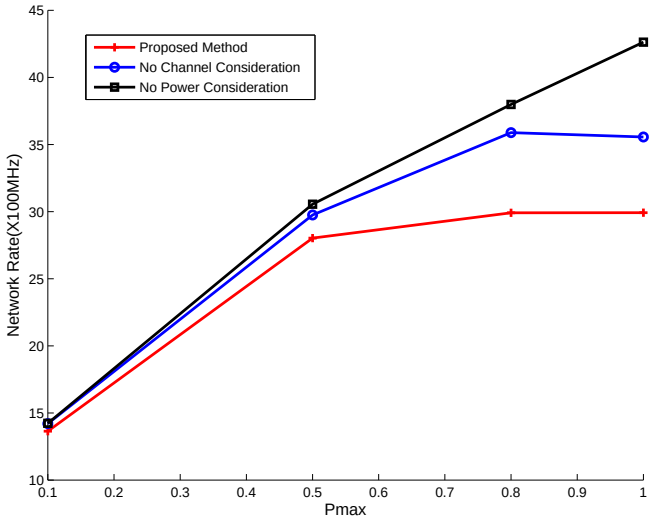

Figure 3.6: The network computing rate as a function of P_{max}. Δr=0.

3.6 Conclusion

This chapter presents a resource allocation scheme for computation oriented WDCNs with a stochastic programming approach. The impact of the system parameters on the performance are investigated in detail based on the proposed algorithm. Channel consideration can improve both the energy efficiency and power efficiency when communication power consumption and computing power consumption are comparable. The low duty cycle and low energy efficient CPU may limit this improvement of power efficiency. Smaller maximum transmission power can always improve energy efficiency, but it cost a lower network computing rate. The reduction of P_{max} leads to a decrease of the supported network computing rate within a small a region. Moreover, a larger buffer size will always increase the network computing rate and improve the energy efficiency. But its impact on the network execution time needs further investigation with different ratios between communication power consumption and computing power consumption.

Proof of (3.28)

For a network with n nodes, let $T_{1,exe}(t) = T_{2,exe}(t) = ... = T_{n-1,exe}(t) = T(t)$, $T(t) \neq T_{n,exe}(t)$, and the $T_{exe}(t) = \frac{W_0}{\sum_{i=1}^{N} \overline{r}_i(t)}$. Assume there is a better workload allocation scheme

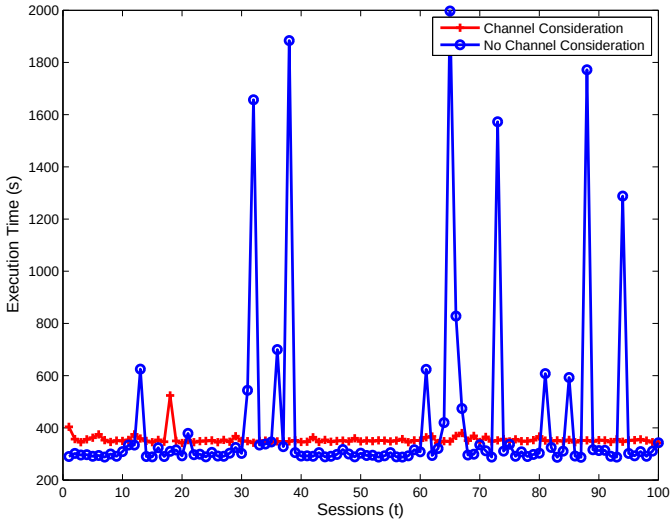

Figure 3.7: Execution time comparison between case 1 and case 2. Case 1 can reduce the variance of execution time and improve the robustness compared with case 2. P_{max}=0.8, $a = 0.15$, Δr=0.

that makes $T(t) < T_{exe}(t)$, then $T_{n,exe}(t) < T_{exe}(t)$. If $T(t) < T_{n,exe}(t)$,

$$\begin{aligned} W_0 &= \sum_{i=1}^{n-1} \overline{r}_i(t)T(t) + \overline{r}_n(t)T_{n,exe}(t) \\ &< \sum_{i=1}^{n} \overline{r}_i(t)T_{n,exe}(t) \\ &< \sum_{i=1}^{n} \overline{r}_i(t)T_{exe}(t) \,(= W_0) \end{aligned} \tag{3.37}$$

Since W_0 can not be smaller than itself, the assumed better solution is not feasible. The similar results can be derived when $T_{n,exe} < T(t)$, which indicates it is infeasible that both $T(t)$ and $T_{n,exe}$ are smaller than $T_{exe}(t)$. For a general case, every node has a different delay and assume there is better solution that $T_{1,exe}(t) < T_{2,exe} < ... < T_{n,exe}(t) < T_{exe}(t)$.

$$\begin{aligned} W_0 &= \sum_{i=1}^{n} \overline{r}_i(t)T_{i,exe}(t) \\ &< \sum_{i=1}^{n} \overline{r}_i(t)T_{n,exe}(t) \\ &< \sum_{i=1}^{n} \overline{r}_i(t)T_{exe}(t) \,(= W_0) \end{aligned} \tag{3.38}$$

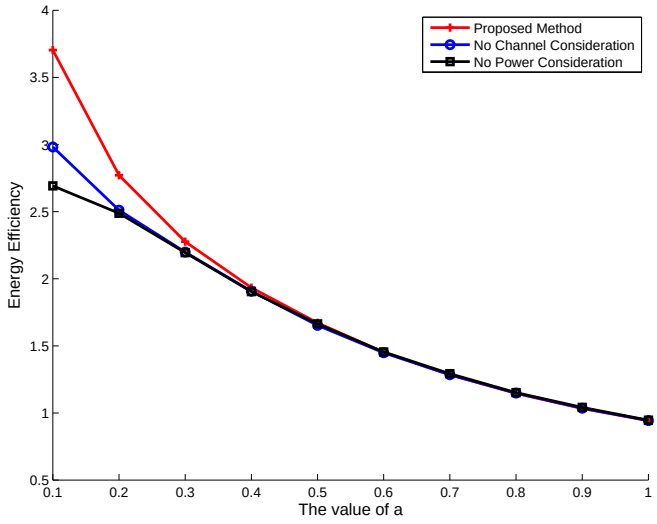

Figure 3.8: The energy efficiency as a function of a, Δr=0, $P_{max} = 0.8$.Smaller a indicates highly power efficient CPU units or a low duty cycle traffic.

which indicates the assumed solution is infeasible and best solution is $T_{1,exe}(t) = T_{2,exe}(t) = T_{3,exe}(t) = ... = T_{n,exe}(t) = T_{exe}(t) = \frac{W_0}{\sum_{i=1}^{N} \overline{r}_i(t)}$.

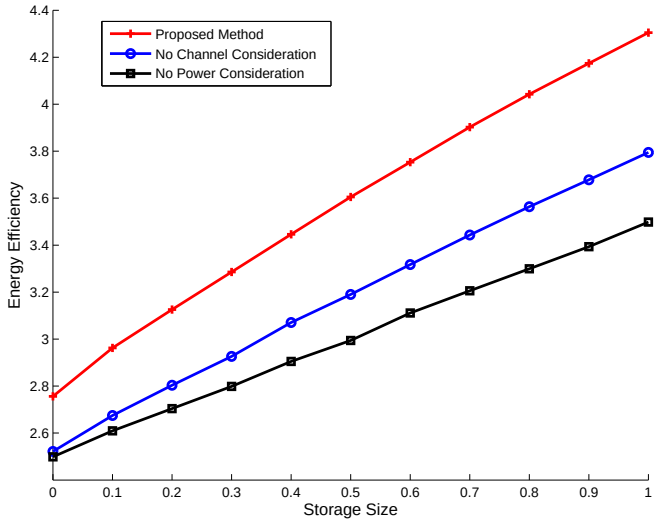

Figure 3.9: The energy efficiency as a function of storage size, a=0.15, $P_{max} = 0.8$.

Chapter 4

Service Access Model For WDCNs

This chapter discusses the service access model for wireless distributed computing networks, which determines the optimal number of service nodes to be accessed by a service request node (SRN) and whether the channel heterogeneity should be considered in order to minimize the average request response time. During service access, a SRN initials service requests to the service node (SN). One SN or multiple SNs respond to the requests with the desired information. This phase could be a preparation for the service provision phase, or it could be a service itself, especially for hardware sharing such as accessing sensors.

The service response time is the key performance indicator during this phase. It includes two components: communication delay and request processing delay. The outage probability of wireless links from a SRN to different SNs may be varying, which leads to different re-transmission times and channel service rates, if we look on each wireless channel as a server with data communication as its service. A heterogeneous queue network can be used to investigate the delay performance of wireless communications. In addition, the delay performance of the service request processing can be modeled with another queue attached to each communication server. Therefore, each SN will be modeled with two cascaded servers, one for communication and one for computing.

When a SRN initials a service request, it needs to make a choice of an optimal number of SNs that it should access in order to minimize the average response time. Generally, the service response time can be reduced with multiple SNs compared with one SN. However, there are several open questions to be answered. For example, what is the optimal strategy for a SRN? How many SNs should be used? How much is the performance gain when a second SN, or even multiple server nodes, are activated? What is the impact of workload density? What is the impact of node geometry? In this chapter, we will combine the interference model using stochastic geometry, the protocol model, and the queuing model in order to provide a framework to analyze the service access performance of WDCNs, especially the average service response time.

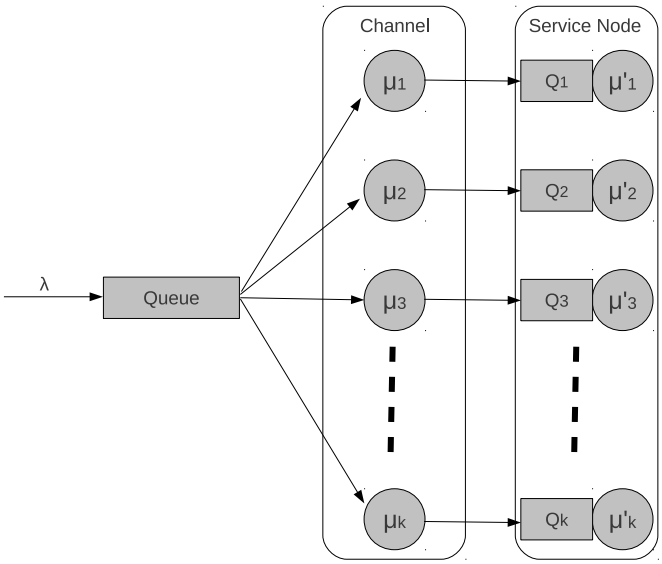

Figure 4.1: Service access model for WDCNs. $\mu_1 \geq \mu_2 \geq \mu_3 ..... \geq \mu_k$, $\mu' >> \mu$.

4.1 Introduction

The service access model can be modeled with the two-layer queue network shown in Figure 4.1. We use λ, μ, μ' to represent arrival rate, service rate of channel, and service rate of request. We assume the request processing time within SNs is much smaller than the channel access time, or $\mu' >> \mu$. Without loss of generality, assume the first server node has the fastest service rate that a SRN can find, defined as μ_{max}. The heterogeneity factor for server node i with service rate μ_i can be defined as

$$H_i \triangleq \frac{\mu_i}{\mu_{max}} \tag{4.1}$$

The larger the value of H_i, the more difference of service rates between server i and the first (fastest) server. The impact of H_i on average response time is the basis of SNs scheduling in order to speed up the service access process. The value of H_i is a function of interference, which relates to node's geometry and over-the-air protocol.

Wireless links can provide an access service for a large area at a relative low cost compared with wired cable systems. However, wired communications, especially optical fiber, are more efficient for longer distance data transmission. Current hierarchical networks, such as cellular and WiFi systems, are based on a structure where multiple lower rate information streams converge to higher rate streams with wireless links and are transferred through wired links. The service access of WDCNs follow a similar pattern. All the service requests converge to SRNs and then request service to SNs within their cluster. We can assume request arrival rate λ follows a Poisson distribution.

One simple idea for a SRN is to always select the SN with the best channel. For example, a SN can be ordered as $\mu_1 \geq \mu_2 \geq \mu_3$. The SRN chooses node 1 and node 2 when it wants to activate two SNs. However, it is not practical, since the desired resource may attached to node 3 instead of node 2. Since the second available SN may be randomly distributed within an area, the impact of the node's geometry on the channel service rate should be considered.

In queueing systems, it makes no difference which server is chosen as long as the servers are homogeneous. Join-the-shortest queue (JSQ) is the optimal method for queueing systems with homogeneous servers [78]. When a job arrives, JSQ observes the queue length of each server and assigns the new job to the server with the shortest queue length. For a heterogeneous servers queue, a threshold (THD) type method is proved to be optimal for two-server case [79]. In this method, one server has a faster service rate than the other. These two servers share a common queue. The scheduling method makes a choice between waiting for the faster server or going to the slower server once a new job arrives. The optimal policy is to always keep the faster server busy and activate the slower server after the queue length exceeds a threshold. Unfortunately, for the multiple-server case, the proof of optimality is incomplete [80], and the math to find the stationary probabilities is tedious because of the curse of dimensionality [81]. Therefore, we will investigate the two-servers case first and then study the upper bound of the multiple-server case. In this chapter, we refer to JSQ as the method without considering the heterogeneity. We refer to THD as the method considering heterogeneity. JSQ treats servers as homogeneous ones which is a special case of THD where the threshold always equals zero.

For heterogeneous servers, the average response time is a function of the service rate heterogeneity H and the arrival density, λ, $\tau = \tau\{H, \lambda\}$ with optimal scheduling methods. This response time includes the impacts of metrics in time domain λ and H. In the spatial domain, the heterogeneity factor is also a function of node density, $H = H(\lambda_s)$, for Poisson point process (PPP) with ALOHA protocol. The distribution of this heterogeneity factor, $f(H)$, can be derived with stochastic geometry theory. The integration of $\tau(H, \lambda)$ and $f(H)$ offers the average performance of the response time for a WDCN with node density of λ_s and request arrival density of λ during the service access phase. This introduces integration, $E\{\tau(\lambda_s, \lambda)\} = \int \tau(H, \lambda) f(H) dH$, which includes the impact of both time domain and spatial domain factors into the performance model.

This chapter is organized according to Figure 4.2. Section 4.2 derives the distribution of the heterogeneity factor, $f(H)$, as a function of node geometry and protocol. The successful probability p_s determines the distribution of re-transmission time which is a geometric distribution. Since a geometric distribution can be approximated with an exponential distribution, the service rate distribution μ for wireless links can be derived based on the outage probability and node geometry. This gives the distribution of the heterogeneity factor $f(H)$ immediately by normalizing the rate μ with maximum service rate μ_{max}. In addition, optimal strategies for multiple heterogeneous servers are investigated in Section 4.3 in order to show the average response time as a function of H and λ. A special case with two SNs is first studied in terms of its performance comparison between optimal methods and the methods

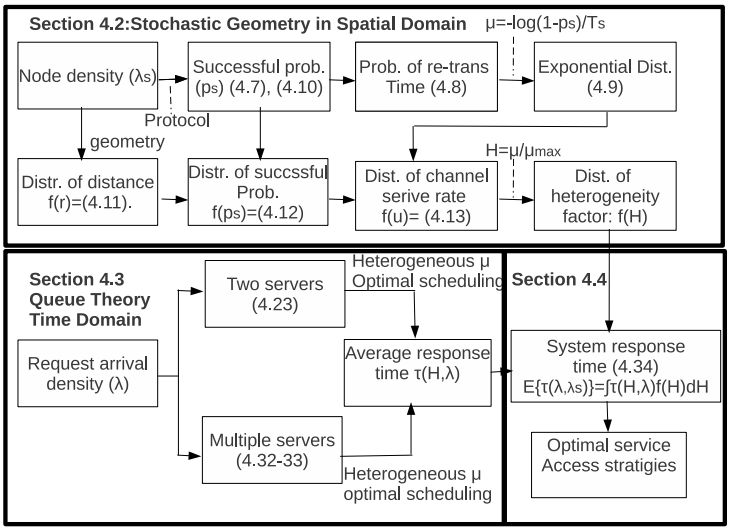

Figure 4.2: Organization of Chapter 4.

that do not consider heterogeneity. For the $K \geq 3$ SNs case, a numerical method based on the Howard iteration is used to find the optimal policy and evaluate $\tau(H, \lambda)$. Finally, Section 4.4 presents optimal strategies for the services access phase of WDCNs with the introduced integration of $\tau(H, \lambda)$ and $f(H)$.

4.2 Channel Heterogeneity Factor

This section is dedicated to modelling the heterogeneity of the channel service time for a give node geometry and ALOHA protocol. The outage probability is introduced based on stochastic geometry theory. The relation between the outage probability and the channel service time are related with an approximation method. The distribution of the heterogeneity factor is then derived.

4.2.1 Outage Probability for Slotted ALOHA Protocol

The outage probability determines the channel service time, which can be derived from stochastic geometry theory [74] [75][76]. For a large network with node density λ_s, each SNR initiates a service request transmission with a transmission probability of p. Assume the random variable X_i denotes the points of a measurement Φ in the Euclidean space $\mathbb{R}^d$

with $d \geq 1$ as in [76]

$$\Phi = \sum_i \delta_{X_i} \tag{4.2}$$

The intensity measure Λ of Φ is defined as $\Lambda(B) = \mathbb{E}\Phi(B)$ for Borel B, where $\Phi(B)$ denotes the number of points in $\Phi \cap B$. The Laplace functional of Φ is defined as [76]

$$\begin{aligned} \mathcal{L}_\Phi(f) &\triangleq \mathbb{E}[e^{-\int_{\mathbb{R}^d} f(x)\Phi(dx)}] = \mathbb{E}[e^{\sum_{x\in\Phi} f(X)}] \\ &= exp(-\int_{\mathbb{R}^d} (1 - e^{-f(x)})\Lambda(dx)) \end{aligned} \tag{4.3}$$

where f is a non-negative function on $\mathbb{R}^d$. The total interference measurement at a point x in the network is given by

$$I(x) \triangleq \sum_{Y\in\Phi_t} \ell(||x - Y||)), \tag{4.4}$$

where Φ_t is a point process of transmitters, and ℓ is a function of the Euclidean norm. Due to Slivnyak's results, $I(x)$ does not depend on a given location of x.

The distances between the origin and X_i in a d-dimensional uniform PPP of intensity λ_s can be defined as $\Phi \triangleq \{R_i = ||X_i||\}$, which is inhomogeneous PPP with an intensity function of $\lambda(r) = \lambda_s c_d d r^{d-1}$. $c_d \triangleq \pi^{d/2}/\Gamma(1 + d/2)$ is the volume of the d-dimensional unit ball. Then, ℓ for distance r can be defined as $\ell(r) = h_r r^{-\alpha}$ with a fading term of h_r and a path loss term of $r^{-\alpha}$. The Laplace transform of the interference is

$$\mathcal{L}_I(s) \triangleq \mathbb{E}[e^{-sI}] = \mathbb{E}\left[\prod_{r\in\Phi} exp(-sh_r r^{-\alpha})\right] \tag{4.5}$$

The expectation takes over both the point process Φ and fading h. The fading is independent of point process and from (4.3), we get

$$\begin{aligned} \mathcal{L}_I(s) &= exp\{-\int_0^\infty \mathbb{E}_h[1 - e^{-shr^{-\alpha}}]\lambda(r)dr\} \\ &= exp(-\mu c_d \mathbb{E}[h^\delta]\Gamma(1 - \delta)s^\delta) \end{aligned} \tag{4.6}$$

where $\delta = d/\alpha < 1$.

With Rayleigh fading and transmission power P, the outage probability over a link of distance

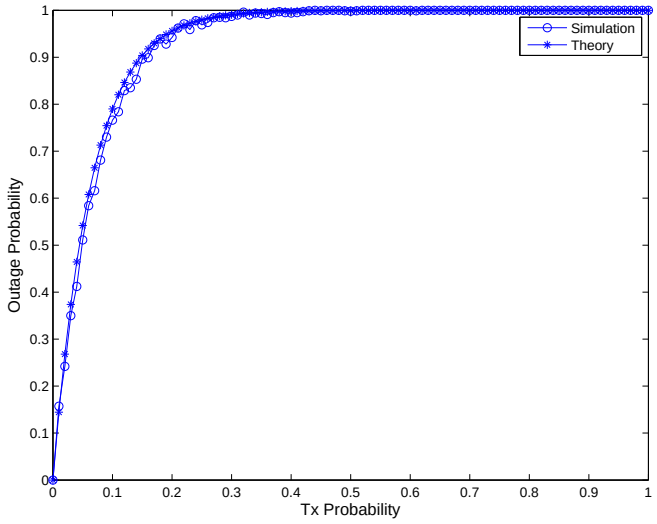

Figure 4.3: Outage probability as a function of transmission probability. Node density $\lambda_s = 1$, SIR threshold $T = 10dB$, coverage range $R = 1$, propagation index $\alpha = 4$, and dimension $d = 2$.

R with a SINR threshold T and noise power W for ALOHA protocol is

$$\begin{aligned} p_o &= 1 - p_s = 1 - \mathbb{P}\{S > T(W+I)\} \\ &= 1 - \mathbb{E}\left\{ Prob.(h^2 > \frac{T(W+I)R^\alpha}{P}) \right\} \\ &= 1 - \mathbb{E}\left\{ \int_{\frac{T(W+I)}{P}}^{\infty} e^{-u} du \right\} \\ &= 1 - \mathbb{E}\left\{ exp(-\frac{T(W+I)R^\alpha}{P}) \right\} \\ &= 1 - exp(-\frac{TR^\alpha W}{P}) \mathbb{E}(e^{-TR^\alpha I}) \end{aligned}$$

The first term of the successful probability relates to reference SNR at distance R. The interference term is the Laplace transform of interference evaluated at $s = TR^\alpha$ for equation (4.6). If noise power W is small enough, the outage probability can be simplified as

$$p_o = 1 - exp(-\lambda p c_d R^d \mathbb{E}[h^\delta] \Gamma(1-\delta) T^\delta) \tag{4.7}$$

A simulation of the outage probability is shown in Figure 4.3. The simulation result matches the theoretic curve plotted with equation 4.7. When the transmission probability increases, more interference and contention occur, leading to higher outage probability.

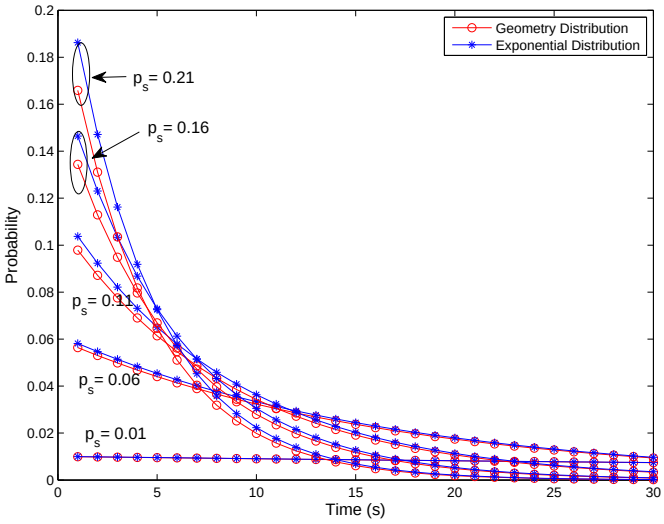

Figure 4.4: Approximation of geometric distribution with exponential distribution, $Ts = 1(sec.)$.

