Multimedia over Cognitive Radio Networks

Algorithms, Protocols, and Experiments

Edited by
Fei Hu and Sunil Kumar

CRC Press
Taylor & Francis Group

Multimedia over Cognitive Radio Networks

Algorithms, Protocols, and Experiments

OTHER COMMUNICATIONS BOOKS FROM AUERBACH

Advances in Biometrics for Secure Human Authentication and Recognition
Edited by Dakshina Ranjan Kisku, Phalguni Gupta, and Jamuna Kanta Sing
ISBN 978-1-4665-8242-2

Advances in Visual Data Compression and Communication: Meeting the Requirements of New Applications
Feng Wu
ISBN 978-1-4822-3413-8

Anonymous Communication Networks: Protecting Privacy on the Web
Kun Peng
ISBN 978-1-4398-8157-6

Case Studies in System of Systems, Enterprise Systems, and Complex Systems Engineering
Edited by Alex Gorod, Brian E. White, Vernon Ireland, S. Jimmy Gandhi, and Brian Sauser
ISBN 978-1-4665-0239-0

Cyber-Physical Systems: Integrated Computing and Engineering Design
Fei Hu
ISBN 978-1-4665-7700-8

Evolutionary Dynamics of Complex Communications Networks
Vasileios Karyotis, Eleni Stai, and Symeon Papavassiliou
ISBN 978-1-4665-1840-7

Fading and Interference Mitigation in Wireless Communications
Stefan Panic, Mihajlo Stefanovic, Jelena Anastasov, and Petar Spalevic
ISBN 978-1-4665-0841-5

Green Networking and Communications: ICT for Sustainability
Edited by Shafiullah Khan and Jaime Lloret Mauri
ISBN 978-1-4665-6874-7

Image Encryption: A Communication Perspective
Fathi E. Abd El-Samie, Hossam Eldin H. Ahmed, Ibrahim F. Elashry, Mai H. Shahieen, Osama S. Faragallah, El-Sayed M. El-Rabaie, and Saleh A. Alshebeili
ISBN 978-1-4665-7698-8

Intrusion Detection in Wireless Ad-Hoc Networks
Edited by Nabendu Chaki and Rituparna Chaki
ISBN 978-1-4665-1565-9

Machine-to-Machine Communications: Architectures, Technology, Standards, and Applications
Edited by Vojislav B. Misic and Jelena Misic
ISBN 978-1-4665-61236

MIMO Processing for 4G and Beyond: Fundamentals and Evolution
Edited by Mário Marques da Silva and Francisco A. Monteiro
ISBN 978-1-4665-9807-2

Network Innovation through OpenFlow and SDN: Principles and Design
Edited by Fei Hu
ISBN 978-1-4665-7209-6

Opportunistic Mobile Social Networks
Edited by Jie Wu and Yunsheng Wang
ISBN 978-1-4665-9494-4

Physical Layer Security in Wireless Communications
Edited by Xiangyun Zhou, Lingyang Song, and Yan Zhang
ISBN 978-1-4665-6700-9

SC-FDMA for Mobile Communications
Fathi E. Abd El-Samie, Faisal S. Al-kamali, Azzam Y. Al-nahari, and Moawad I. Dessouky
ISBN 978-1-4665-1071-5

Security and Privacy in Smart Grids
Edited by Yang Xiao
ISBN 978-1-4398-7783-8

Security for Multihop Wireless Networks
Edited by Shafiullah Khan and Jaime Lloret Mauri
ISBN 978-1-4665-7803-6

Self-Healing Systems and Wireless Networks Management
Junaid Ahsenali Chaudhry
ISBN 978-1-4665-5648-5

The State of the Art in Intrusion Prevention and Detection
Edited by Al-Sakib Khan Pathan
ISBN 978-1-4822-0351-6

Wireless Ad Hoc and Sensor Networks: Management, Performance, and Applications
Jing (Selina) He, Mr. Shouling Ji, Yingshu Li, and Yi Pan
ISBN 978-1-4665-5694-2

Wireless Sensor Networks: From Theory to Applications
Edited by Ibrahiem M. M. El Emary and S. Ramakrishnan
ISBN 978-1-4665-1810-0

ZigBee® Network Protocols and Applications
Edited by Chonggang Wang, Tao Jiang, and Qian Zhang
ISBN 978-1-4398-1601-1

Multimedia over Cognitive Radio Networks

Algorithms, Protocols, and Experiments

Edited by
Fei Hu and Sunil Kumar

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20141014

International Standard Book Number-13: 978-1-4822-1487-1 (eBook - PDF)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

To Fang Yang, Gloria Yang Hu, Edwin and Edward (twins)

Also dedicated to Prof. Sunil Kumar's family (Rajni, Paras, and Shubham)

Contents

Preface

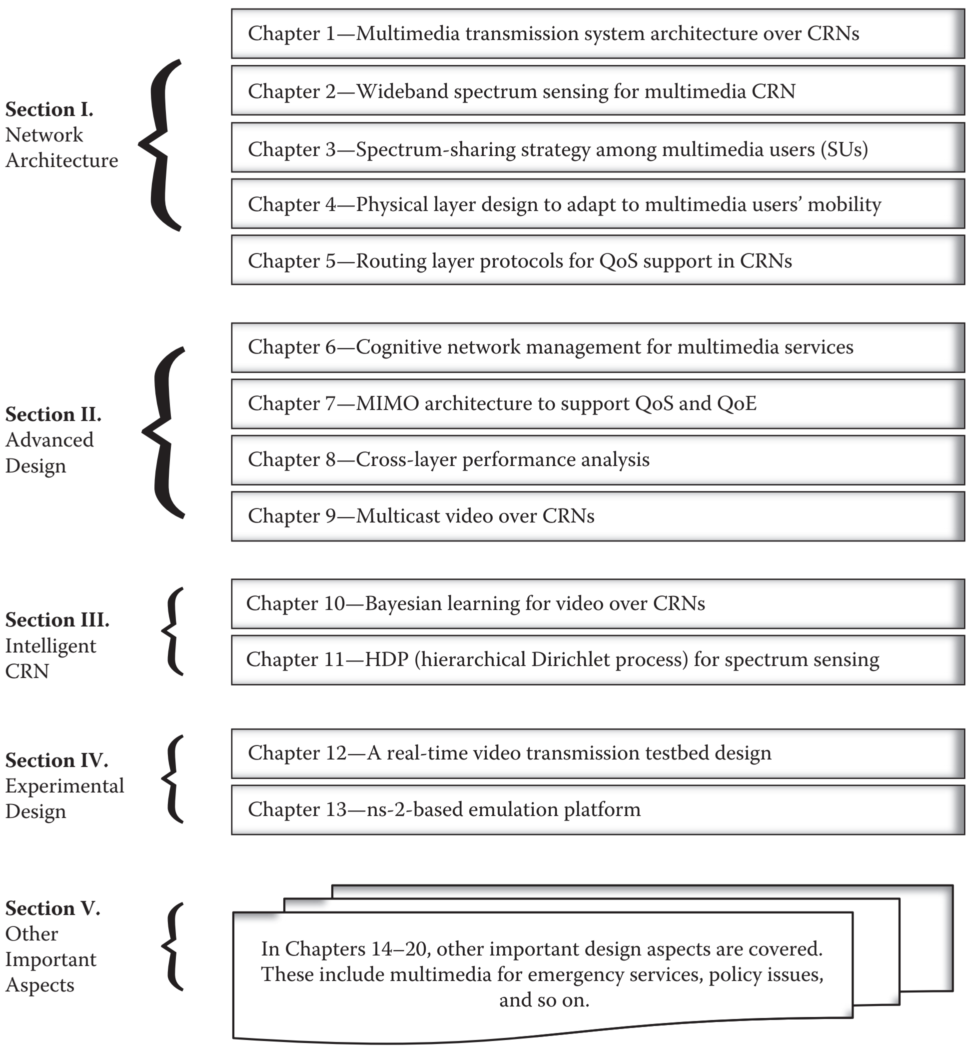
Rapid development of multimedia applications (such as video streaming) over wireless networks requires wide channel bandwidth. Even when the available radio spectrum is allocated, large portions of it are not used efficiently. Lately, cognitive radio networks (CRNs), which use intelligent dynamic spectrum access techniques to use the unoccupied spectrum, have received much attention. In CRN, the unlicensed secondary users (SUs), also known as cognitive users, search intelligently the vacant spectral bands and access them to maximize SUs performance. An SU needs to hand off to other vacant channels if its current channel is needed by licensed primary users (PUs). The main issues in CRN design involve s*pectrum analysis* and *spectrum decision*. The aim of *spectrum analysis* is to use the signal processing and machine learning algorithms to extract the patterns of the sensed idle spectrum (from the *spectrum-sensing* module). Examples of patterns include channel quality changes in a time window, channel holding time, and the probability of control channel saturation in the next time slot. These patterns are used by the *spectrum decision* module to generate accurate network control *actions* based on the reasoning of the system *state* in the current time slot. Examples of *actions* include the spectrum hand-off, allocation of spectral bands to SUs, adjustment of the video source rate, data allocation in the orthogonal frequency division multiplexing (OFDM) channel, and spectrum scanning rate.

Many challenging issues related to multimedia transmission over CRN still remain unresolved. Examples of these issues include QoS (quality of service) support for delivering a video stream over dynamic channels that change each time the PU takes the channel back, using queuing theory to model the delay/jitter parameters when an SU performs spectrum hand-off, QoE (quality of experience) support that reflects the users' satisfaction with a video clip's resolution, achieving higher-priority transmission for users, and integrating the video coding with CRN protocols in order to achieve a better QoS/QoE performance. This book covers the algorithms, protocols, and experiments for the delivery of multimedia traffic over CRNs.

Features of the Book: Compared to other books on CRNs, this book has the following two special features: (1) Emphasis on understanding video streaming in the dynamic spectrum access environment in CRNs: This book covers the important CRN protocol designs for transmitting video flows over CRNs. It has physical/MAC/routing layer protocols for a mobile and dynamic spectrum environment. In the physical layer, it considers different modulation/encoding schemes; in the MAC layer, it focuses on new algorithms for scheduling and communication among neighboring users in the context of spectrum hand-off; in the routing layer, it provides multichannel, multihop, spectrum-adaptive routing algorithms. (2) Explanation of both theoretical and experimental designs: Most CRN books provide just the protocol and algorithm details without linking them to the experimental design. This book explains how the universal software radio peripheral (USRP) boards could be used for real-time, high-resolution video transmission. It also discusses how a USRP board can sense the spectrum dynamics and how it can be controlled by GNU software. A separate chapter discusses how the ns-2 could be used to build a simulated CRN platform.

Target Audience: This book is suitable for the following types of readers:

(1) *College students*: This book can serve as a textbook or reference book for college courses on CRNs, which could be offered in computer science, electrical and computer engineering, and information technology and science departments. It can also be used as a part of a wireless network course.
(2) *Researchers*: Because the book explains multimedia over CRNs from protocols to experimental design, it will be very useful for researchers (including graduate students) who are interested in multimedia applications in a dynamic spectrum environment.



(3) *Computer scientists*: Algorithms on video coding/transmission over CRNs as well as Python programming for USRP boards are provided in this book. Computer scientists could refer to these principles in their own design.
(4) *Engineers*: Many useful CRN design principles are explained in the book, which would be useful for engineers in industry in their product design work. The first 10 chapters provide concrete CRN protocol details.

Book Organization: This book consists of 20 chapters (five sections) that cover the most important aspects of multimedia over CRNs. These chapters include CRN architecture, protocols, algorithms, and experimental design.

