Virtualization-Based Cognitive Radio Networks

Mahmoud Al-Ayyoub[a], Yaser Jararweh[a], Ahmad Doulat[a], Haythem A. Bany Salameh[b], Ahmad Al Abed Al Aziz[a], Mohammad Alsmirat[a], Abdallah A. Khreishah[c]

[a]*Department of Computer Science, Jordan University of Science and Technology, 22110 Jordan,*
e-mail: {maalshbool,yijararweh}@just.edu.jo, {ahmad.doulat, ahmad.mma87}@gmail.com, masmirat@just.edu.jo
[b]*Yarmouk University, Irbid, Jordan. e-mail: haythem@email.arizona.edu*
[c]*New Jersey Institute of Technology, NJ, USA. e-mail: abdallah@njit.edu*

Abstract

The emerging network virtualization technique is considered as a promising technology that enables the deployment of multiple virtual networks over a single physical network. These virtual networks are allowed to share the set of available resources in order to provide different services to their intended users. While several previous studies have focused on wired network virtualization, the field of wireless network virtualization is not well investigated. One of the promising wireless technologies is the Cognitive Radio (CR) technology that aims to handle the spectrum scarcity problem through efficient Dynamic Spectrum Access (DSA). In this paper, we propose to incorporate virtualization concepts into CR Networks (CRNs) to improve their performance. We start by explaining how the concept of multilayer hypervisors can be used within a CRN cell to manage its resources more efficiently by allowing the CR Base Station (BS) to delegate some of its management responsibilities to the CR users. By reducing the CRN users' reliance on the CRN BS, the amount of control messages can be decreased leading to reduced delay and improved throughput. Moreover, the proposed framework allows CRNs to better utilize its resources and support higher traffic loads which is in accordance with the recent technological advances that enable the Customer-Premises Equipments (CPEs) of potential CR users (such as smart phone users) to concurrently run multiple applications each generating its own traffic. We then show how our framework can be extended to handle multi-cell CRNs. Such an extension requires addressing the self-coexistence problem. To this end, we use a traffic load aware channel distribution algorithm. Through simulations, we show that our proposed framework can significantly enhance the CRN performance in terms of blocking probability and network throughput with different primary user level of activities.

Keywords: Wireless Network Virtualization, Cognitive Radio Networks, Software Defined Radio, Coexistence Problem, Resource Allocation

---

1. Introduction

Due to the low cost and widespread acceptance of wireless communication devices and applications, the available radio spectrum became insufficient to accommodate the needs of such a large number of wireless devices [1]. Users of wireless networks are generally viewed as either Primary Users (PUs) or Secondary Users (SUs). PUs are those licensed users, for which a part of the available spectrum is reserved for their own services. Such reserved spectrum bands are not fully utilized. Thus, to improve the spectrum utilization, SUs are allowed to opportunistically utilize the licensed bands provided that they do not affect the PUs. Cognitive Radio (CR) is a technology that enables a CR node (a SU) to sense its operational environment and change its transmission parameters according to the acquired information with the goal of increasing spectrum utilization while minimizing the introduced interference to PUs. For example, a CR node can sense its environment for the available spectrum (spectrum holes), divides it into a set of channels and selects the best channel according to a predefined policy.

The IEEE 802.22 standard for Wireless Regional Area Network (WRAN) was proposed as the first attempt to enable commercial applications based on CR technology [2, 3]. According to the IEEE 802.22 standard, a network consists of a base station (BS) and a set of Customer Premises Equipment (CPEs). The available spectrum is divided using Orthogonal Frequency Division Multiple Access (OFDMA) modulation scheme into a set of orthogonal channels and each BS is responsible for allocating available channels to the CPEs it is servicing. While such an approach avoids having interfering concurrent transmissions between CPEs of the same cell, it does not avoid other kinds of interference.

Co-existence of different wireless networks and interference management are challenging problems in CR environment. There are two different types of co-existence; incumbent co-existence (between licensed and unlicensed users) and self-coexistence (between SUs in multiple overlapping WRANs cells). To overcome the self-coexistence problem in WRANs, many fixed and dynamic channel assignment techniques have been proposed [3, 16].

The emerging virtualization techniques introduced in many computing and communication systems provide a paradigm shift in resource management of these systems. Virtualization abstracts the underlying physical system components into virtual components that can be efficiently allocated for system applications. While several previous studies have focused on computing systems and wired network virtualization, the field of wireless network virtualization is not well investigated. Moreover, the virtualization of CRNs is still in its early stages with a lot of room for exploiting virtualization benefits to support CRNs operations. CRNs virtualization can benefit both inter-cell and intra-cell operations. With virtualization, each physical node is able to run multiple virtual networks instead of one network in the case of a non virtualized frameworks [4, 5].

In this work, we propose two frameworks. The first framework, Single-Cell Cognitive Radio Network (SDSC-CRN), provides a virtualization based resource allocation approach for CRNs. In this approach, we advocate delegating some of the management responsibilities of the BS to the users and allow them to make local decisions. By doing so, we aim to reduce the users' reliance on the BS and improve network performance by reducing the unneeded control overhead. The resource management process is totally software based without the need for any network administrator interventions. The distinct features/advantages of SDSC-CRN are, (1) cooperative resource management over wireless cognitive networks, (2) a centralized BS controls and manages the resource allocation using a centralized manager called the Global Hypervisor (GH) among the different cognitive users joining the network without affecting the PUs, and (3) virtualizing of the physical radio nodes to have several instances for different virtual networks, each of which having an intermediate layer called the Local Hypervisor (LH). The LHs support the GH in distributing the resources to minimize the control overhead at the BS. These features/advantages play a significant role in improving the overall network throughput, as well as minimizing the management overhead at the BS. The second framework, Software Defined Multi-Cell Cognitive Radio Network (SDMC-CRN), integrates the concepts of virtualization into multi-cell CRNs. This framework distributes the resources among the different cells intelligently and, within each cell, uses SDSC-CRN to minimize the control overhead and blocking ratio while maximizing the network throughput.

The rest of this paper is organized as follows. In Section 2, we present some background knowledge necessary to understand the proposed frameworks. We present the Software Defined Single-Cell Cognitive Radio Network (SDSC-CRN) framework in Section 3. In Section 4, we describe the Software Defined Multi-Cell Cognitive Radio Network (SDMC-CRN) framework. We evaluate our proposed SDSC-CRN and SDMC-CRN frameworks through simulations in Section 5. Related works are presented in Section 6. Finally, Section 7 provides conclusion remarks.

2. Work Preliminaries

In this section, we present some background knowledge necessary to understand the proposed frameworks. Since the basic idea of proposed framework is to integrate virtualization into CRNs, we dedicate the following subsections to discuss each one of these two fields.

2.1. Virtualization

The term virtualization has been extensively discussed in literature. It refers to the process of creating a number of logical resources using the set of the available physical resources, in a way that allows the user to use them as if he/she was using the physical resources directly. This way, the physical resources are better utilized since virtualization allows more users to share them. Moreover, this provides an additional layer of security since the user's application has no direct control over the physical resources.

The original idea aimed to virtualize the computer system's physical resources such as the memory, the processors, the network interfaces, and the storage unit into separate sets of logical instances. Each set of these virtual instances is assigned to different users who see them as separate entities by themselves. The goal is to maximize the physical resources' utilization while keeping the required performance [6]. Computer system virtualization has three properties: (1) isolation between users, (2) service customization and (3) increasing resource efficiency.

The term virtualization was later introduced into the field of wired networks by introducing the framework of virtual private networks (VPNs) at the service provider networks level using different layers such as optical wavelength, multi-protocol label switching (MPLS). In these techniques, the virtualization was deployed by partitioning the existing physical network into a set of logical network instances. This process requires a careful management in order to maintain the Quality of Service (QoS) and the required level of security by each user in the network. The management tasks are the responsibility of the service provider.

Recent research studies are focusing on applying the virtualization concept in wireless networks, where things become more complicated and new challenges appear. Examples of such challenges include: (1) the lack of infrastructure in many types of wireless networks, (2) spectrum sharing, (3) the different geographic coverage regions of the wireless networks, and (4) the mobility of users and the dynamic topologies in such networks [7].

2.2. Cognitive Radio Networks (CRNs)

CR has been presented as a key technical improvement to the legacy wireless communications technologiesto improve spectrum utilization. To be more specific, CR can setup their own CRNs by opportunistically utilizing the available spectrum (spectrum holes) in the licensed bands. These spectrum holes represent fragments of some PU's licensed bands that are not fully utilized for some time [8]. In order for these CR nodes to utilize these spectrum holes without interfering with the PUs, they should have some sensing capabilities through which channel availability is determined. Moreover, they should be flexible enough to quickly vacate a channel as soon as its owner (PU) starts using it. In such cases, CR nodes must quickly and seamlessly move their ongoing communication sessions into another available channel. The capabilities of the CR can be listed as follows [9]:

- *Spectrum sensing*: a CR node should be able to sense the surrounding environment to detect the PUs activities and determine which channels are available and which channels can cause minimum interference with the PUs.
- *Spectrum sharing*: a CR node should have the ability to share the available spectrum as well as the gathered information from the surrounding environment with other CR nodes or a BS. The BS is responsible for coordinating CR nodes' access to available channels without interfering with PUs. This is known as cooperative sensing.
- *Self-organization*: CR should be able to create ad-hoc networks and organize the communication sessions carried on them.
- *Re-configurability*: the ability of a CR node to configure its operating parameters (including the power and frequency) according to the gathered information.