4.2.2 Distribution of Heterogeneity Factor

With a successful transmission probability of p_s, the probability of the event that the kth transmission is the first success follows the geometric distribution as

$$Pr(X = k) = (1 - p_s)^{k-1}(p_s) \tag{4.8}$$

the corresponding transmission time is kTs assuming Ts is the time for one transmission. For a exponential distribution with service rate μ or average service time of $1/\mu$, the pdf of Y is

$$f(y, \mu) = \mu e^{-\mu y}, y \geq 0 \tag{4.9}$$

With a relation of $\mu = -log(1 - p_s)/Ts$, the geometric distribution can be approximated with exponential distribution as $X = \lfloor Y/T_s \rfloor$, where $\lfloor\rfloor$ is the floor function. Figure 4.4 shows an example of this approximation. This approximation works better for high outage probabilities since the derivation of the two pdfs increases when the successful probability has a relatively small value. Therefore this method offers an upper bound of channel service rate μ. From equation (4.7), the transmission successful probability is a function of the coverage range r

$$p_s = exp(-Ar^d) \tag{4.10}$$

where $A = \lambda p c_d \mathbb{E}[h^\delta]\Gamma(1 - \delta)T^\delta$. For PPP and two dimensional node distribution $d = 2$, it is easy to show the pdf of the distance, r, as

$$f(r) = 2r \tag{4.11}$$

Combining (4.10) and (4.11), the pdf of p_s can be represented as

$$\begin{aligned} f(p_s) &= \left| \frac{d\sqrt{\frac{lnp_s}{A}}}{dp_s} \right| 2\sqrt{\frac{lnp_s}{A}} \\ &= \frac{1}{Ap_s} \end{aligned} \tag{4.12}$$

where $f(p_s) = \left|\frac{dr}{dp_s}\right| f(r)$ is used since $f(r) > 0$. With the relation of $\mu = -log(1-p_s)/Ts$ and (4.12), the pdf of μ can be represented as

$$\begin{aligned} f(\mu) &= \left| \frac{d(1-e^{-\mu Ts})}{d\mu} \right| \frac{1}{A(1-exp^{-\mu Ts})} \\ &= \frac{e^{-\mu T_s} T_s}{A(1-e^{-\mu T_s})} \end{aligned} \tag{4.13}$$

The impact of node density λ_s on the CDF of μ in a two dimensional area is shown in Figure 4.5. The simulation results match the theoretical curve according to equation 4.13. Larger values of node density (λ_s) cause more interference and higher outage probability, which shifts the curve to the left. Normalizing μ with μ_{max} gives the distribution of H, $f(H) = f(\mu/\mu_{max})$ which will be used to evaluate the optimal scheduling strategies and system average response time in Section 4.4. The heterogeneity increases with the increase of node density if the node with μ_{max} is always treated as the first activated SN. In other words, if a SRN needs to activate two SNs, the channel service rate difference between the second SN and the first (fastest) SN increases as the node density λ_s increases.

4.3 Scheduling Algorithms for Heterogeneous Servers

Activating multiple servers reduces the average response time. When the servers are homogeneous, a closed-form equation about the average response time is widely used [78]. However, there is currently no close-form equation for a multiple server queueing system with heterogeneous service rates. The service rate difference among servers makes it complex for arrivals to make a choice between waiting for the faster server in a queue and accessing the slower server immediately. Although an approved optimal solution for the two server case is given in [79], the proof of the optimal scheduling for more than three server cases is incomplete [80]. In this section, we will first introduce the performance analysis for the homogenous servers queue, which will be the baseline for the performance evaluation. Then the optimal scheduling methods for heterogeneous servers are discussed and evaluated in order to show the impacts of heterogeneity factor H and request arrival density λ on the average response time τ.

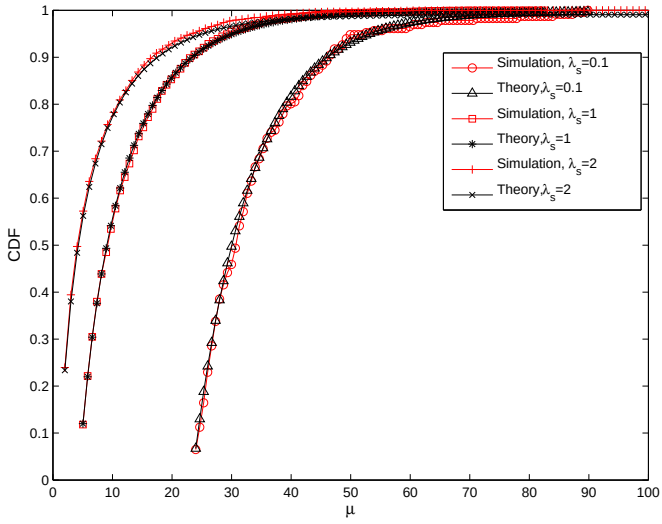

Figure 4.5: The impact of node density λ_s on the distribution of μ. $Ts = 0.1second$, p=0.1, T = 6 dB, α= 4; d =2.

If the channel service rate is unavailable, the SRN has to treat all SNs as the same, or homogeneous servers. If a SRN has this information, it can make a scheduling decision based on the heterogeneity factors H, or normalized service rate as in 4.1. The performance speedup between these two can be represented as

$$S1 = \frac{Performance\ Considering\ Heterogeneity}{Performance\ without\ Considering\ Heterogeneity} \tag{4.14}$$

It is important whether the channel heterogeneity should be considered for multiple server case since collecting and maintaining channel status uses more resources and increases system complexity.

The results of using multiple SNs is also compared to the single server case, which can be represented as

$$S2 = \frac{Performance\ with\ Two\ SNs}{Performance\ with\ One\ SNs} \tag{4.15}$$

The value of $S2$ determines whether multiple servers should be activated in addition to one server with the fastest service rate.

4.3.1 Mean Performance for Queue System

In a queue system with general arrival and service rate and one server ($G/G/1$), the occupation rate (server utilization) per server is $\rho = \lambda E\{B\}$, where λ is the arrival rate and $E\{B\}$

is the server busy time. When the capacity of the system is sufficient to deal with arrivals, the mean sojourn time $E\{S\}$, the mean number of jobs in the system $E\{L\}$, and λ have the relation

$$E\{L\} = \lambda E\{S\} \tag{4.16}$$

Similarly, the average number of jobs in the queue, $E\{L^q\}$, and the waiting time, $E\{W\}$, has the relation as

$$E\{L^q\} = \lambda E\{W\} \tag{4.17}$$

For a $M/M/1$ queue system, since the average number of customers in the system seen by an arriving job equals $E(L)$ and each of them has a service time with mean $1/\mu$ and the job has to wait for its own service time, the arrival relation can be represented as

$$E(S) = (E(L) + 1)\frac{1}{\mu} \tag{4.18}$$

Combining equation (4.16) and (4.18), we have service response time

$$\tau \triangleq E\{S\} = \frac{1/\mu}{1-\rho} \tag{4.19}$$

where the expectation is over arrivals along a time scale. Since ρ is the mean number of customers in service, the mean queue length can be represented as

$$E\{L^q\} = E\{L\} - \rho = \frac{\rho^2}{1-\rho} \tag{4.20}$$

The waiting time can be derived by

$$E\{W\} = E\{S\} - 1/\mu = \frac{\rho/\mu}{1-\rho} \tag{4.21}$$

With a similar procedure, the mean performance for the multiple server queue system $(M/M/k)$ with k servers can be derived as

$$E\{W\} = \prod\nolimits_W \frac{1}{1-\rho}\frac{1}{k\mu} \tag{4.22}$$

where $\prod_W$ is the delay probability which can be solved with numerical methods. The equations in this section offer the theoretic results of the average response time for the single server case (M/M/1) and homogeneous multiple servers (M/M/K), which will be used for performance comparison in the following sections.

4.3.2 Two Servers Case

Although the optimal control theory for the heterogeneous system is still under study ([80]), the optimal control for the two server case has been solved by [79]. In a two server queue system, the buffer has a Poisson arrival with a rate of λ. It is served by two servers with different mean service time. The service time is exponentially distributed with a rate of μ_i (i=1,2). We assume $\mu_1 < \mu_2$ and $\lambda < \mu_1 + \mu_2$.

The optimal policy that minimizes the mean sojourn time is of the threshold type. The faster server should be kept busy whenever it is available, while the slower server should be activated once the queue length exceeds a threshold. This optimal threshold can be computed by minimizing the cost function J_{t_m} [79]

$$J_{t_m} = \left\{ \begin{array}{l} \frac{\sum_{i=1}^{2} c_i \eta_i^m \left\{ [\frac{\rho}{1-\rho} - \frac{\eta_i}{1-b}]m + \frac{\eta_i}{(1-b)^2} b^{-m} + [\frac{1}{(1-\rho)^2} - \frac{\eta_i}{(1-b)^2} - \frac{1}{1-\eta_i}] \right\}}{\sum_{i=1}^{2} \left\{ \frac{\eta_i}{b(1-b)} b^{-m} + [\frac{\rho}{1-\rho} - \frac{\eta_i}{1-b}] \right\}} \; if \lambda \neq \mu_1 \\ \frac{\sum_{i=1}^{2} c_i \eta_i^m \left\{ \frac{\eta_i}{2} m^2 + [\frac{\rho}{1-\rho} + \frac{\eta_i}{2}]m + [\frac{1}{(1-\rho)^2} - \frac{1}{1-\eta_i}] \right\}}{\sum_{i=1}^{2} c_i \eta_i^m \left\{ \frac{\eta_i}{2} m + [\frac{\rho}{1-\rho} + \frac{\eta_i}{2}] \right\}} \; if \lambda = \mu_1 \end{array} \right\} \tag{4.23}$$

where $c_1 = \frac{1-\eta_1}{\eta_2-\eta_1}$, $c_2 = \frac{\eta_2-1}{\eta_2-\eta_1}$, $\eta_1 = \frac{1-\sqrt{1-4\mu_1\lambda}}{2\mu_1}$, $\eta_2 = \frac{1+\sqrt{1-4\mu_1\lambda}}{2\mu_1}$, $\rho = \frac{\lambda}{\mu_1+\mu_2}$, $b = \frac{\lambda}{\mu_1}$. Since the minimum of J_{t_m} is the average number of jobs in the system, combining the above equation with equation 4.16 gives the average sojourn time. The optimal threshold is a function of the heterogeneity of the two service rate heterogeneity $H = \frac{\mu_2}{\mu_1}$ and the arrival density λ.

Figure 4.6.a shows the average response time with the optimal threshold method for two heterogeneous servers using equation 4.23. The performance as a function of workload density ρ illustrates that a larger workload ρ increases the average response time. In addition, when there is no heterogeneity, $H = 1$, the average response time equals the results of equation 4.22, which indicates the system degrades to a homogeneous $M/M/2$ queueing system.

An example of the optimal threshold of the queue length as the function of H and workload ρ is shown in Figure 4.6.b according to equation 4.23. The stairway shape of the optimal threshold indicates whether to consider heterogeneity factor, H, is determined by both the range of H and ρ. For example, thresholds equal zeros when $H \geq 0.5$ in this figure, which indicates that two servers could be treated as homogeneous servers and it is not necessary to consider the channel heterogeneity factor within this range. When the rate difference is large and H is small, optimal thresholds of queue length increase quickly for a light workload with a small value of ρ. As the threshold is approaching infinity, a single server is preferred.

The average response time reduction when considering the heterogeneity factor is shown in Figure 4.7. When the rate difference between two servers decreases, the optimal thresholds approach zero, or the homogeneous solution. With a smaller value of H, or larger heterogeneity, the optimal threshold provides higher performance speed up compared to the no threshold method. For a light loaded system, such as $\rho < 0.5$, considering the heterogeneity factor can improve average response time significantly. However, a heavy loaded system has limited performance improvement by the optimal threshold method compared to treating

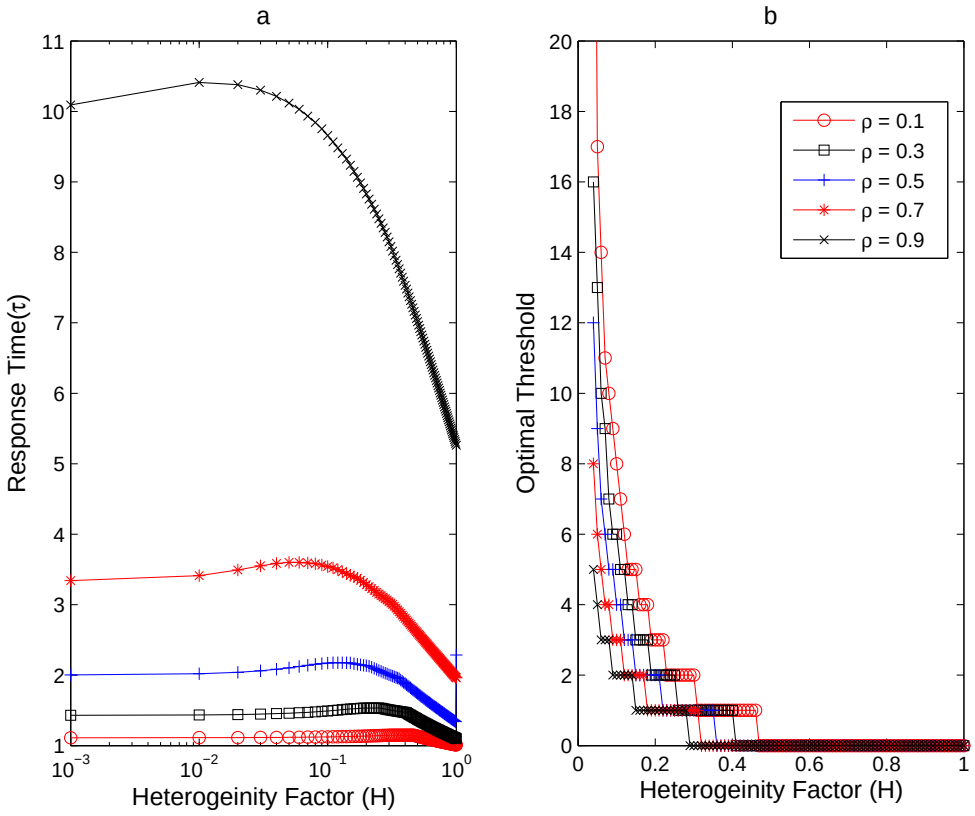

Figure 4.6: The average response time (a) and optimal threshold of queue length. (b) as functions of H and ρ for two heterogeneous server queueing systems.

two servers as homogeneous servers.

In summary, considering the channel heterogeneity factor offers a significant improvement of the average response time for a light loaded system with a high service rate heterogeneity for the two server case.

4.3.3 Multiple Servers Case

This section will investigate the performance with more than three SNs, or $K \geq 3$. The optimal control of the queue system makes a decision about the arrival based upon the observation of queue and servers status, which can be modeled as an Markov decision process (MDP). A numerical solution for MDP, such as the Horvard iteration, can be used to evaluate the optimal threshold for the multi-server case [81]. Based on this evaluation, the average response time considering the heterogeneity factor can be shown.

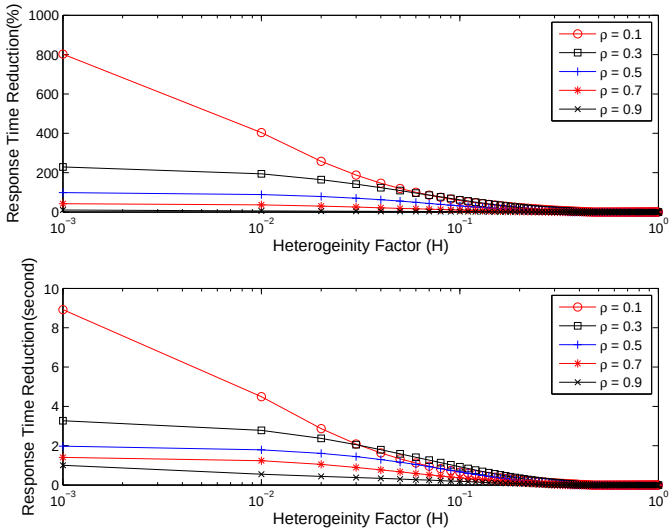

Figure 4.7: Response time reduction as a function (S1) of heterogeneity factor H and workload ρ for the threshold method over the homogeneous method.

Consider a queue system with K servers, and the processing rate of μ_i has a a descent order $\mu_1 \geq \mu_2 \geq \mu_3 ... \geq \mu_K$. A vector with $K+1$ bits is used to indicate the status of the system. The first bit of this vector indicates the queue length (q) which could be any integer smaller than $N-K$. The lower K bits are used to indicate the server status d_{K+1-k}, where 1 indicates the server is occupied or busy and 0 indicates the server is idle. The Kth bit indicates the status of Server 1 with the fastest service rate of μ_1.

With a queue length of $q(x)$, the system status can be represented with an integer

$$x = q2^K + \sum_{1 \leq k \leq K} 2^k d_{K+1-k},\ 0 \leq x \leq I-1 \tag{4.24}$$

where the total number of the states for the system is $I = (N-K)2^K$. The cost for each state is

$$c(x) = q(x) + \sum_{1 \leq k \leq K} d_k \tag{4.25}$$

The cost function to be minimized is

$$Y(t) = \int_0^t c(u)du \tag{4.26}$$

where $c(u)$ is the cost calculated with equation 4.25 at time u. The minimum average cost for a policy f can be represented as

$$g(f) = \inf g(t;f) = \inf \lim_{t \to \infty} \frac{1}{t} \mathbb{E}_x^f \{Y(t)\} \tag{4.27}$$

where x is the system status.

For a MDP problem with minimizing average cost, the optimal value of the *Bellman* function satisfies

$$v(x) = \min_{a \in \mathbb{A}(x)} b(x, a; v(x)) \tag{4.28}$$

where a is an action from an action set $A(x)$. An optimal policy is

$$f^* = \underset{a \in \mathbb{A}(x)}{\text{argmin}} b(x, a; v(x)) \tag{4.29}$$

If t is large enough , or $t \to \infty$, the value of the Bellman function is time independent as

$$v(x, t) = g(f)t = v(x) \tag{4.30}$$

which indicates the system enters a stationary status. Since both the arrival process and service process are Poisson processes, the derivative of $v(x, t)$ satisfies the following

$$\frac{\partial v(x,t)}{\partial t} = c(x) - (\lambda + M_1(x))v(x,t) + \lambda T_0 v(x,t) + \sum_{j \in J_1(x)} \mu_j T_j v(x,t) \tag{4.31}$$

where M_1 is the rate sum for all the busy servers with $d = 1$, T_0 is the action for arrival, and T_j is the action for departure; $J_1(x)$ is the set for busy servers.

Combining equations 4.29, 4.30, and 4.31 offers a method to find an optimal threshold for heterogeneous servers queueing systems.

Assume there are multiple policies to evaluate in a policy set $\mathbb{F}$, for the nth policy, $f_n \in \mathbb{F}$, a minimum value of the Bellman function satisfies

$$\begin{aligned} v_n(x) &= \frac{1}{\lambda + M}[c(x) + \lambda v_n(x + e_{f_n(x)}) + M_0(x)v_n(x) \\ &+ 1_{q(x)=0} \sum_{j \in J_1(x)} \mu_j v_n(x - e_j) \\ &+ 1_{q(x)>0} \sum_{j \in J_1(x)} \mu_j v_n(x - e_j - e_0 + e_{f_n(x - e_j - e_0)}) - g_n(f)] \end{aligned} \tag{4.32}$$

where e_j is a unit vector with the jth bit equals to one, which represents actions. g_n is the long term processing cost, or average number of jobs in the system, plus the number of busy servers for policy n. The first term denotes the jobs in the system for status x. The second term indicates the events of arrivals. The third term indicates the events of no change. The fourth and fifth terms indicate departure with queue that is empty and not empty, respectively. Equation 4.32 can be solved with an iterative approach [81] for each

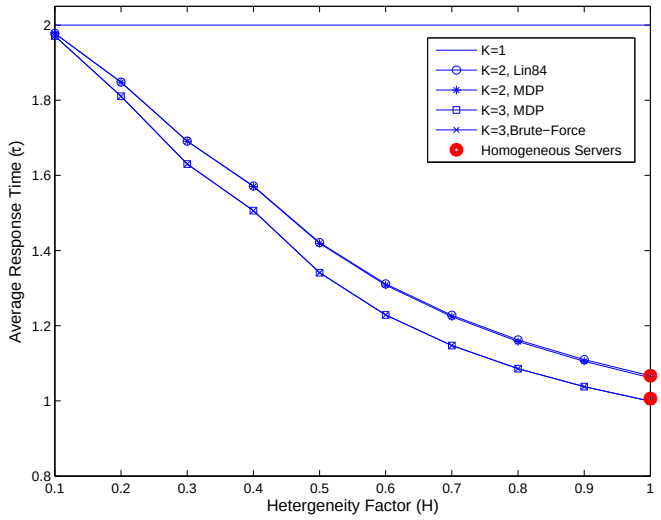

Figure 4.8: Average response time τ as a function of heterogeneity factor, $\lambda = 0.5$. Comparison between the results of equation 4.23 and MDP methods.

policy as

$$\begin{aligned} g_n^{m+1} &= \frac{1}{\lambda + M}[\lambda v_n^m(e_{f_n(0)}) - g_n^m] \\ v_n^{m+1}(x) &= \frac{1}{\lambda + M}[c(x) + \lambda v_n^m(x + e_{f_n(x)}) + M_0(x) v_n^m(x) \\ &+ 1_{q(x)=0} \sum_{j \in J_1(x)} \mu_j v_n^m(x - e_j) \\ &+ 1_{q(x)>0} \sum_{j \in J_1(x)} \mu_j v_n^m(x - e_j - e_0 + e_{f_n(x-e_j-e_0)}) - g_n^{m+1}] \end{aligned} \tag{4.33}$$

which gives the value of $v_n(x)$.

Changing the policy from f_n to f_{n+1} and evaluating each policy $f_n \in \mathbb{F}$, the optimal threshold and $v(x)$can be determined. Then the average response time can be derived with equation 4.16. Compared with the algorithm in [81], this method exploits the threshold structure and evaluates every possible policy, instead of setting up a table of state transitions, in order to find the optimal thresholds, which has a faster speed.

For the two server case, the evaluation results of the MDP method and the close form solution according to equation 4.23 match well, as shown in Figure 4.8. When $H = 1$, both methods approach the results of homogeneous multiple server systems according to equation 4.22. For the more servers case, we first consider a three-server case with $\mu_1 = 1$ and $\mu_2 = \mu_3 < 1$. Since $\mu_1 \geq \mu_2 \geq \mu_3 ... \geq \mu_K$, the best possible value for μ_3 is $\mu_3 = \mu_2$, which will provide

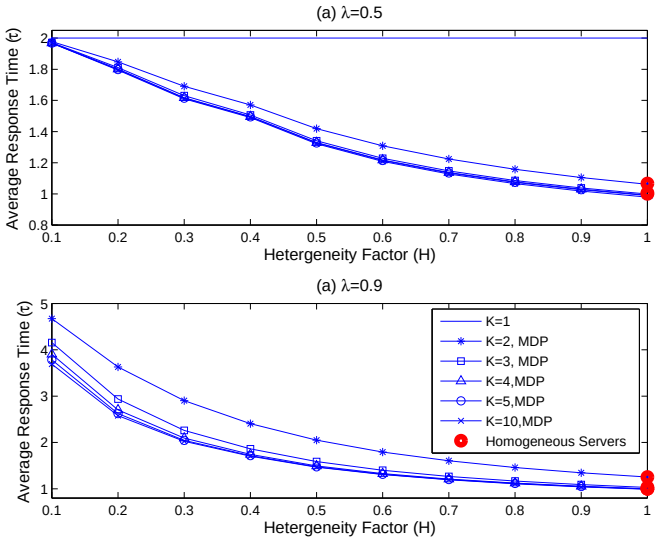

Figure 4.9: Average response time τ as a function of heterogeneity factor (H) and number of servers K, (a) $\lambda = 0.5$ (b) $\lambda = 0.9$.

an upper bound of the performance. Figure 4.8 also shows the average response time as a function of the heterogeneity factor for the $K = 3$ case. The line with squares shows the results for the case when the thresholds of Server 2 and Server 3 are the same. The brute-force method evaluates all the policies with different combinations of thresholds for the three servers. The results show that the same threshold for Server 2 and Server 3 is actually optimal.