Editors

Dr. Fei Hu is currently an associate professor in the Department of Electrical and Computer Engineering at the University of Alabama, Tuscaloosa, Alabama. He obtained his PhD degrees at Tongji University (Shanghai, China) in the field of signal processing (in 1999) and at Clarkson University (Potsdam, New York) in electrical and computer engineering (in 2002). He has published more than 160 journal/conference papers and books. Dr. Hu's research has been supported by U.S. National Science Foundation, Cisco, Sprint, and other sources. His research expertise can be summarized as *3S: Security, Signals, Sensors.*

Dr. Sunil Kumar is currently a professor and Thomas G. Pine Faculty Fellow in the Department of Electrical and Computer Engineering at San Diego State University (SDSU), San Diego, California, USA. He received his PhD in electrical and electronics engineering from the Birla Institute of Technology and Science (BITS), Pilani, India, in 1997. From 1997 to 2002, Dr. Kumar was a postdoctoral researcher and adjunct faculty at the University of Southern California, Los Angeles. He also worked as a consultant in industry on JPEG2000- and MPEG-4-related projects, and was a member of the US delegation in JPEG2000 standardization activities. Prior to joining SDSU, Dr. Kumar was an assistant professor at Clarkson University, Potsdam, New York (2002–2006). He was an ASEE Summer Faculty Fellow at the Air Force Research Lab in Rome, New York, during the summer of 2007 and 2008, where he conducted research in Airborne Wireless Networks. Dr. Kumar is a senior member of IEEE and has published more than 125 research articles in international journals and conferences, including three books/book chapters. His research has been supported by grants/awards from the National Science Foundation, U.S. Air Force Research Lab, Department of Energy, California Energy Commission, and other agencies. His research areas include wireless networks, cross-layer and QoS-aware wireless protocols, and error-resilient video compression.

Contributors

Bashar I. Ahmad
Department of Engineering
University of Cambridge
Cambridge, United Kingdom

Gianmarco Baldini
European Commission
Joint Research Centre
Institute for the Protection and Security
of the Citizen
Ispra, Italy

Ke Bao
Department of Electrical and Computer
Engineering
University of Alabama
Tuscaloosa, Alabama

Tamer Başar
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
Champaign, Illinois

Abdur Rahim Biswas
CREATE-NET (Center for REsearch
And Telecommunication
Experimentation for NETworked
communities)
Trento, Italy

Faouzi Bouali
Department of Signal Theory
and Communications
Universitat Politécnica de Catalunya
Barcelona, Spain

Athina Bourdena
Department of Informatics Engineering
Technological Educational Institute of Crete
Heraklion, Greece

Emmanuel Casseau
University of Rennes 1, IRISA, INRIA
Rennes, France

Abdelaali Chaoub
Laboratory of Electronic and Communication
Mohammadia School of Engineers
Mohammed V-Agdal University
Rabat, Morocco

Chihkai Chen
Department of Electrical Engineering
University of California, Los Angeles
Los Angeles, California

Xiang Chen
Aerospace Center
Tsinghua University
Beijing, China

Nan Cheng
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada

Markus Fiedler
Department of Communication Systems
Faculty of Computing
Blekinge Institute of Technology
Karlskrona, Sweden

Matthieu Gautier
University of Rennes 1, IRISA, INRIA
Rennes, France

Fei Hu
Department of Electrical and Computer Engineering
University of Alabama
Tuscaloosa, Alabama

Xin-Lin Huang
School of Electronics and Information
Tongji University
Shanghai, China

Elhassane Ibn-Elhaj
Department of Telecommunication
National Institute of Posts and Telecommunications
Rabat, Morocco

Reema Imran
Department of Electrical Engineering
University of Jordan
Amman, Jordan

Muhammad Awais Javed
School of Electrical Engineering and Computer Science
The University of Newcastle
Callaghan, Australia

Jamil Yusuf Khan
School of Electrical Engineering and Computer Science
The University of Newcastle
Callaghan, Australia

Sunil Kumar
Department of Electrical and Computer Engineering
San Diego State University
San Diego, California

Cong Ling
Department of Electrical and Electronic Engineering
Imperial College London
London, United Kingdom

Ning Lu
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada

Jon W. Mark
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada

George Mastorakis
Department of Informatics Engineering
Technological Educational Institute of Crete
Heraklion, Greece

Constandinos X. Mavromoustakis
Department of Computer Science
University of Nicosia
Nicosia, Cyprus

Yakim Y. Mihov
Faculty of Telecommunications
Technical University of Sofia
Sofia, Bulgaria

Arumugam Nallanathan
Department of Informatics
King's College London
London, United Kingdom

Ricardo Neisse
European Commission
Joint Research Centre
Institute for the Protection and Security of the Citizen
Ispra, Italy

Duy Trong Ngo
School of Electrical Engineering and Computer Science
The University of Newcastle
Callaghan, Australia

Homayoun Nikookar
Faculty of Electrical Engineering
Mathematics and Computer Science
Delft University of Technology
Delft, The Netherlands

Maha Odeh
Interdisciplinary Centre for Security
Reliability and Trust
University of Luxembourg
Luxembourg City, Luxembourg

Ganda Stéphane Ouedraogo
University of Rennes 1, IRISA, INRIA
Rennes, France

Evangelos Pallis
Department of Informatics Engineering
Technological Educational Institute of Crete
Heraklion, Greece

Jordi Pérez-Romero
Department of Signal Theory and Communications
Universitat Politécnica de Catalunya
Barcelona, Spain

Adrian P. Popescu
Department of Communication Systems
Faculty of Computing
Blekinge Institute of Technology
Karlskrona, Sweden

Alexandru O. Popescu
Department of Communication Systems
Faculty of Computing
Blekinge Institute of Technology
Karlskrona, Sweden

Barbaros Preveze
Department of Electrical and Electronics Engineering
Cankaya University
Ankara, Turkey

Mickaël Raulet
INSA of Rennes, IETR
Rennes, France

Mubashir Husain Rehmani
Department of Electrical Engineering
COMSATS Institute of Information Technology
Wah Cantt, Pakistan

Walid Saad
Wireless@VT
Bradley Department of Electrical and Computer Engineering
Virginia Tech
Blacksburg, Virginia

Yasir Saleem
Department of Computer Science and Networked System
Sunway University
Selangor, Malaysia

Oriol Sallent
Department of Signal Theory and Communications
Universitat Politécnica de Catalunya
Barcelona, Spain

Olivier Sentieys
University of Rennes 1, IRISA, INRIA
Rennes, France

Xuemin (Sherman) Shen
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada

Hongjian Sun
School of Engineering and Computing Sciences
Durham University
Durham, United Kingdom

Alberto Trombetta
DiSTA
University of Insubria
Varese, Italy

Christos Verikoukis
Telecommunications Technological Centre of Catalonia (CTTC)
Barcelona, Spain

Jing Wang
Research Institute of Information Technology
Tsinghua National Laboratory for Information Science and Technology
Tsinghua University
Beijing, China

Jun Wu
School of Electronics and Information
Tongji University
Shanghai, China

Yang Yan
Research Institute of Information Technology
Tsinghua National Laboratory for Information Science and Technology
Tsinghua University
Beijing, China

Kung Yao
Department of Electrical Engineering
University of California, Los Angeles
Los Angeles, California

Yong Yao
Department of Communication Systems
Faculty of Computing
Blekinge Institute of Technology
Karlskrona, Sweden

Hervé Yviquel
INSA of Rennes, IETR
Rennes, France

Ning Zhang
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada

Ming Zhao
Research Institute of Information Technology
Tsinghua National Laboratory for Information Science and Technology
Tsinghua University
Beijing, China

Xiaofeng Zhong
Research Institute of Information Technology
Tsinghua National Laboratory for Information Science and Technology
Beijing, China

and

Department of Electronic Engineering
Tsinghua University
Beijing, China

Haibo Zhou
Department of Electrical Engineering
Shanghai Jiao Tong University
Shanghai, China

Nizar Zorba
Department of Electrical Engineering
Qatar University
Doha, Qatar

I

NETWORK ARCHITECTURE TO SUPPORT MULTIMEDIA OVER CRN

Chapter 1

A Management Architecture for Multimedia Communication in Cognitive Radio Networks

Alexandru O. Popescu, Yong Yao, Markus Fiedler, and Adrian P. Popescu

Department of Communications and Computer Systems, School of Computing, Blekinge Institute of Technology, Karlskrona, Sweden

Contents

1.1 Introduction

As mobility and computing become ever more pervasive in society and business, new mobile services are emerging for multimedia communication that are creating new challenges for telecommunication operators. New advanced infrastructure needs to be created to coexist with legacy infrastructures, which are expected to progressively fade out over a longer time period. Studies by Ericsson (November 2011) indicate, for instance, that mobile data traffic is expected to grow tenfold by 2016 (Release, Ericsson Press 2011). In particular, Internet traffic, especially video, is expected to drive the increase in mobile traffic by nearly 60% per year.

In addition, considering the rate at which the spectrum share of mobile devices is increasing and the customer base is expanding in densely populated areas, overcrowding in the operating bands is unavoidable, particularly for multimedia communication. Today, there are several possible solutions to solve the problem of spectrum crunch.

Long-Term Evolution (LTE) and LTE-Advanced (LTE-A) systems have been designed to support high data rates and a large number of users. However, even though the design objectives of LTE-A and WiMAX are to support high data rates and a large number of users, high-mobility devices located at the edge of a cell may experience degraded service levels [1]. The main reasons for this are limited possibilities of reconfiguring terminals and networks depending on spectrum availability, ineffective spectrum usage, and nonoptimal use of radio resources as well as insufficiently flexible deployment of base stations (BSs). Accordingly, these limitations of 4G systems, often referred to as 4G bottlenecks, are expected to become a major issue for quality of service (QoS) over the next couple of years.

Combining LTE and Wi-Fi in a single service is an interesting solution, although the slow adoption of femtocells by mobile industry has left the market open for low-cost (and often free) Wi-Fi service. The fact that femtocell technology still does not work properly has led mobile operators to promote the use of Wi-Fi to offload traffic from their networks. As of 2009, the US-based operator Sprint actively mandated that all smartphones used within its network should also be Wi-Fi capable, helping to relieve network load. In addition, this is a worldwide trend, where, for instance, KDDI Japan is already offloading as much as 50% of its wireless traffic to its 220,000 public hotspots, whereas AT&T has surpassed 2.7 billion Wi-Fi hotspot connections. Even developing countries in Africa and Southeast Asia are heavily investing in public access Wi-Fi.

Subsequently, Wi-Fi systems began to be used in combination with ad hoc mode, where aggregated Wi-Fi networks continue to expand provided that smartphones have the capability of directly connecting to other Wi-Fi enabled smartphones in ad hoc mode, bypassing the access point. This creates a small ad hoc network that can be extended to span several hops by including more Wi-Fi-enabled phones in each path. Finally, ad hoc networks spanning entire cities can be envisioned that would allow the mobile device to communicate without using any cellular infrastructure. Naturally, this would pose a serious threat to the revenues of current mobile operators.

Accordingly, new management architectures and business models must be developed to merge the interests of individual users, who want to make free or very cheap calls, with the operator objectives of retaining control and sources of revenues.

The basic idea of cognitive radio networks (CRNs) is to allow unlicensed users use underutilized spectrum in highly dynamic radio environments. Key features include, among others, pervasive wireless computing and communications, cognitive radio (CR) technology, IPv6, wearable devices with artificial intelligence (AI) facilities, and a unified global standard [2,3]. Nevertheless, the typical decentralized approach to CR functionality has dictated that adaptations to the operational parameters of cognitive radio devices (CRDs) have mainly been based on local observations made by individual CR users.

However, peer discovery without network support is typically time and energy consuming, requiring beacon signals with sophisticated scanning and security procedures [1]. By using LTE network assistance in terms of synchronization, identity and security management, beacon signal configuration, and reserving peer discovery resources, the discovery process of devices can be made more user-friendly and energy efficient. This is true regardless of the subsequent device-to-device communications taking place in the cellular spectrum or the use of noncellular technologies. Accordingly, it is important to mention that 3GPP has already done studies on the advantages of network-controlled discovery and communications for LTE Release-12 [1].