2.2.1. IEEE 802.22 Wireless Regional Area Network (WRAN)

IEEE 802.22 standard for Wireless Regional Area Network (WRAN) is the first effort to enable commercial applications based on CR technology [2, 3]. According to the IEEE 802.22 standard, a network consists of a BS and a set of Customer Premises Equipment (CPEs). The available spectrum is divided using Orthogonal Frequency Division Multiple Access (OFDMA) modulation scheme into a set of orthogonal channels. To provide better knowledge of the availability of channels, the users sense the spectrum availability in their vicinity and periodically send their sensing reports to the BS. The IEEE 802.22 standard makes use of the white spaces in the VHF/UHF TV broadcast service, ranging from 54 MHz to 862 MHz. These white spaces are portions of the underutilized spectrum allocated to service the television broadcast. They are provided to the CPEs who do not have licensed bands provided that they do not interfere with the PUs who have licenses for these bands. The IEEE 802.22 standard was aimed to present broadband access to hard-to-reach areas with low population density, which makes it suitable for rural environments.

2.2.2. The Coexistence Problem

The coexistence problem is one of the most challenging problems facing multi-cell CRNs. Multiple overlapping cells may suffer from harmful interference resulting from two kinds of coexistence [10]: (1) incumbent coexistence, in which SUs affect the transmissions of PUs and (2) self-coexistence, which occurs between SUs located within different overlapping cells. In both types the resulting interference causes a significant degradation in the system performance. In this paper, we are interested in the self-coexistence problem. see Figure 1.

Figure 1 represents a system with two cells A and B. Suppose that SU1 belongs to cell A , SU2 belongs to cell B and channel one (CH1) is available to both cells. If the BS of cell A assigns CH1 to SU1 and the BS of cell B assigns CH1 to SU2, then interference may occur between the two SUs which affects the system performance. To cope with this problem, BSs must cooperate to determine the set of operational channels for each cell. Different approaches have been introduced to solve this issue. One of these approaches is the

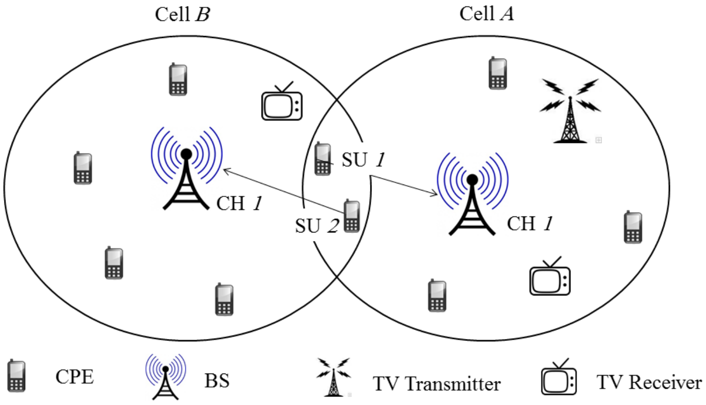


Figure 1: Self-coexistence problem.

Fixed Channel Allocation (FCA) scheme in which channels are permanently allocated to each cell. The main drawback of this approach is that it does not take into account the amount of traffic loads within the cell. A traffic load aware channel allocation algorithm is employed in this work.

3. The Single-Cell Cognitive Radio Network (SDSC-CRN) Framework

We now discuss the SDSC-CRN framework. We consider a network with one BS and n physical radio nodes (PNs) with varying sets of resources. For simplicity, we assume that these resources include a number of Radio Interfaces (RIs) at each PN and a set of orthogonal channels[1] available for specific periods of time based on the PU's activity along with constraints on the power levels that can be used for transmission in order to achieve several non-conflicting concurrent transmissions. Additional resources such as coding schemes can be easily incorporated into our framework. The existence of multiple channels and multiple RIs at each PN requires some access coordination which is achieved using multiple transreceivers per node [12].

Each one of the PNs hosts a set of virtual nodes (VNs). VNs residing in different PNs need to communicate with each other. To facilitate such communications, VNs request resources from their hosting PNs. The merits of our scheme are most evident when we are dealing with a heavily loaded network, which is characterized by two ratios. The first one is the ratio of the number of VNs residing at any PN to the number of available RIs at the same PN (VNs - RIs). This ratio can be smaller or equal to one, where each VN will be assigned one or more RIs, or it can be larger than one, where a number of VNs will have to take turns using the same RI, which can be achieved using any time sharing policy such as Time Division Multiple Access (TDMA). The second ratio is the ratio between the number of VNs residing in any PN and the number of available channels at the same PN (VNs - CHs). Again, this ratio can be smaller than or equal to one, where each VN will be assigned one or more channels. It can be larger than one, where a number of VNs will have to take turns using the same channel. This can be done using Frequency Division Multiple Access (FDMA) or TDMA policies. We consider the more interesting and challenging scenario in which both ratios are greater than one.

Different VNs residing on different PNs form virtual networks (VNets). See Figure 2. VNs of the same VNet communicate with each other using the physical resources of the PNs. The goal of our proposed framework is to coordinate the access to these resources with minimal control overhead.

Note that the discussion in this work is meant to be as general as possible without enforcing a specific meaning of VNets. Nonetheless, to give a concrete example, consider a scenario in which PN j_1 has a stream of packets to be broadcasted to a set of PNs, $j_2, \ldots, j_n$. Then, for each PN j_i, where $i = 2, \ldots, n$, j_1 creates a VNet consisting of two VNs: one residing in j_1 and one residing in j_i.

To achieve our goal of resource sharing with minimal control overhead, we propose a two-tier management scheme where the resources of the entire network is managed by a middle layer at the BS called the Global Hypervisor (GH). The GH allocates resources to the PNs where they will be managed by another middle layer at the PN called the Local Hypervisor (LH). LHs allocate the available resources to the different VNs running on them. Obviously, the resources are allocated based on their availability as well as the requests made by the VNs. A critical aspect to the success of this approach is the ability to avoid both the

---

[1] The unrealistic assumption of orthogonality between channels is made here for simplicity. It can be relaxed by introducing guard bands [11].

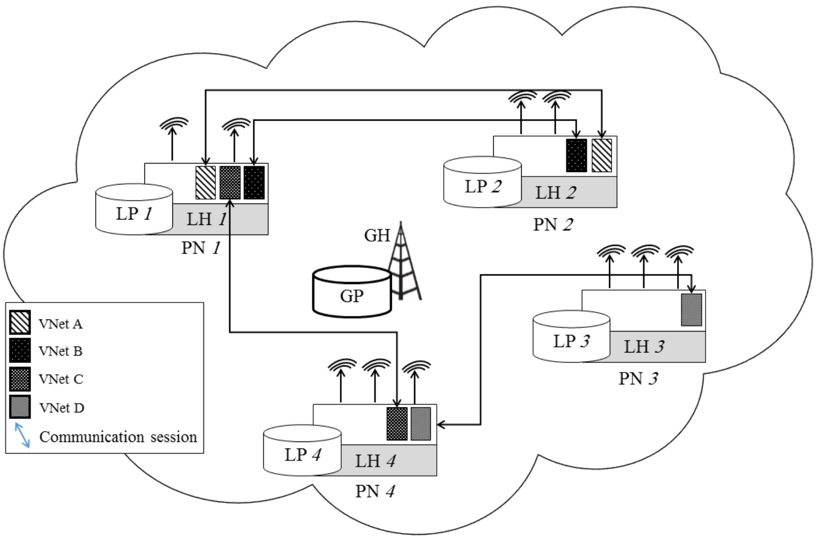


Figure 2: Virtual network architecture.

contention between SUs and any potential conflict with PUs. To address the former issue of coordinating SUs' access to the resources, the GH must assign the resources to the LHs in a way that allows each LH to make local decisions on how to utilize the resources assigned to it without interfering with other LHs. GH can use any fairness scheme to distribute the available resources among the LHs. As for the latter issue of avoiding interference with PUs' communication sessions, cooperative sensing is exploited in which each SU sends periodic sensing reports to the BS.

The following subsections discuss the steps of the proposed resource allocation framework as depicted in Figure 3. The discussions therein pertain to specific resources: the RIs and the available channels; similar techniques can be applied to other resources.

3.1. Tier 1: Global Hypervisor (GH)

As mentioned earlier, the GH resides at the BS and its purpose is to manage all of the resources within the BS's cell while the local management of the resources is left to LHs. In this subsection, we discuss the details of how this can be achieved.