Figure 4.9 shows the average response time τ as a function of the heterogeneity factor H for arrival rate $\lambda = 0.5$ and $\lambda = 0.9$, respectively. All servers with index greater than two use the threshold of Server 2 in this evaluation. The results show that the performance improvement from Server 2 and Server 3 is significant while the increase of the number of SNs for $K \geq 4$ only provides marginal benefit. When heterogeneity factor H is small, all the lines overlap with each other which indicates the threshold to activate the second server is large and single server solution is preferred. This section introduces the optimal scheduling methods for queueing systems with multiple heterogeneous servers, which identifies two important rules as follows

- Consideration of heterogeneity: When the heterogeneity factor is small, the optimal scheduling method considering heterogeneity is actually approaching the single server case. When the heterogeneity factor is large, the optimal scheduling method considering heterogeneity is actually approaching a multiple homogeneous system. Therefore, it is not necessary to consider channel heterogeneity in these cases. However, considering channel heterogeneity for H within the medium range provides significant performance gain.

- The number of servers: The performance speedup diminishes when more servers are activated. The second server and the third server provide the most significant performance gain. When the arrival density is low, multiple servers can provide more speedup when compared to the case with heavier arrival density.

4.4 Optimal Scheduling in Service Access

In Section 4.3, the optimal scheduling strategies for a multiple servers queueing system are discussed as a function of the arrival density of λ and the heterogeneity factor H. Section 4.2 derived the distribution of H as a function of the node density λ_s. This section will combine the results from the previous two sections in order to investigate the performance improvement for the average response time and optimal scheduling strategies where both λ and λ_s are considered.

Equation (4.13) defines the distribution of the heterogeneity factor for a Poisson point process with ALOHA protocol. The average response time can be evaluated with equation (4.23) and (4.33). When we combine these two, the average system response time for WDCN can be represented as:

$$E\left\{\tau(\lambda_s, \lambda)\right\} = \int_{i \in \mathbb{S}} \tau(H, \lambda) f(H) dH \tag{4.34}$$

There are two integrations in equation 4.34. The first expectation of the response time τ defined in equation 4.19 is taken over time. When combining this with the distribution of H, the second expectation $E\{\tau\}$ takes the expectation over the set of nodes $\mathbb{S}$.

Figure 4.10 shows the comparison of average response time according to equation 4.34 for the two server case. The transmission probability of the ALOHA protocol is $p = 0.1$. This figure illustrates that average response time is a function of the densities in both the spatial domain and the time domain. For arrival density $\lambda < 0.5$, the average response time of the threshold method remains similar with different values of node density λ_s while the response time of the JSQ method is varying with different λ_s. As λ increases, the difference between the performance considering heterogeneity and no heterogeneity consideration methods become similar, which indicates the impact of arrival density in time domain dominates and the impact of node density in the spatial domain diminishes.

Figure 4.10 also compares the performance of the two SNs case with the single SN case. The THD method has to be used if the information of μ is available and the arrival density λ is not heavy. If the workload is heavy, the method without consideration of the heterogeneity factor is sufficient to provide similar average response time performance compared to the homogeneous method. Figure (4.10) shows that it is necessary for a SRN to search for the second SN only when arrival rate λ exceeds a certain level, $\lambda \geq 0.4$ in this case. The SRN should request only one SN with the fastest μ when $\lambda \leq 0.4$ whatever the node density λ_s.

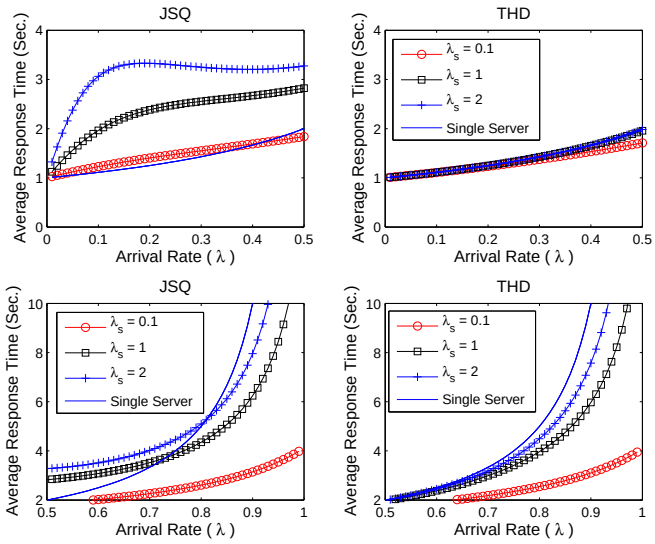

Figure 4.10: Average performance as a function of arrival rate λ and node density λ_s. $T = 6dB$, $p = 0.1$. Comparison of the system average response time between single server and multiple servers ($S2$) as well as that between considering channel heterogeneity and no channel heterogeneity consideration, $S1$.

Figure (4.11) shows the performance speed up of $E\{\tau\}$ between considering heterogeneity and when no channel heterogeneity is considered, $S1$, defined in equation 4.14, is a function of node density λ_s and workload λ. When the node density λ_s is low and heterogeneity is small as showed in Figure (4.5), considering the heterogeneity factor provides smaller performance gain compared with the method where channel heterogeneity is not considered. For higher node density, considering heterogeneity factor with optimal threshold can provide a significant reduction of average response time, especially with a medium value of λ. The speed up of $S2$, defined in equation 4.15, decrease as the node density λ_s increases which indicates that higher nodes density leads to more interference and a worse channel service rate.

Figure 4.10 and 4.11 together illustrate that it is important for a SRN to access more than two SNs when the request arrival density, λ, is within a medium range, $T1 \leq \lambda \leq T2$. When $\lambda \leq T1$, a SRN should access only one SN. When $\lambda \geq T1$, a SRN should access multiple SNs. When $\lambda \geq T2$, a SRN can treat multiple SN as homogeneous servers. In the given example in these two figures, $T1 = 0.4$, $T2 = 0.8$.

Increasing the number of SNs from one to two will reduce the average response time only when the arrival rate exceeds a threshold with ALOHA protocol and PPP. Using equation 4.34, the average performance of the service access model $E\{\tau\}$ can be found for $K \geq 3$ cases with a similar method for the two server case. $E\{\tau\}$ as a function of the number of SNs, K, and the request arrival rate λ is shown in Figure 4.12. When node density λ_s equals one,

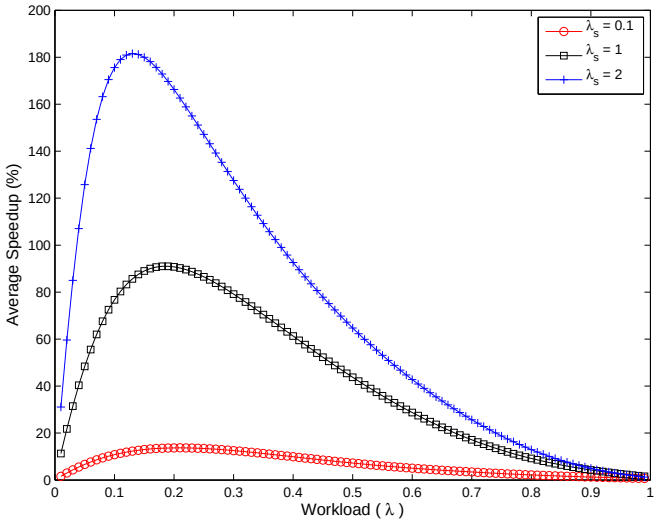

Figure 4.11: The performance speedup of $E\{\tau\}$ between considering heterogeneity and no heterogeneity consideration as a function of node density λ_s and workload λ. $T = 6dB$, $p = 0.1$.

increasing the number of SNs improves $E\{\tau\}$ only when workload $\lambda > 0.5$, approximately. The increase of the number of SNs provides marginal performance speedup for $K \geq 4$.

The impact of node density (λ_s), request density (λ) and the number of SNs (K), on service access strategies can be summarized as

- $S1$, the performance gain between considering H and not considering H, increases as λ_s increases.
- $S2$, the performance gain between multiple SNs and a single SN, decreases as λ_s increases.
- Considering H offers the optimal average response time with different values of λ.
- Considering H methods are valuable for $T1 \geq \lambda \geq T2$.
- $K \geq 4$ servers provides a diminishing performance gain.

4.5 Conclusion

This chapter introduced a framework to analyze the service access performance with a combination of stochastic geometry and queue theory. For the ALOHA protocol and Poisson

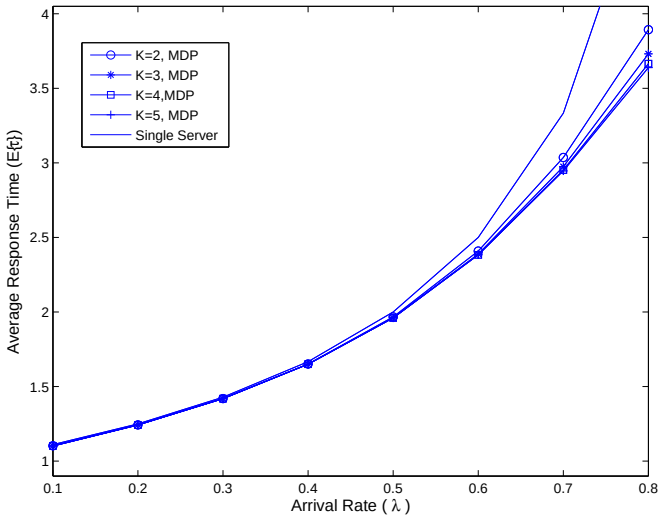

Figure 4.12: System average response time $E\{\tau\}$ as a function of workload (λ) and number of servers K considering channel heterogeneity, $\lambda_s = 1$.

Point Process, the optimal strategy is to choose the best three servers and threshold type of scheduling methods in order to dramatically improve the average performance. When request arrival density λ is low, considering heterogeneity factors with the THD method can improve the average performance over that of homogeneous methods for the multiple SNs case. A SRN can actually access to one SN instead of multiple SNs since the optimal threshold for the second server is large. When arrival density is high, all SNs can be treated as homogeneous nodes by using the same threshold as zero for all nodes. The increase of SNs provides a marginal performance gain for $K \geq 3$ cases. The optimal strategy for other protocols and node distributions may be evaluated in a similar way as proposed.

The future direction of this chapter is to consider the job size distribution for the service process time. There are two types of scheduling for queue systems, node scheduling and job scheduling. Node scheduling, sometimes called router or dispatcher, determines which SN accepts the service requests. Job scheduling determines which job is to be processed first. In this chapter, we only investigated the performance improvement with node scheduling. The service request processing time of SNs can vary due to the different job size even with the same processing rate. For example, the sizes of files on one disk are different, which leads to different amount of time to access different files. The file access time will have a fat tail of pdf instead of an exponential one. When $\mu' \simeq \mu$, variant job scheduling may have different impacts on the average response time [77]. With higher priority to jobs with smaller processing time, the average performance could be further improved.

Chapter 5

Network Initialization and Synchronization

WDCNs can group wireless devices to share hardware and software resources through different types of wireless connections. Due to the heterogeneous wireless links and different communication protocols among nodes, reconfiguration capability is essential. Although software defined radio (SDR) may leverage this problem to some extent, how to synchronize the reconfiguration process in a dynamic spectrum access (DSA) environment with heterogeneous nodes remains an open problem. This chapter will first discuss the reconfiguration requirements of WDCNs. Then, design challenges for logic clock synchronization are introduced and analyzed in terms of the lower bound and practical design issues. A synchronization scheme is proposed for multi-cluster and multi-channel WDCNs. The comparisons between proposed method and the reference design are given at the end. The simulation results show that the proposed method can be used for SDR based WDCNs in DSA environment without a significant increase of time complexity compared with the reference algorithm. Some material in this chapter was published in [82].

5.1 Introduction

WDCNs can group different types of resources and share them within the grids through wireless links. These resources, such as mobile smart phones, laptops, and even some dummy devices like sensors, printers and cameras, could be attached to the heterogeneous devices. To share resources, these devices need to communicate with each other through wireless links. Since these devices may have different wireless links, a reconfiguration of PHY/MAC layers is needed to maintain a ubiquitous connection. The reconfiguration of the SDR enables the WDCNs to connect different types of mobile platforms. In addition, resource sharing within a grid is inherently a wireless distributed system. Depending on the application, the

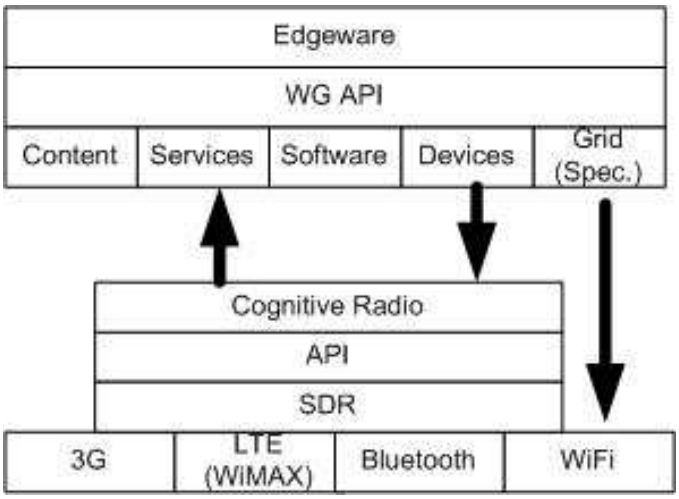

Figure 5.1: Software architecture for SDR based wireless distributed computing nodes

specifications of resource sharing may vary. For example, the synchronization and timing schemes for sensor sharing and CPU sharing are different from each other. These differences may also require the reconfiguration of higher layer protocols in addition to PHY/MAC.

5.2 SDR Based WDCNs

WDCNs are defined as ad-hoc dynamic sharing of physical and virtual resources among heterogeneous devices. In a typical WDCN, a service requesting node initializes a resource sharing request to the grid. A node with the requested resource becomes the service node for the requesting node. It will respond to the service request and provide service for its requesting node. The communication among nodes goes through wireless links, which may be different from each other. The typical software architecture of a WDCN based on SDR is shown in Figure 5.1 [83]. The SDR builds the foundation for integrating different wireless standards. The grid specifications define the interfaces, function requirements and integration processes when mapping the upper layer protocols to the wireless links underneath. It also offers the regulation and sanity check to guarantee the wireless links reconfiguration works in an expected manner. The resources shared within a grid could be content, software, services, and even hardware devices. In addition, the two API layers support their upper layers with functionalities. The edgeware is the access point when forming a WDCN and it can provide the initialization and tearing down functionality, network monitoring and management, and the language when nodes talk to each other.

WDCNs can reduce the distance between services and their potential users, which reduces the traffic through backhaul networks and servers. This may also lead to reduction of network power consumption and response time (RT). Moreover, WDCNs can provide ubiquitous

access to the desired resources and services. This loose coupling nature leads to a more flexible grouping strategy.

Recently, several WDCNs applications have begun to emerge, such as cloud printing, home group, distributed content grid, and Global Information Grid [84]. Researchers in the cellular industry are also discussing the possibility of a cloud cell phone with LTE technologies [85]. Moreover, Virginia Tech and Syracuse University, as well as several other universities and organizations, are working together on a Wireless innovation testbed (WiGiT) in order to find the insights into the potential applications of WDCNs and to generate open specifications [83]. In other words, WDCN is a promising research area that combines wireless communication technologies and distributed computing.

Before the wide application of WDCNs, there are several important open questions that need to be solved, within which reconfiguration and synchronization are two important topics and will be discussed in the following sections.

Hardware and software reconfigurations are the most important strengths of SDR. Flexible RF hardware, such as the small wide-band antenna, RFIC, low power DSP processors, and matching circuits design, are currently under research and development [86][87][88]. The software communication architecture (SCA) and GNU radio are the two famous software architectures for SDR [89]. The reconfiguration capability determined by both the hardware and software is an essential component for WDCNs.

However, reconfiguration capability is also the weakness of a SDR based WDCN in terms of the synchronization process. A WDCN has some requirements for the reconfiguration of SDR, such as stability, concurrence, speed, and cost. The reconfiguration itself is actually a disturbance to a WDCN from the networking point of view. The reconfiguration of different nodes should converge to a certain steady state without manual tuning. Therefore, it also needs to have a global timing reference for all the nodes to behave in an organized way instead of random reconfiguration which hurdles the concurrence. Moreover, the reconfiguration process should be fast enough in order to resume the disrupted service and avoid the interference to other nodes. Finally, the lower cost is always preferred considering the device heterogeneity.

The reconfiguration process has to be coordinated among nodes. For example, both WiFi and Bluetooth work in the 2.4 GHz ISM band and have several channels overlapping with each other [90]. If the reconfiguration process of one node lags behind or progresses much faster than the other nodes, the networks will need more time to be synchronized with each other and make an agreement on the logic clocks. Moreover, the un-synchronized nodes will introduce extra interference and disruption. This is even worse for a WDCN with dynamic spectrum access capability since the false alarm of channel occupation will increase.

A successful reconfiguration of wireless links is critical for SDR based WDCNs. Each reconfiguration process of WDCNs is correspondent to an initialization process, of which synchronization is a building block. Therefore, synchronization will influence the stability, speed,

and concurrence of a reconfiguration directly. A careful design of the synchronization scheme is one of the requirements and the basis of a successful reconfiguration.

5.3 Synchronization Scheme

There are several levels of synchronization for radios, such as carrier, bit, slot, frame/superframe, and protocol. The rest of this chapter discusses the synchronization of the logic clock in the initialization phase of WDCNs. The logic clock of a node depends on the hardware clock and the message it receives, based on which a round number is generated. For example, each node in a WDCN maintains a round number during communication, and this round number defines the timing behavior of protocols. Once they have proper round numbers, synchronization is achieved and the grid is formed.

Synchronization is more important, but difficult for WDCNs than conventional wired distributed computing networks due to the openness and randomness of the wireless channel, the random locations of nodes, and the ad-hoc nature of WDCNs. The synchronization skew among nodes happens for three reasons: the uncertainty of message delay, the drifts of hardware, and the network diameter and size [91]. Although the uncertainty of message delay can be mitigated with a careful design of clock rate increase or decrease [92], dynamic wireless environments and heterogeneous devices will deteriorate the synchronization performance and even compromise this type of methods. In addition, the wireless protocol parameters, such as re-transmission times, communication frequency band, and node degrees, will also impact synchronization performance. For the SDR platform, the interface between the RF front-end and GPP may add more uncertainty to the process of synchronization [93].

The easiest way to achieve synchronization is GPS [91]. But this solution requires a line of sight reception of the satellite signal. In addition, not every device is capable of embedding a GPS receiver, especially wireless sensors. Moreover, the non-determinism in transmission time caused by the MAC layer leads to the difficulties for a coordinated time increment.

There are five requirements for synchronization: validity, synch. commit, correctness, agreement, and liveness [94]. Once synchronized, every node knows it is synchronized and the round numbers can continue increasing without losing the synchronization, which means that each node will always generate a consistent round number after the synchronization with high probability.

5.3.1 Performance Bound

The performance bounds for wireless synchronization are listed as follows.

- Single Channel: It can be shown that no algorithm can prevent a clock skew of $\Omega(log_b D)$

for neighboring nodes with a diameter of D [95], where the value of b relates to the maximum clock drifting bound. D is the maximum uncertainty between any two nodes. The upper bound has been improved to this value also for a special case [95]. A skew between any two nodes is bounded by $\Omega(\sqrt{D})$ with high probability.

- Multiple Channels with Interference: For a regular network [94], its synchronization protocol requires at least

$$\Omega(\frac{ln^2N}{(F-t)lnlnN)} + Ft\frac{lnN}{F-t} \tag{5.1}$$

where F is the total number of channels, t is the maximum number of the possible disrupt channels, N is the upper bound of participating nodes, or network size, and $F-t$ is the number of clear channels.

5.3.2 Practical Designs

The synchronization performance bounds are based on the worst case analysis. In practice, synchronization may vary according to system assumptions. Optimal synchronization protocols may have better performance than the lower performance bound by adapting to different scenarios instead of only considering the worst case. Different types of synchronization schemes have been suggested for wireless sensor networks in practice [96][97]. The most famous ones are Reference Broadcasting Synchronization (RBS) and the flooding time synchronization protocol (FTSP). RBS uses physical time-stamps to remove the non-determinism of the transmitter and form precise relative timing without an external reference [98]. By utilizing MAC time-stamping and time skew estimation, FTSP can mitigate the uncertainty of message delay to a few clock ticks [99]. These schemes are suitable for synchronization during communications.

Some of the current proposed synchronization algorithms for network initialization are based on leader election to group the nodes since leader election is a better choice for multi-hop and dynamic networks compared with spanning tree [94][95]. Once a leader is elected successfully, all the other nodes within a cluster can synchronize themselves to their leader. If the nodes location are modeled as a two dimensional Poisson point process, a leader election or synchronization process can be finished in $T = 3e\delta lnN$ with a high probability no less than $1 - N^{-3}$. Here, δ is the maximum number of a nodes neighbors (network degree), and N is the total number of nodes (network size) [100]. For a multi-channel and single hop scenario, an adaptive protocol terminates synchronization within $O(t'ln(3N))$ depending on the actual disrupt frequency number, $t' < t$, instead of the worst case with t disrupted channels [94].

Table 5.1: Frequency Table Update Algorithm

1: $If(collisionhappens)$
2: $\quad Score(i) = Score(i)(F-1);$
3: $\quad Score(j) = Score(j) + 1; j \neq i$
4: $\quad q(k) = Score(k)/sum(p(k)); k = 1, 2F.$
5: $elseif(Rxsuccessfully)$
6: $\quad Score(i) = Score(i) + 1;$
7: $\quad Score(j) = Score(j) - 1; j \neq i$
8: $\quad q(k) = Score(k)/sum(p(k)); k = 1, 2, \ldots, F.$
9: end

5.3.3 Proposed Algorithm

SDR based WDCNs require synchronization before and after reconfiguration. Moreover, different wireless links may have different frequency channels. Sometimes, these channels may be overlapped, such as between WiFi and Bluetooth. Therefore, multi- channel is a requirement of the reconfiguration process of SDR based WDCNs. The results of the synchronization process for SDR based WDCNs should be multi-hop and multi-channel with clusters as building blocks.