Moreover, CR functionality for multimedia communication requires the development of flexible terminals and a rethinking of the classical wireless network architecture [4,5]. For purposes such as flow control in spectrum sensing, selection, and sharing of radio environments, cooperation among protocols in different layers is necessary [4,6]. CRNs push the CR concept a step further by looking at the network as a association of CRDs. Typically, CRN architectures cover the whole TCP/IP protocol stack and do not conform to the strictly enforced traditional layered approach. To support video and audio streaming, there is a need to adopt a cross-layer design for information exchange at different layers in the protocol stack. New architectural models must be developed for communication management in CRNs [7]. Subsequently, a novel management architecture designed and developed at the application layer is advanced for horizontal spectrum sharing, also known as infrastructure-based CRN (currently under standardization by IEEE P.1900). For the objective of integration with low-power BSs in 4G networks, a centralized entity called the support node (SN) provides necessary functionality in terms of specific hardware and software [7].

The architectural solution enables expansion of the technological limits in 4G mobile systems to include CR facilities, and makes it possible to increase spectrum utilization within particular geographical areas. The theoretical spectral efficiency improvement of 4G systems has been shown to suffer from performance constraints in practical scenarios [1]. To increase the throughput, higher bandwidths together with transmissions over shorter distances are necessary. Integration with multihop CRNs can in this case yield higher throughput (i.e., more bandwidth through dynamic spectrum access) and also minimize the transmission distances (because direct communication with the BS is no longer necessary).

The CRN management architecture suggested in this chapter incorporates sensing and prediction, addressing and routing, and middleware and decision making. Accordingly, to coordinate spectrum access and exchange of signaling information, a common control channel (CCC) is necessary to enable cooperation among CR users [4]. Dimensioning of a CCC is considered in this chapter with regard to a specific CRN scenario. Particular focus is placed on the overhead required to maintain a network topology, which enables end-to-end (e2e) optimization of routing paths according to user and environmental constraints.

The rest of this chapter is organized as follows. Section 1.2 considers related work in the field. Section 1.3 describes the operations performed by SN and network members. Section 1.4 presents a basic CRN management scenario. Section 1.5 introduces the architecture for the suggested management framework. Section 1.6 is about CCC dimensioning aspects as well as implementation details for the network model used to communicate among network hosts. Further, with regard to secondary user (SU) service completion, Section 1.7 investigates the performance of a basic communication scenario through numerical analysis and simulation. Finally, Section 1.8 concludes the chapter.

1.2 Related Work

Recent research in the CR area has focused on new solutions for energy and spectral efficiency in wireless communication (green operation), spectrum assignment for opportunistic spectrum access, routing and handover mechanisms, and decision making and prediction algorithms to minimize the impact of SUs on primary users (PUs) [8–10].

For instance, Peng et al. [11] present a general model and utility functions for optimizing utilization in spectrum allocation. Given the spectrum heterogeneity, a framework for solving the spectrum access problem is suggested where the problem is reduced to a global optimization scheme based on graph theory. It is further pointed out that, because the global optimization problem is Non-deterministic Polynomial-time-hard, a heuristic approach to vertex labeling is required. Accordingly, a set of approximation algorithms is described for both centralized and distributed approaches to spectrum allocation, each with its specific advantages and drawbacks. Nevertheless, this study only considers a static network environment and focuses on optimizing a snapshot of the network at a particular time moment. For a dynamic network environment, the spectrum allocation problem becomes more complex, given the need for recomputations as the topology changes.

On the contrary, Won-Yeol and Akyldiz [12] advance a spectrum decision framework to allocate a set of spectra to CR users by considering the application requirements together with the dynamic nature of the spectrum bands. Two categories of applications are considered. These are real-time and best-effort applications where the suggested spectrum decisions are classified into minimum-variance-based spectrum decision and maximum-capacity-based spectrum decision. The aim is to minimize the capacity variance of the selected spectrum bands for real-time applications and maximize the total network capacity for best-effort applications. In addition, a dynamic resource management scheme is developed to adaptively coordinate the spectrum decisions dependent on the time-varying CRN capacity. This is achieved through a centralized resource manager located at the BS, which gathers information from CR users and accordingly performs spectrum decisions. Basically, CR users carry out two main tasks, which are spectrum sensing and quality monitoring. Subsequently, this enables a so-called event detection that allows the CRN to reconfigure its resource allocation as needed in order to maintain the service quality. Simply put, the advanced spectrum decision framework provides a hierarchical QoS guaranteeing scheme that consists of spectrum sharing (to assign channel and transmission power for short-term service qualities) and spectrum decision (to determine the best spectrum for sustaining the service quality over the long term).

In Ref. [13], the authors describe a spectrum-aware mobility management scheme for CR cellular networks. A novel network architecture is developed to support spectrum mobility management, user mobility management, and intercell resource allocation. Essentially, infrastructure-based CRNs are considered, which consist of multiple cells where each cell has a single BS with corresponding SUs in its coverage area. The cell coverage of each BS is furthermore regarded as the PU activity region. Hence, in order to avoid interference with the PU operation, each BS determines appropriate actions in support of an upper-level control node through observations and also through reporting of CR users located within the cell coverage area. This is possible because CR users are assumed to have a single wideband radio frequency (RF) transceiver that can simultaneously sense multiple contiguous spectrum bands without RF reconfiguration. The network determines a suitable spectrum band and also the target cell according to both the current spectrum utilization and the stochastic connectivity model. Increased cell capacity is, in other words, achieved through bandwidth aggregation by sharing spectrum owned by other cellular operators, or by opportunistically utilizing unused spectrum bands otherwise licensed to PUs. In addition, a number of four different hand-off schemes are defined for CRNs with regard to the mobility management of users and spectrum. Ultimately, a cost-based hand-off decision mechanism is suggested to minimize QoS degradation caused by user mobility.

Moreover, in Ref. [14] the authors focus on providing a systematic overview of CR communication and networking from the standpoint of the lower layers. In particular, the study investigates how the physical, medium access control, and network layers are involved in enabling multihop or relay communication in CRNs. It is concluded that the design of routing and control mechanisms in a multihop CR scenario requires cooperative CR users to optimize the overall detection capability of the CR network. In other words, spectrum-aware routing requires cooperative spectrum sensing that can take advantage of the dynamically available spectrum holes and adapt the path computations to the changing environment. Naturally, spectrum-aware routing, also called opportunistic spectrum routing, further requires support from intermediate (relay) nodes.

1.3 Network Model

A cognitive network is partitioned into a number of so-called CRNs. Every CRN is served by an SN and contains a number of SUs referred to as CRDs in the remainder of this chapter. The SN is responsible for populating the available spectrum opportunities (SOPs) within its geographical coverage area and also for keeping track of the current network members and operational conditions. A knowledge database is used to maintain all relevant CRN information. A CRD wishing to join a CRN must first contact the responsible SN by sending a join message to the SN requesting an operational zone (called a device spectrum opportunity [DSOP] in the continuation [15]) within an available spectrum opportunity in the particular CRN. SOPs can be partitioned in three categories with reference to the activity and holding times of licensed users: static, dynamic, and opportunistic [4,5,16]. However, in our CRN communication framework, only static/dynamic SOPs are considered because the holding time of an e2e path must be long enough to allow it to be traversed.

The functions provided by an SN for basic support of multimedia communication in a CRN are as follows:

1. Collection of information regarding the specific CRN, where data are collected from all network elements. This is used to identify and represent the available SOPs and to build a statistics database for decision making.
2. Support for two CRN routing mechanisms, greedy and optimized [7]. The routing mechanisms must also accommodate support for intra- and inter-CRN routing decisions, that is, computation of e2e routing paths among CRDs within the same or different CRNs.

3. Provision of bootstrapping and adaptation support for SUs. While SOPs are available in the CRN coverage area, an SN continues to allow SUs to join the CRN and to provide adaptations to network members for maintained service.
4. Spectrum sensing for the geographic coverage area of the CRN.

The functions provided by a CRD are as follows:

1. Spectrum sensing and monitoring regarding the used resources (i.e., channel utilization/occupancy). The collected information is used to detect possible conflicts in the CRN and update the deserving SN.
2. Routing and support for crossing routes, based on e2e routing decisions received from the particular SN. That is, network hosts are required to reserve resources for crossing communication paths to meet the e2e goals of different sources and destinations.
3. End-to-end decision making for opportunistic routing purposes. If no more static/dynamic SOPs are available in the particular geographical area, arriving CRDs must make their own decision regarding communication, which entails per-packet dynamic routing over opportunistically available channels with limited or no QoS guarantees. This means that SUs may be required to handle elastic traffic (i.e., share the same channel), which further degrades the existing QoS owing to resource sharing among a varying number of SUs generating different traffic.

A CRN is designed to cover an area in the range of a microcell network, that is, a radius of around 1000 meters. The goal is to provide added spectrum utilization and improved capacity at cells in crowded metropolitan areas such as office centers, airports, and malls. Compared to typical microcells where the spatial multiplexing interference created by the subdivision of cells is managed with the help of optimized power controls, a CR approach makes it possible to use the entire frequency spectrum, not just the fixed spectrum chunks allocated for operators [17]. This allows for optimization in both transmit power and the frequency domain, thus ensuring a considerable increase in the available communication capacity.

1.4 Basic Scenario

Four fundamental operations need to be controlled in the management of a CRN [4]: spectrum sensing, spectrum decision, spectrum sharing, and spectrum mobility. Another CRD-associated operation is mobility. Hence, ad hoc algorithms can be combined with spectrum mobility algorithms to provide an e2e solution for communication. Collaboration between CRDs and the SN is therefore vital and provides us with the necessary operations to communicate within CRNs. The joining of unlicensed users is done in a first-come, first-serve order through a scheduler to fairly partition the available DSOPs. If no static/dynamic DSOPs are available or predicted to be available in the CRN coverage area, the SN rejects the joining of newly arriving CRDs. Otherwise, after checking the available resources in its locally stored database, the SN computes a reply to the requesting CRD containing all the necessary information to allow it to join the CRN. A reply message typically contains parameters such as geographic point, frequency assignment, maximum power output, DSOP expected duration time, and a list of neighboring CRDs (already bootstrapped network hosts). Moreover, the SN constructs a database with information collected from all network elements. The collected data are used to identify and represent available DSOPs and to thus build a statistics database for future decision making [15].

The SN performs global spectrum sensing of the entire CRN coverage area to detect SOPs available for SU operation. Moreover, to identify local conflicts within the CRN, network members are expected to perform their own spectrum sensing for their respective channel assignments (depicted here as a DSOP, provided that an unknown modulation is used in the CRN, where every SU needs a portion of an available SOP to operate in). Any detected inconsistency must be communicated back to the SN for centralized decision making. This implies that a conflict in the DSOP of a network member may require an adaptation of its operational parameters, which is decided by the SN [15]. In addition, a CRD has the facility to communicate with another CRD in the CRN by conveying a communication request to the local SN. Depending on the type of request, unoptimized or optimized, the reply is either the virtual identifier (VID) of the destination or a complete optimized e2e path to reach it. That is, all intermediate hops and channel assignments along the path are optimized according to the requirements of the source.

1.5 Communication Architecture

The architecture demands an intelligent wireless communication system able to

- Monitor the radio environment continuously.
- Learn from the radio environment.
- Facilitate communication among multiple users.
- Adapt the e2e performance to statistical variations of the radio environment as well as user preferences and behavior.
- Facilitate self-adapting methods for communication.
- Solve diverse conflicts among users.
- Provide self-awareness.

The so-called cognitive engine, defined as a set of algorithms that together perform sensing, learning, optimization, and adaptation control of the CRN, is used [18]. Hence, our framework can be seen as a software suite implemented at BSs and CRDs with specific radio capabilities. The main components of the CRN management architecture (depicted in Figures 1.1 and 1.2) are middleware, software-defined radio (SDR), and overlays.