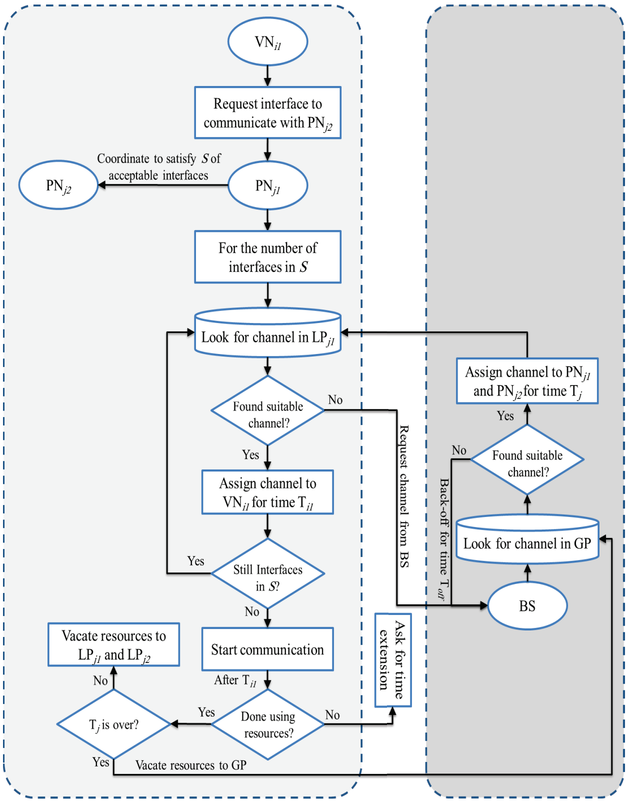
When the GH receives a request from the LH of some PN j (LH$_j$) for a certain number of channels to be used for a certain period of time (request time), it will check whether enough channels are available at the Global Pool (GP). If not, then the GH will wait for a specific period of time to determine if some channels can be vacated from other nodes in the network to satisfy the request. When the waiting period is expired, the request is dropped.

Now, if there are enough channels to satisfy the request, the GH will assign them to LH$_j$. For each channel c assigned to LH$_j$, the GH sets an upper limit on the transmission power that can be used by j. The transmission power is selected such that for each transmission from PN s (to some PN t) that belongs to the set J_c of concurrent transmissions on channel c, the following SINR equation is satisfied [13, 14]:

$$\frac{\frac{P_s}{d_{s,t}^{\gamma}}}{N + \sum_{j \in J_c} \frac{P_j}{d_{j,t}^{\gamma}}} \geq \beta, \tag{1}$$

where P_s is transmission power from s, N is the white noise, $d_{s,t}$ is the distance between s and t, γ is the path loss exponent usually assumed to be greater than 2 and β is the SINR threshold that depends on the desired data rate, the modulation scheme, etc. The GH resource allocation control is depicted in the right-hand side of Figure 3.

Note that the GH assigns the upper limits on the power levels such that even if each transmitter uses the maximum allowed power, Equation 1 will still be satisfied at each receiver of all concurrent transmissions. This means that the GH need not be aware of the details of every communication session taking place in its region as long as every node is restricting itself to the assigned power limits. Nonetheless, if the GH is informed of the current transmissions, it will be able to utilize the channels more. This can be achieved by piggybacking such information into the control messages sent from the LHs to the GH such as sensing information, and requests for more resources.



(a) Tier 1 (LH) allocation control (b) Tier 2 (GH) allocation control

Figure 3: Allocation flow control.

A channel should be vacated in different scenarios such as when the PU starts using it, when the request time is over, when the communication session is over before the end of the request time, etc. In some cases, the LHs might ask for an extension of the request time, such requests are granted provided that no starvation or interference is caused.

3.2. Tier 2: Local Hypervisor (LH)

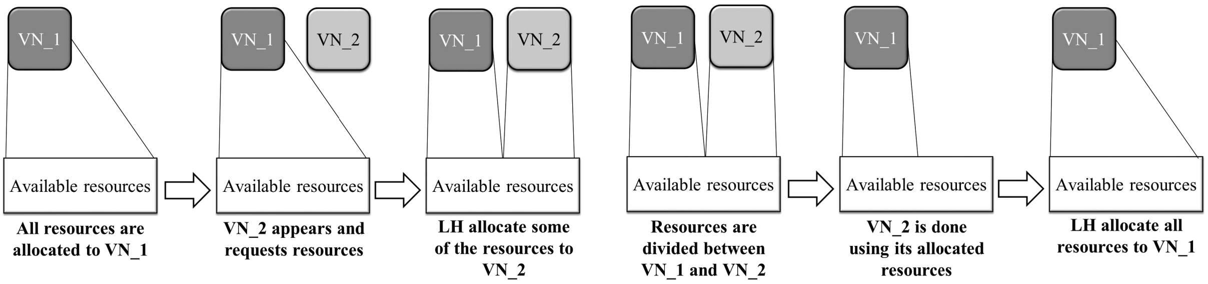
A LH at a certain PN is responsible for allocating the available resources to the VNs running on the PN. In accordance with the common terminology in the virtualization literature, we say that each LH has a local pool (LP) of resources obtained from the GH's global pool (GP) of resources. Obviously, the resources are allocated based on their availability as well as the requests made by the VNs.

When a VN i_1 (residing in PN j_1) wants to communicate with another VN i_2 (residing in PN j_2) in the same VNet,[2] both i_1 and i_2 have to request the resources necessary to complete their communication sessions from their respective LHs. For simplicity, we assume that these resources include the number of RIs from j_1, the same number of RIs from j_2, the number of channels that are available at both j_1 and j_2 and the ability to transmit at an appropriate power level at both j_1 and j_2. Additional resources such as coding schemes can be incorporated easily into our framework. These resources are requested for a specific period of time. The LHs have a specific time period during which they must satisfy the request or drop it. Note that the request can be *single-minded* (where the request must be satisfied completely or rejected) or *best-effort* (where the request is made for specific amounts of each resource, but it is acceptable to be assigned fractions of these amounts). Either way, there will always be a minimum set of resources acceptable for any request.

To satisfy the minimum requirements of resources for a given request, the LH at j_1 must coordinate with the LH at j_2. The two LHs should be willing to assign enough RIs to each VN and they should agree on a set of channels to be used for communication at acceptable power levels. These channels must be available at the local pool of each LH. If this is not the case, the channels (along with the acceptable power

---

[2] Instead of assuming that one VN sends packets and the other VN receives them, we assume here the more interesting case of two-way communication.



(a) Scale down resources.

(b) Scale up resources.

Figure 4: Scale down/up resources.

level) must be requested from the GH by the LH of the sender, j_1. If there are enough channels available at the GH, then they will be assigned to the LHs. Otherwise, the GH will wait for a specific time period to determine if some channels can be vacated to satisfy the request. When the waiting period is expired, the request will be dropped. The LH resource allocation control is depicted in the left-hand side of Figure 3.

3.3. Dynamic Resource Allocation

An important aspect of virtualization is the ability to dynamically allocate the available resources to competing entities. In our framework, this means that when a VN is trying to establish a communication session on a PN with a small total of requested resources, then this VN might get all the resources it requests. As more VNs emerge and request resources, the amount of resources available to each VN decreases. This scenario of *scaling down* the resources is depicted in Figure 4(a). On the other hand, if some communication sessions are completed and their allocated resources are vacated, the amount of resources available for each VN increases. This scenario of *scaling up* the resources is depicted in Figure 4(b).

It should be noted that the procedures of scaling up and down the resources are not as straightforward as they seem due to the potential interference with other concurrent transmissions (e.g., if a channel c is currently being used by i_1, it cannot be simply used by another VN, i_2, if i_2 requires a transmission power level that is high enough to interfere with other concurrent transmissions over c). Now, several options to handle the newly available resources at the LH are considered. The first option is to perform a scale up procedure and allow other VNs at the same PN to utilize these resources. The other option is to vacate the resources and return them to the global pool. This option depends mainly on the fairness policy. A thorough exploration of this issue is the subject of future research.

4. The Multi-Cell Cognitive Radio Network (SDMC-CRN) Framework

In this section we discuss the details of SDMC-CRN framework, which is concerned with inter-cell as well as intra-cell resource allocation. For simplicity, the only inter-cell resource considered here is the radio spectrum, which is divided into a set of orthogonal channels. As for the intra-cell resources, more details are presented below. We assume that each cell, i, consists of a BS, BS_i, and n_i PNs equipped with varying sets of resources. For simplicity again, we assume that these resources include a number of radio interfaces at each PN and a set of orthogonal channels available for specific periods of time based on the PUs activities along with constraints on the power levels that can be used for transmission in order to allow several non-conflicting concurrent transmissions within each cell. Additional inter-cell and intra-cell resources such as coding schemes can be easily incorporated into our framework.

From the discussion of the previous paragraph, the proposed framework consists of two levels of resource management, intra-cell and inter-cell. For the former level, the SDSC-CRN framework is used, which is different from typical WRAN intra-cell management of resources in the sense that it does not require the involvement of the BS in every decision. As for the latter level, we exploit the channel distribution algorithm proposed in [3, 15]. The following paragraphs give more details about the inter-cell resource management level.

We consider a CRN consisting of multiple overlapping and virtualized cells. One of the major problems facing multi-cell networks is the *self-coexistence* problem between multiple SUs of different cells causing a significant degradation in the network performance. To cope with this problem, the BSs of overlapping cells must use some coordination scheme to determine the set of operational channels for each cell. Different approaches have been introduced to address this issue; one of these approaches is the Fixed Channel Allocation (FCA) scheme in which channels are permanently allocated to each cell. Applying such approaches may cause significant increase in the request blocking rate in heavily loaded cells while there are extra channels assigned to lightly loaded cells. This problem causes significant blocking to SUs transmissions which minimizes the network throughput and channels utilizations. Our goal here is to distribute the set of available channels between cells such that no cell will gain channels more than its demand based on the prevailing traffic load. Then, each cell uses its share to serve its clients. We use the algorithm illustrated in [3] to distribute channels among different cells as discussed in the following section.