However, the existing synchronization schemes do not satisfy these requirements. For example, [94] only considered a multi-hop wireless network with a single channel. Although [100] discussed the multiple channel cases, it only considered a single hop scenario. We assume that the minimum unit for the communication is a slot. Each node can switch frequency channels from slot to slot. The receiver can receive a message successfully when there is only one transmitter transmitting. That is, nothing can be received if collision happens. Each transmitter transmits according to a probability of p, which is an ALOHA type transmission.

Moreover, each node maintains a frequency table and scores for each frequency channel. The score indicates the channel status. When collision happens, the node will reduce the score for that channel. When the node receives a message successfully on a channel, it will increase its score. This score is normalized to generate the probability for frequency selection. When a node needs to switch the channel, it always chooses the better channel with higher probability. Note that spectrum sensing is not included with this algorithm. This frequency table actually provides channel cognition even without spectrum sensing. The details for constructing the frequency table are shown in Table 5.1.

This multi-channel cognition capability may be integrated with the multi-hop clustering method in [100] in order to form a synchronization scheme for a SDR based WDCN in DSA environment, which is included in Appendix A. Each node can be one of three possible roles: master node, quasi-slave node, and slave node. All the nodes start as quasi-slave nodes. They will change their roles to quasi-leader nodes if they have not receive any contention after a certain period. If quasi-leader nodes observes any signal for quasi-leader or leader nodes after

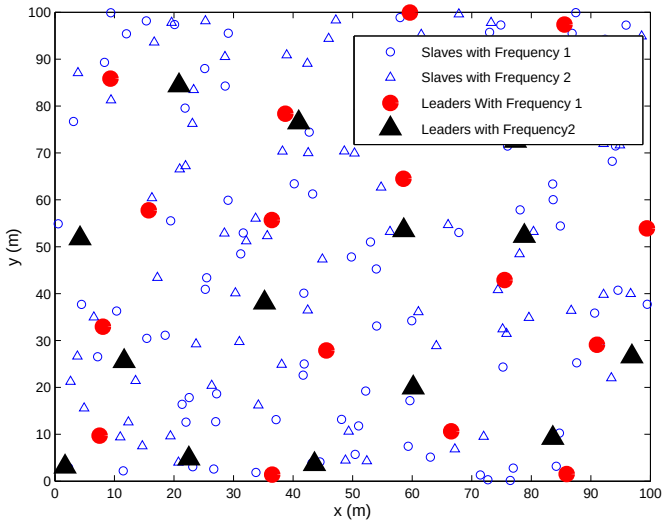

Figure 5.2: Location distribution with proposed method. 200 nodes distributed within 10000 square meters with $R = 20m$ and $\epsilon = 10^{-3}$

a certain period, they will change their roles to leader nodes. Otherwise, quasi-leader nodes will change its role to quasi-slave node when they receive a signal from quasi-leader or leader nodes. Each node changes its role by itself based on the signal they received. In addition, frequency channel status is updated according to the method described in 5.1. Quasi-leader nodes can select a transmission frequency using the frequency channel status based on the observed contention information. The algorithm terminates when there are only two types of nodes, leaders and slaves. Theoretical analysis can be performed following the similar steps in [100].

5.4 An Example

In this section, we will simulate a synchronization process based on the proposed method for SDR based WDCNs in DSA environments. In order to compare the proposed method with the reference algorithm in [100], we follow a similar definition of parameters.

If the nodes are located within an area with a fixed node density of λ_s, and transmission range of R, the maximal number of the nodes neighbors δ can be derived with an accuracy no lower than $1 - \epsilon$, where $\epsilon > 0$ and $\epsilon \to 0$ [101]. The optimal transmission probability can be shown as

$$p = 1 - \delta^{\frac{1}{1-\delta}} \tag{5.2}$$

The timer threshold for status transitions are $T_1 = 3\delta lnN$ and $T_2 = 3e\delta lnN$, where N is the

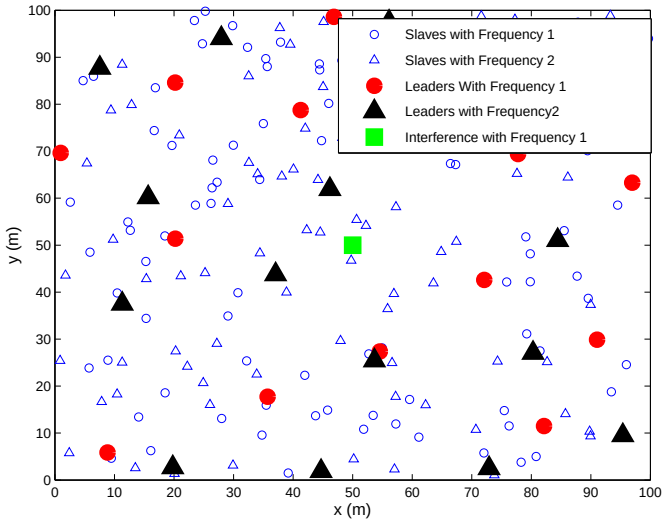

Figure 5.3: Location distribution with interference. 200 nodes distributed within 10000 square meters with $R = 20m$ and $\epsilon = 10^{-3}$

network size [100]. If there are 200 nodes located within an area of 100X100 square meters, $\lambda_s = 0.02$, $R = 20$ meters, and $\epsilon = 10^{-3}$, then $\delta = 60$, $p = 0.067$, $T_1 = 954$, and $T_2 = 2592$ [100]. We assume that the total number of the frequency channels is $F = 2$.

Figure 5.2 shows a simulation result with the above parameters. The slave nodes are uniformly distributed within the transmission range of the leader nodes. Figure 5.3 shows a result when an interference source is located in the center of the area with a transmission range of 20 meters. The nodes within the inferences neighborhood avoid the disrupted frequency channel and use the second channel.

In addition, the number of nodes in channel 1 is equal to that of channel 2 as shown in Figure 5.4. Both the number of nodes and the number of leaders are almost evenly distributed among frequency channels when the number of frequency channels increases, as illustrated in 5.5. Figure 5.6 compares the convergence performance of the proposed method with the method in [100]. The modified method does not increase the running time too much. It only increases the number of leaders. Since the coverage range R is fixed, multiple channels split the network into more clusters in order to cover the same area size with multiple channels. The number of leaders is approximately proportional to the number of available channels if R remains the same as shown in Figure 5.4, 5.5, and 5.6.

Figure 5.7 shows the running time for different network sizes. The running time for those two methods are at the same level and increases as the network size increases. Both the proposed method and reference method are robust to the network size since the running time is a concave function of the network size.

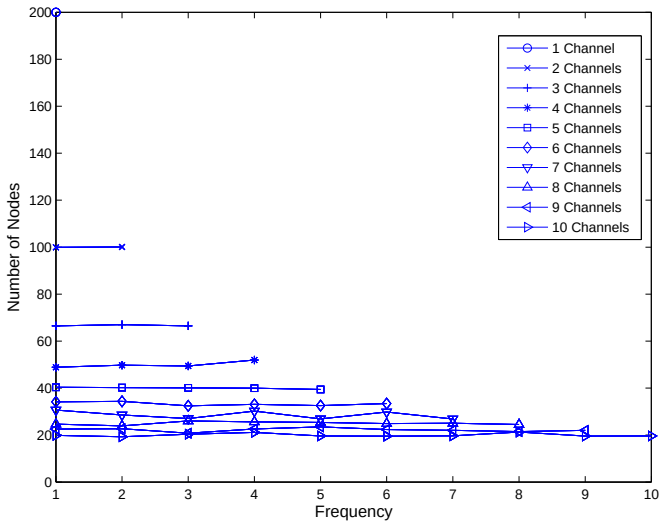

Figure 5.4: Node distribution among frequency channels. Nodes are evenly distributed across channels.

5.5 Conclusion

This chapter discusses the network initialization function requirements for SDR based WDCNs, especially reconfiguration and synchronization. A synchronization method with frequency channel cognition is proposed to satisfy the multi-hop and multi-channel requirements of SDR based WDCNs in DSA environments. The simulation example shows that the proposed method can help form the WDCN within a limited running time. The proposed method can distributed the nodes evenly amongst different frequency channels.

This work does not consider the impact of mobility. Both the convergence time and time complexity of the synchronization algorithm will be increased if the wireless nodes are moving. More research is needed to better understand the impact of mobility on the algorithm.

Table 5.2: Network Synchronization Algorithm

$Fun\_send()$: *Send packet with prob.p.* $Fun\_recv()$: *receive packet in current slot.*	$Freq\_update()$: *update freq. table* $Freq\_selection()$: *select freq.*
1: *State of node i* : $Si = quasi - leader$	24: $Si = Slave; Freq\_update(); break$
2: $While$ (1)	25: *end*
3: $Timer++$	26: $If timer = T1$
4: $If\ Si = quasi - leader then$	27: $Si = quasi - leader; timer = 0;$
5: $Fun\_send()$	28: *end*
6: $If\ Fun\_recv() = quasi\_leader$	29: $If\ collision = 1$
7: $S_i = quasi\_slave; timer = 0;$	30: $Freq\_update(); Freq\_Selection();$
8: $Freq\_update();$	31: *end*
9: $else\ if\ Fun\_recv() = leader$	32: *end*
10: $S_i = slave; Freq\_update(); break;$	33: $If\ Si = leader$
11: *end*	34: $Fun\_send();$
12: $If\ timer = T1$	35: $If\ timer = T2$
13: $S_i = leader; timer = 0;$	36: $Break;$
14: *end*	37: *end*
15: $If\ collision = 1$	38: *end*
16: $Freq\_update(); Freq\_selection();$	39: $If\ Si = slave$
17: *end*	40: $Fun\_recv();$
18: *end*	41: $If\ collision = 1$
19: $If\ Si = quasi - slave$	42: $Freq\_update();$
3 20: $If\ Fun\_recv() = quasi - leader$	43: *end*
21: $Timer = 0;$	44: *end*
22: $Freq\_update();$	45: *end*
23: else if Fun_recv()=leader	

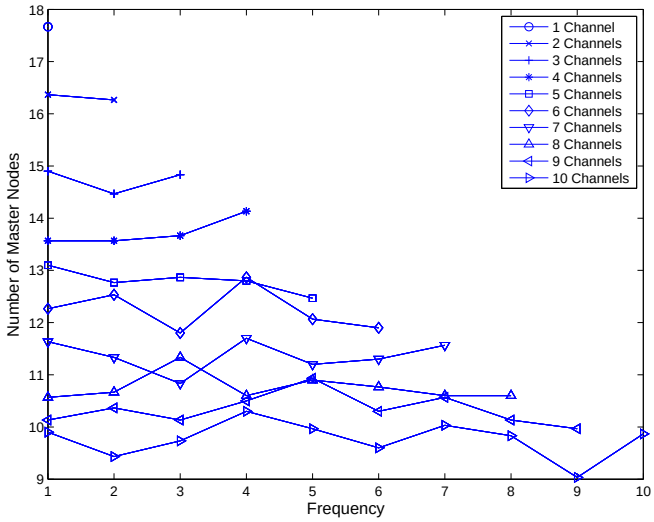

Figure 5.5: Leader distribution among frequency channels. Leader nodes are evenly distributed across channels.

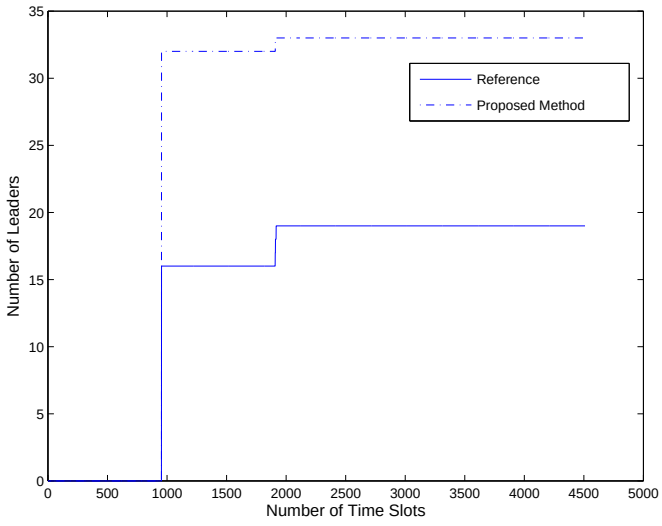

Figure 5.6: Convergence Rate comparison.

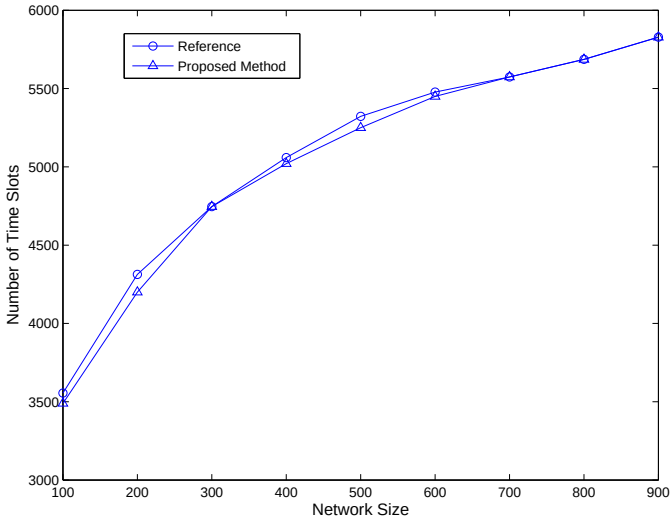

Figure 5.7: Time complexity comparison.

Chapter 6

Dynamic spectrum access in WDCNs

Dynamic spectrum access concept provides one of the enabling tools to implement WDCNs. Current spectrum policy is usage-oriented which means each spectrum block is assigned and dedicated to a particular application. However, spectrum access for one application has temporal and spatial patterns which leads to a low spectrum utilization. Therefore the spectrum can be statistical re-used by multiple applications in order to improve the spectrum utilization.

As discussed in Chapter 1 and Chapter 5, there may be heterogeneous devices in one WDCNs with different radios capabilities. DSA helps WDCNs accommodate all kinds of radios with different over-the-air interfaces in a more flexible way. Moreover, some white space spectrum blocks are available as the transition from analog TV to digital TV recently. These white space frequency blocks are potential for novel applications where WDCNs fits in as the secondary user systems. One of the most important problems for WDCNs in DSA environment is how to protect a primary user while maintain the links for the second user (SU). In the rest of this chapter, we refer to WDCN nodes as SU.

This chapter will discuss a primary user detection method for WDCNs in DSA environments. A classifier based on support vector machine (SVM) is proposed to reduce the channel vacation time during the communications. Some material of this chapter was published in [102].

6.1 Introduction

DSA can increase spectrum use by exploiting unoccupied spectrum. The secondary user (SU) in a DSA system has to sense and avoid the primary user (PU) of the spectrum. PU detection is the fundamental functionality of dynamic spectrum access systems, especially for the SU. There are at least two purposes for PU detection. During the connection setup

phase, PU detection will determine whether the desired channel is vacant and can be used by the SU. Once the connection is set up, the SU will periodically sense the channel and will release the channel once the presence of the PU is detected.

Current PU detection requires the SU to stop communications when sensing the spectrum. For example, in IEEE 802.22 systems idle time is used by the customer premises equipment (CPE) for spectrum sensing to determine whether there is a TV signal within a particular area. If sensing slots are used for primary user detection, lots of signal/interference detection methods can be used to fulfill this spectrum sensing task. There are three major types of techniques for signal detection: the matched filtering scheme, the energy detection scheme, and statistical analysis based feature detection, e.g. cyclostationary [114, 115, 112]. Recent developments in this field, using wavelets and compressed sensing, are promising in handling the resolution issues of wide band signals [134, 135, 119]. Mixed signal sensing, multidimensional signal analysis, and distributed cooperative sensing are also introduced to relax the bandwidth and sensitivity requirement of the RF-front end in [137, 117, 124, 113]. Optimal MAC protocols have been discussed in the literature as a way to balance the tradeoff between maximizing detection probability and minimizing detection delay by adopting the sensing slot length and sensing period [138, 120].

The detected signal or interference sometimes needs to be classified to know the modulation scheme and to make better decisions about the presence of primary users. Signal classification techniques can be divided into two major categories: feature based (FB) classification and likelihood based (LB) classification [108]. LB methods offer optimal results in the Bayesian sense but suffer from computation complexity. The average likelihood ratio test (ALRT), generalized likelihood ratio test (GLRT), and hybrid likelihood ratio test (HLRT) are the three major proposed LB methods depending on how the unknown parameters are treated and whether the probability density function is known [107]. Compared with LB approaches, FB approaches are easier to implement and have less computation complexity. In addition to the wavelet transform approach and cyclostationary analysis approach mentioned above, some other signal feature extraction approaches are also used to identify signal format, such as high order statistics [132, 106], instantaneous magnitude, phase, and frequency [103, 127, 109], zero crossing [118], signal constellation recovery [125] and spectral features [130]. Some of these features may be organized to form a hierarchical system for optimal classification, thus satisfying the tradeoff between performance and complexity.

However, the approach based on idle time, or sensing slot, is not sufficient if a primary user appears during the communication of secondary users. The secondary users' receivers cannot vacate the occupied channel immediately since they have to wait for the idle time to sense the existence of the primary user. Thus, this idle time reduces spectrum efficiency and increases the channel vacancy response time when a PU appears. If the secondary receivers can detect the primary user during the communication, the usage of idle time can be reduced, and the channel shift response time may be improved. Detecting PU during SU communication corresponds to detect interference in the presence of the desired signal.

Several approaches can be used to detect interference beneath a signal. The state-of-the-art technology to detect the interference beneath a signal is the digital phosphor technology (DPX) display approach in the real time spectrum analyzer by [133]. If the spectrum processing speed is higher than the data acquisition speed, the interference beneath a signal can be determined by the color. Unfortunately, high speed spectrum generation is too expensive for mobile terminals to be applicable to DSA systems. Another approach of detecting interference beneath the signal relates to the cluster reduction in radar systems by [131]. The basic idea is to use a cluster signal as a reference signal when the echo from the target does not exist. For example, ground clusters can be removed in the Doppler domain due to their zero velocity although they cannot be separated from the signal in the temporal domain. When the cluster echo has similar Doppler velocity, such as the ocean cluster with echo from moving ships, the assumption that the characteristics of cluster in adjacent bins are similar can be used to remove the cluster from the desired signal. Three hypothesis testing criteria are used for signal detection, such as max Bayes, minmax, and Neyman-Pearson, after removing the interference. All of these detection approaches require knowledge of the probability density function (PDF) of the received signal.

For communication systems, interference rejection, interference cancellation, and multi-user detection (MUD) are potential solutions to detect interference beneath a desired signal. Methods to detect interference beneath a desired signal have been investigated for a long time, and a lot have been proposed. A good summary of interference rejection for communication systems can be found in [121] and [128]. Adaptive equalization technologies, e.g. the constant-modulus algorithm, the decision-directed adaptive filter, neural networks, time-dependent filtering, and nonlinear filters have been proposed for interference rejection [121]. Some techniques that exploit orthogonality in the temporal or space domain are also commonly used, such as multi-user detection (MUD) in CDMA systems and beamforming in MIMO systems [122, 126, 136, 129].

However, there is no bandwidth difference and orthogonality between a PU signal and SU signal for SU receivers. PU and SU are sharing the same frequency band. A SU normally has a narrower bandwidth compared with PU bandwidth in cognitive radio. The received PU signal and SU signal have the same bandwidth at the SU receiver, which keeps the adaptive filters from detecting a PU signal underneath a SU signal. Moreover, time/spectrum correlation methods cannot work well since there is no orthogonality between SU and PU waveforms. Interference cancellation is difficult since it is hard for the SU to estimate the channel between the PU and the SU. In addition, beamforming and generalized beamforming are not applicable, since we assume there is only one antenna for SUs.

The only difference between the PU signal and the SU signal at SU receivers is the power amplitude. The major difficulty in the detection of the interference co-channel with a signal is that the received signal power is the sum of the signal (S), interference (I), and noise (N). It is difficult to separate them when no reference interference is available. When the ISR is low, it would be even harder to detect the interference co-channel with a desired signal. In addition, there are situations where S or I can be absent. Detection approaches need to

classify these possible combinations when trying to detect the interference beneath a desired signal. Instead of detecting the interference from the signal directly, we propose to record the statistical patterns of different combinations of S, I, and N. These statistical patterns are used to train a SVM. Then the combinations of the three components are classified by the SVM during the communication phase. These combinations can be detected through the classification process.

Channel vacancy time is critically important in determining whether the primary system operators are willing to share their spectrum for secondary usage. Most PU detection methods or dynamic spectrum sensing methods focus on how to find an empty spectrum by listening to the spectrum during the communication setup phase. The interference temperature model used in cognitive radio is also based on interference measurement using idle time [105]. Several publications about MAC protocol design discuss the channel vacancy time by considering the dynamic presence of primary users during SU communication [138, 120]. However, all are based on sensing slots where there is a tradeoff between maximizing the usage of opportunity channels and minimizing channel shifting time. The purpose of this chapter is two-fold. First, we propose a method to detect PUs by SUs during their communication which reduces the required numbers of sensing slots. This helps to improve channel usage and channel vacancy time when compared with the current methods. In addition, a SVM based classifier is formulated to show that it is feasible to detect the PU during communication. Good general capability, low model complexity independent on the input dimensionality, and fast speed with shorter classifiers, makes SVM a better solution for this problem compared with other pattern recognition methods.

The chapter is organized as follows. Section 6.2 introduces the basic assumptions and the system model. Section 6.3 discusses how to reformulate the detection problem with a classification approach, and introduces the proposed SVM-based approach. We demonstrate the performance of the proposed method using theory analysis and simulation examples in Section 6.3 and Section 6.4. A conclusion of our study is made in Section 6.5.

6.2 System Model

The system scenario is shown in Figure 6.2. In DSA systems, PU detection is required for both the connection setup phase and the communication phase. In this chapter, we only consider interweave DSA systems, such as IEEE 802.22 systems, without considering overlay and underlay paradigms. Since the PU always has priority, the SU needs to sense the spectrum before it can use a particular channel. The SU can use the spectrum block when it is not being used by the PU. After the connection is set up, the SU needs to monitor the spectrum periodically to avoid interference to the PU system. When the PU appears, the SU has to stop communication or jump to a different channel. Both SU and PU are assumed to have a transmitter and a receiver for each node. There are two kinds of interference for SU in DSA systems, intra-system interference and inter-system interference. Intra-system

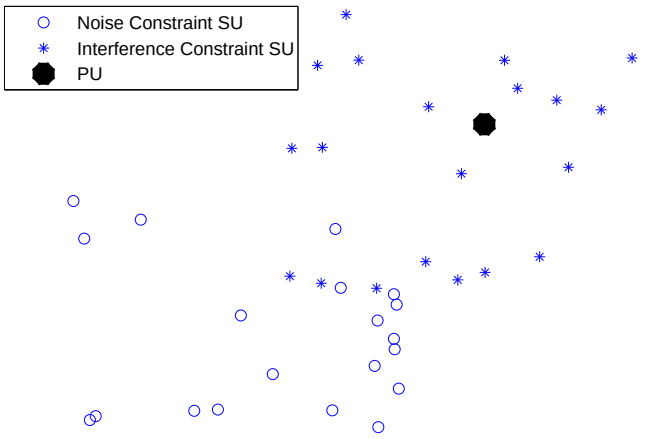

Figure 6.1: The system scenario.

interference is caused by the increase of the SU node density or reduced distance among links. Inter-system interference refers to the interference from the PU link to SU link. In this chapter, we only consider inter-system interference.