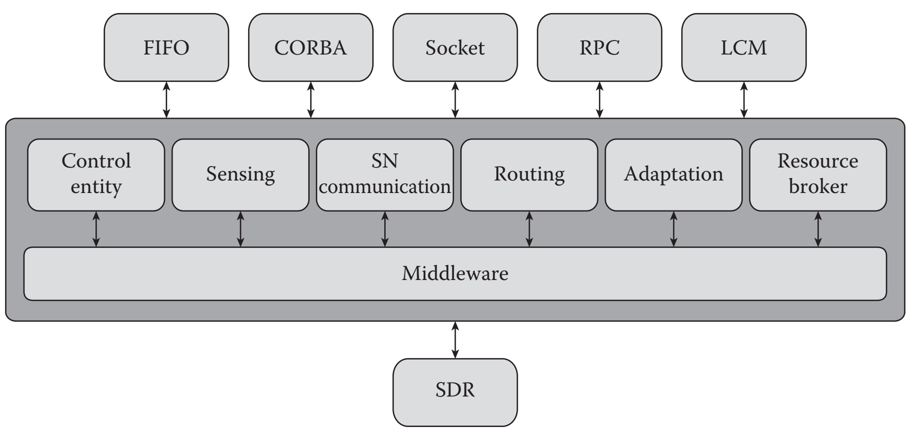


Figure 1.1 Cognitive radio device.

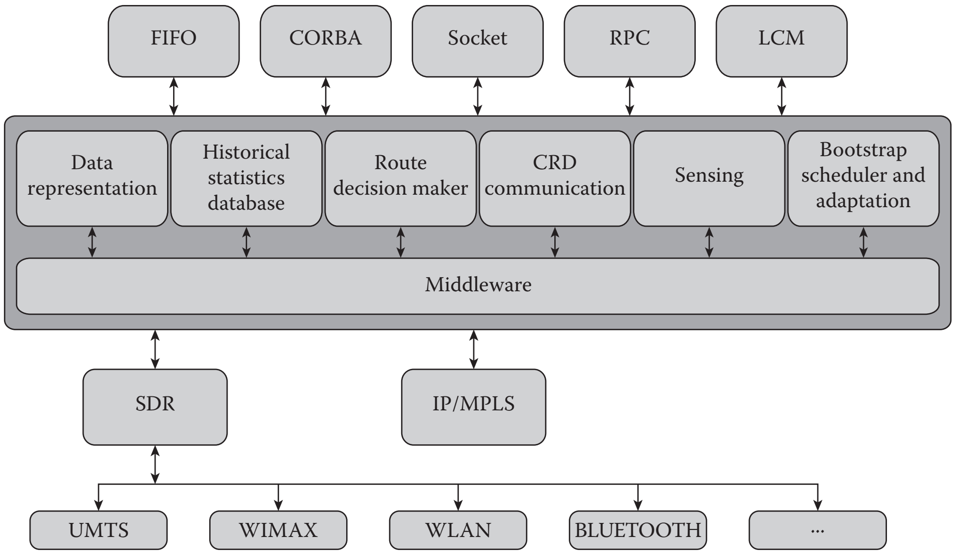


Figure 1.2 Support node.

The CRN architecture has been partially developed and implemented at Blekinge Institute of Technology. A short description of the components is as follows.

1.5.1 Middleware

Operator and user expectations in the CR area have governed the research for solving the complexity of management, security, and scalability provisioning. New BS designs need to be developed that can incorporate CR functionality and increase spectrum utilization through the use of dynamic spectrum access [15]. Because CRNs use low-layer (link layer and physical layer) information to improve radio communication performance, it is necessary to point out that full control of the lower layers is often limited owing to hidden information in the form of proprietary hardware and software. However, cooperation among protocols at different layers is necessary in a CRN for purposes such as flow and control over spectrum sensing (RF and transmit power output observations), spectrum selection, and spectrum sharing [4,6]. This type of direct communication between nonadjacent layers indicates the need for a cross-layer approach to ensure a functional implementation.

Because this approach conflicts with the strictly enforced traditional layered approach, a specific middleware, as presented in Figures 1.1 and 1.2, is used to address some of the stated problems. The middleware is software that bridges and abstracts underlying components of similar functionality, exposing it through a common application programming interface (API) originally developed for seamless handover purposes [19]. This offers the advantage of flexibility in present and future development of new services and applications. Furthermore, by using a middleware-based architecture with various overlays and underlays, the convergence of different technologies is simplified. The goal in this case is to develop a framework for communicating in CRNs, which requires minimal changes to the applications using the platform.

1.5.2 Software-Defined Radio

SDR is a class of reconfigurable radio where the physical layer behavior can be significantly changed as a consequence of changes in software; that is, the same hardware entity can perform different functions at different times. Simply put, an SDR is a device that provides RF and intermediate frequency functionality, including waveform synthesis in the digital domain [20]. This enables a CRD to relay messages to other CRDs and SNs over different channels (Figure 1.3).

1.5.3 Support Node

The SN is a centralized entity, implemented in the form of a software suite, that provides the following functionalities (in the form of overlays).

Data representation: A knowledge database for the data collected in the particular multihop CRN. In our case, this is based on the geometric structure of a datacube composed from several two-dimensional [0,1] × [0,1] Cartesian coordinate spaces (CCSs) [21]. Each CCS represents a different CR-dimension, where all dimensions are functions of time. Thus, the operational parameters of each network host can be described in every CR-dimension, for example, frequency (depicted as channels, depending on the available SOPs and used modulation [15]), power (transmit output power), and geographic point mapped to a set of virtual space coordinates (x,y) referred to as VID Φ. Because the selection is on the basis of the actual geographical location of the hosts in the physical CRN coverage area, visualization can be used to determine if additional CRDs can join the particular CRN or not. Every mapped device is represented in the data cube with a geometric portion of the space dimension CCS called zone, which surrounds VID Φ [7,21]. In addition, such devices have information about adjacent zones and the network member devices responsible for them. Because network hosts with geographical coordinates located close together access resources (such as SOPs) in the same area of the CRN, their VIDs are also mapped close

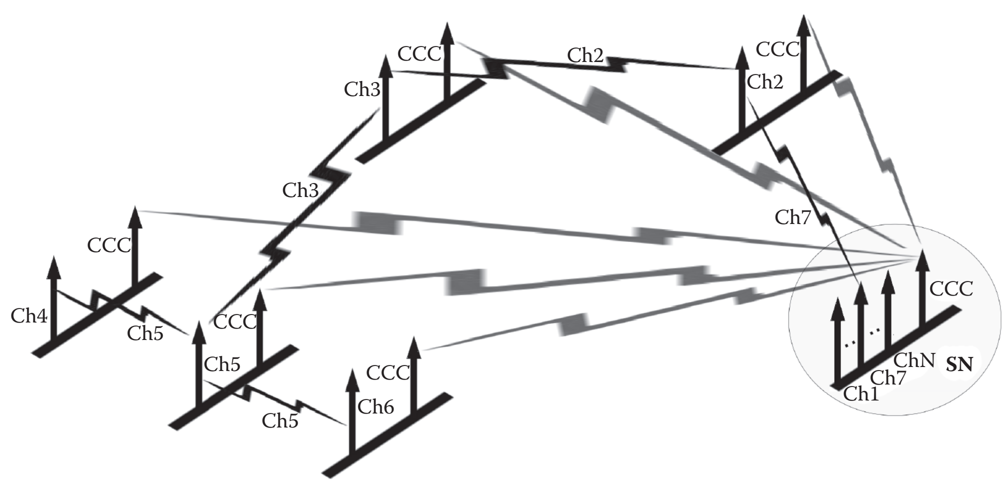


Figure 1.3 Message relaying between hosts in the CRN.

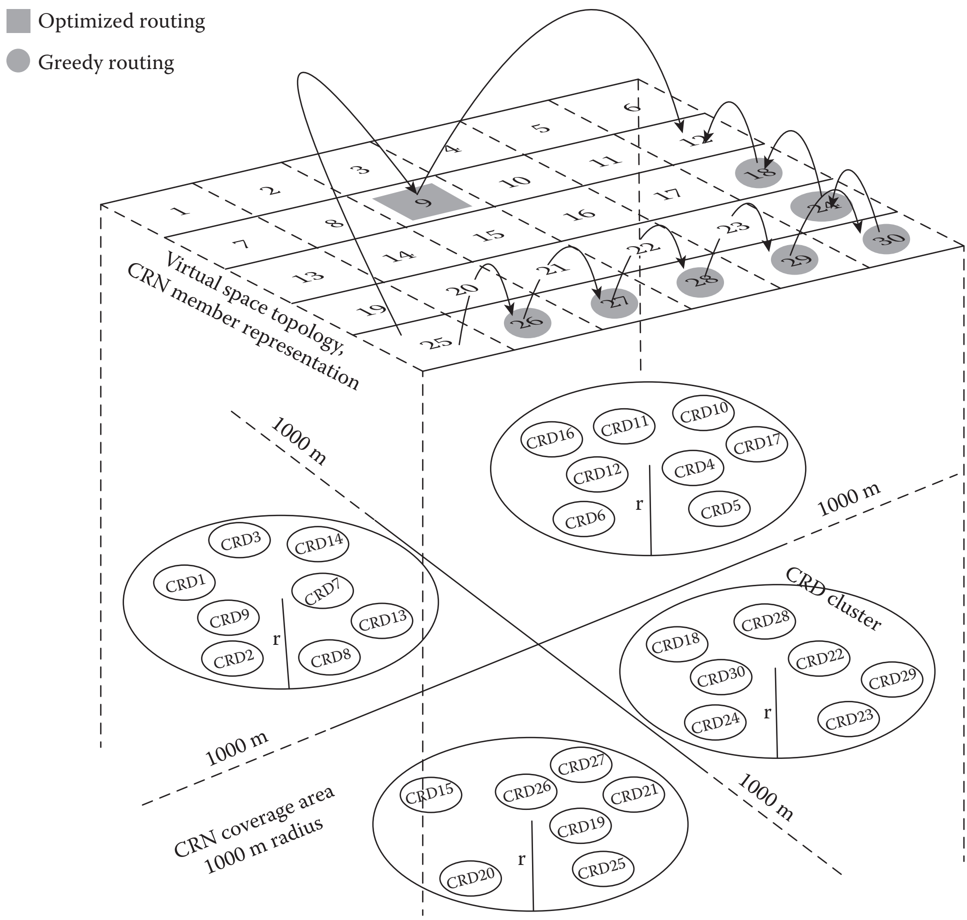


Figure 1.4 CRN overview depicting CRD clusters, optimized and greedy routing (From A. O. Popescu, et al., Communication mechanisms for cognitive radio networks, *11th IEEE Pervasive Computing and Communication*, San Diego, CA, March 2013.).

to each other in the virtual space dimension representation. Simply put, the SN imposes a particular geometric structure on its CRN coverage area to create a structured CRN topology (Figure 1.4).

Historical-statistics database (HSDB): The collected CRN data are subject to constant changes, where moving devices means that PUs may come and go, altering communication relationships in the CRN. Therefore, e2e path computations should not be based solely on current available operational states, but also on associated statistics computed and measured over long time periods for the particular CRN coverage area. Prediction models can be used to determine the holding times of DSOPs, which an optimized e2e path is set to traverse. Thus, having historical statistical data recorded in the HSDB overlay enables the decision maker algorithms to learn from experience and to more accurately predict future changes through methods such as the ones suggested in Refs. [22,23].

Decision maker: This is used for e2e route computation. The operational data collected by the SN and network members, together with HSDB-retained information, allow the decision maker overlay to compute an e2e route according to user preferences. The e2e path is subject to centralized decision making with reference to a particular temporal framing and various QoS/QoE parameters, for example, cost, throughput, delay, service availability, security, and privacy levels. Different mechanisms can be used for decision making, for example, context-aware, fuzzy logic, analytic hierarchy processing [22,24]. The computed e2e path is propagated back to the requesting CRD.