In [3], the authors presented a max-min weighted fair algorithm for channel distribution called TAECA with the goal of minimizing the interference in WRAN environments. They compared their algorithm against the FCA scheme, in which the available sets of channels are permanently distributed between cells regardless of the prevailing traffic load. This means that cells with low traffic loads get the same share of channels as cells with high traffic loads causing a waste in resources and an increase in the number of blocked requests. Their results showed that their algorithm outperforms FCA under unbalanced traffic load scenarios. We use this algorithm to distribute the channels among different virtualized cells in our framework with minimum control overhead while at the same time minimizing SU's transmissions blocking rate which maximizes the resource utilization.

We now discuss the details of the distribution algorithm. Each BS runs a copy of TAECA to maintain cooperative and distributed management. The operational time is divided into a set of time intervals called T_{win}, which contains two intervals, *beaconing* interval (BI) and *data transmission* interval. BI is the time in which different BSs can communicate and share control information between each other. Each beacon contains information regarding the set of all available channels at a given time t, $C(t)$, the expected number of SU transmissions during the next time interval, $\widehat{N}_i(t)$, and the sequences of beacon transmissions. During BI, each BS sends its beacon to other BSs on all of the available channels. Using the received information, each BS maintains a K-entry table called the traffic load table. The i^{th} entry in this table corresponds to the expected number of SU transmissions of cell i during the next observation window $[t, t+T_{\text{win}}]$. note that these beacon transmissions are very light-weight in comparison with the actual transmissions since it carries only information about the set of all available channels at a given time t, $C(t)$, the expected number of SU transmissions during the next time interval, $\widehat{N}_i(t)$, so the cost for such beacon transmissions can be ignored.

Initially, the set of available channels are evenly distributed between cells. Then at any time t, each BS, i, counts the number of SU transmissions, $N_i(t)$, during the current time interval. Using this information and the information from the traffic load table, each BS can calculate the expected traffic load for the next time interval using the following formula [3]:

$$\widehat{N}_i(t) = \alpha N_i(t) + (1-\alpha)\widehat{N}_i(t - T_{\text{win}}),$$

where α is the *forgetting factor*. Then, based on the calculated $\widehat{N}_i(t)$, each BS gets a share of channels which will be used during the next time interval. The algorithm continues as follows. [3].

1. Calculate the weight w_i for each cell i by finding the minimum $\widehat{N}(t)$ and then normalize the other $\widehat{N}(t)$s according to it. $w_i = \frac{\widehat{N}}{N_{\min}}$, where $\{i = 1, 2, \ldots, K\}$. Let $S = \sum_i w_i$ be the sum of all cell's weights.
2. Divide the set of available channels into S shares such that the size of each share is $v = \frac{C(t)}{S}$ channels.
3. Distribute channels over cells according to each cell's share $S_i = v \cdot w_i = \lfloor S_i \rfloor + \{S_i\}$, where $\lfloor Si \rfloor$ and $\{S_i\}$ are the integer part and the fractional part of S_i, respectively.
4. Assign the remaining unassigned channels $C_{\text{rem}} \in \{1, 2, \ldots, K-1\}$ to the cell with the maximum fractional value where $C_{\text{rem}} = |C(t)| - \sum_{i=1}^{k} \lfloor S_i \rfloor$.

The previous steps are repeated on each BS every T_{win}.

As an illustrative example, consider a network with $K = 5$ cells and a set of available channels $C(t) = 100$ channels. Suppose that the expected traffic loads for each cell are {25, 54, 10, 77, 40}, then the normalized traffic loads $w_i = \{2.5, 5.4, 1, 7.7, 4\}$, the total number of shares is $S = 20.6$ and the share value v is $\frac{100}{20.6} = 4.854$. Thus, the number of channels assigned to the first cell is $S_1 = 4.854 \times 2.5 = 12.135 = 12 + 0.135 = 12$. Similarly, the numbers of channels assigned to each cell are $S_2 = 26$, $S_3 = 4$, $S_4 = 37$ and $S_5 = 19$.

After assigning these values to each cell, the remaining channels, $C_{\text{rem}} = 100 - 98 = 2$ channels, are added to cell 3 since it has the maximum fractional value. The algorithm guarantees that no cell will get more channels than its demand according to the prevailing traffic load.

5. Performance Evaluation

We evaluate our SDSC-CRN and SDMC-CRN frameworks using our own discrete event simulator written in C++. The results shown below are based on ten independent runs.

5.1. SDSC-CRN Performance Evaluation

This experiment's setup include a single-cell CRN with a varying number of PNs scattered within a region of 500 × 500 meters, a varying number of channels, and different problem scale ratios ((VNs - RIs) and (VNs - CHs)).

Four main simulation scenarios are used. The first scenario is concerned with the effect of varying the number of VNs residing at each PN while fixing the number of PNs, channels and RIs per PN. The considered values are (1, 3, 5 and 7). Note that the case in which each PN has a single VN is equivalent to the non-virtualized framework. The simulation parameters for this scenario are shown in Table 1. This scenario compares the performance between the virtualized framework and non-virtualized framework, and how increasing the number of VNs per PN affects the performance of SDSC-CRN.

Table 1: Simulation parameters for scenario 1 (SDSC-CRN scheme)

Parameter	Value
Number of PNs	100
Number of VNs/PN	1, 3, 5 and 7
Number of RIs/PN	2 and 4
Number of channels	120
Message length	300 packets
Packet size	1 KB
Simulation time	10,000 time slot/single run

The second scenario is carried out by changing the number of RIs available at each PN while fixing the number of VNs residing at each PN. Three different values are considered (1, 2 and 4). The simulation parameters for this scenario are shown in Table 2. This scenario shows the significant improvements achieved when increasing the availability of the resources for each PN.

Table 2: Simulation parameters for scenario 2 (SDSC-CRN scheme)

Parameter	Value
Number of PNs	50
Number of VNs/PN	5
Number of RIs/PN	1, 2 and 4
Number of channels	120
Message length	300 packets
Packet size	1 KB
Simulation time	10,000 time slot/single run

The third and the fourth scenarios are carried out to evaluate the effect of considering different values for the (VNs - RIs) ratio (in scenario 3), as well as different values for the (VNs - CHs) ratio (in scenario 4). These two scenarios are done by interchangeably fixing one ratio and using different values for the other one. Different values for both ratios were used (less than, equal to or greater than 1). The simulation parameters for the third and the fourth scenarios are shown in Tables 3 and 4 respectively.

The results of scenario 1 are shown in Figure 5. Figure 5(a) shows how SDSC-CRN framework improves the average network throughput. Intuitively, allowing multiple VNs to coexist and share the available resources on a single PN as well as reducing the need to communicate with the BS per request saves the time required to communicate with the BS and uses that time for transmitting packets between the source and the destination in the different PNs. This increases the network throughput when using the virtualized framework for up to 300% (when using 7 VNs per PN). On the other hand, the results show that increasing the number of RIs for each PN results in a significant improvement on the network throughput for SDSC-CRN framework since in SDSC-CRN setting, increasing the number of RIs means that the LHs try to

Table 3: Simulation parameters for scenario 3 (SDSC-CRN scheme)

Parameter	Value
Number of PNs	30
Number of VNs/PN	2,4,6 and 8
Number of RIs/PN	4
Number of channels	20,40,60 and 80
Message length	300 packets
Packet size	1 KB
(VNs - CHs)	(2 - 1)
Simulation time	10,000 time slot/single run

Table 4: Simulation parameters for scenario 4 (SDSC-CRN scheme).

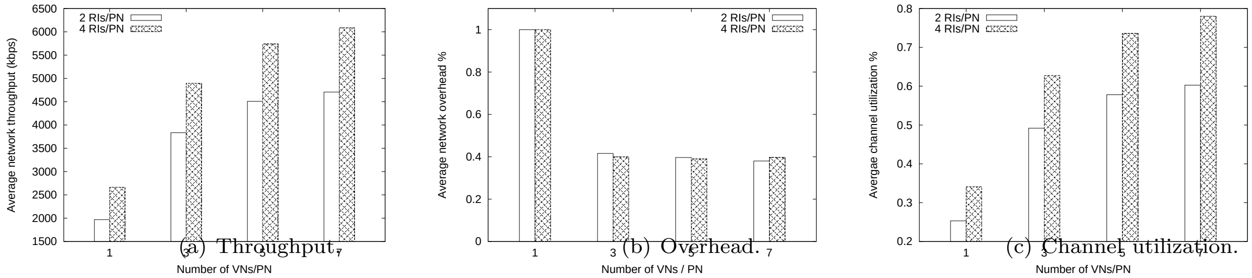
Parameter	Value
Number of PNs	20
Number of VNs/PN	4
Number of RIs/PN	4
Number of channels	40,80 and 160
Message length	300 packets
Packet size	1 KB
(VNs - RIs)	(1 - 1)
Simulation time	10,000 time slot/single run

retrieve more channels from the BS for each request (to be used for future local requests) as well as allowing multiple VNs at the same PN to share these acquired resources.

Figure 5(b) shows the improvement achieved using SDSC-CRN framework on the network overhead, which is computed by taking the ratio between the number of control messages sent by SDSC-CRN and the number of control messages sent by a non-virtualized framework. In most traditional frameworks, every request must be forwarded to the centralized BS to be replied with suitable resources or blocked if the minimum requested resources are not available. On the other hand, in SDSC-CRN, irrespective of the number of VNs residing at each PN, the results show that the network overhead is almost fixed after a specific period of time. This is due to the observation that each PN keeps using the channels available on its own LP, based on the fairness policy discussed earlier, which reduces the control packets forwarded to the BS.