At the SU receiver, the desired signal power and interference power are assumed to be larger than the noise power, that is $S > N$ and $I > N$. The received signal is degraded by Additive White Gaussian Noise (AWGN). It is also assumed that SU receivers have no mobility issues and are sparsely distributed within an area. The interference measurement of the connection setup remains the same as in IEEE 802.22 standards. The proposed approach is designed for the scenario when the SU already occupies the channel and monitors the spectrum continuously to detect the emergence of the PU.

A frame structure is shown in Figure 6.2. SU 1 and 2 have already found an available channel and set up a communication link. When a PU appears during their communication, the SU can detect the presence of the PU during the sensing period at the end of each frame in the traditional interference detection approach. The collision period is shown as collision region 1. If the SU can detect the PU at the receiving phase, the collision between PU and SU can be reduced to collision region 2, as shown in Figure 6.2. SUs can also shift to another available channel more quickly if SU 2 informs SU 1 with a pre-defined message.

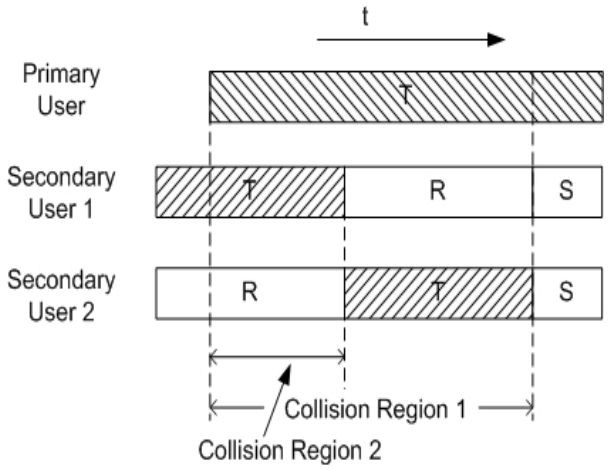

Figure 6.2: The diagram of one frame. T: Transmission phase; R: Recieving phase; S: sensing phase.There is always one secondary user monitoring the presense of primary users with the proposed approach.

6.3 Algorithm Development

The detection of primary users during communication requires that the receiver of secondary users can detect the signal from the primary user with the presence of the desired secondary signal. If the received signal of the primary user is considered as the interference to the receiver of secondary users, the problem can be formulated to detect interference with the presence of the desired signal.

6.3.1 Problem Formulation

The possible combinations of the three components are shown in Figure 6.3. The BER information is monitored during the communication phase and can be used as the first feature to classify the possible combinations. The signal to interference and noise ratio $SINR$ is related to the signal to noise ratio SNR, interference to noise ratio INR, and interference to signal ratio ISR as

$$\begin{aligned} SINR &= SNR/(1+INR) \\ &= (1/SNR + 1/SIR)^{-1} \\ &= (1/SNR + ISR)^{-1}. \end{aligned} \tag{6.1}$$

When the BER is low, the desired signal can be subtracted from the receiving signal. Then

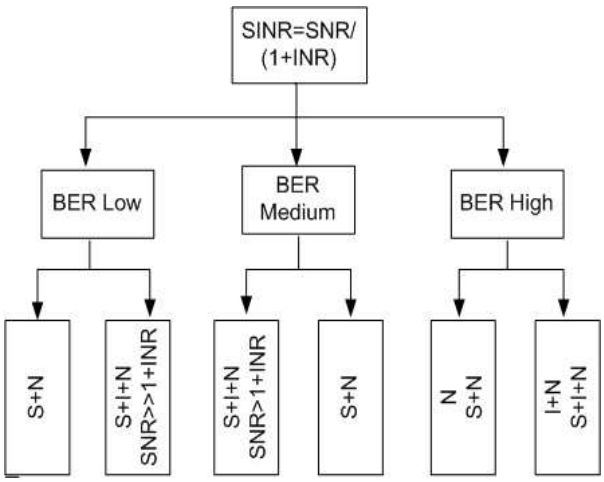

Figure 6.3: The problem formulation. The interference detection problem is classified as three categories based on BER range. The medium BER class constrains the detection performance.

the classification between $I+N$ and N can be performed to determine whether interference exists. This classification can be easily implemented since the variance and mean magnitude of the received signal with interference are different from that of the noise only case. When BER is high, the SNR of the $S+N$ case is low and can aggregate with the N only case. If both INR and ISR are high, it can be included with the $I+N$ case. These two patterns can also be classified with the above two features. A medium BER may be caused by two possible patterns, namely, $S+N$ and $S+I+N$. These two patterns are the hardest to determine, especially when ISR is low. The solution of this particular classification will constrain the performance of the whole classification problem, which this chapter is dedicated to solving.

An example of BPSK modulation through an AWGN channel is shown in Figure 6.4. $ISR = -5dB$ and $ISR = -20dB$ are the high BER case and low BER case, respectively, for $SNR = 10dB$. The S component can be either ignored or eliminated to help detect the interference. For $SNR = 10dB$ and $ISR = -10dB$ with medium BER, neither S nor I can be eliminated from the detection process. The relation between $SINR$ and BER may change with different modulation/coding schemes and diversity techniques. in practice, the BER region may need calibration, which is considered as part of the future work and is beyond the scope of this chapter.

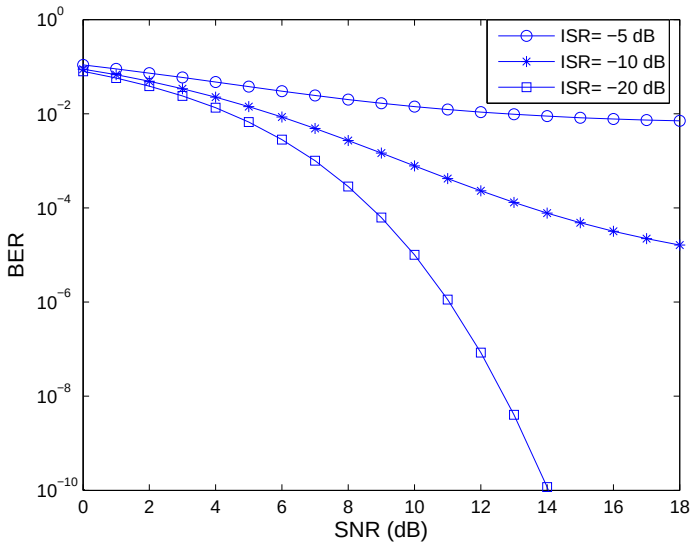

Figure 6.4: An example of BER classification for BPSK through AWGN channel. The BER classification may vary for different QoS requirement, modulation/coding scheme, and diversity technique in practice.

6.3.2 Classifier Design

From the above discussion, the worst case is to classify the S+N and S+I+N when ISR is low with a medium BER. This section focuses on how to design a classifier based on SVM technology to tackle this extreme situation and evaluate its performance.

SVM is an important invention in pattern classification and has several advantages compared with the traditional pattern recognition approaches in [104]. SVM has good general capability by balancing the tradeoff between accuracy and capacity of the classifier. The model complexity is independent of the input dimensionality in SVM. This approach relates to data regularization and convex optimization and the global optimum is guaranteed. Finally, the speed of the classifier can be very fast with shorter classifiers, or support vectors, for the classification process. However, the mislabeling issue of the training samples is the major disadvantage of SVM for its application in our problem. The decision hyperplane is not optimal once there are some mislabeling samples, which reduces the classification correctness. SVM is more complex than the previous interference detection methods. For the low BER and high BER cases, traditional interference detection methods may be sufficient to detect the interference. But for medium BER, pattern recognition has to be used, since previous methods have some limits in DSA scenarios, as discussed in the introduction.

SVM is originally introduced as a linear binary classifier when trying to separate hyperplanes with the largest margin. Suppose we have a training set $\{\mathbf{x}_i, y_i\}$, $i = 1, 2, ..., N$, and $\mathbf{x}_i \in \mathbf{R}^M$. The label value, $y_i = \{1, -1\}$, represents the class that the sample belongs to. If there exists

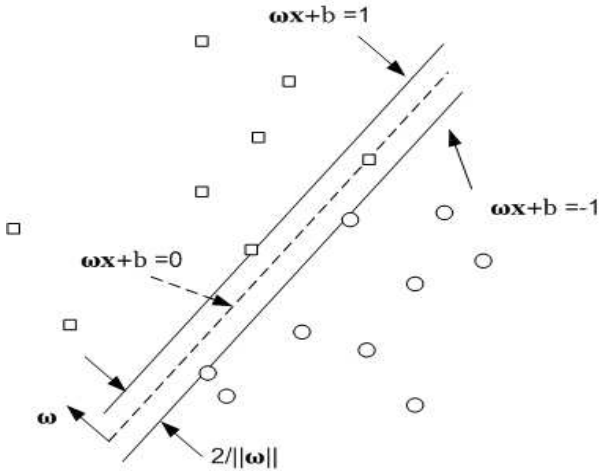

Figure 6.5: The principle of SVM. The classification margin depends on the distance between two support planes consisting of the support vectors.

a hyperplane that can separate the positive samples from the negative ones, the points that lie on this hyperplane satisfy

$$\mathbf{w} \cdot \mathbf{x}_i + b = 0 \tag{6.2}$$

where, w is the norm of the hyperplane, and b is a distance parameter of the hyperplane to the origin. The closest distance between this hyperplane to the nearest positive/negative sample can be defined as the margin. The classification can be summarized as

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geq 0, \quad \forall i. \tag{6.3}$$

The training process of SVM is to find the $\mathbf{w}$ and b that maximize the margin $2/||\mathbf{w}||$ with training samples as shown in Figure 6.6. This problem can be formulated as a convex optimization problem

$$\begin{aligned} &maximize && 2/||\mathbf{w}|| \\ &subject\ to && y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geq 0, \quad \forall i \end{aligned} \tag{6.4}$$

and its dual formulation is given by

$$L_D = \sum_i \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j \mathbf{x}_i \mathbf{x}_j. \tag{6.5}$$

The training phase is used to determine a multi-dimensional curve to fit the training samples. When the two patterns are nonlinear separately, a soft margin ξ_i as well as a parameter C

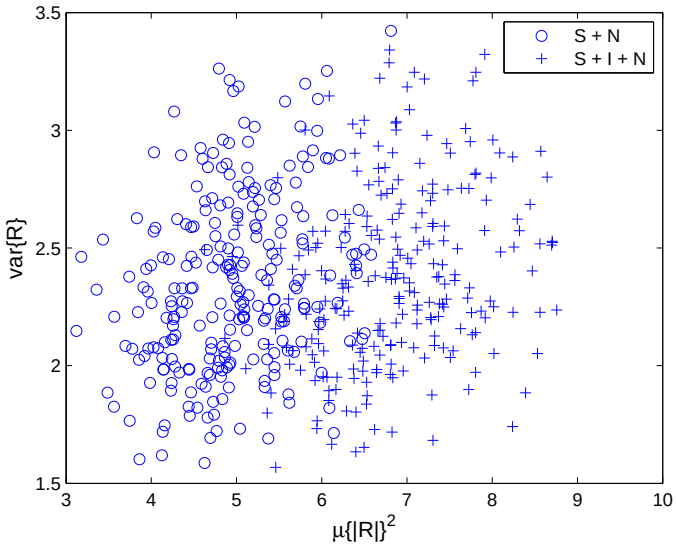

Figure 6.6: The patterns normalized by noise power with $\sigma_p = \sigma_s = 1$, $SNR = 5dB$ and $ISR = -15dB$. The two patterns have a better chance to be separated with two-dimensional classifier as compared with a one-dimensional power separation approach.

representing the penalty of mismatching is introduced. Then, the objective function is changed to

$$minimize \quad ||\mathbf{w}||/2 + C\sum_i \xi_i. \tag{6.6}$$

The classifier design needs to make a choice of kernel function and initial parameter in order to solve the discussed problem. Kernel functions are commonly used to transfer the data to a higher dimension according to the Cover's Theorem and Mercer's Theorem in [110]. The choice of the kernel function is the most challenging design problem in an SVM classifier. We choose the Gaussian radial basis function (RBF) as the kernel function since we are trying to include the training samples within the smallest ellipsoid. The RBF function can be represented as

$$K(\mathbf{x}, \mathbf{y}) = e^{-||\mathbf{x}-\mathbf{y}||^2/2/\sigma^2}. \tag{6.7}$$

The dual problem is then changed to

$$L_D = \sum_i \lambda_i - \frac{1}{2}\sum_{i,j} \lambda_i \lambda_j K(\mathbf{x}_i \mathbf{x}_j). \tag{6.8}$$

Two initial parameters of the SVM classifier need to be determined, σ of the RBF and the penalty of mismatching C. The choice of these parameters will impact classifier performance and is normally an empirical design problem. Here, we choose the variance of the

Table 6.1: Proposed Approach For PU Detection

1: Spectrum sensing in connection phase
2: If Interference detected
3: Store the samples as training samples for "+1"
4: Shift to another channel.
5: else
6: Store the samples as training samples for "-1"
7: Start communication and train the classifier
8: Switch (Interference is found)
9: Case: 0
10: Monitor BER, $var\{R\}$, and $E\{R\}^2$.
11: Test these features with trained classifier.
12: Case: 1
13: Stop communication
14: Store the samples as training samples for "+1"
15: Break; Go back to Step 1.
16: end
17:end

training samples' features, $var\{\mathbf{x}\}$, as the value of σ, and $C = 1/N$. N is the number of training samples. Lagrangian SVM is exploited as the solution to determine $||\mathbf{w}||$ and b by [123]. LSVM relaxes the objective function of the original SVM problem in order to use the Sherman-Morrison-Woodbury identity, which will reduce the computation complexity of inverting the matrix, as in [116].

The proposed approach is shown in Table 1. We represent the class without interference and the class with interference with -1 and +1 respectively. Three parameters of the received signal R, BER, $var(R)$ and $E\{R\}^2$, are selected as the features for classification. Each storage operation will store these three features and its labels, e.g., -1 and $+1$. The training samples are collected both during the connection phase and the communication phase. The classifier will learn its interference environment when more training samples are collected, and the relative location of the PU and SU do not change, such as in IEEE 802.22 systems.

6.3.3 Performance Analysis

The SVM approach uses the training samples to determine an optimal decision boundary. The classification margin is determined by both the center and the spread of each cluster. Those two metrics can be represented by the sum of the distance and the sample variance, both of which are functions of SNR and ISR. In this section, we will analyze the impact of SNR and ISR on classification performance.

The interference to noise ratio INR has the relationship with SNR and ISR as

$$INR = SNR \times ISR = \alpha \times \beta. \tag{6.9}$$

Since ISR represents the relative power of the interference compared with the signal power, we will evaluate the performance of the proposed approach as a function of SNR and ISR. To simplify the analysis, assume that BPSK is the modulation scheme for both the SU and PU. Channel gains G_s and G_p for the SU link and the interference link are uncorrelated and have a Rayleigh distribution with parameters σ_s and σ_p respectively. The relation between BER and SINR is then determined by the modulation scheme under the Rayleigh fading channel. The received signal can be represented as

$$R = \begin{cases} G_s S + G_p I + N, & PU. \\ G_s S + N, & No \;\; PU. \end{cases} \tag{6.10}$$

The variance and mean-square magnitude of the receiving signal can be represented as follows

$$var\{R\} = \begin{cases} \frac{4-\pi}{2}\sigma_s^2 \alpha P_n + \frac{4-\pi}{2}\sigma_p^2 \alpha\beta P_n + P_n, & PU. \\ \frac{4-\pi}{2}\sigma_s^2 \alpha P_n + P_n, & No \;\; PU. \end{cases} \tag{6.11}$$

and

$$\mu\{|R|\}^2 \leq \begin{cases} (\sqrt{\pi/2\alpha}\sigma_s + \sqrt{\pi/2\alpha\beta}\sigma_p)^2 P_n, & PU. \\ \pi/2\alpha\sigma_s P_n, & No \;\; PU. \end{cases} \tag{6.12}$$

where P_n is the noise power. The sum of the distance along two dimensions between the centers of two patterns normalized by the noise power is upper bounded by

$$\begin{aligned} |d|_{\sum} &= (4-\pi)/2\sigma_p^2\alpha\beta + (\pi)/2\sigma_p^2\alpha\beta + \pi\sigma_s\sigma_p\alpha\sqrt{\beta} \\ &= 2\sigma_p^2\alpha\beta + \pi\sigma_s\sigma_p\alpha\sqrt{\beta}. \end{aligned} \tag{6.13}$$

The above equation indicates that the value of $|d|_{\sum}$ relates to the noise floor, channel conditions, SNR and INR. Better classifier performance may be achieved with a larger value of $|d|_{\sum}$ as shown in Figure 6.7. Severe fading with larger σ_p may actually help classification.

The cluster size is determined by the sample variance. The expected variance can be estimated with the sampling variance distributions as in [111]

$$\begin{aligned} \sigma_v^2 &= var\{var\{R\}\} \\ &= \frac{(M-1)^2}{M^3}\mu_4 - \frac{(M-1)(M-3)}{M^3}\mu_2, \end{aligned} \tag{6.14}$$

and

$$\sigma_m^2 = var\{\mu\{R\}\} = \mu_2/M. \tag{6.15}$$

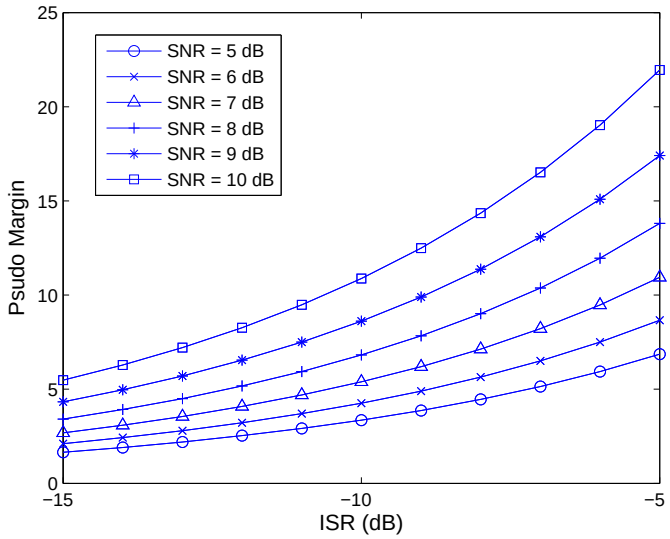

Figure 6.7: The pseudo margin as a function of ISR and SNR with $M = 100$, $\sigma_p^2 = \sigma_s^2 = 1$.

where, μ_4 and μ_2 are the forth-order central moment and 2nd-order central moment respectively. M is the number of samples that generate the two statistic parameters. It is easy to show that μ_4 can be expressed as

$$\mu_4 = \begin{cases} \sum \mu_{4,i} + 6(\mu_{2,S}\mu_{2,I} + \mu_{2,S}\mu_{2,N} + \mu_{2,I}\mu_{2,N}), \\ \mu_{4,S} + \mu_{4,N} + 6\mu_{2,S}\mu_{2,N}, \end{cases} \tag{6.16}$$

and

$$\mu_2 = \begin{cases} \mu_{2,S} + \mu_{2,I} + \mu_{2,N}, & PU. \\ \mu_{2,S} + \mu_{2,N}, & No \quad PU. \end{cases} \tag{6.17}$$

where, $i \in \{S, I, N\}$. $\mu_{4,i}$ and $\mu_{2,i}$ are the 4th-order and 2nd-order central moment for signal, interference, and noise samples respectively. Those moments can be expressed as

$$\mu_{4,i} = \begin{cases} \frac{32-3\pi^2}{4}\sigma_s^4\alpha^2 P_n^2, & i = S \\ \frac{32-3\pi^2}{4}\sigma_p^4\alpha^2\beta^2 P_n^2, & i = I \\ 3P_n^2, & i = N. \end{cases} \tag{6.18}$$

and

$$\mu_{2,i} = \begin{cases} \frac{4-\pi}{2}\sigma_s^2\alpha P_n, & i = S \\ \frac{4-\pi}{2}\sigma_p^2\alpha\beta P_n, & i = I \\ P_n, & i = N. \end{cases} \tag{6.19}$$

Combining Equation 6.14-6.19, the spread of each cluster can be determined. The pseudo margin of the classification normalized by the noise power can be defined as the sum of the distance subtracted by the sum of the variances. That is

$$D = |d_{\sum}| - \sum_k (\sigma_{m,k} + \sigma_{v,k}) \tag{6.20}$$

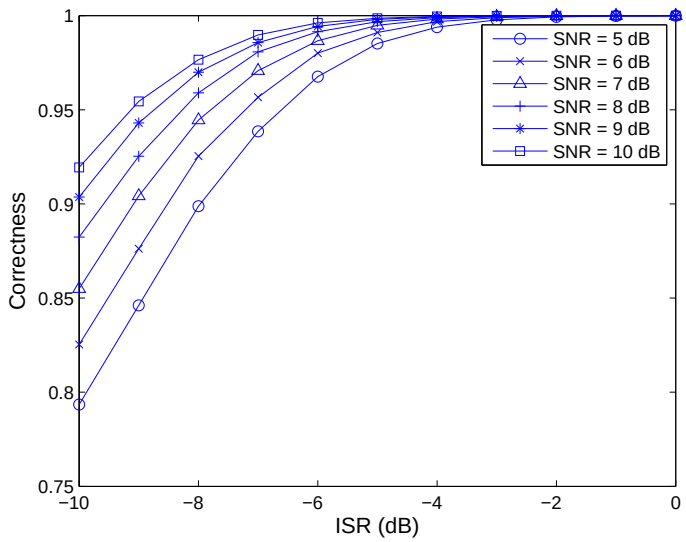

Figure 6.8: The classifier performance with $5dB \leq SNR \leq 10dB$, $-10dB \leq ISR \leq 0dB$, $\sigma_p^2 = 1$, and $\sigma_s^2 = 0.1$.

where, k is the number of patterns, which equals to two in Figure 6.6. D is an approximation of the classification margin since the shape of the two clusters is not a sphere and the maximum spread instead of the variance of spread determines the two decision boundaries. The larger D indicates possible improved classification performance. Notice that this is an approximation of the classification margin because of the approximation of equation 6.20. Figure 6.7 shows the impact of SNR and ISR on D. It can be predicted that the increase of SNR and ISR will improve the classification performance.

6.4 Simulation

Classifier performance is limited by the extreme case discussed in Section III. The performance of detection of interference with the presence of a signal is largely limited with the medium BER case, which can be represented with two possibilities: medium SNR with low ISR or high SNR with medium ISR for the S+I+N class. In this section we will demonstrate a simulation example and evaluate the performance of the proposed approach for this scenario.