CRD communication: This function handles all communication between the SN and the CRD terminal. Specific messages are exchanged over a CCC, which is conceptualized to handle all necessary data exchange with the SN, for example, bootstrap request, adaptation request, update request, communication request, and resource reply. This overlay collects all required information from the necessary overlays and properly formats the CRD messages before they are transmitted out on the CCC.

Additional functionalities: The SN provides other functionalities, for example, bootstrapping procedures, for joining unlicensed CR users, keeping the data representation up to date through global spectrum sensing of the CRN coverage area and updates received from the network members, maintaining a member list of all current network hosts and associated operational parameters in different CR-dimensions, and handling inter-domain communication hand-off to other SNs. Because a device is bootstrapped one SN at a time, handover schemes for devices moving from one CRN to another must also be developed.

1.5.4 Cognitive Radio Device

The following overlays are used in the CRD architecture (software suite).

Control entity: It handles the actions related to user control, contextual-awareness, and security facilities. This refers to informing the SN about the user context and service-defined preferences, that is, the type of service as well as other relevant information for the e2e route computation. This overlay offers the end user the possibility of defining preferences for best decisions to be taken with the help of generic models, distributed measurements, and data exchange.

Sensing: This handles actions related to multiband operation and fast frequency scanning in collaboration with the SDR underlay. Gathering spectrum usage information enables forecasting for licensed and unlicensed users as well as for the user itself (e.g., user activity, available resources, movement prediction). This requires solving a number of challenging research questions such as the hidden PU problem, detecting spread spectrum PUs, sensing duration and frequency, cooperative sensing, and security.

SN-communication entity: This handles signaling communication between the terminal itself and the SN. This is done over a CCC and through specific message types formatted to handle all necessary data exchange, for example, bootstrap reply, adaptation reply, update reply, communication reply, and resource request.

Routing: This function controls the routing so that the destination is reached. Depending on the type of requested communication, that is, unoptimized or optimized, the SN replies either with a message containing the VID of the destination or with a complete optimized e2e path. The e2e path is computed according to user preferences, and the reply message from the SN has all the information necessary to reach the destination, that is, intermediate CRDs (all hops) together with channel assignments along the path that are optimized according to, for example, cost, throughput, delay, and security level [7].

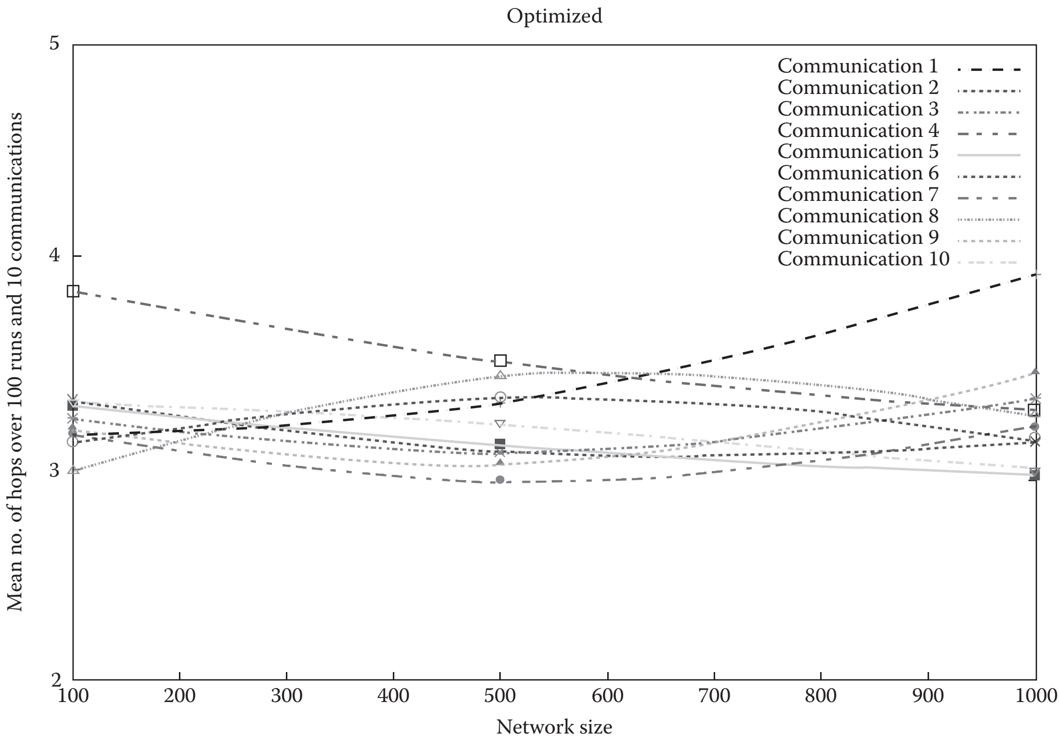


Figure 1.5 CRN cluster-optimized routing performance.

Unoptimized routing employs a distributed hop-by-hop routing model (greedy routing, Figure 1.5) toward the VID of the destination without a comprehensive network overview. That means we have in this case limited or no QoS guarantees. In other words, a source only requires the VID of the destination from the SN and routes toward it according to a rectilinear distance geometry along the *x* and *y* dimensions. The VIDs of neighbors in the virtual space topology are cross-checked with the VID of the destination, and the neighbor closest to the destination is selected as next hop [7]. Two network members are considered neighbors if the coordinate spans of their geometric zones overlap along one dimension and abut along the other dimension. On the contrary, an SN-computed and -optimized routing path allows for the e2e goals of the source to be achieved at the cost of larger overhead and computational latency. The SN imposes different constraints on the routing path, denoting ad hoc packet routing with aggregated throughput obtained by employing suitable multiobjective, multiconstrained optimization algorithms [7,15]. The destination can thus be reached according to specific QoS framings that depend on the particular domain, networking conditions, and user requirements. If an intermediate device (a CRN network host) between a particular source and destination fails during an ongoing communication, a recomputation of the e2e path may be required. The routing overlay at the affected device (i.e., for which its next hop is no longer available) must contact the SN to request an updated route to the destination, if one is still available. This may be necessary if a PU takes over the DSOP of an SU that is part of the CRN and the particular e2e path. However, if the SN fails or no other network host with a suitable DSOP holding time is available, the routing overlay at individual devices must make its own decision regarding communication. This entails a fully distributed per-packet dynamic routing over opportunistically available channels, that is, limited or no QoS guarantees as in the unoptimized routing case.

Adaptation: While sensing the operational environment, a network host may discover that adaptations are needed in one or more CR-dimensions to maintain service continuity [15]. This can occur owing to node mobility, where a moving device is no longer able to access its assigned resources (such as a DSOP) and needs to update them according to its new geographical location. The particular device sends an adaptation request to the local SN and waits for a reply. When a reply message is received, the actual changes to the operational parameters are executed by the "Adaptation" overlay of the device. The changes may require adaptations to be done in different CR-dimensions, for example, frequency, power, and space, and entail collaboration with the SDR underlay, which allows for the physical layer behavior to be modified as a consequence of these changes.

Resource broker: It monitors the available resources of the particular CRD. Every network host is required to keep track of its vital operational parameters, for example, remaining processing capacity, remaining battery capacity, and available bandwidth of its assigned data channel (DSOP). It is necessary to point out that even if all data channels have the same initial capacity (assuming a uniform modulation), the available bandwidth may vary depending on the geographical positions of the network hosts (subject to interference from other devices and obstacles affecting some DSOPs more than others), ongoing communications (current channel load due to, for example, elastic traffic [25]), and other environmental factors. Hence, the actual channel capacity available to a network host at a specific time must be determined either through periodic or on-demand updates among the network members.

1.6 Common Control Channel

The CCC is intended for CRN control purposes, which means that it must meet the signaling needs between SNs and CRDs in the CRN framework. Toward this goal, we use a CCC that is available at all CRDs (global coverage) in the CRN all the time [4,26]. To avoid interference with the licensed bands, CCC is placed in out-of-band frequencies such as the unused parts of the UHF band recently released by the discontinuation of analog TV. Unlike in-band CCC, where channels used for data transmissions are also used to convey control messages, an out-of-band CCC separates control data from the user data transmission. To access the CCC, network members are thus required to have a dual-mode radio, that is, an extra narrow-band low-bit-rate transceiver to exchange control messages with the SN. Even though out-of-band CCC frequencies are used, it is assumed that they may change with respect to geographical domains, because no worldwide harmonization of the CCC exists today, and different frequencies may be available in different countries. Three different possibilities are available for CCC detection by SUs: out-of-band, in-band, and combined solution [27]. The suggested CRN framework is based on a centralized cognitive framework and aimed for BS implementation. Typically, the cognitive framework can be implemented in different ways, such as centralized, distributed, or mixed (a hybrid between centralized and distributed), each with specific advantages and drawbacks [28]. An in-band CCC detection method is considered most suitable for our purposes [4].

In addition, information about the modulation (radio access technology) used for the particular out-of-band CCC must be conveyed via the initial in-band signaling as well. This allows a CRD that wishes to join a CRN to directly tune into the out-of-band CCC, thus removing the need to sense the entire frequency spectrum to discover it. A typical scenario occurs when users wish to join the CRN because of service starvation owing, for example, to overcrowding in the regular operating bands of a particular operator. Because CRN management requires an exchange of data among network members, the information carried over the CCC consists of on-demand compiled messages; that is, information is only communicated over the CCC by a terminal request.

1.6.1 CCC Model

The exchanged signaling messages among network members are mainly generated owing to node mobility and the complexity of the particular CRN scenario, thus creating the need to update the data (topology) maintained at the local SN. The frequency of these updates is related to the complexity of the particular scenario; that is, it depends on the CRN coverage area characteristics and population size. Hence, different CRNs may need to use the CCC differently. Some CRNs may require a more frequent use of the CCC, whereas others may require less frequent use. This indicates that the bandwidth needed for a CCC can vary with different geographical domains. Given that message types of various sizes are exchanged over the CCC (between the SN and network members), the amount of data necessary to represent the different parameters (in bits) is computed as follows.

We assume the following information is available:

1. The device id D_{id} identifies and differentiates between network hosts (CRDs). To keep the identifier size small, the id is the phone number of a user, because the aim is to implement CR functionality at BSs with CR-available capabilities. A 67-bit number is used to represent the (phone) numbers, 34 bits of which are used for the actual number (allowing for 10-digit numbers), and the remaining bits are reserved for prefixes if a host in the local CRN wishes to communicate with non-CR users located in other geographical domains.
2. The frequency band interval between F_{min} and F_{max}, which specifies the DSOP where the available communication channel for a particular host is located. For simplicity, frequency modulation is assumed, and the CRN operating frequencies are in the range of 0 Hz to 10 GHz [27]. Thus, two variables of 34 bits each can be used to define the minimum and maximum frequencies.
3. Geographic point information, G_{gps}, given by the global positioning system (GPS) latitude and longitude coordinates. The latitude is defined as XX° YY′ ZZ″ together with a north/south pointer G_{NS} that can be represented by 1 bit. The range of latitude degrees XX° is from 0° to 90°, and it can be encoded by 7 bits. One degree of latitude is 60 arc minutes (approximately 100 km), implying a range of 0 to 59 arc minutes and seconds, respectively. This requires 6 bits each. An additional 7 bits are used to represent the hundredths of a second (from 0 to 99) for increased positioning precision. Meridians of longitude are similar, though ranging from 0° to 180° instead, thus adding another bit to the set of coordinates. Moreover, a west/east pointer G_{WE} represented by 1 bit is used to identify on which side of the prime meridian (the zero line, which passes through Greenwich, UK) the longitude coordinate is located in. Altogether, this results in a total of 55 bits to represent a complete G_{gps} set of coordinates with a positioning precision of approximately 0.3 m.