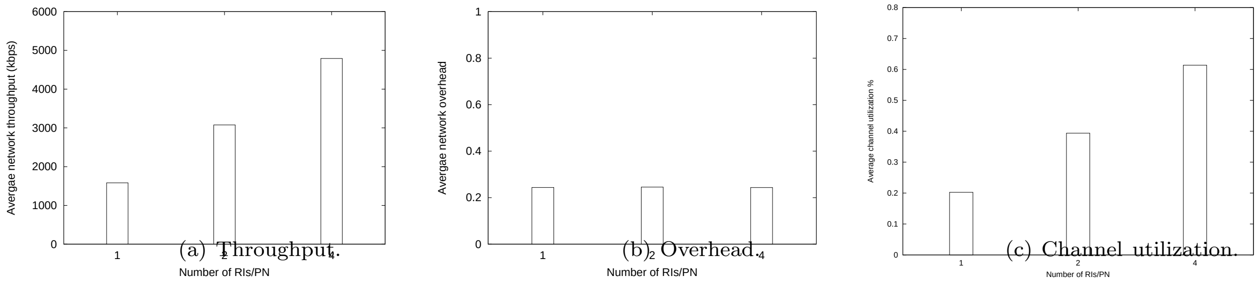
From the results shown in Figure 5(c), channel utilization is increased from about 34% when using 1 VN per PN to about 78% when using 7 VNs per PN with the same number of available RIs per PN. Intuitively, allowing more instances of VNs to instantiate their own communication sessions within a single PN plays a significant role in improving the channel utilization. This is because, in SDSC-CRN, more concurrent transmissions can be achieved simultaneously, which means maximizing the benefits gained from the available resources in hand and more traffic is allowed for a specific period of time.

We now discuss the second scenario in which we study the network performance with respect to the



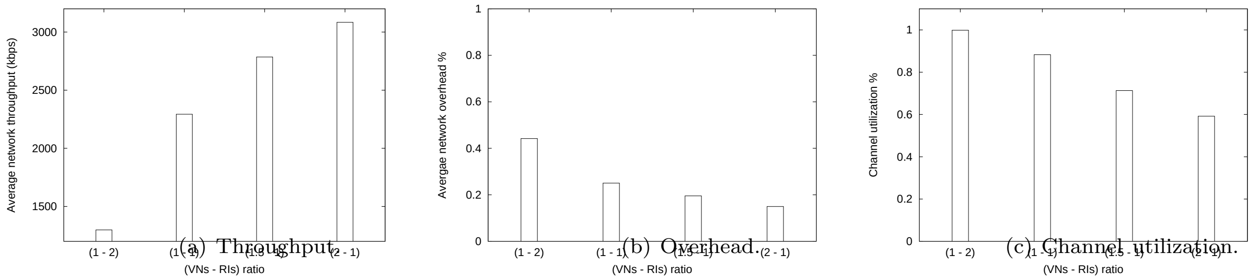
(a) Throughput. (b) Overhead. (c) Channel utilization.

Figure 5: Comparing SDSC-CRN and non-virtualized framework.



(a) Throughput.
(b) Overhead.
(c) Channel utilization.

Figure 6: SDSC-CRN's performance with different values of (RIs - PN).



(a) Throughput.
(b) Overhead.
(c) Channel utilization.

Figure 7: SDSC-CRN's performance with different values of (VNs - RIs).

number of RIs per PN. The results are shown in Figure 6. Figures 6(a) and 6(c) show that the average network throughput as well as the channel utilization are increased with the increase in the number of available RIs. In fact, by increasing the RIs from one to four, both the throughput and utilization are tripled. Figure 6(b) shows that, regardless of the number of RIs per PN, the network overhead is almost fixed after a specific period of time since each PN keeps using the channels available on its own LP. This reduces the control packets forwarded to the BS.

We now discuss the third scenario, in which the network performance is measured in terms of (VNs - RIs) ratio. In this scenario, we fix (VNs - CHs) to be (2 - 1). This allows us to determine the effect of the network performance caused by the availability of RIs per PN.

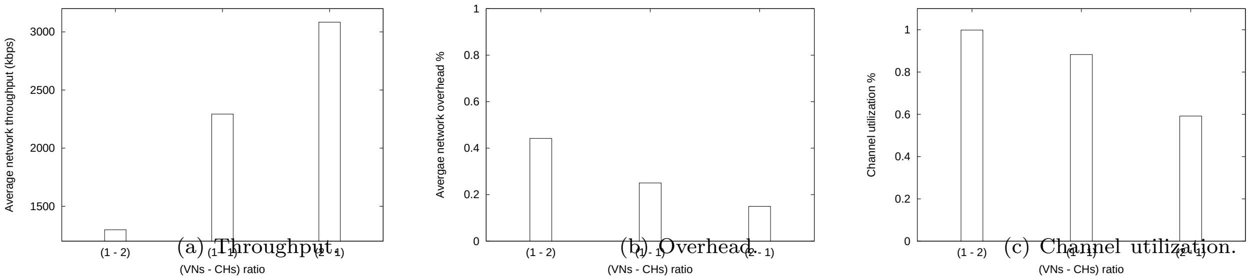
Figure 7(a) shows that the greater the (VNs - RIs) ratio the greater the improvement of the average network throughput. Obviously, having more RIs available for VNs allows more traffic which in turn results in maximizing the network throughput for about up to 230% rising from (1 - 1) up to (2 - 1) ratio.

One of the most important aspects of the proposed SDSC-CRN framework is the reduction in network overhead, which is presented in Figure 7(b). Again, the more the demand on the available resources (channels and RIs) the more the amount of the available resources on the LP for each PN which results in more options for future requests that can be satisfied using these resources with minimum number of packets needed to communicate with the GH. So, in this scenario allowing more VNs to coexist within a single PN means more demand on resources which in turn increases the availability of resources on the LP.

Figure 7(c) shows that the channel utilization is decreased using greater ratios. Since, using (1 - 2) ratio allows maximum utilization, which comes from the fact that using a greater number of RIs decreases the number of blocked requests caused by the unavailability of RIs. This is because, in SDSC-CRN, the request might be blocked either because of the unavailability of RIs or the unavailability of channels, which in turn allows more traffic to be achieved within a specific period of time.

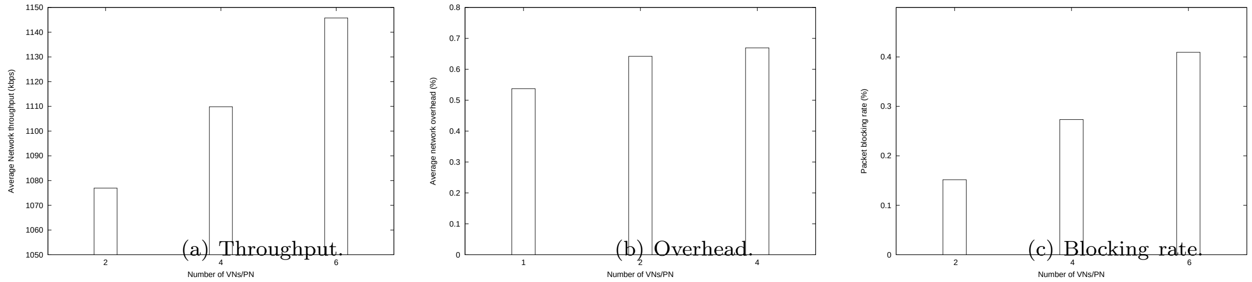
Finally, in the fourth scenario, we study the network performance with respect to the (VNs - CHs) ratio. In the following discussion, we fix the (VNs - RIs) ratio to be (1 - 1), to ignore the effect of the network performance caused by the unavailability of RIs. This allows us to determine effect caused only by the unavailability of channel for each VN.

Figure 8(a) shows that the average network throughput is increased while increasing the number of available channels for each VN. This is a result of the number of the blocked request when using small number of channels, since they are both under same traffic load. It is obvious that having more resources leads to a higher throughput; however, this means that the channel utilization will be affected under such



(a) Throughput. (b) Overhead. (c) Channel utilization.

Figure 8: SDSC-CRN's performance with different values of (VNs - CHs).



(a) Throughput. (b) Overhead. (c) Blocking rate.

Figure 9: SDMC-CRN's performance with different values of (VNs - PN).

circumstances, because we need to get the most benefit of the available resources for a specific period of time.

Figure 8(b) shows that the network overhead is decreased while increasing the number of available channels for each VN residing on each PN, from about 45% when using (1 - 2) ratio to about a 15% when using (2 - 1) ratio. Here again, having more resources means more chances for LHs to have more candidates on its own LP to select from, leading to a lower overhead.

Figure 8(c) shows that using a ratio bigger than 1 increases the channel utilization to be closer to the peak amount that the available resources can handle, by trying to keep the channels as busy as possible. The channel utilization was decreased significantly using smaller ratios to reach less than 60% when using (2 - 1) ratio.

5.2. SDMC-CRN Performance Evaluation

For the SDMC-CRN, the experiment's setup include multiple overlapping cells ($K = 5$). Each cell contains 20 PNs and each PN has a number of RIs, $W \in \{1, 2, 4\}$. The packet size is 1 Kbit. We assume that we have a set of M available channels and that the bandwidth of a channel c is enough to fulfill a transmission of a SU. We set the channel distribution time $T_{\text{win}} = 2$ seconds in accordance with the IEEE 802.22 standard. We test the proposed framework under four different simulation scenarios in which we vary the values of (i) VNs per PN, (ii) RIs per PN, (iii) available channels and (iv) PU activity level.