In this simulation, $N = 100$ samples are used as the training patterns and 4000 testing samples are used to evaluate classification performance. Each sample is generated by $M = 100$ data. Each figure is generated with 1000 experiments. Since the BER is a function not only of these two parameters but also of the modulation and coding scheme, some particular range of SNR and ISR is assumed to control the BER in the medium range in

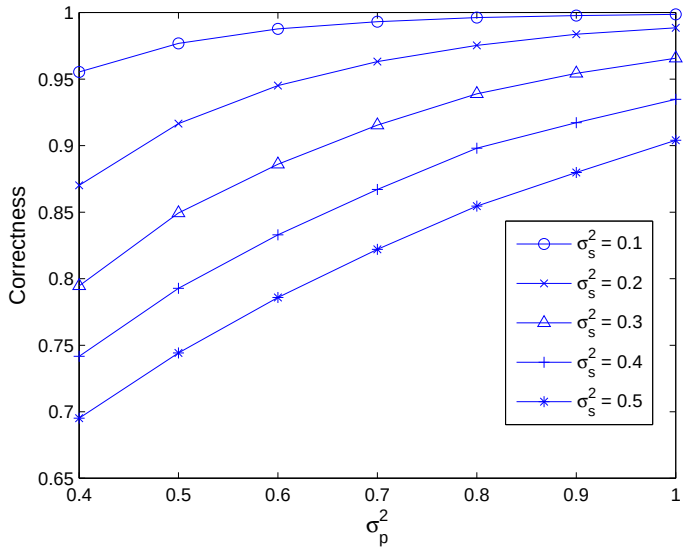

Figure 6.9: The classifier performance with $0.1 \leq \sigma_s^2 \leq 0.5$, $0.4 \leq \sigma_p^2 \leq 1$, $SNR = 10dB$, and $ISR = -5dB$.

the simulation. The specific BER thresholds may need to be calibrated for a real system.

The performance of the proposed approach in the case with medium BER is shown in Figure 6.8 and Figure 6.9. Figure 6.8 shows that the performance of the medium BER scenario is limited by the low ISR case. Higher SNR can help achieve better classification performance. The correctness of classification can achieve more than 76% when $ISR \geq -10$ dB if the SU link and PU link have large difference as indicated in Figure 6.8. For the $INR > 0$ case, the correctness can be more than 90%. Figure 6.9 shows the impact of fading on the classification results. The recognition correctness is an increasing function of σ_p^2 and a decreasing function of σ_s^2, which indicates that different fading between the SU and PU link may help classification.

The following simple example is given to illustrate the advantage of the proposed PU detection approach. There is $T = 120$ s multi-media data to be exchanged between the secondary user link. The traffic is symmetric. A super frame consists of β transmission and receiver slots paired with $T = 10$ ms and $R = 10$ ms. Each super frame will have one sensing slot (idle time) of $S = 2$ ms. If a super frame has $\beta = 1$, as shown in Figure 6.2 in Section II, there is a total of 6000 super frames for this data transmission task. Depending on the value of β, the number of super frames may vary for this task. A primary user appears during the communication of the secondary users. The presence time point of the primary user is assumed to follow a uniform distribution within an interval of T. The detection probability of the proposed method is P_d. It is assumed the secondary user's receiver can detect the presence of the primary user with a probability of 1 within the idle time. If the secondary receiver does not detect the primary user during the communication, it will be detected

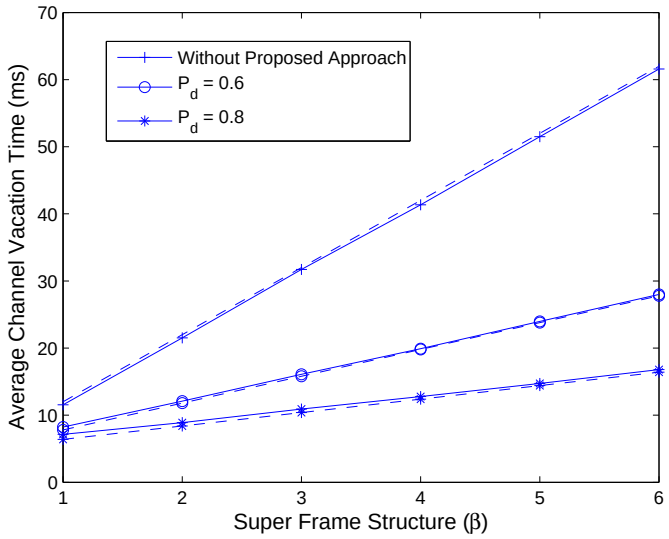

Figure 6.10: The average channel vacation time as a function of β and P_d. Solid line represents the simulation results. Dash line represents the theory results. The channel vacation time can be reduced by 33% (β=1,P_d=0.6) to 73% (β=6,P_d=0.8) with the proposed approach.

within the idle time at the end of each super frame with a channel collision time of ($T_{c,1}$), as collision region 1 in Figure 6.2. If a primary user is detected by the secondary user during the communication with the proposed method, the receiver side will issue a channel vacation message to its peer. This will cause a channel collision time of $T_{c,2}$ as collision region 2 in Figure 6.2. Thus the channel vacation time can be shown as

$$T_{vacation} = (1 - P_d)T_{c,1} + P_d T_{c,2}. \tag{6.21}$$

It can be easily shown that the average value of $T_{c,1}$ and $T_{c,2}$ is

$$\begin{aligned} E\{T_{c,1}\} &= \frac{\beta(T+R)}{2} + S. \\ E\{T_{c,2}\} &= \frac{(T+R)}{4}. \end{aligned} \tag{6.22}$$

The usage of the idle time is defined as

$$\eta = \frac{S}{\beta(T+R)+S}. \tag{6.23}$$

The simulation results in Figure 6.10-6.11 show that the proposed approach can reduce both the average channel vacation time and the usage of the sensing slot or idle time. The performance improvement depends on the super frame structure and the detection probability of

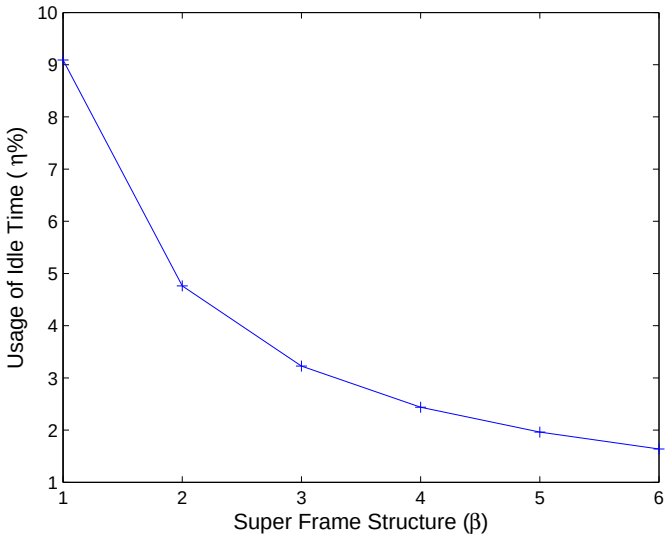

Figure 6.11: The usage of idle time as a function of β. Larger values of β reduce the usage of idle time.

the proposed method. To minimize the average channel vacation time, traditional methods assign an idle time at the end of each frame with $\beta = 1$ which has the largest η, as shown in Figure 6.11. Although a larger value of β can increase the spectrum efficiency and reduce the usage of idle time, it also increases the average channel vacation time, as shown in Figure 6.10. The proposed approach can suppress this increase of channel collision if a sufficient P_d is available. The average channel vacation time can be reduced by 33%-73% in this particular task depending on the value of β and P_d.

6.5 Conclusion

This chapter proposed a SVM-based interference detection approach to detect PU during the communication of SUs, or wireless nodes in WDCNs. Idle time can be used to train the SVM during the connection initialization phase. During the communication phase, WDCNs will continuously receive the signal and detect the interference, or the primary user's signal, beneath the desired secondary signal. The proposed approach can help reduce the usage of the idle time and the channel vacation time for WDCNs. Future research includes consideration of the mobility of the SU and/or the PU, with more sophisticated modulation and coding schemes.

Chapter 7

An Example of WDCN on CORNET

In this chapter, we will introduce a demo of WDCNs developed on the Virginia Tech COgnitive Radio NEtwork Testbed (CORNET). Some of the proposed concepts in previous chapters can be demonstrated with the software developed in this chapter.

7.1 Background

7.1.1 CORNET

CORNET is a collection of 48 radio nodes deployed throughout a research building on the Virginia Tech campus. All the radios are reconfigurable based on software defined radio (SDR) technologies. The architecture of CORNET is shown in Figure 7.1. There are 12 nodes on each floor. Each node consists of one RF front-end, USRP2, and a host server. Each USRP has a daughter board and a mother board. The mother board communicates with the host server through Ethernet. Each server has a Quadcore 2.13GHz CPU and 3GB RAM in order to support signal processing for the base band signal. These 48 servers form a server cluster through Ethernet connections and switches. The whole testbed is managed by five management servers which can provide the remote control and the access capabilities to the testbed through a secure shell from anywhere in the world.

The mother board of USRP2 contains a Virtex Spartan3 FPGA. The receiver chain has a 14-bit analog-to-digital converter with 100 M samples/second speed and the RF chain has a 16-bit, 400 M samples/second digital-to-analog converter. This structure provides a capability of up to 50 MHz bandwidth with 16 bit/complex samples. Each mother board can support up to two daughter boards. The current daughter board is a wide-band transceiver which supports the frequency range from 50 MHz to 2.2 GHz. It has a maximum transmission power of 20 dBm and a noise figure of 5 dB. A custom designed daughter board using Motorola's RFIC is under development. RFIC is a multi-band direct conversion RF chip

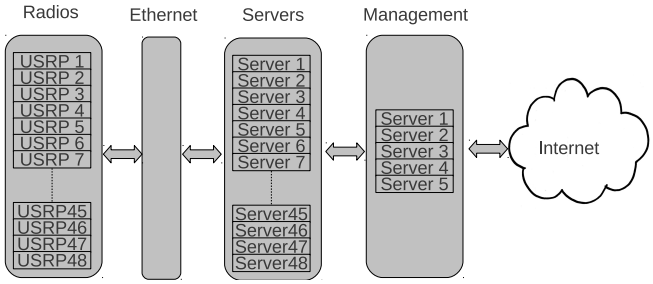

Figure 7.1: The system architecture of CORNET.

with 90 nm CMOS technology developed by Motorola Research Lab [139]. Its carrier is tunable from 100 MHz to 2 GHz with variable bandwidth of 10 kHz to 20 MHz. In the future, low-power mobile nodes will also be available.

Both GNU radio and Open Source SCA Implementation-Embedded (OSSIE) are installed in each host server. GNU radio is an open source software for SDR development, which provides libraries for radio reconfiguration and control. OSSIE follows the Software Communication Architecture (SCA) developed for Joint Tactical Radio System (JTRS) for military software defined radios. The programs developed with these two are not generally compatible.

All host servers and management servers are mounted in a 45U server rack cabinet located in a control room. CAT6 cables are used for the connection between the USRPs' ethernet port and the servers' ethernet port. It is easier to maintain and manage the servers in a centralized architecture than in a distributed architecture. CORNET provides an open research and development platform for medium scale cognitive radio networks research such as distributed CR operation, CR network security, dynamic spectrum access and spectrum coordination. The 3-D distribution of CORNET nodes is shown in 7.2. Detailed information about CORNET can be found in [140].

The demonstration of wireless distributed computing with CORNET has the flexibility that a network interface card (NIC) can not provide. The radio parameters are reconfigurable and multiple PHY and MAC protocol may be developed with the same software tool. Different types of WDCNs can be demonstrated easily by replacing the computing applications. The server cluster has the potential for data mining. Mobile nodes may be integrated into the demo in order to study the impact of heterogeneous devices.

However, there is a limitation of the node structure currently used by CORNET. The digitized base band signal from USRPs is processed in the user domain of the host server. All the decisions and controls are done at the general purpose processor (GPP) of the host server. Although this structure has the best flexibility in terms of software/hardware reconfiguration,

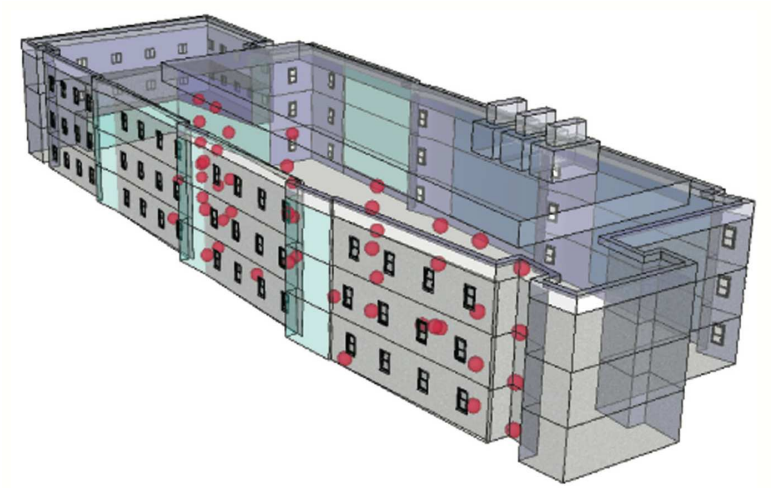

Figure 7.2: Node Distribution of CORNET. (Reprinted with permission from paper [140]. copyright ©IEEE 2010)

it also introduces latency and jitter issues [93]. It is highly possible that GPP is interrupted by other processes during its processing of USRP data. Kernel level delay measurements indicate a latency range from 600 μs and 15 ms, which is too high for a contention-base protocol. When the number of nodes increases, contention increases due to the high latency and jitters which leads to failures in decision making and multiple access control (MAC). A possible solution is to split the functionalities of the MAC protocol between the USRP and host server, which requires programming FPGA and DSP on a USRP [142]. Another possible solution is to use a TDMA protocol with guarding time slots. In this demo, we will choose the second solution.

7.1.2 Scenario

The computing task in this demo is a video compression with high compressing ratio, such as 40 times. A H.264 encoder is used for this purpose [141]. A service request node (SRN) has a certain number of frames to be compressed. It has the option to compress by itself or distribute to service nodes through wireless links over USRPs. A SRN can distribute the video frames evenly without considering the channel condition or unevenly with consideration of the channel heterogeneity. These two strategies may have different impacts on the system delay and power efficiency. This demo shows that considering channel heterogeneity is the key to building power efficient and robust WDCNs.

Among the links between a SRN and SNs, some have an outage probability of zero and the others have an outage probability of p_o. This difference in outage probability is called channel heterogeneity in this chapter. The workload allocation considering channel heterogeneity calculates the number of frames that each SN should compress according to the method pro-

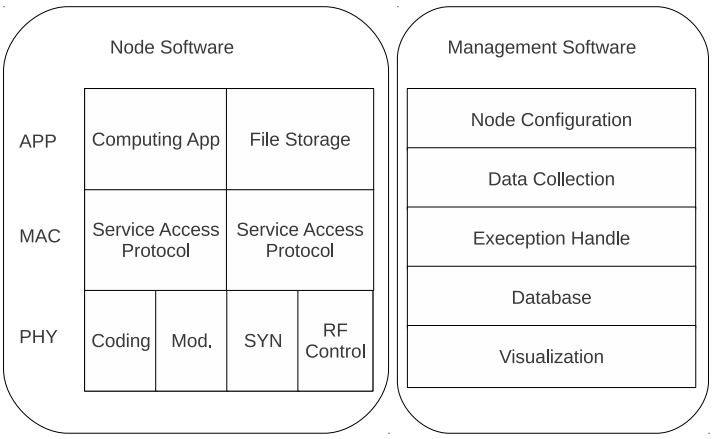

Figure 7.3: Software architecture of WDC demo.

posed in Chapter 2. This will reduce both the mean and the variance of the total computing time compared with the case frames are allocated evenly among SNs, which is the method without considering channel heterogeneity. The demo developed in this chapter is expected to show this result.

7.2 Software Design

The software architecture is shown in Figure 7.3. Each node runs a node software which integrates the communication process and computing process. The physical layer is based on the Liquid Radio DSP library [143] with OFDM capabilities. The reconfigurable parameters include: frequency, bandwidth, sub-carrier space, modulation, coding, transmission power, and packet length. The MAC protocol uses the TDMA protocol based on the polling and reporting mechanism. Service request nodes poll the status of service nodes and make decisions about workload allocation and transmission scheduling based on reports from the service nodes.

The MAC protocols for service access and service provision are different since service access does not require activating the computing process and accessing the file system. During service provision, H.264 video compressing is the computing application. All the compressed data collected by a SRN can be de-compressed and compared with the original raw data. It is easy to replace video compression with other desired computing tasks as required. The node software is written in the C language. A UHD driver is required in order to control USRPs.

A management software is also implemented to coordinate the nodes configuration, data collection, and visualization. It is cumbersome to log in to every CORNET node and make the configuration. Therefore, the *Expect* language is used to automatically log in and configure multiple CORNET nodes. This network management software controls and monitors the nodes through Ethernet and is deployed on one control node while node software is deployed on each CORNET node.

The node software for each CORNET node can be divided into two parts: the software for service request nodes and the software for service nodes. The flow charts for these two are shown in Figure 7.4 and 7.5. During the service access phase, the SN will initialize the configurations of radios and create two processes, one for communication and the other for computing. The SRN will send out service request messages. Once it gets responses, the SRN will store the node ID of the SNs into a table and distribute the workload to SNs based on the proposed resource allocation method.

Then the service provision phase follows and the computing process on SNs starts compressing the raw data. The SRN can poll the status of the SNs periodically during service provision phase. The communication process on the SN responds to these inquiries with its own status. Once the data compression is finished, the SN will report to the SRN and request data transmission. The SRN will make a decision about the priority of the transmission based on its record about the status of the SN nodes. Once it receives the compressed data from one SN node successfully, the SRN will remove the ID of this service node from its polling table and inquire the computing status of the rest of the nodes, which will speed up the response time. Once all the compressed data is received, the SRN will record the statistics of the experiment. It can also de-compress the raw data and play it back.

The node software encapsulates the service access and service provision into four functions: $wdc\_control\_tx$, $wdc\_control\_rx$, $wdc\_txbuffer$, and $wdc\_rxbuffer$. $wdc\_control\_tx$ and $wdc\_control\_rx$ provides the service access capabilities. The other two provide service provision capabilities. In this way, the performance of these two phases can be investigated independently.

Multi-process programming is used to implement the functions of the SNs. The communication process and computing process are affiliated to separate CPU cores in order to run independently without interfering with each other. The two processes can communicate with each other about their status using a signal library. When an exception occurs, the exception handling module of the network management software will kill the process in order to re-install the normal operation. Network management software can also enable the debug mode of the node software which helps protocol modification in the future.

A storage file can be used to decouple the communication process and computing process. The SN creates two processes responsible for communication and computing, respectively. When the SN responds to the SRN, the most important status the SRN needs is whether the computing process is finished. Instead of using the signal library to update the status between the two processes, the existence of a compressed data file can be used as a control

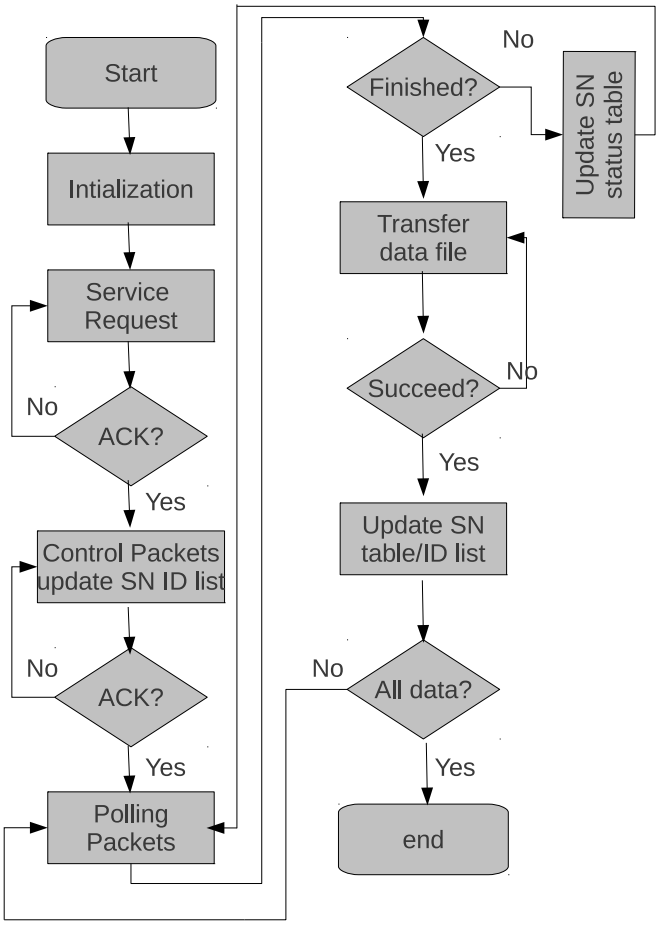

Figure 7.4: Flow chart of SRN software.

message between processes. The encoder writes the compressed data into a temporary file.

Once the encoder compresses all the raw data assigned to it, it will rename the data file with a new file name consisting of the node ID and a file suffix of "*snr*". The communication process checks the existence of this file in order to determine whether the computing process is finished. Although this may increase the delay compared with the implementation with a signal library, it has no impact on the communication process when replacing the video compressing application with any other computing applications, which makes it much easier to demo other WDCN implementation in the future.

Figure 7.6 shows the format for one packet. The first 128-bits are used for the purpose of synchronization. The packet header with 8 bytes is reserved for control information and protected with the strongest modulation and coding scheme. The definition of the first 8

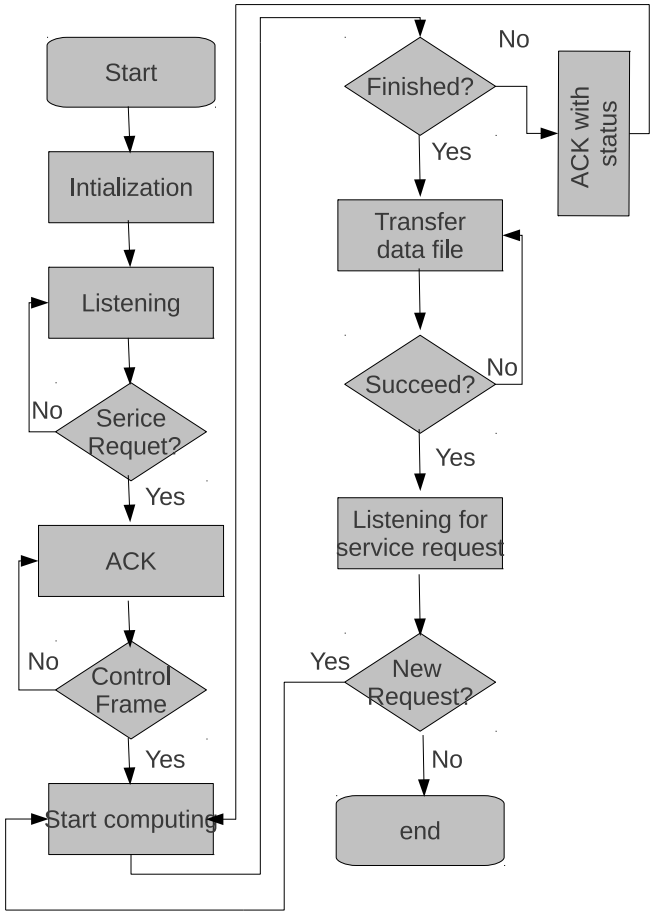

Figure 7.5: Flow chart of SN software.

bytes header is shown in Table 7.1. Source ID and Destination ID are used to identify the wireless link. Packet Type has four values namely for the control packets, data packets, and their ACKs. The number of packets and payload information help buffer allocation and decisions about the status of the data transmission. The payload length can vary according to different requirements.