4. The maximum transmit power is used by the SN to set the output level for network members accessing specific DSOPs in the CRN coverage area. The power control makes it possible for network hosts to operate in the CRN with a minimum of interference to other devices. However, not all devices usually have the same maximum output power capabilities, and the required power classes (maximum output power level) typically vary with the used bands and modulations. Considering power class 1 mobile devices, a maximum transmit output power of 2000 mW is feasible, which requires 11 bits to define the output power variable P_{max}.
5. The expectation time, which is the time period during which a particular SOP is expected to be available for SU operation. This information can, for example, be obtained based on long-term traffic measurements and analysis of a particular CRN coverage area as well as a statistical prediction method. A SOP duration foresight of maximum 1 day, or 86,400 seconds, is assumed, which requires two variables to be communicated, t_{start} and t_{end}. t_{start} depicts (in seconds) the time moment when a particular SOP becomes available, and t_{end} depicts (in seconds) the time moment when a particular SOP's availability expires. Thus, 17 bits are needed to represent each of the two variables t_{start} and t_{end}, that is, 34 bits in total. To ensure that unpredicted SOPs expire the next day, the t_{end} variable of the reply contains a maximum value of 86,400. Assuming no unexpected events (such as PUs unexpectedly taking over the used SOP), a stationary host accessing a DSOP (within a static SOP) may continue to use the same DSOP even after the first expectation time expires. However, an updated expectation time must first be requested from the SN, thus implying that every CRN member must contact the SN at least once a day to exchange control information.
6. The starting t_{min} and ending t_{max} time moments define the time interval when a network host requires an optimized e2e path. This is different from the DSOP expectation time interval described earlier. For simplicity, a maximum waiting time of 100 seconds is assumed, which requires 7 bits to represent each of the two variables t_{min} and t_{max} (i.e., 14 bits in total, where $t_{min} < t_{max}$).
7. Virtual space coordinates are imposed by the SN to create a structured CRN topology. Identifiers of current network members in the CRN geographical coverage area are mapped to a set of virtual space coordinates (x,y) (in the CCS representation of the CRN space dimension) referred to as VID Φ. The Φ virtual coordinates are represented by 8 bits per CCS axis, x and y; that is, a total of 16 bits are required to represent a complete set of CRN virtual space coordinates $VID_{x,y}$. This is enough to represent the virtual space coordinates for a network population of thousands of devices, and it can easily be expanded for larger network populations.
8. The virtual cluster radius length is referred to as R_{length}. Because several hosts may be available throughout different geographical areas of a CRN (for a microcell coverage area), virtual clusters are created by the SN, which groups together network members depending on location and accessed resources (such as SOPs) in the particular CRN geographic area. That is, the CRN coverage area is subdivided into clusters, where each cluster comprises a number of member devices. The radius of the clusters is not a fixed number, and it may vary from one CRN to another. A device is considered to belong to a particular cluster if its geographic coordinates are within the radius of that particular cluster. The virtual cluster radius is represented by 17 bits, for a maximum cluster radius of 700 m represented in centimeters. Naturally, the most basic cluster scenario is when the entire CRN area (a microcell of 1000 m radius) is covered by only two clusters (Figure 1.6).

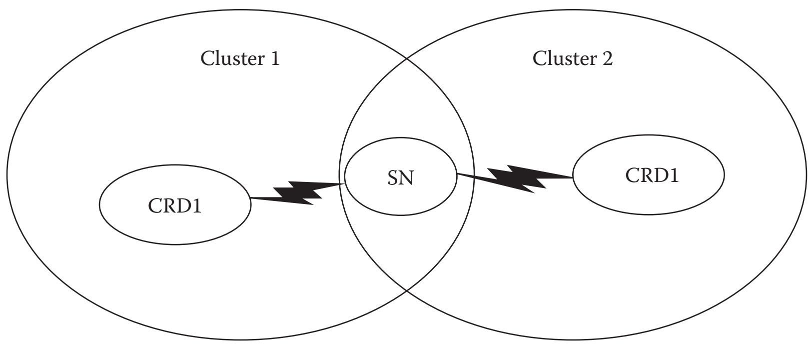


Figure 1.6 Basic scenario with two clusters.

Table 1.1 CRD Message Types

Update request	$D_{id}(i)$, $D_{id}(n)$, A
Adaptation request	$D_{id}(s)$, $G_{gps}(s)$, A
Bootstrap request	$D_{id}(s)$, $G_{gps}(s)$, A
Communication request	$D_{id}(s)$, $D_{id}(d)$, A, t_{min}, $t_{max}(s)$, $Cv(s)$
Resource reply	$D_{id}(s)$, $Rv(s)$, A

Devices in the same cluster are represented as neighbors in the CRN virtual space topology (illustrated in Figure 1.4). Considering the geographical proximity between adjacent clusters, devices therein are assumed to be able to communicate directly without any intermediate relays. However, this requires an initial dimensioning of the cluster sizes in CRNs with regard to the specific environmental constraints to ensure limited signal degradation and maximum throughput for the available channels, during inter- and intracluster communication among devices. The number of clusters used in a specific CRN depends on the cluster sizes, and we consider factors such as local, geographical, or regulatory aspects; required uplink/downlink data rates by devices in the particular geographical area; rated power output of devices in the CRN; and fading characteristics of the channels in the coverage area caused by obstacles [7,15].

Furthermore, because devices in the CRN are not always stationary, this process can potentially be very disruptive given the need to update the network topology and the used resources [15]. A communicating device moving away from the location where it accesses an available DSOP may have insufficient signal strength to maintain the connection. This can cause signal-to-noise ratio (SNR) and bit-error-rate (BER) problems. Thus, the SN must provide topology updates, initiated on demand by requests from network hosts. In other words, network hosts are required to keep track of their own movements and cluster boundary crossings, which further implies that every device must be aware of its own current location and the coverage area of the cluster it resides in. This can be done through bootstrap and adaptation messages where the center coordinates and cluster radius are provided together with other necessary information, as shown in Table 1.1. In such a case, a simple comparison of the current geographical coordinates in relation to the center coordinates and radius of the cluster allows a particular network host to be aware of when the cluster boundaries are crossed. The geographical coordinates can be any type of positioning coordinates, although for simplicity purposes we only consider GPS coordinates.

The parameters described earlier are sufficient to compute most of the control messages used in the CRN management architecture. Table 1.1 depicts the CRD-generated messages *bootstrap request, adaptation request, update request*, *communication request*, and *resource reply*. Each message type is used by network hosts to pass a number of relevant parameters to the SN, where the variable A can take values that satisfy $0 \leq A \leq 4$, and it is used to inform the SN about the type of CRD message received:

Update request: The value $A = 0$ informs the SN that this is an update request message from intermediate device i with ID $D_{id}(i)$. This message type is used if a network host detects that one of its neighbors is no longer available. As such, this requires an update of the particular host's neighbor list and the network topology maintained at the SN. For instance, if an intermediate device along an e2e path fails in its attempts to forward packets to one of its neighbors, the neighbor is considered void. An update neighbor list request is send to the deserving SN, which also allows it to update the network topology. $D_{id}(n)$ is the id of neighbor n (in the neighbor list of intermediate device i) that is no longer reachable.

Adaptation request: The value $A = 2$ informs the SN that this is an adaptation request message from source device s with ID $D_{id}(s)$, where $G_{gps}(s)$ contains its current geographical position. This message type is used if a network host requires adaptations to its operational parameters as a result of mobility or a PU taking over its DSOP. Provided the particular device has moved to a new geographical location (located in a different cluster), its entire neighbor list must be updated as well.

Bootstrap request: The value $A = 2$ informs the SN that this is a bootstrap request message from source device s with ID $D_{id}(s)$, where $G_{gps}(s)$ contains its current geographical position. This request is similar to the adaptation request, although with a lower priority assigned to it. This is because preserving service continuity for already bootstrapped network members is considered to be more important than bootstrapping new arrivals. That is, network hosts have priority in accessing available resources in the CRN, whereas newly arriving CRDs are only bootstrapped if resources are available and not needed for any pending adaptation requests in the particular area of the CRN.

Communication request: In the greedy routing approach, no path optimization mechanisms are employed. This means that no e2e goals are defined in the context vector $Cv(s)$ of source device s. Subsequently, only the source $D_{id}(s)$ and the destination $D_{id}(d)$ IDs are included in the communication request message to the deserving SN. However, in the case of an optimized routing request, the user context parameters (such as service-defined preferences, i.e., type of service) for the path must also be conveyed. QoS requirements are in such a case enclosed in the context vector $Cv(s)$ of source device s and include the following parameters: minimum required throughput Th_{min} for the service, the starting t_{min} and ending t_{max} times for the time interval when the resources required by source device s must become available (this is different from the DSOP availability time interval t_{start} and t_{end}), the maximum delay Δ_r tolerated for the route, the maximum allowed cost Co_{max} for the particular service, and the minimum expected security level Se_{min} [15]. The value $A = 3$ informs the SN that this is a communication request from a source device with ID $D_{id}(s)$ to a destination device with ID $D_{id}(d)$.

Resource reply: This is used to reply to an SN-initiated resource request. Every network host included in the initial request must relay its currently available resources to the SN. Thus, the parameters relayed in resource vector $Rv(s)$ by the source device s comprise the device's remaining processing capacity, battery capacity, security capabilities, and the available channel capacity of its assigned data channel. The data channel characteristics of a network host are typically described by several metrics, for example, throughput, delay, and security level. However, the security level over a channel depends on the capabilities of the particular transmitter and receiver, and it can be determined by the SN from a consideration of their commonly supported protocols. The value $A = 4$ informs the SN that this is a resource reply message from the source device with ID $D_{id}(s)$.

Table 1.2 shows the SN-generated messages (depending on the received requests), which are *update reply, adaptation reply, bootstrap reply*, *communication reply*, and *resource request*. Each message type is used by the SN to pass a number of relevant parameters to network hosts, where the variable B can take values satisfying $0 \le B \le 4$, and it is used to inform the hosts of the type of message received.

Update reply: $D_{id}(s)$ is the ID of source device s that the response is aimed at. $D_{id}(n)$ is the ID of the new neighbor n of source device s. $Ch_{Fmin,Fmax}(n)$ comprises the channel assignment of neighbor n (given frequency modulation as mentioned earlier). The value $B = 0$ indicates that the received message is an update message.

Table 1.2 SN Message Types

Adaptation Reply
$D_{id}(s)$, $Ch_{F_{min},F_{max}}(s)$, $P_{max}(s)$
$VID_{x,y}(s)$, $t_{start}(s)$, $t_{end}(s)$, $D_{id}(n \mid n = 1, 2,\ldots x)$
$Ch_{F_{min},F_{max}}(n \mid n = 1, 2,\ldots x)$
$VID_{x,y}(n \mid n = 1, 2,\ldots x)$, $G_{gps}(c)$, $R_{length}(c)$, B
Bootstrap Reply
$D_{id}(s)$, $Ch_{F_{min},F_{max}}(s)$, $P_{max}(s)$
$VID_{x,y}(s)$, $t_{start}(s)$, $t_{end}(s)$, $R_{length}(c)$, B
$D_{id}(n \mid n = 1,2,\ldots x)$, $G_{gps}(c)$
$Ch_{F_{min},F_{max}}(n \mid n = 1, 2,\ldots x)$, $VID_{x,y}(n \mid n = 1, 2,\ldots x)$
Communication Reply
$D_{id}(s)$, $D_{id}(d)$, $D_{id}(i \mid i = 1, 2,\ldots x)$
$Ch_{F_{min},F_{max}}(i \mid i = 1, 2,\ldots x)$, B
Update Reply: $D_{id}(s)$, $D_{id}(n)$, $Ch_{F_{min},F_{max}}(n)$, B
Resource Request: $D_{id}(s)$, B

Adaptation reply and bootstrap reply: These messages are identical, although the computational priority at the SN is higher for the adaptation message. $D_{id}(s)$ is the ID of source device s that the response targets. $P_{max}(s)$ is the maximum allowed transmit power output for the requesting device, whereas $Ch_{Fmin,Fmax}(s)$ is its new channel (i.e., DSOP) assignment [20]. $t_{start}(s)$ and $t_{end}(s)$ define the time interval during which the assigned DSOP is predicted to be available for SU operation, which in this particular case is source device s. $\text{VID}_{x,y}(s)$ is the VID of the requesting source device. $D_{id}(n \mid n = 1, 2,\ldots x)$ is the neighbor list of the source device, provided there are x neighbors. $Ch_{Fmin,Fmax}(n \mid n = 1, 2,\ldots x)$ denotes the channel assignments for the neighbors, and $\text{VID}_{x,y}(n \mid n = 1, 2,\ldots x)$ denotes the VIDs of the neighbors. $G_{gps}(c)$ denotes the center GPS coordinates for virtual cluster c, and $R_{length}(c)$ is the radius in centimeters for virtual cluster c, within which the particular source device is located. The value $B = 1$ indicates that the received message is an adaptation message, whereas the value $B = 2$ indicates a received bootstrap message (differentiation is needed as the two messages have different priorities).