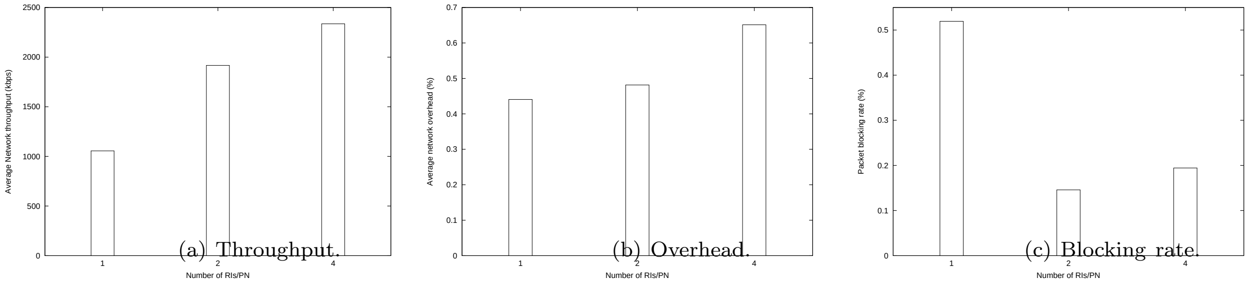
In the first scenario, we evaluate the performance of the proposed framework under different numbers of VNs per PN. Figure 9 shows the results of these experiments. We fixed the number of RIs for each PN and the number of channels. The simulation parameters for this scenario are shown in Table 5.

Figure 9(c) shows that the blocking rate increases with the increase in the number of VNs per PN. This is because when we increase the number of VNs, the competition for the available and limited number of RIs also increases. For example, if we have 4 VNs competing for 2 RIs, then 2 PNs will each get a single RI and the other 2 will be blocked. On the other hand, if we have 6 VNs competing for 2 RIs, 4 VN requests will be blocked. Also, increasing the number of VNs per PN will increase the traffic load which needs more channels to serve the increasing demands.

Figure 9(b) shows that the overhead increases while increasing the number of VNs per node due to the significant increase in the demand for channels for each node and hence the node's LH cannot fulfill the needs of its VNs. In this case, the LH has to resort to the GH to ask for channels which increases the overall overhead.

Table 5: Simulation parameters for scenario 1 (SDMC-CRN scheme)

Parameter	Value
Number of cells	5
Number of PNs	20
Number of VNs/PN	2,4 and 6
Number of RIs/PN	2
Number of channels	120
Message length	300 packets
Packet size	1 KB
Simulation time	18,000 time slot/single run



(a) Throughput. (b) Overhead. (c) Blocking rate.

Figure 10: SDMC-CRN's performance with different values of (RIs - PN).

Figure 9(a) shows the average network throughput. We can notice that in spite of the increase in the values of both the average blocking rate and the average overhead while increasing the number of VNs per node, we can get a significant increase in the average network throughput. This is because we allow multiple VNs to share the resources of a PN which makes the resources always busy. On the other hand, minimizing the communication with the BS saves the time needed to send control messages which make the resources available for data transmissions.

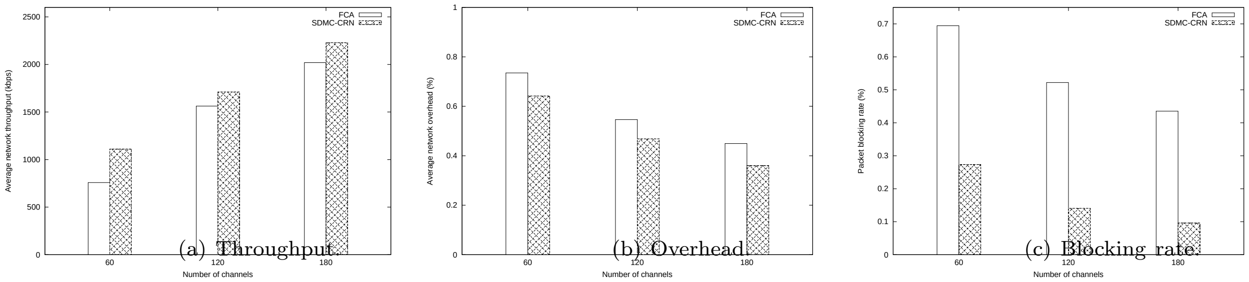
Now we discuss the second scenario, in which the network performance is measured in terms of varying the number of RIs per PN while fixing the number of VNs per PN and the number of channels. This allows us to determine the effect caused by the availability of RIs per PN. The simulation parameters for this scenario are shown in Table 6. The results are shown in Figure 10.

Table 6: Simulation parameters for scenario 2 (SDMC-CRN scheme)

Parameter	Value
Number of cells	5
Number of PNs	20
Number of VNs/PN	4
Number of RIs/PN	1, 2 and 4
Number of channels	120
Message length	300 packets
Packet size	1 KB
Simulation time	18,000 time slot/single run

From figure 10(c), we can conclude that we can gain a significant decrease in blocking rate when we use 2 and 4 RIs instead of using a single RI. That is because when we have a single RI, a single VN from the set of 4 VNs reserves the RI until it finish its transmission session. During this period, any request from the other 3 VNs will be blocked because the LH cannot scale down the resources of the VN reserving the RI. Also note that the blocking rate is increased when moving from 2 RIs to 4 RIs. Our explanation is that when we have 4 RIs, the PN needs more channels (one channel for each RI) which increases the blocking on the channel level.

The overall overhead is shown in Figure 10(b). We can see that the average overhead increases while



(a) Throughput. (b) Overhead. (c) Blocking rate.

Figure 11: SDMC-CRN's performance with different numbers of channels.

increasing the number of RIs because of the increasing number of times the LH visits the GH asking for channels for each RI. For example, when the PN has 2 RIs, the possibility of visiting the GH to ask for channels is less then when the PN has 4 RIs.

Figure 10(a) shows the average network throughput. As we can see the average throughput increases significantly when increasing the number of RIs per node. since increasing the number of RIs will increase the amount of requested channels and hence increase the number of served requests which consequently improves channel utilization.

For the third scenario, we compare the performance of our proposed scheme against the FCA under unbalanced traffic load scenarios. We assume that the SU request activity follows a Poisson distribution with rate λ_i packet/time slot, which is the same for all SUs within the cell i. We randomly change λ for each cell every 20 second and we assume that λ is uniformly distributed over [0,1]. We set the time window T_{win} to be 2 seconds. We list the parameters of this scenario in Table 7.

Table 7: Simulation parameters for scenario 3 (SDMC-CRN scheme)

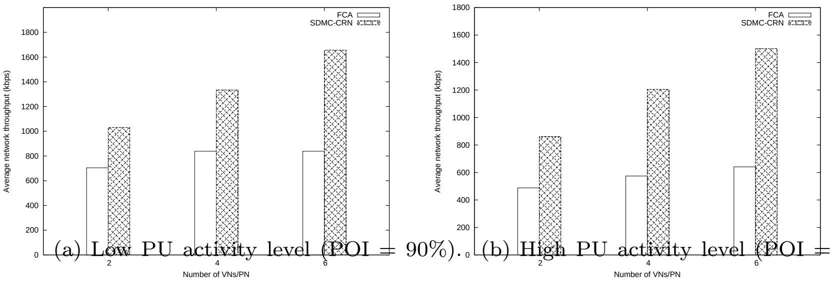
Parameter	Value
Number of cells	5
Number of PNs	20
Number of VNs/PN	4
Number of RIs/PN	4
Number of channels	60, 120 and 180
Message length	300 packets
Packet size	1 KB
Simulation time	18,000 time slot/single run

Figure 11(c) shows the blocking rate of both schemes while increasing the number of available channels. Notice that the proposed scheme outperforms the FCA scheme. It is also clear that while increasing the number of available channels, the performance of SDMC-CRN becomes more efficientthan FCA scheme. This is because SDMC-CRN allocates channels to cells according to the prevailing traffic loads so each cell gets enough channels to serve its SU requests. The proposed scheme achieves about 35% better results than FCA.

We can see from Figure 11(b) that the enhancement we gained in the blocking rates reflected on the values of the average overhead. Our interpretation is that when the set of channels are periodically redistributed between cells based on their demands, the cells with high traffic loads would have enough channels to satisfy the requests of its PNs. So each LH will get the required share of resources. Therefore, the possibility of a LH's access to the GH will be reduced. Our proposed scheme is about 13% better in terms of the average overhead compared with FCA.

Figure 11(a) shows the average network throughput for both schemes. We can easily notice that our SDMC-CRN scheme achieves better values than the FCA (on average, 15% better) because each cell gets a share of channels proportional to its demands which will increase channels utilization and hence increases the network throughput.

For the fourth scenario, since PU activity is an important parameter in CRN, we evaluate the performance of our proposed framework across different PU activity levels in the fourth scenario. We represent



(a) Low PU activity level (POI = 90%). (b) High PU activity level (POI = 50%).

Figure 12: Average network throughput with different PU activity levels.

the PU activity as a Probability Of Idleness (POI) for each channel (i.e. for each time slot during simulation the probability of each channel to be idle depends on the level of PUs' activities). Based on this, we conduct two experiments to compare the performance of SDMC-CRN scheme with the performance of FCA scheme in terms of network throughput. These experiments are done using different number of VNs per PN, fixed number of RIs per PN and fixed number of channels. The first experiment is concerned with low PU activity level (POI = 90%), and the second experiment is concerned with high PU activity level (POI = 50%). The simulation parameters for these experiments are shown in Table 8.