7.3 A Demo

One cluster with a SRN and multiple SNs is used to demo the concepts of WDC, especially the channel impact on the delay performance. Due to the limitation of latency and jitters, the real-time channel estimation is currently impossible. A channel emulator has been built

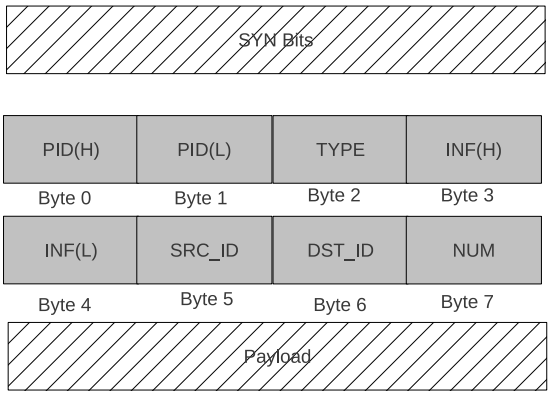

Figure 7.6: Format of one packet for WDC demo.

Table 7.1: Packet Head For One WDC Packet

$Byte0$	Packets ID	$Byte4$	Payload Info.
$Byte1$	Packet ID	$Byte5$	Source ID
$Byte2$	Packet Type	$Byte6$	Destination ID
$Byte3$	Payload Info.	$Byte7$	number of Packets

into the software to demo the impact of channel heterogeneities on the performance. Due to different outage probabilities, the number of re-transmissions may vary, which leads to a random communication delay. When the outage probabilities vary from link to link, the heterogeneous wireless channels are emulated.

7.3.1 Illustrations

The debug mode is shown in Figure 7.7. This figure is captured during the service provision phase. The SRN, with $Node\_ID = 10$, on the left is polling the status of SNs according to its SN_List. SNs, with $Node\_ID \in \{1, 2, 3, 4\}$, respond with their computing status. The SNs windows throw the exceptions as needed, such as a cyclic redundancy check error for SN 1. Each SN has a counter to show the number of inquiries that they receive. Both the SRN and SNs initialize timers at proper locations in the software in order to record communication delay and computing delay, which will be used to analyze the performance later.

Figure 7.8 shows the over-the-air waveform captured by another CORNET radio with a function from GNU radio [144] when the demo is running. The waveform has the shape of

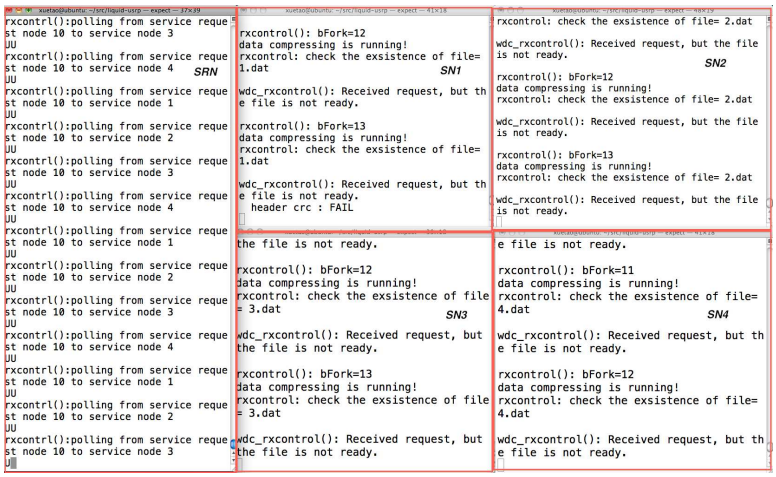

Figure 7.7: Debug Mode of the WDC demo captured during service provision phase.

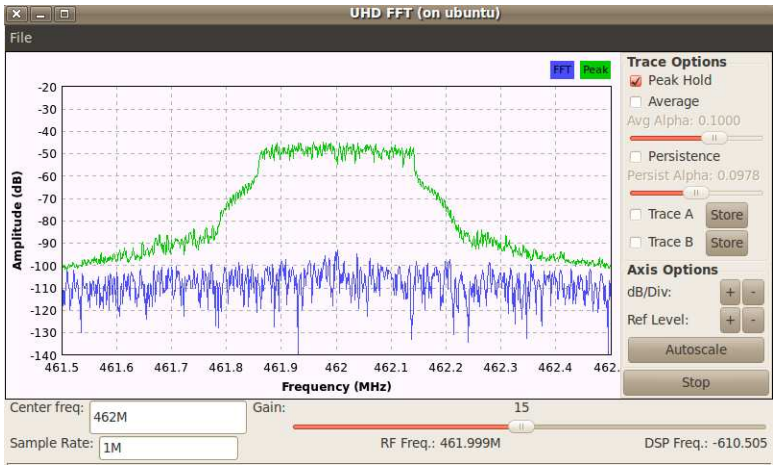

Figure 7.8: Over-the-air (OTA) waveform.

an OFDM signal with bandwidth of $B = 300\,kHz$ and center frequency of $f_c = 462MHz$. These RF parameters can be changed as needed. Currently, CORNET nodes support a frequency range from 50 MHz to 2.2 GHz.

It should be noted that the amplitude in Figure 7.8 is a relative one since all the RF chains of the USRPs are not calibrated. The daughter board needs to be calibrated as follows in order to have the absolute value of power in dBm. First, the thermal noise floor should be calculated and calibrated. Then the gain errors of the RF chain and thermal noise can be de-coupled with an injected reference RF signal, which will convert the unit of received power from dBi to dBm.

7.3.2 Workload Allocation Algorithm

Workload allocation methods considering channel heterogeneity follow the concepts in Chapter 2. If there are N service nodes in a demo, half of them, $n \in \mathbb{S}_b$, have bad channels with an outage probability of p_o while the other half, $n \in \mathbb{S}_g$, have good channels with an outage probability of 0. For each service node, n, the compressing time $\tau(n, m)$ is a function of the number of frames, m. The communication time, $\tau(n, p_o, k)$, is a function of the outage probability p_o and the number of packets k.

If each transmission costs T_s seconds, the average execution time $T(f)$ for each possible workload allocation scheme f can be represented as follows

$$\begin{aligned} T1(f) &= \sum_{n\in\mathbb{S}_b} \frac{N}{2(1-p_o)} T_s k(n) + \tau(n, m) \\ T(f) &= \max\{T1(f), \tau(n, \frac{2M}{N} - m)\} + \sum_{n\in\mathbb{S}_g} \frac{N}{2} T_s k(n) \end{aligned} \tag{7.1}$$

The optimal workload allocation policy f^* can be found as

$$f^* = \underset{f\in\mathbb{F}}{\operatorname{argmin}}\, T(f) \tag{7.2}$$

where $\mathbb{F}$ is the set of workload allocation schemes. The reference workload allocation is evenly distributed as $m = M/n$ frames to each SN.

7.3.3 An Experiment

In this demo, the workload is a video with $M = 12$ frames. SN 2 and SN 4 have an outage probability of zero while SN1 and SN3 have an outage probability of p_o. A larger p_o indicates greater difference between bad and good links, or larger channel heterogeneity for the whole network. The method considering channel heterogeneity allocates frames with equation (7.2). For the method without considering channel heterogeneity, $m = 3$ for all $n \in \mathbb{S}$. This demo intends to show the impact of this channel heterogeneity on delay performance and workload allocation methods.

Figure 7.9 shows the analysis results for average delay performance. It compares the results between the reference workload allocation method and the method considering channel heterogeneity calculated as equation (7.1) and (7.2). When p_o increases, the difference between good and bad links increases and will have more impact on delay performance. When p_o is smaller than a threshold around, $p_o = 0.75$, channel heterogeneity has no improvement on delay performance. In other words, there is no need to consider the channel heterogeneity when $p_o \leq 0.75$ in this demo setup.

Figure 7.10 shows the optimal workload allocation for $n \in \mathbb{S}_b$. $m = M/n = 3$ frames is the reference workload allocation scheme. When $p_o \leq 0.75$, the method considering channel

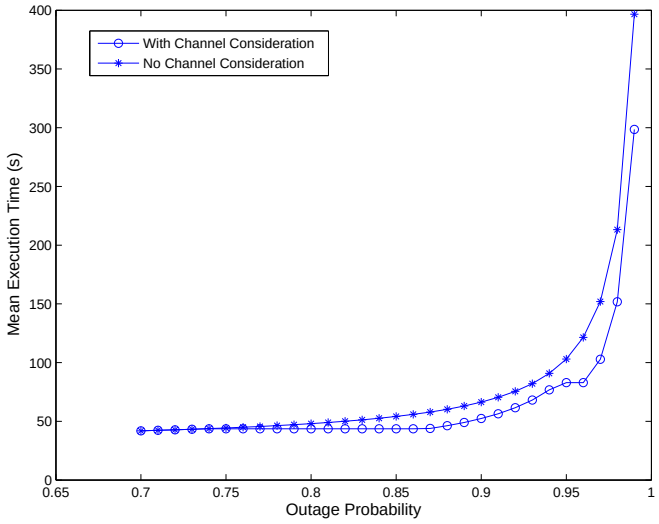

Figure 7.9: Estimated average delay performance as a function of channel heterogeneity and workload allocation methods.

heterogeneity is coincident with the method without considering channel heterogeneity. It has a staircase shape similar to the results in Chapter 4 since the minimum divisible unit for the workload is one frame in this demo.

Table 7.2 and Figure 7.11 show the experimented data collected from CORNET. They show the impact of channel heterogeneity on the mean and the variance of the execution time during the service provision phase. There are four test cases of SN 1 and SN3 having an outage probability of $p_o \in \{0.8, 0.85, 0.9, 0.95\}$, respectively. When the outage probabilities of SN1 and SN3 increase from 0.8 to 0.95, the channel heterogeneity increases. The method without considering channel heterogeneity distributes workload evenly among four SNs. The method with channel consideration allocates the frames according to the results in Figure 7.10. As expected, the method with channel consideration will reduce both the mean and the variance of the execution time. Both the mean and variance increase when the channel heterogeneity increases.

7.4 Summary

An implementation and a demonstration of WDCNs are shown in this chapter. A software dedicated to WDC based on SDR technologies is designed and implemented. The resource allocation ideas in Chapter 2 and Chapter 4 are demonstrated. The experimented data matches with the analysis results of the delay performance. The results of our experiments also demonstrated that channel heterogeneity must be considered for WDCNs in order to

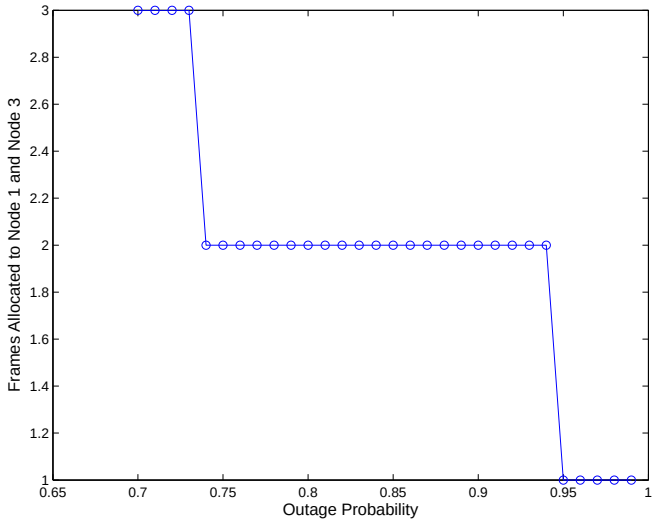

Figure 7.10: Optimal workload allocation as a function of channel heterogeneity.

Table 7.2: Experiment Data for Service Provision Time in Seconds

	Reference Method		Proposed Method	
p_o	Mean	Variance	Mean	Variance
0.8	51.3	23.2	48.5	1.5
0.85	56.0	66.1	51.6	2.6
0.9	66.4	116.8	57.2	69.2
0.95	94.0	319.6	83.9	94.7

reduce both the mean and variance of the execution time. The combination of SDR and multi-process programming makes it easy, based on this demo, to implement all kinds of WDCNs with different computing applications in the future.

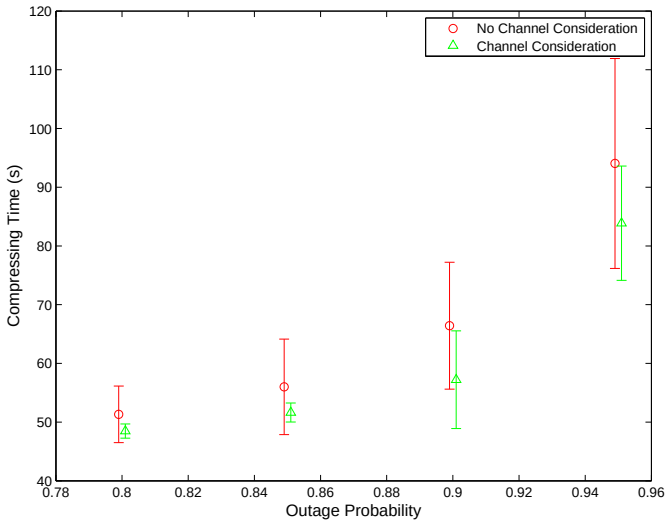

Figure 7.11: Experiment data for two work allocation methods in seconds. The mean and variance of execution time as a function of outage probability.

Chapter 8

Conclusion

8.1 Conclusions

We are in the darkness before the dawn when wireless networks will become a utility the same as power and water. Their heterogeneous nature and new applications will drive the complexity both within individual nodes and over a whole network. Theoretical analyses and computational simulations, besides experimental methods, are necessary to gain insights into design and implementation of wireless distributed computing networks and to investigate their performance under different scenarios. Enabling wireless distributed computing requires advanced nodes, a more complex network architecture, novel resource allocations. This dissertation explores the resource allocation for WDCNs which divides a service session into two parts, the service access phase and the service provision phase, based on different traffic patterns and requirements. Based on identifying several performance metrics, tradeoffs, and research challenges, this dissertation presents resource allocation research in several aspects of WDCNs: power, rate, and workload allocation for the service provision phase, optimal service access, network initialization and synchronization, and channel switching in DSA. In addition, an example design has been implemented to demo the resource allocation problems and their solutions proposed in this dissertation.

More specifically, Chapter 2 and Chapter 3 focus on the service provision phase. Chapter 2 investigates the impact of communication channel conditions on the average execution time of the computing tasks within WDCNs. We then discuss how to increase the robustness and power efficiency for WDCNs subject to the impact of channel variance and spatial heterogeneity. A workload distribution algorithm is proposed to mitigate the effects of channel variation. Moreover, we analyze the requirements for different types of WDCNs. Based on that discussion, a resource allocation solution for computation oriented WDCNs is then introduced in detail. By maximizing the network computing capability and minimizing the communication power consumption, the majority of the power supply can be used to sup-

port the computing process. The impact of the devices' components on system performance, such as the buffer length, and CPU type, are also discussed. While Chapter 2 discusses the system performance in a large time scale and in an average sense, Chapter 3 investigates the performance in a smaller time scale and dynamic sense. A two time scale resource allocation frame is also proposed to combine the power-rate and workload allocation.

In addition, Chapters 4 to 6 discuss performance issues for the service access phase. Firstly, Chapter 4 combines stochastic geometry, the protocol model, and queue theory to analyze the average performance of service response time and designs the optimal strategies during the service access phase. It provides a framework to analyze whether the channel heterogeneity should be considered and the optimal strategies to access service nodes in order to reduce the response time. The channel heterogeneity needs to be considered only when the request arrival rate is high or the channel heterogeneity is high. Increasing the number of SNs provides a marginal reduction of average response time. The optimal access strategy for a SRN is to access best three channels and their corresponding SNs.

Further, in Chapter 5, network initialization and synchronization are investigated in order to form multiple cluster WDCNs in a DSA environment. The proposed algorithm explores contention information to build a status table with which each node can increase or decrease their roles and responsibilities after a certain time period. The convergence rate and time complexity remains similar to the reference design without DSA capabilities. Then, Chapter 6 presents an efficient primary user detection method for WDCNs in a DSA environment. It can reduce the channel vacation latency by detecting the primary user during communications.

Finally, an example of WDCN on CORNET is shown in Chapter 7. It designs and implements an experiment to demo several resource allocation methods presented in this dissertation. The experiment data shows that not considering channel heterogeneity will increase both the mean and the variance of execution delays, which matches with the theoretical analysis.

8.2 Future Work

This section describes the author's thoughts about future research work, which includes workload decomposition methods, security, job scheduling, and device heterogeneity.

Breaking the whole workload into several tasks and identifying dependencies is defined as decomposition. There are three types of workload decomposition methods based on task, data, and data flow, respectively. Task decomposition breaks the workload into several tasks by the functions' nature. If two functions are independent, they can be divided and assigned to two separate processes, or different processes work on different functions. Data decomposition assigns different data blocks to different processes running the same functions. Moreover the data flow decomposition method partitions a problem by how data flows between tasks with the consumer and producer model, which requires consideration of the initial and shut

down latency. Excessive latency occurs with bad implementation when the producer finishes its job and waits for the consumer. The optimal strategy for workload decomposition is determined by the nature of computing applications and resource constraints. The empirical timing and evaluation, instead of theoretical analysis, are explored in the parallel computing community to make an design choice from these three. With more complexities introduced by the wireless channel, workload decomposition in a WDCN requires even more careful consideration, especially for the channel heterogeneity. The research and development in this dissertation provides a starting point for workload decomposition combining data decomposition and task flow decomposition as described in Chapters 2, 3, 4, and 7. However, different workload blocks are assumed to be independent. This is not always the case in practice. For example, the same number of video frames may have different computing densities since the video compressing algorithm may use the information between frames. It requires further efforts to investigate the performance of workload decompositions in WDCN with different computing applications and channel heterogeneity. Criteria should be developed to assess the applications suitable for WDCN implementations.

Security issues are critical to any application with potential commercial success. This is especially true for WDCNs with a hardware and software sharing purpose. The power/energy consumption model defined in Chapter 3 can be used to build a security mechanism for WDCNs, especially for those based on SDR technologies. By monitoring power efficiency, illegal reconfiguration of the hardware may be detected with hardware onboard. Since the power efficiency for RF chain is a function of the reconfigurable parameters such as frequency, bandwidth, modulation and coding, and transmission power, monitoring the ratio between the output power feeding into the antenna and the DC power draining from the power supply will provide a hint about the correctness of the reconfigurations. In addition, measuring system power efficiency, the ratio between communication power consumption and the total power consumption, may inhibit malicious software component in the computing application. A simple hardware security circuit consisting of power meters and some simple logic circuits may be built to measure those two power efficiencies in order to secure SDR based WDCNs. The challenges are the accuracy of the power measurements and the interaction effects between radio components and computing components.

The framework developed in Chapter 4 describes the patterns in both the spatial and time domains, which may also be extended for security applications, such as intrusion detection. For example, when one node is hacked, it will initiate some requests in a greedy way in order to destroy the network as soon as possible. This will change the pattern of the over-the-air interface such as the transmission probability, the density of the requests, and the average response time. These parameters are actually related with each other and can be analyzed within the framework developed in Chapter 4. Based on this idea, the cognitive engine may be integrated to monitor over-the-air (OTA) activities in the physical layer and detect the intrusion when it occurs.

The service access model in Chapter 4 can be extended to include the service processing time of SNs and investigate the optimal strategies for job scheduling. First-In-First-Output

(FIFO) is not optimal for WDCNs to minimize the average response time when the job size distribution has a fat tail. Recent research of server farms suggests that allowing preemptive scheduling and giving priority to jobs with shortest job size may reduce the average response time even further. Although it is hard to build a closed form performance analysis, the idea is promising for WDCNs. This will add extra complexity to the established model in Chapter 4 which requires more research and investigation.

In addition, the current models in Chapters 3 and 4 in this dissertation assume SRN requests to devices with a homogeneous processing rate. The service rate heterogeneity of devices needs more sophisticated resource allocation methods. The optimal strategy for both service access and service provision may be changed to adopt this rate of heterogeneity for devices. The optimal threshold in Chapter 4 needs to be modified in order to consider the status of both the channels and devices when channel service rate and request process rate are comparable.

Finally, the works in Chapters 5 and 6 can be extended to include a mobile network environment for WDCNs in DSA environments. It is more critical to improve the algorithm convergence rate for the proposed method in Chapter 5 if the mobility is included in the model. In addition, mobility may reduce the correctness of the classifier in Chapter 6. Future research is needed to solve these problems.

Bibliography

[1] Google phone webpage: http://www.google.com/phone

[2] FierceWireless, "Beyond the cell phone:create the next frontier in wireless," white paper, Jan. 2010.

[3] K. Doppler, M. Rinne, C. Wijting, C. B. Ribeiro, and K. Hugl, "Device-to-device communication as an underlay to LTE-advanced networks," *IEEE Commun. Mag.* vol. 47, no. 12, pp. 42- 49, Dec. 2009.

[4] Global Information Grid, National Security Agency: http://www.nsa.gov.

[5] F. H. P. Fitzek, and F. Reichert, "Mobile phone programming and its application to wireless networking," Chapter 13, Springer Press.

[6] Thomas Tsou, and Jeffrey H. Reed, "Software architecture for cooperative applications," SDR Forum 2009, Washington DC, Dec. 2009.

[7] X. Fan, W.D. Weber, and L. A. Barroso, "Power provisioning for a warehouse-size computer," *The 34th ACM ISCA*, 2007.

[8] B. H. Calhoun, and A. P. Chandrakasan, "Ultra-dynamic voltage scaling (UDVS) using sub-threhold operation and local voltage dithering," *IEEE Journal of Solid-State Circuits,* vol. 41, no. 1, pp. 238-245, 2006.

[9] S. C. Cripps, *RF power amplifiers for wireless communications,* Artech House, Boston, MA, 2002.

[10] A. I. Pressman, K. Billings, and T. Morey, *Switching power supply design,"* McGraw-Hill Professional, 3rd Ed. 2009.

[11] K. Akkaya and M. Younis, "A Survey of Routing Protocols in Wireless Sensor Networks," *Elsevier Ad Hoc Network Journal*, Vol. 3/3 pp. 325-349, 2005.

[12] C. E. Jones, K. M¿ Sivalingam, P. Agrawal and J. C. Chen, "A survey of energy efficient network protocol for wireless networks," *Journal of Wireless Networks*, pp. 343-358, 2001.

[13] H. Li, L. Zhong, and K. Zheng, "Drowsy transmission:physical layer energy optimization for transmitting random packet traffic," IEEE Infocom 2009, Brazil, April 2009.

[14] E. Modiano, "An adaptive algorithm for optimizing the packet size used in wireless ARQ protocols," *Wireless Networks,* vol. 5, no. 4, pp.279-286, July 1999.

[15] Y. Wei, F. Yu, M. Song, and V. Leung, "Energy efficient distributed relay selection in wireless cooperative networks with finite state Markov channels," IEEE Globecom 2009, Honululu, HI, Dec. 2009.

[16] L. Dai, and V. Chan, "Impact of signal processing energy and large bandwidth on infrastructureless wireless network routing and scalability," IEEE Trans. on Wireless Communications, vol. 8, no. 6, June 2009.

[17] R. Berry and R. Gallager,"Communication over Fading Channels with Delay Constraints, IEEE Transactions on Information Theory, vol. 48, no.5, pp. 1135-1149, May 2002.

[18] E. Uysal-Biyikoglu, B. Prabhakar, and A. E. Gamal, Energy-efficient Packet Transmission over a Wireless Link, IEEE/ACM Transactions on Networking, vol. 10, no. 4, pp. 487-499, Aug. 2002.