Communication reply: $D_{id}(s)$ is the ID of source device s that the response is aimed at. $D_{id}(d)$ is the ID of the destination device d in the requested route. $D_{id}(i \mid i = 1, 2,\ldots x)$ denotes the IDs of all intermediate nodes i along the path to the destination, and $Ch_{Fmin,Fmax}(i \mid i = 1, 2,\ldots x)$ encloses the channel assignment for each respective intermediate hop. The variable $B = 3$ indicates that the received message is a communication message. The routing path is compiled by the SN according to e2e goals by using the optimization algorithms in Ref. [15].

Resource request: This is used by SN to request the present state of available resources at a particular source device s with ID $D_{id}(s)$. This message is typically transmitted to network hosts located throughout regions that an optimized e2e path must traverse. The value $B = 4$ indicates that the received message is a resource request message.

A computational example of the size of an adaptation reply message (ARM) compiled by the SN is as follows:

$$\begin{aligned} ARM = {} & (D_{id}(s)) + (Ch_{Fmin,Fmax}(s)) + (P_{max}(s)) + (D_{id}(n \mid n = 1, 2,\ldots x)) + (Ch_{Fmin,Fmax}(n \mid n = 1, 2,\ldots x)) \\ & + (t_{start}, t_{end}(s)) + (VID_{x,y}(s)) + (VID_{x,y}(n \mid n = 1, 2,\ldots x)) + G_{gps}(c) + R_{length}(c) + (B) \end{aligned}$$

Assuming that a network host has four neighbors in its new geographical location, the size (in bits) of ARM can be calculated as follows:

$$ARM(4) = 80 + (44 + 44) + 14 + 4 \times 80 + 4 \times (44 + 44) + (20 + 20) + 16 + 4 \times 16 + 55 + 17 + 3 = 1049$$

Likewise, a bootstrap reply message can be computed with similar information and size, whereas other control messages may contain more or less data. For instance, in the case of a communication reply message (CRM) where an e2e path optimized by the SN is replied back to the requesting host, the distance between the source and the destination can vary for different requests. A variable number of intermediate hops along the path implies routing overheads of different sizes. Typically, the overhead contained in an optimized CRM needs to accommodate device IDs (for source, destination, and intermediate hops), channel assignments (for every intermediate hop to the destination), and message type identifier. Channel assignments are necessary for optimized routes given that network hosts only have such information for adjacent neighbors. In our case, an optimized path entails bypassing the adjacent neighbors and communicating directly with network hosts located in a neighboring cluster, naturally within the limits of a device's capabilities (such as, transmit power output). Hence, an optimized routing path typically reaches the destination through less intermediate hops as compared to a greedy routing approach (Figure 1.4), even for distant spacing between the source and destination. Put differently, the overhead enclosed in the CRM that describes an optimized path is lightweight, thanks to an optimization mechanism that reduces the number of intermediate hops from the source to the destination [7].

1.6.2 Implementation Details

Assuming equally sized clusters with a radius of not less than 150 m, the maximum number of viable virtual clusters in the CRN coverage area described earlier (a microcell with 1000 m radius) is 36 [7]. Depending on the particular CRN environment, clusters with both smaller and larger radii may be necessary. It is important to mention that, in the case of a smaller cluster radius, the signaling requirements may also increase owing to node mobility and the exchange of ARMs. Hence, the CCC must be dimensioned to handle signaling loads in accordance with the particular scenario, that is, the used number of clusters.

To estimate the amount of overhead required for optimizing communications with a particular 36-virtual-cluster solution (i.e., a medium-cluster-size scenario), we have implemented a topology construction algorithm for the two-dimensional $[0,1] \times [0,1]$ CCS representation of the CRN space dimension [7]. The virtual space topology construction algorithm has been implemented with the C++ object-oriented programming language, and it is used to simulate the mean number of hops (path lengths) required for 10 random communications over 100 runs and different network populations (n) of 100, 500, and 1000 simulated devices. The communications per simulated run and network population are done for randomly selected network hosts, where each communication is optimized according to the shortest path and least cost constraints. In Figure 1.5, it can be observed that, for each network population, the mean path length per simulated communication over 100 runs is between 3 and 4 hops, regardless of the network population. In other words, provided the CRN coverage area is subdivided into virtual clusters where each cluster is responsible for the hosts within its area, the mean path lengths converge according to a uniform distribution, regardless of the network population. That is, the routing tables no longer grow with the network population [7,15].

Thus, the average path length (regardless of the network population) that can be expected for a random communication in a CRN optimized according to a 36-virtual-cluster subdivision of its coverage area, is computed with

$$\overline{H_n} = \frac{1}{30}\sum_{c=1}^{30}\overline{h_c} \tag{1.1}$$

$\overline{H_n}$ is the average path length, computed from the mean path lengths of all 10 simulated communications. $\overline{h_c}$ indicates the mean path length for each respective communication taken over all runs per network population. Assuming 10 simulated communications and network populations (n) of 100, 500, and 1000 simulated devices, a total of $10 \times 3 = 30$ mean path lengths is obtained. Accordingly, from the mean path lengths presented in Table 1.3, the grand average path length for this particular case (computed with Equation 1.1) is 3.25. In addition, over the 100 simulated runs, 95% of the communication path lengths for each network population (n) of 100, 500, and 1000 simulated devices are within ±3.1%, ±3.0%, and ±3.2%, respectively, of the grand average path length. The 95% confidence interval (CI) is computed for each network population:

$$CI = \overline{H_n} \pm 1.96\frac{\sigma}{\sqrt{n}} \tag{1.2}$$

Table 1.3 Mean Path Length per Communication over All Runs and Different Network Populations

Network Population	*Com1*	*Com2*	*Com3*	*Com4*	*Com5*	*Com6*	*Com7*	*Com8*	*Com9*	*Com10*
100	3.17	3.33	3.24	3.84	3.30	3.13	3.17	3.00	3.20	3.32
500	3.32	3.09	3.09	3.51	3.12	3.34	2.95	3.44	3.03	3.22
1000	3.92	3.13	3.33	3.28	2.97	3.15	3.20	3.26	3.46	2.99

$\overline{H_n}$ is the average path length, computed from the mean path lengths of all 10 simulated communications. The critical value for the 95% CI is 1.96, with a standard normal (0,1) z-distribution. σ is the standard deviation, given by $\sigma = \sqrt{Var(x)}$, where $Var(x)$ is the variance of the grand average path length for each network population, taken over all communications and runs. n is the total number of communications performed over all runs; that is, given that 10 communications are simulated for each run, the total number of communications over 100 runs per network population is $100 \times 10 = 1000$. Subsequently, the amount of CRM overhead for this particular case can be computed as follows:

$$CRM = (D_{id}(s)) + \left(\bar{x} \times (D_{id}(i \mid i = 1,2,\ldots x)) + (Ch_{F_{min},F_{max}}(n \mid n = 1,2,\ldots x))\right) + (B)$$

$D_{id}(s)$ is the source device id, and $D_{id}(i \mid i = 1, 2,\ldots x)$ is the id of the intermediate hops, given a total number of x intermediate hops (including the destination, which is the last hop). $\bar{x}$ is the average path length for a random optimized communication according to the suggested virtual cluster solution. $Ch_{Fmin,Fmax}(i \mid i = 1, 2,\ldots x)$ denotes the operating frequencies (channel assignments) of all intermediate hops along the path, including the destination. We assume a frequency modulation where each network member occupies a channel in the frequency range F_{min} to F_{max}. The value $B = 3$ indicates that the received message is of type CRM. Given an average path length of 3.25 hops, the amount of overhead (in bits) required to optimize communications according to the particular 36-virtual-cluster solution is

$$80 + 3.25 \times (80 + (44 + 44)) + 3 = 629$$

It is important to mention that for a scenario with differently dimensioned cluster sizes, the grand average path length may also vary, which requires a different optimization overhead.

1.7 System Performance

Depending on several factors, which are presented in Section 1.6, the cluster sizes in the CRN coverage are dimensioned to ensure limited signal degradation and maximum throughput for the available channels during inter- and intracluster communication among devices. As such, different-sized clusters in various numbers can be available throughout different CRNs. However, because all inter-CRN communication (regardless of where in the CRN the source is located) must go through the BS, only the network hosts located within the clusters in the immediate vicinity of the BS can transmit data directly to it. Subsequently, in this case we present the system model for a basic scenario where only two clusters are located near the BS, and each cluster accommodates only one transmitting host at a time within its area. With this particular model, only one hop is necessary to achieve communication among the network hosts and the BS. This enables us to study the direct impact PU activity has on SU service completion, which further allows an optimization problem to be formulated and addressed.

1.7.1 Queuing Modeling

As indicated in Figure 1.6, two clusters denoted as *CL*-1 and *CL*-2 are assumed to share a single radio channel that is licensed to PUs and is denoted as *c*. The activity of PUs in this channel is assumed to be spatially invariant within the geographical area of the two clusters. In other words, at a particular time moment, PU's activity (i.e., being either present or absent in channel *c*) is observed to be the same at every SU located within the two clusters. To avoid radio interference between the two clusters during SU communication, we assume that channel *c* is divided into two identical subchannels denoted as c_1 and c_2. In our case, for simplicity, this is done by frequency modulation techniques such as frequency division multiple access (FDMA) [15]. Subsequently, c_1 and c_2 are allocated to *CL*-1 and *CL*-2, respectively, for use by SUs.

Typically, channel *c* is said to become available for SUs when PUs are absent. Thus, in such a case, SUs in *CL*-1 and *CL*-2 can opportunistically access c_1 and c_2 in each respective cluster. Moreover, it is important to mention that channel *c* can only be used by one PU at a time. This means that if channel *c* is already used by one PU, a newly arrived PU is blocked from transmitting data. Accordingly, if channel *c* is available for SUs, each of the two subchannels c_1 and c_2 can only be used by one SU at a time. Similarly, if the available subchannel in each respective cluster is used by an SU, data transmission for newly arrived SUs within each particular cluster is blocked by the SN. Naturally, when PUs return to channel *c*, SUs using either c_1 or c_2 are dropped by the SN. Because our current system model only assumes one subchannel per cluster (i.e., no adaptation is possible for SUs to retain service), the data transmissions of dropped SUs are forced to be terminated.

We assume that PU arrivals to channel *c* follow the Poisson stream with mean rate λ_p. The time periods of PUs transmitting data are assumed to be exponentially distributed with mean value $1/\mu_p$. For each cluster *CL*-1 and *CL*-2, the arrivals of SUs at subchannels c_1 and c_2 are assumed to independently follow Poisson streams with mean rates γ_1 and γ_2. Accordingly, the time periods of SUs transmitting data via the two respective subchannels are assumed to independently follow exponential distributions with mean rates δ_1 and δ_2.