Table 8: Simulation parameters for scenario 4 (SDMC-CRN scheme)

Parameter	Value
Number of cells	5
Number of PNs	20
Number of VNs/PN	2,4 and 6
Number of RIs/PN	4
Number of channels	120
Message length	300 packets
Packet size	1 KB
POI	90% and 50%
Simulation time	18,000 time slot/single run

Figure 12(a) shows the average network throughput for both schemes in terms of low PU activity level (POI = 90%). We can easily notice that SDMC-CRN scheme achieves better values than the FCA scheme, because each cell gets a share of channels proportional to its demands which will increase the available channels' utilization and hence increases the network throughput.

Figure 12(b) shows the average network throughput for both schemes in terms of high PU activity level (POI = 50%). Again, SDMC-CRN scheme shows better results than FCA scheme. Regardless of the activity level of PUs, the proposed scheme is capable of adjusting the available resources (mainly the available channels) based on each cell's demands. Hence we can say that the proposed scheme can adapt to the number of available channels, and opportunistically utilize these channels effectively while they are idle. On the other hand, the FCA scheme distributes the available channels evenly among different cells, which makes it more likely to be affected as PU activity level increased.

Finally, in the last simulation scenario, we evaluate our proposed SDMC-CRN scheme against Traffic-aware Self-Coexistence Management in IEEE 802.22 WRAN Systems (TASCM) of [16] as well as the FCA scheme. In [16], the authors proposed an approach to overcome the self-coexistence problem in WRAN systems. This approach provides interference-free environment with minimum cooperation overhead at pre-specified blocking probability requirements. The main idea of their proposed scheme is that the idle channels are redistributed among neighboring overlapped WRAN cells, by taking into account the prevailing traffic loads at these cells. Initially, all idle channels are evenly allocated to the overlapped cells, then, for the j^{th} time period of length T$_s$, each WRAN cell computes the blocking rate at time period j-1 according to Equation 2.

$$P_B^{(i)}(j-1) = \frac{NB_i^{(j-1)}}{NB_i^{(j-1)} + NS_i^{(j-1)}}, i = 1,2,3 \tag{2}$$

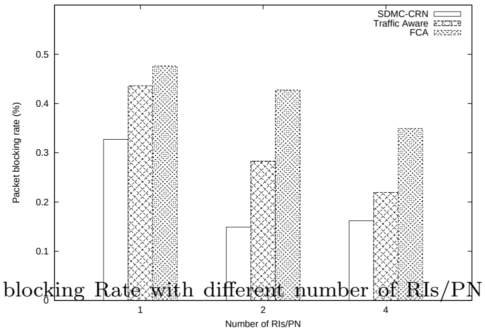


Figure 13: Packet blocking Rate with different number of RIs/PN in non-virtualized environment.

where $NB_i^{(j-1)}$ and $NS_i^{(j-1)}$ represent the number of blocked and served requests in the $(j-1)^{\text{th}}$ time period, respectively. Then each cell determines if there is a need to request channel redistribution based on the computed blocking rate and a predefined threshold blocking probability. The share value for each cell is calculated according to Equation 3.

$$C_i^{(j)} = \left\lfloor \frac{C_i^{(j-1)} + NB_i^{(j)}}{N + \sum_{i=1}^{3} NB_i^{(j)}} N \right\rfloor, i = 1, 2, 3 \tag{3}$$

where $C_i^{(j-1)}$ and N represent the number of currently allocated channels to the current cell and the total number of channels in the system, respectively. After calculating the share value for each cell, the remaining idle channels are allocated to the most heavily loaded cell (i.e. the cell with the maximum share value).

For the last scenario, we compare the performance of our SDMC-CRN scheme with both the TASCM scheme and the FCA scheme, We assume that a system with three overlapped WRAN cells, each cell contains 50 PNs, each of which has different number of RIs, and a total number of 60 channels, according to [16], we set the system blocking rate requirement to 1%. The simulation parameters of this scenario are depicted in Table 9.

Table 9: Simulation parameters for scenario 5 (SDMC-CRN scheme)

Parameter	Value
Number of cells	3
Number of PNs	50
Number of VNs/PN	1
Number of RIs/PN	1, 2 and 4
Number of channels	60
Message length	300 packets
Packet size	1 KB
Simulation time	18,000 time slot/single run

Figure 13 shows the blocking rate of the three schemes with different number of RIs/PN within a non-virtualized environment. Notice that the SDMC-CRN scheme outperforms both the FCA scheme and the TASCM scheme. The main reason that SDMC-CRN shows better performance compared with the TASCM scheme comes from the fact that in the TASCM scheme, the channel redistribution process is based on the calculated blocking rate in the previous time period. Note that such a policy neglects to take into account that some requests get blocked because of the unavailability of RIs, not because of the unavailability of channels in each cell. This forces the cell to request more channels even though it does not really need these channels; it really needs more RIs. On the other hand, the redistribution process in SDMC-CRN scheme depends mainly on the number of requests made in the previous time period, regardless of the number of blocked requests. This provides the cell with better awareness of the actual traffic demands for the next time period.

6. Related Work

The contributions of our proposed frameworks lie mainly in integrating the concept of virtualization into CRNs for both single-/multi-cell networks, which requires addressing the self-coexistence problem. The following subsection discusses network virtualization in general and its application to CRNs. The other subsection discusses the current approaches to handle the coexistence problem in CRNs.

6.1. Network Virtualization

To the best of our knowledge, the concept of virtualization in CRNs has not been investigated well in the literature. The keyword "virtual" was used in several contexts. For example, the authors of [17] used it in the context of how to use CR technology to re-use allocated (but unused) spectrum. The authors of [18] used the virtual concept to define a virtual wireless networks (VWN), which is a network with no physical resources of its own. A VWN provides services to the clients using the resources of other networks in a CR-based fashion, while still ensuring that the QoS reliability requirements, such as blocking and dropping guarantees, are achieved. Their system consisted of a set of PUs networks over which a set of virtual networks could be created. A PU request can only be served in its own home primary network while a cognitive user request can be served in any one (or more) of these networks provided that there are enough available channels there. Otherwise, the request will be blocked.

Other works have used the word "virtual" in a more general context. In [12], the authors proposed a platform where different virtual radio networks running on top of the physical nodes can share the available spectrum. The access to the spectrum was managed according to a multiple access scheme like CDMA, TDMA or FDMA.

The authors of [19] presented a platform called AMPHIPIA, which exploits the integration between CRN and network virtualization. It provides a cooperative resources management over wired and heterogonous wireless networks. The wired part is managed through the use of slice-based virtualization and the standard CR control mechanism is used for the wireless part. The authors also aimed to provide a virtualized BS with CR capabilities. AMPHIPIA consists of three main players. The first one is the service provider which provides a certain service to end users. The second part is the radio terminals. The service provider asks AMPHIPIA through the radio terminals to provide a virtual network with the required QoS. The third part is the infrastructure provider which provides AMPHIPIA with the required virtual network including BSs and radio terminals with cognitive radio capabilities.

In a follow-up work [20], the authors implemented a system prototype for AMPHIBIA including the implementation of a cognitive virtualization manager, which controls the process of network reconfiguration. The authors demonstrated that a virtual machine-based virtual network including a virtual Cognitive Base Station (vCBS) that can be dynamically established, expanded, and removed and the streaming services can be flexibly deployed on demand on the virtual network. They also showed that AMPHIBIA is capable of creating and reconfiguring vCBSs in a nearly short time interval. They showed the node setup for the vCBS, which is hosted on a host BS, and dealt with as a virtual machine. It also contains a node manager and a number of vCBSs. The node manager is responsible of managing the available resources and the access control in the host base station. On the other hand, the node manager has an interface for the network virtualization manager, and initiates and removes the vCBS based on the instructions from the network virtualization manager. Each vCBS has all the functionalities of the original cognitive BS, in which it contains a Cognitive Base Station Reconfiguration Manager (CBSRM), a Cognitive Base Station Measurement Collector (CBSMC), and Cognitive Base Station Reconfiguration Controller (CBSRC).

The authors of [21] proposed an airtime based resource control technique for wireless network virtualization in which the available resources are allocated among competing virtual networks while keeping their programmability. Their approach was applied to WLAN systems by adopting the IEEE 802.11e Enhanced Distributed Channel Access (EDCA) mechanism. The proposed solution focused on controlling airtime usage rather than controlling bandwidth usage. For each virtual network a dedicated Virtual Access Point (VAP) was created on a physical access point (AP). The VAP enables its corresponding virtual network to reserve and control the required resources on the physical wired network.

The authors of [22, 23] proposed to virtualize LTE networks. They proposed the use of a virtual base station called enhanced Nobe-B (eNB). Such nodes have a middle layer called hypervisor. This hypervisor is responsible for distributing resources among various virtual instances implemented on the higher layer. Such resource scheduling depends on different metrics including conditions for the user channel, the traffic loads, the priorities, the QoS requirements and the contract information for each virtual operator. They also proposed to divide the available spectrum into small and similar units called Physical Resource Blocks (PRBs), which will be allocated to the virtual operators. They implemented two algorithms for their scheme. The first one is a static algorithm, in which the spectrum is divided and assigned for each virtual operator which can keep using it for the whole time. The second algorithm is a dynamic version in which the resource allocation has to be done during the runtime. Therefore this allocation and the amount of resources can vary upon time, based on the load of the operator's traffic.