[19] C. Li and M. J. Neely, "Energy-Optimal Scheduling with Dynamic Channel Acquisition in Wireless Downlinks, IEEE Transactions on Mobile Computing, vol. 9, no. 4, April 2010.

[20] F. Yao, A. Demers, and S. Shenker, "A scheduling model for reduced CPU energy," 36th FOCS, Milwaukee, WI, Oct. 23-25, 1995

[21] E.N. Elnozahy, M. Kistler, and R. Rajamony, "Energy-efficient server cluster," Proceeding of the 2nd Workshop on Power-Aware Computing Systems,2002.

[22] X. Fan, W. Weber, and L. A. Barroso, "Power provisioning for a warehouse-sized computer," Proceedings of ACM ISCA, San Diego, CA, June 2007.

[23] T. Horvath, T. Abdelzaher, etc., "Dynamic voltage scaling in multi-tier web servers with end-to-end delay control," IEEE Trans. on Computers, vol. 56, no. 4, pp. 444-458, April 2007.

[24] R. Nathuji, C. Isci, and E. Gorbatov, "Exploiting platform heterogeneity for power efficient data center," IEEE ICAC 2007, Jacksonville, FL, June 11-15, 2007.

[25] C. Bekas, A. Curioni and I. Feduova, "Low Cost High Performance Uncertainty Quantification", Worskhop on High Performance Computational Finance, Supercomputing 09, Portland, USA, Nov. 2009.

[26] Y. Zhang, D. Parikh, K. Sankaranarayanan, K. Skadron, "HotLeakage: a temperature-aware model of sub-threshold and gate leakage for architects," In Tech Report CS-2003-05, Univ. of Virginia, 2003.

[27] E. Pinheiro, W.D. Weber, and L. A. Barroso, "Failure trends in a large disk drive population," *FAST'07*, pp. 17-29, Feb. 2007.

[28] D. Abts, M. Marty, P. Wells, P. Klausler, and H. Liu, "Energy-proportional datacenter networks," *IEEE ISCA 2010*, Saint-Malo, France, Jun. 16-19 2010.

[29] J. Kim, W. Dally, S. Scott, and D. Abts, "Cost-efficient dragonfly topology for large scale systems," *IEEE Micro*, vol. 29, no. 1, pp. 33-40, Feb. 2009.

[30] P. Gupta, P.R. Kumar, "The capacity of wireless networks," *IEEE Trans. on Inform. Theory*, vol. 46, pp. 388-404, Mar. 2000.

[31] A. O zgur, O. Leveque, and D. Tse, Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks, *IEEE Trans. Inform. Theory*, vol. 53, no. 10, pp. 35493572, Oct. 2007.

[32] M. Franceschetti, M.D. Migliore, and P. Minero,The capacity of wireless networks: information-theoretic and physical limits," *IEEE Trans. Inform. Theory*, vol. 55, no. 8, pp. 3413-3424, 2009.

[33] J. Ghaderi, L. Xie, and X. Shen, "Hierarchical cooperation in ad hoc networks: optimal clustering and achievable throughput," *IEEE Trans. Inform. Theory*, vol. 55, no. 8, pp. 3425-3436, 2009.

[34] V. Cadambe, S. Jafar, "Interference alignment and degree of freedom for the K-user interference channel," *IEEE Trans. on Inform. Theory*, vol. 54, no. 8, pp.3425-3441, 2008.

[35] M. Haenggi, J. G. Andrews, F. Baccelli, O. Dousse, and M. Franceschetti, "Stochastic geometry and random graphs for the analysis and design of wireless networks," *IEEE JSAC*, vol. 27, no. 7, pp. 1029-1046, Sept. 2009.

[36] J. Andrew, S. Shakkottai, R. Heath, N. Jindal, M. Haenggi, R. Berry, D. Guo, M. Neely, S. Weber, S. Jafar, and A. Yener, "Rethinking information theory for mobile ad hoc networks," *IEEE Commun. Mag.*, vol. 46, no. 12, pp. 94-101, 2008.

[37] D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris, *"Fundamentals of queueing theory"*, 4th Edition, John Wiley&Sons, Inc, Hoboken, NJ, 2008.

[38] M. Mitzenmacher, and B. Vocking, "The asymptotics of selecting the shortest of two, improved," *Proc. of the 37th AACCC*, pp. 326-327, 1999.

[39] M. Mitzenmacher, "Studying balanced allocations with differential equations," *Combinatorics, Probability, and computing*, vol. 8, pp. 473-482, 1999.

[40] G. M. Amdahl, "Validity of the single-processor approach to achieving large-scale computing capabilities," *Proc. Am. Federation of Information Processing Societies Conf.,* AFIPS Press, pp. 483-485, 1967.

[41] J. L. Gustafson, "Reevaluating Amdahl's Law," *Communications of the ACM*, vol. 31, no. 5, pp.532-533, 1998.

[42] Di Wu, Y. Liu, and K. Ross, "Modeling and analysis of multi-channel p2p live video systems," *IEEE Infocom 2009*, Rio de Janeiro, Brazil, April 19-25 2009.

[43] Hadoop, online software, http://hadoop.apache.org/

[44] Erlang, online resource, http://www.erlang.org/

[45] X. Chen, T. Bose, S.M. Hasan, and J. H. Reed, "Resource allocation for wireless computing networks," to be submitted.

[46] M. J. Neely, "Energy optimal control for time varying wireless networks," *IEEE Trans. on Info. Theory*, vo 52, no. 7, July 2006, pp. 1 - 18.

[47] M. J. Neely, "Stochastic network optimization with application to communication and queueing sytsems," *Synthesis Lectures on Communication Networks*, Morgan&Claypool Publishers, 2010

[48] A. Ozgur, R. Johari, D. Tse, and O. Leveque, "Information theoretic operating regimes of large wireless networks," *IEEE Trans. Inform. Theory*, vol. 56, no. 1, pp. 427-437, 2010.

[49] X. Chen, S.M. Hasan, T. Bose, and J. H. Reed "Cross-layer resource allocation for wireless distributed computing networks," *IEEE RWS 2010*, New Orleans, LA, Jan. 2010.

[50] X. Chen, T. R. Newman, D. Datla, T. Bose, and J. H. Reed, "The impact of channel variances on wireless distributed computing networks," *IEEE Globecom 2009*, Honolulu, HI, Dec. 2009.

[51] Z. Ma, H. Hu and Y. Wang, "On Complexity Modeling of H.264/AVC Video Decoding and Its Application for Energy Efficient Decoding," submitted to IEEE Trans. CSVT, Jan. 2010.

[52] P. Ranganathan,P. Leech, D. Irwin, and J. Chase, "Ensemble-level power management for dense blade servers," *In Proc. of ISCA 2006*, pp. 66-77, July 2007.

[53] A. Wierman, L. L. H. Andrew, and A. Tang, "Stochastic analysis of power-aware scheduling," *IEEE ICCC 2008*, pp. 1278-1283, Sept. 2008.

[54] R. Nathuji, C. Isci, and E. L. Gorbatov, "Exploiting platform heterogeneity for power efficient data center," *IEEE ICAC 2007*, Jacksvill, FL, June 2007.

[55] J. Huang, and S. Lee, "A heterogeneity-aware approach to load balancing of computational tasks: a theoretical and simulation study," *Cluster Computing*, vol.11, no.2, June 2008 pp.133-149.

[56] S. Dhakal, M. Hayat, M. Elyas, J. Ghanem, and C. Abdallah, "Load balancing in distributed computing over wirless LAN: effects of network delay," *IEEE WCNC 2005*, pp. 1755-1760, March 2005.

[57] A. Goldsmith and S. B. Wicker, "Design challenges for energy-constrained ad hoc wireless networks," *IEEE Wireless Communications Mag.*, vol. 9, no. 4, Aug. 2002, pp. 8-27.

[58] C. Anderson, D. Freeman, I. James, A. Johnston, and S. Ljung, *Mobile Media and Applications, From Concept to Cash: Successful Service Creation and Launch*, Wiley, 2006.

[59] M. Williams, LG, Samsung develop solar-powered cell phones, Webpage:http://www.pcworld.com/article/159507/lg_samsung_develop _solarpowered _cell_phones.html.

[60] T. Burd and R. Brodersen, " Energy efficient CMOS microprocessor design," *In Proc. of 28th Hawaii Int'l Conf. on System Sciences 1995*,pp. 288-297, Jan. 1995.

[61] R. J. LaDuca, J. Sharkey, D. V. Ponomarev, "Hiding communication delays in clustered mrcroarchitectures," *SBAC-PAD 2008*, pp. 107-114, Oct. 2008.

[62] V. Rodoplu, and T.H. Meng, "Minimum energy mobile wireless networks," *IEEE JSAC*, vol. 17, no. 8, pp. 1333-1344, July 1999.

[63] J. G. Proakis, *Digital Communications*, 4th Ed. New York: McGraw- Hill, 2000.

[64] ARM1156 processor, online available at: http://www.arm.com/.

[65] J. Mo and J. Walrand, "Fair end-to-end window based congestion control," *IEEE Trans. Networking*, vol. 8, no. 5, Oct. 2000.

[66] R. Bellman, *Dynamic Programming*, Princeton University Press, 2003.

[67] S. Boyd, and A. Mutapcic, web resource,http://www.stanford.edu/class /ee364b/notes/stoch_subgrad_notes.pdf

[68] N. Shor, *Nondifferentiable Optimization and Polynomial Problems,* Nonconvex optimization and its applications, Kluwer Academic Publishers, Boston, MA, 1998.

[69] D. O'Neill, B. S. Thian, A. Goldsmith, and S. Boyd, "Wireless NUM: rate and reliability tradeoffs in random environments," *IEEE WCNC 2009*, 2009, Budapest Hungary, 2009.

[70] Z. Han, K. J. Ray Liu, *Resource Allocation for Wireless Networks,* Cambridge University Press, July 2008.

[71] A. Ribeiro, and G. B. Giannakis, "Separation principles in wireless networking," *IEEE Trans. on Information Theory*, vol. 56, no. 9, Sept. 2010, pp. 4488-4505.

[72] X. Wang, and N. Gao, "Stochastic resource allocation over fading multiple access and broadcast channels," *IEEE Trans. on Information Theory*, vol. 56, no. 5, May. 2010, pp. 2382-2391.

[73] D. Datla, X. Chen, T. R. Newman, T. Bose, and J. H. Reed, "Power efficiency in wireless network distributed computing," *IEEE VTC- Fall 2009*, Anchorage, AK, Sept. 2009.

[74] Martin Haenggi, "Mean interference in hard-core wireless networks," *IEEE Communication Letters*, vol. 15, no. 8, pp. 792-794, Aug. 2011.

[75] Francois Baccelli and Bartlomiej Blaszczyszyn, *Stochastic Geometry and Wireless Networks*, Foundations and Trends in Networking, NOW Publishers, Volume 3, Issue 3-4. 2011.

[76] M. Haenggi, J. G. Andrews, F. Baccelli, O. Dousse and M. Franceschetti, "Stochastic Geometry and Random Graphs for the Analysis and Design of Wireless Networks", *IEEE Journal on Selected Areas of Communications*, Sept. 2009

[77] Harchol-Balter. "Queueing Disciplines," Wiley Encyclopedia Of Operations Research and Management Science, 2009.

[78] I. Adan, and J. Resing, "Queueing Theory," Feb. 2002, available at: www.win.tue.nl/ iadan/queueing.pdf.

[79] Woei Lin, and P. Kumar, "Optimal control of a queueing system with two heterogeneous servers," *IEEE Trans. on Automatic Control*, vol. 29, no. 8, pp. 696-703, Aug. 1984.

[80] F. de. Vericourt and Y.P. Zhoud, "On the incomplete results for the multi-server slow server problem," *Queue Systems*, vol. 52, no. 3, pp. 189-191, 2006.

[81] V. Rykov, and D. Efrosinin, "Optimal control of queueing systems with heterogeneous servers," *Queue Systems*, vol. 36, pp. 389-407, 2004.

[82] X. Chen, S.M. Hasan, T. Bose, and J. H. Reed, "Software defined radio based wireless grids," Wireless Innovation Forum and Product Exposition, Washington, DC, Nov. 2010.

[83] Wireless Grid Innovation Testbed (WiGiT) Open Specifications, online resource: http://wglab.net/

[84] Global Information Grid, Wikipedia, online resource: http://en.wikipedia.org.

[85] K. Doppler, M. Rinne, C. Wijting, C. B. Ribeiro, and K. Hugl, "Device-to-device communication as an underlay to LTE-Advanced networks, *IEEE Commun. Mag.*, Vol. 47, No. 12, pp. 42-49, Dec. 2009.

[86] G. Cafaro, T. Gradishar, J. Heck, S. Machan, G. Nagaraj, S. Olson, R. Salvi, B. Stengel, and B. Ziemer, "A 100 MHz 2.5 GHz Direct Conversion CMOS Transceiver for SDR Applica-tions, *IEEE Radio RFIC Symp. 2007*, pp.189-192, June 2007.

[87] K. Lim, and J. Laskar, "Emerging opportunities of RF IC/system for future cognitive radio wireless communications," *IEEE RWS*, pp. 703 706, Jan. 2008.

[88] S. E. Sussman-Fort, "Matching network design using non-Foster impedances, *International Journal of RF Microwave Computer-Aided Engineering*, pp. 13542, March 2006.

[89] J. Hoffman, D.A. Ilitzky, A. Chun, and A. Chapyzhen-ka, "Architecture of the Scalable Communications Core," *NOCS 07.*, pp.40-52, May 2007.

[90] U. Ramacher, "Software-Defined Radio Prospects for Multistandard Mobile Phones, *IEEE Computer Mag.*, Vol. 40, No. 10, pp. 62-69, Oct. 2007.

[91] C. Lenzen, T. Locher, P. Sommer, and R. Wattenhofer, "Clock Synchronization: Open Problem in Theory and Practice, *SOFSEM 2010*, pp. 61-70, 2010

[92] R. Fan, and N. Lynch, "Gradient Clock Synchronization, *PODC04*, St. Johns, Newfoundland, Canada, July, 2004.

[93] T. Tsou, and J. H. Reed, "Software architecture for cooperative applications, *SDR Forum*, Washington DC, Dec. 2009.

[94] S. Dolev, S. Gilbert, R. Guerraoui, F. Kuhn, and C. Newport, "The Wireless Synchronization Problem, *PODC09*, Calgary, Alberta, Canada, Aug. 2009.

[95] P. Sommer, and R. Watternhofer, "Gradient Clock Synchronization in Wireless Sensor Networks, *IPSN09*, San Francisco, CA, USA, Apr. 2009.

[96] B. Sundararaman, U. Buy, and A. D. Kshem, "Clock Synchronization for Wireless Sensor Networks: A Survey, *Ad Hoc Networks*, Vol. 3, No. 3, pp. 281-323, May 2005.

[97] S. Fikret, and Y. Bulent, "Time Synchronization in Sensor Networks: A Survey, *IEEE Networks*, Vol. 18, No. 4, pp. 45-50, 2004.

[98] J. Elson, L. Girod, and D. Estrin, "Fine-Grained Network Time: Synchronization using Reference Broadcasts, *OSDI 02*, Boston, MA, Dec. 2002.

[99] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The Flooding Time Synchronization Protocol, *Proc. 2dn ACM Conf. on Embedded Networked Sensor Systems*, pp. 39-49, 2004.

[100] P. Guo, T. Jiang, K. Zhang, and H.-H. Chen, "Clustering Algorithm in Initialization of Multi-hop Wireless Sensor Networks, *IEEE Trans. on Wireless Commun.*, Vol. 8, No. 12, Dec. 2009.

[101] S. Iyer and D. Manjunath, "Topological Properties of Random Wireless Networks, *Proc. Indian Academy Sciences*, vol. 31, no. 2, pp. 117-139, Apr. 2006.

[102] X. Chen, T. Bose, S.M. Hasan, and J. H. Reed, "Efficient detection of primary users in cognitive radio networks," *International Journal of Communication Networks and Distributed Systems*, vol. 8, no.3/4, 2012, pp. 267 - 285.

[103] E. E., Azzouz, and A. K, Nandi, "Automatic Identification of Digital Modulation Types', *IEEE Signal Processing*, Vol. 47, No. 1, pp.55–69, 995.

[104] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge*, Vol. 2, pp.121–167, 1998.

[105] T. C. Clancy, and W. A. Arbaugh, "Measuring interference temperature," *Virginia Tech Wireless Personal Communications Symposium 2006*, 2006.

[106] O. A. Dobre, Y. Bar-Ness, and W. Su, "Higher Order Cyclic Cumulants for High Order Modulation Classification," *Proc. IEEE MILCOM*, pp.112–117, 2003.

[107] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Blind modulation classification: a concept whose time has come," *IEEE Advances in Wired and Wireless Communication*, pp.223–228, 2005.

[108] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of Automatic Modulation Classification Techniques: Classical Approaches and New Trends," *IET Communication*, Vol. 1, No. 2, pp.137–156, 2007.

[109] I. Druchmann, E. I. Plotkin, and M. N. S, Swamy, "Automatic Modulation Type Recognition', *IEEE Canadian Conference on Electrical and Computer Engineering*, Vol. 1, pp.65–68, 1998.

[110] R. O. Duda, *Pattern Classification*, New York:The John Wiley & Sons Inc., 1998.

[111] J. F. Kenney, and E. S. Keeping, *Mathematics of Statistics*, Princeton:Van Nostrand, 1951.

[112] K. Kim, I. A. Akbar, K. K. Bae, J.-S Um, C. M. Spooner, and J. H. Reed, "Cyclostationary approaches to signal detection and classification in cognitive radio," *IEEE DySPAN '07*, April, 2007.

[113] G. Ganesan, and Y. Li, "Cooperative spectrum sensing in cognitive radio, part I: two user networks," *IEEE Trans. on Wireless Commun.*, Vol. 6, No. 6, pp.2204–2222, 2007.

[114] W. A. Gardner, "Signal interception: a unifying theoretical framework for feature detection," *IEEE Trans. Commun.*, Vol. 36, pp.897–906, 1988.

[115] W. A. Gardner, A. Napolitano, and L. Paura, "Cyclostationarity: Half a century of research', *Signal Processing*, Vol. 86, pp.639–697, 2006.

[116] G. H. Galub, and C F. Loan, *Matrix Computations*, Baltimore:The John Hopkins Univeristy Press, 1996.

[117] A. M. Hisham, and H. Arslan, "Multidimensional signal analysis and measurements for cognitive radio systems," *IEEE RWS '08*, pp. 639–643, 2008.

[118] S. Z. Hsue, and S. S. Soliman, "Automatic Modulation Classification Using Zero Crossing," *IEE Proc. Radar and Signal Processing*, Vol. 137, No. 6, pp. 459–465, 1990.

[119] Y. Hur, G. Park, C. H. Lee, H. S. Kim, and J. Laskar, "A cognitive radio (cr)-based mobile interactive digital broadcasting application adopting a multi-resolution spectrum-sensing (MRSS) technique," *IEEE 66th. Vehicular Tech. Conf.*, pp. 1912–1916, 2007.

[120] H. Kim, and K. G. Shin, "Efficient Discovery of Spectrum Opportunities with MAC-Layer Sensing in Cognitive Radio Networks," *IEEE Trans. on Mobile Computing*, Vol. 7, no. 5, pp. 533–545, 2008.

[121] J. D. Laster, and J. H. Reed, "Interference rejection in digital wireless communications," *IEEE signal processing magazine*, Vol. 14, No. 3, pp. 37–62, 1997.

[122] U. Madhow, and M. L. Honig, "MMSE interference suppression for direct-sequence spread spectrum CDMA," *IEEE Trans. On Commun.*, Vol. 42, No. 12, pp.3178–3188, 1994.

[123] O. L. Mangasarian, and D. R. Musicant, "Lagrangian support vector machine," *Journal of Machine Learning Research*, Vol. 1, pp.161–177, 2001.

[124] S. M. Mishra, A. Sahai, and R. W. Brodersen, "Cooperative Sensing among Cognitive Radios," *IEEE ICC '06*, pp. 1–4, 2006.

[125] B. G. Mobasseri, "Constellation Shape as a Robust Signature for Digital Modulation Recognition," *Proc. IEEE MILCOM*, pp. 442–446, 1999.

[126] S. Moshavi, "Multi-user detection for DS-CDMA communications," *IEEE Commun.Mag.*, Vol. 34, No. 10, pp. 124–137, 1996.

[127] A. K. Nandi, and E. E. Azzouz, "Algorithms for Automatic Modulation Recognition of Communication Signals," *IEEE Trans. on Commun.*, Vol. 46, No. 4, pp. 431–436, 1998.

[128] H. V. Poor, and G. W. Wornell, *Wireless communications: signal processing perspectives*, Chapter 2, Prentic-Hall, Inc.: Upper Saddle River, NJ, 1998.

[129] J. G. Proakis,and M. Salehi, *Digital communications*, 5th ed., McGraw-Hill, Boston, MA, 2008.

[130] C. Schreyogg, K. Kittel, U. Kressel, and J. Reichert, "Robust Classification of Modulation Types Using Spectral Features Applied to HMM," *Proc. IEEE MILCOM*, pp. 1377–1381, 1997.

[131] M. Skolnik, *Introduction to Radar Systems*, New York:McGraw-Hill Higher Eduction Press, 2001.

[132] A. Swami, and B. M. Sadler, "Hierarchical Digital Modulation Classification Using Cumulants," *IEEE Trans. Commun.*, Vol. 48, No. 3, pp. 416–429, 2000.

[133] Tektronix Electronics. "DPX acquisition technology for spectrum analyzers fundamentals," Obtained through the Internet: http://www2.tek.com/.

[134] Z. Tian, and G. B. Giannakis, "A wavelet approach to wideband spectrum sensing for cognitve radios," *IEEE 1st Int. Conf. on Cognitive Radio Oriented Wireless Networks and Communications*, pp. 1–6, 2006.

[135] Z. Tian, and G. B. Giannakis, "Compressed sensing for wideband Cognitive Radios," *Proc. of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1357–1360, 2007.

[136] B. D. Van Veen, and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE ASSP Mag.*, Vol. 5, pp. 4–24, 1998.

[137] Z. Yu, S. Hoyos, and B. M. Sadler, "Mixed-signal parallel compressed sensing and reception for cognitive radio," *IEEE ICASSP '08*, pp. 3861–3864, 2008.

[138] Q. Zhao, L. Tong, and A. Swami, "Decentralized Cognitive MAC for Dynamic Spectrum Access," *IEEE DySPAN '05*, pp. 224–232, 2005.

[139] G. Cafaro et al., "A 100 MHz2.5 GHz Direct Conversion CMOS Transceiver for SDR Applications, in *Proc. IEEE Radio Frequency Integrated Circuits (RFIC) Symposium*, Honolulu, Hawaii, Jun. 2007.

[140] T. Newman, S.M.S Hasan, D. Depoy, T. Bose, and J. Reed, "Designing and deploying a building-wide cognitive radio networks," *IEEE Communication Magazine*, vol. 48, no. 9, Sept. 2010.

[141] H.264 Reference Software, online resource: http://iphome.hhi.de

[142] G. Nychis, T. Hottelier, Z. Yang, S. Seshan, and P. Steenkiste, "Enabling MAC protocol implementations on software-defined radios," NSDI 2009, Berkeley, CA, 2009.

[143] Joseph Gaeddert, Liquid Radio DSP library, online resource: www.ganymede.ece.vt.edu

[144] GNU Radio, online: http://gnuradio.org