With regard to the above assumptions, a loss-system-based continuous time Markov chain (CTMC) queuing model is constructed. We let a set of three integers (i_1,i_2,j) denote a system state such that i_1 SUs are using subchannel c_1, i_2 SUs are using subchannel c_2, and j PUs are using channel *c*. Therefore, the system state space is obtained as $S = \{(i_1,i_2,j)\}$, where the three integers are constrained by $i_1 \in \{0,1\}$, $i_2 \in \{0,1\}$, $j \in \{0,1\}$, $(i_1 + j) \in \{0,1\}$, and $(i_2 + j) \in \{0,1\}$. The system state diagram is shown in Figure 1.7, whereas the system state transitions are described as follows:

- Both $(0,0,0) \leftrightarrow (1,0,0)$ and $(0,1,0) \leftrightarrow (1,1,0)$ mean that an SU in *CL*-1 accesses and releases c_1 with rates γ_1 and δ_1, respectively.
- Both $(0,0,0) \leftrightarrow (0,1,0)$ and $(1,0,0) \leftrightarrow (1,1,0)$ means that an SU in *CL*-2 accesses and releases c_2 with rates γ_2 and δ_2, respectively.

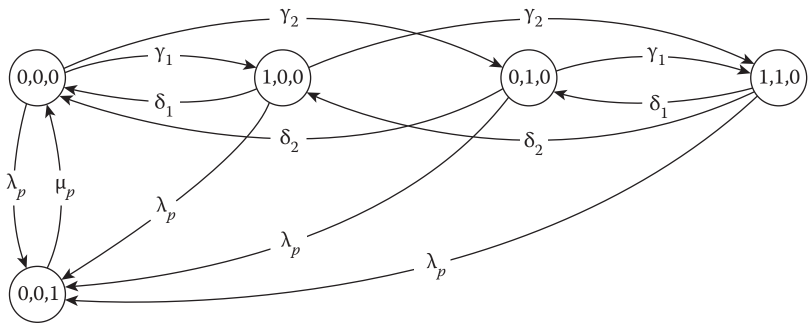


Figure 1.7 State diagram of the modeled system.

- (0,0,0) ↔ (0,0,1) means that a PU accesses and releases channel c with mean rates λ_p and μ_p, respectively.
- Transitions from (1,0,0), (0,1,0), and (1,1,0) to (0,0,1) mean that a PU occupies channel c with rate λ_p, so that the SUs using either c_1 or c_2 are forced to be terminated.

Let $\neq_{i_1,i_2,j}$ denote the steady-state probability of state (i_1,i_2,j). We have the balance equations:

$$\pi_{0,0,0} = \frac{\pi_{1,0,0}\delta_1 + \pi_{0,1,0}\delta_2 + \pi_{0,0,1}\quad{}_p}{(\gamma_1 + \gamma_2 + \lambda_p)} \tag{1.3}$$

$$\pi_{1,0,0} = \frac{\pi_{0,0,0}\gamma_1 + \pi_{1,1,0}\delta_2}{(\gamma_2 + \delta_1 + \lambda_p)} \tag{1.4}$$

$$\pi_{0,1,0} = \frac{\pi_{0,0,0}\gamma_2 + \pi_{1,1,0}\delta_1}{(\gamma_1 + \delta_2 + \lambda_p)} \tag{1.5}$$

$$\pi_{1,1,0} = \frac{\pi_{1,0,0}\gamma_2 + \pi_{0,1,0}\gamma_1}{(\delta_1 + \delta_2 + \lambda_p)} \tag{1.6}$$

$$\pi_{0,0,1} = \frac{(\pi_{0,0,0} + \pi_{1,0,0} + \pi_{0,1,0} + \pi_{1,1,0})\lambda_p}{{}_p} \tag{1.7}$$

where all steady-state probabilities are subject to the constraint

$$\pi_{0,0,0} + \pi_{1,0,0} + \pi_{0,1,0} + \pi_{1,1,0} + \pi_{0,0,1} = 1 \tag{1.8}$$

With Equations 1.3 to 1.8, a set of linear equations can be constructed, and the steady-state probabilities of all states can be computed.

1.7.2 Performance Metrics for SUs

With the foregoing queuing model described, we study four performance metrics: blocking probabilities of SUs in clusters *CL*-1 and *CL*-2, dropping probability of SUs, and service-completion throughput of SUs.

Given the system state at (i_1,i_2,j), we know that the SUs in both clusters *CL*-1 and *CL*-2 are blocked when channel c is occupied by PUs; that is, $j = 1$. If $j = 0$, blocking of newly arrived SUs in *CL*-1 and *CL*-2 occurs for $i_1 = 1$ and $i_2 = 1$, respectively, that is, when subchannels c_1 and c_2 are already in use by other SUs. Let $P_{bl,1}$ and $P_{bl,2}$ denote the probabilities of blocking SUs in *CL*-1 and *CL*-2, respectively:

$$P_{bl,1} = \pi_{1,0,0} + \pi_{1,1,0} + \pi_{0,0,1} \tag{1.9}$$

$$P_{bl,2} = \pi_{0,1,0} + \pi_{1,1,0} + \pi_{0,0,1} \tag{1.10}$$

For a system state (i_1,i_2,j), the dropping of SUs occurs when we have three mutually considered conditions: (1) $j = 0$, (2) a PU occupies channel c, and (3) $(i_1 + i_2) > 0$. In other words, the case of dropping SUs refers to three particular system states: (1,0,0), (0,1,0), and (1,1,0). In each of the three states, the rate of dropping SUs is equal to λ_p, as shown in Figure 1.7. Thus, let P_{dr} denote the dropping probability of the SUs. To compute P_{dr}, we divide the total rate of dropping SUs by the actual arrival rate of SUs into the system. For the system at the three states (1,0,0), (0,1,0), and (1,1,0), the numbers of dropped SUs due to channel occupancy by PUs are equal to 1, 1, and 2, respectively. Hence, the total rate of dropping SUs equals $\lambda_p(\pi_{1,0,0} + \pi_{0,1,0} + 2\pi_{1,1,0})$. Given that the actual arrival rate is $(\gamma_1(1 - P_{bl,1}) + \gamma_2(1 - P_{bl,2}))$, we have

$$P_{dr} = \frac{\lambda_p(\pi_{1,0,0} + \pi_{0,1,0} + 2\pi_{1,1,0})}{\gamma_1(1-P_{bl,1}) + \gamma_2(1-P_{bl,2})} \tag{1.11}$$

We define the service-completion throughput of SUs as the average rate of SUs completing the transmission using subchannels. Thus, let R denote the service-completion throughput of SUs. According to the expectation definition $\sum_{i=1}^{\infty} x_i \Pr(x_i)$, R is computed as follows:

$$\sum_{\forall i_1,i_2,j}^{(i_1,i_2,j)\in S} (i_1\delta_1 + i_2\delta_2)\pi_{i_1,i_2,j} \tag{1.12}$$

1.7.3 Performance Evaluation

Both numerical and simulation results are reported; the simulation experiments were conducted to validate the numerical analysis.

1.7.3.1 Parameter Settings

We consider the parameters of the PU arrival rate as $\lambda_p \in \{0.03,0.06,0.09,0.12\}$ s^{-1} and the service rate as $\mu_p = 0.16$ s^{-1} [23]. Naturally, for SUs, the two clusters cover particular geographical areas, where the arrival rates of SUs are considered equal: $\gamma_s = 0.68$ s^{-1}; that is, $\gamma_1 + \gamma_2 = \gamma_s = 0.68$ s^{-1}. This furthermore entails that the two clusters may cover differently sized geographical areas, which means that the arrival rates of SUs to them may be different. Therefore, we introduce a selection probability $\alpha \in \{0.1,0.3,0.5\}$, by which γ_1 and γ_2 are constrained as $\gamma_1 = \alpha\gamma_s$ and $\gamma_2 = (1-\alpha)\gamma_s$, respectively [23]. Assuming that the SU service rates over two identical subchannels c_1 and c_2 are the same, they are considered equal to 0.82 s^{-1}; that is, $\delta_1 = \delta_2 = 0.82$ s^{-1}. For each parameter setting,

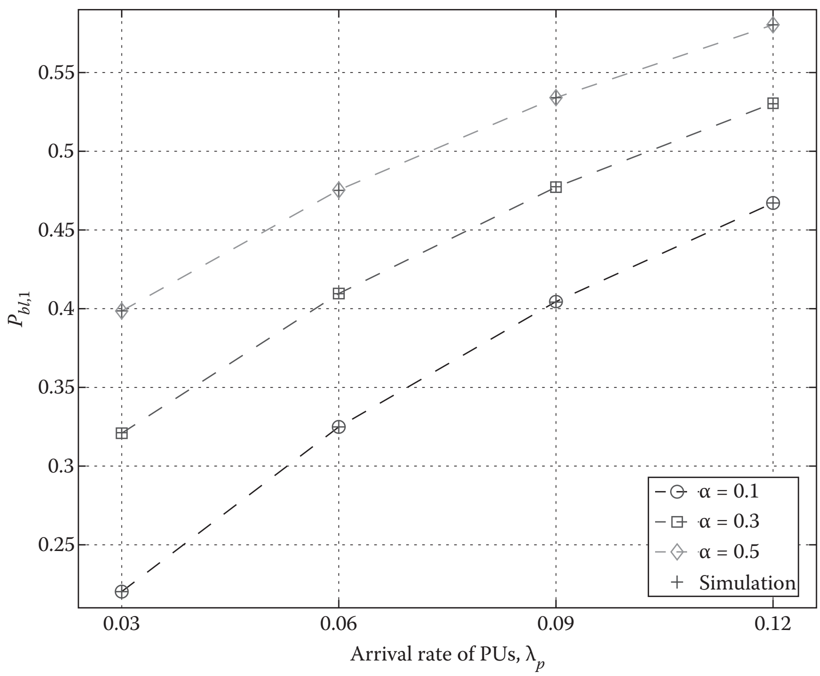


Figure 1.8 Blocking probability of SUs in *CL*-1 versus λ_p.

the simulator runs for a simulation time of 10^7 s. The numerical results of $P_{bl,1}$, $P_{bl,2}$, P_{dr}, and R are shown in Figures 1.8, 1.9, 1.10, and 1.11, respectively. The simulation results are indicated with the marker "+," and we observe that they closely match the numerical results. Further, we report in Table 1.4 the numerical results of the actual arrival rate of SUs into the system. A more detailed discussion on the obtained results follows.

1.7.3.2 Given Identical Selection Probability

Figures 1.8, 1.9, and 1.10 indicate that $P_{bl,1}$, $P_{bl,2}$, and P_{dr} increase with λ_p, respectively. This is because more PUs are requesting channel resources when λ_p is increasing. Hence, the availability of the two subchannels for SUs is reduced. As a result, the service-completion throughput of SUs, R, is decreased with λ_p, as observed in Figure 1.11.

1.7.3.3 Given Identical Arrival Rates

Figure 1.8 shows that $P_{bl,1}$ increases with α. Figure 1.9 shows on the contrary that $P_{bl,2}$ decreases with α. The reason for this is that when α is increasing, the competition among SUs becomes higher for subchannel *c*1 and lower for subchannel *c*2. Put differently, given $\alpha < 0.5$, the competition among SUs in *CL*-1 is always lower than in *CL*-2.

However, when α is close to 0.5, the arrivals of SUs are more evenly distributed between the two clusters. This means that the available resources are better used; the total blocking probability of SUs decreases with increasing α, and it is smallest when $\alpha = 0.5$. This can be observed in Table 1.4, where the actual arrival rate of SUs into the system increases with α.