The authors of [24] proposed a framework called ENVIRAN to integrate the concept of network virtualization with the concept of CRNs. ENVIRAN is composed of two major elements: access service and cloud service. The former is used to efficiently utilize the available resources of the network (including spectrum, network physical hardware and the energy) and it is composed of two classes, one for terrestrial segment and the other for satellite segment. As for the cloud service, it used to increase the efciency and awareness of the entire system. The major objectives were to provide an efficient management of the system, making the system more flexible by using a centralized management of the network resources, and to reduce the sustainable operational expenditures.

Several works have proposed to extend Software Defined Network (SDN) concepts to cover wireless networks. The earliest example is [25]. Another example for Wireless Sensor Networks (WSNs) is SD-WSN, which provides an SDN based framework to handle WSNs common management problems such as manual reconfiguration of WSN [26]. In [27], the authors claim that SDN can make cellular networks much simpler and easier to manage. The first integration between SDN and CRN was presented in [28]. Neither virtualization was used nor resource allocation algorithm was proposed in [28].

6.2. The Coexistence Problem

In order to address the coexistence problem, Bany Salameh et al. [3] proposed TAECA, a max-min weighted fair algorithm for channel distribution in multi-cell WRAN systems. Since we make use of this algorithm in our proposed framework, we explain it in details in Section 4.

Several other solutions have been proposed to solve the coexistence problem. The authors of [29] proposed a resource sharing algorithm which coordinates the transmission of multiple overlapping WRAN cells. They used a graph coloring mechanism for channel distribution between cells. In [30], the authors introduced four schemes to reduce the effects of the coexistence problem. The first two schemes are based on using omnidirectional RIs at the BS whereas the BSs in the other two schemes use directional RIs. These four schemes are basically variants of the Dynamic Frequency Hopping (DFH). CASS [31] is a channel assignment algorithm to solve the self-coexistence problem. CASS supports the two spectrum sharing mode: exclusive and non-exclusive. Depending on the channel condition and after performing channel evaluation, CASS can switch between the two modes.

All the previously mentioned algorithms are distributed algorithms where all the BSs cooperate to make the channel allocation decision. In the subsequent works, a centralized approach was followed. In the centralized approach the whole network is divided into communities and one BS is elected to be a coordinator. The authors of [32] introduced DFH community (DFHC) which aims to operate between multiple WRAN cells in the DFH mode to ensure efficient frequency usage with reliable channel sensing. ESC [33] is a resource sharing algorithm that works in a multi-channel environment. It aims at assigning channels to WRAN cells in such a way that satisfies their requests and avoids any interference among them. In ESC, cells form a group and elect a coordinator among them based on priority, as explained in [32]. The coordinator collects information about each cell and compiles an assignment table. It guarantees QoS among cells but has the disadvantage that only a certain cell can use channels exclusively. In [34], the authors used the same techniques as in [32, 33]. The network area is divided into groups and each cell belonging to a group share the information about network status including the appearance of PU. The MAC layer in this approach uses a super frame structure to control data communication and permit a number of cognitive functions for licensed incumbent protection, synchronization and self-coexistence.

7. Conclusion and future work

In this paper, we proposed the SDSC-CRN framework, a virtualization based semi-decentralized resource allocation scheme for single-cell CRNs that uses the concept of multilayer hypervisors to reduce the CRN users' reliance on the CRN BS and to improve the network performance by reducing the control overhead. We then extended it into multi-cell systems taking into account the self-coexistence problem. The simulation results showed that using the virtualized frameworks resulted in significant increases in the network throughput (up to 300% when using 7 VNs per PN in a single cell) since it allows multiple VNs to coexist and share the available resources on a single PN as well as reduces the need to communicate with the BS per request, which reduces the control packets and allows for more data packets to be transmitted . Moreover, the results showed significant improvements of the virtualized frameworks over the non-virtualized ones in terms of control overhead, channel utilization and blocking rate. Finally, deploying the proposed frameworks on a testbed and evaluating their performance under real-life settings is the major next step for this work.

8. References

[1] H. Bany Salameh and M. Krunz. Channel access protocols for multihop opportunistic networks: Challenges and recent developments. *IEEE Network-Special Issue on Networking over Multi-hop Cognitive Networks*, July 2009.

[2] Carlos Cordeiro et al. IEEE 802.22: An introduction to the first wireless standard based on cognitive radios. *JCM*, 1(1):38–47, 2006.

[3] H. Bany Salameh, Y. Jararweh, T. Aldalgamouni, and A. Khreishah. Traffic-driven exclusive resource sharing algorithm for mitigating self-coexistence problem in WRAN systems. In *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1933–1937, 2014.

[4] Y. Jararweh, M. Al-Ayyoub, A. Doulat, A. Al Abed Al Aziz, H. Bany Salameh, and A. Khreishah. SD-CRN:software defined cognitive radio network framework. In *Proceedings of the 2014 IEEE International Conference on Cloud Engineering*, pages 592–597. IEEE Computer Society, 2014.

[5] Yaser Jararweh et al. Software defined cognitive radio network framework: Design and evaluation. *International Journal of Grid and High Performance Computing (IJGHPC).* To appear.

[6] Charles Kalmanek. The essential elements of successful innovation. *ACM SIGCOMM Computer Communication Review*, 42(2):105–109, 2012.

[7] Xin Wang, Prashant Krishnamurthy, and David Tipper. Wireless network virtualization. In *International Conference on Computing, Networking and Communications (ICNC), 2013*, pages 818–822. IEEE, 2013.

[8] Joseph Mitola and G Maguire. Cognitive radio: making software radios more personal. *IEEE Personal Communications*, 6(4):13–18, 1999.

[9] Xuemin Hong, Zengmao Chen, Cheng-Xiang Wang, Sergiy A Vorobyov, and John S Thompson. Cognitive radio networks. *Vehicular Technology Magazine, IEEE*, 4(4):76–84, 2009.

[10] Chittabrata Ghosh, Sumit Roy, and Dave Cavalcanti. Coexistence challenges for heterogeneous cognitive wireless networks in TV white spaces. *Wireless Communications, IEEE*, 18(4):22–31, 2011.

[11] H. Bany Salameh, M. Krunz, and D. Manzi. An efficient guard-band-aware multi-channel spectrum sharing mechanism for dynamic access networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–5. IEEE, 2011.

[12] Joachim Sachs and Stephan Baucke. Virtual radio: a framework for configurable radio networks. In *Proceedings of the 4th Annual International Conference on Wireless Internet*, page 61. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[13] Mahmoud Al-Ayyoub and Himanshu Gupta. Joint routing, channel assignment, and scheduling for throughput maximization in general interference models. *IEEE Transactions on Mobile Computing*, 9(4):553–565, 2010.

[14] A. Khreishah, C. C. Wang, and N. Shroff. Cross-layer optimization for wireless multihop networks with pairwise intersession network coding. *IEEE Journal on Selected Areas in Communications*, 27(5):606–621, 2009.

[15] Ahmad Doulat, Ahmad Al Abed Al Aziz, Mahmoud Al-Ayyoub, Yaser Jararweh, Haythem A Bany Salameh, and Abdallah A Khreishah. Software defined framework for multi-cell cognitive radio networks. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2014 IEEE 10th International Conference on*, pages 513–518. IEEE, 2014.

[16] Mohammad Bani Hani, Haythem Bany Salameh, Yaser Jararweh, and Achraf Bousselham. Traffic-aware self-coexistence management in ieee 802.22 wran systems. In *GCC Conference and Exhibition (GCC), 2013 7th IEEE*, pages 507–510. IEEE, 2013.

[17] Danijela Čabrić, Shridhar Mubaraq Mishra, Daniel Willkomm, Robert Brodersen, and Adam Wolisz. A cognitive radio approach for usage of virtual unlicensed spectrum. In *The 14th IST Mobile and Wireless Communications Summit*, 2005.

[18] Brent Ishibashi, Nizar Bouabdallah, and Raouf Boutaba. QoS performance analysis of cognitive radio-based virtual wireless networks. In *The 27th Conference on Computer Communications INFOCOM 2008. IEEE.* IEEE, 2008.

[19] Kiyohide Nakauchi, Kentaro Ishizu, Homare Murakami, Akihiro Nakao, and Hiroshi Harada. AMPHIBIA: a cognitive virtualization platform for end-to-end slicing. In *2011 IEEE International Conference on Communications (ICC)*, pages 1–5. IEEE, 2011.

[20] Kiyohide Nakauchi, Kentaro Ishizu, Homare Murakami, Yasunaga Kobari, Yuji Nishida, Akihiro Nakao, and Hiroshi Harada. Virtual cognitive base station: Enhancing software-based virtual router architecture with cognitive radio. In *2012 IEEE International Conference on Communications (ICC)*, pages 2827–2832. IEEE, 2012.

[21] Kiyohide Nakauchi, Yozo Shoji, and Nozomu Nishinaga. Airtime-based resource control in wireless LANs for wireless network virtualization. In *ICUFN*, pages 166–169. IEEE, 2012.

[22] Yasir Zaki, Liang Zhao, Carmelita Görg, and Andreas Timm-Giel. A novel LTE wireless virtualization framework. *Mobile Networks and Management*, pages 245–257, 2011.

[23] Liang Zhao, Ming Li, Yasir Zaki, Andreas Timm-Giel, and Carmelita Görg. LTE virtualization: from theoretical gain to practical solution. In *Proceedings of the 23rd International Teletraffic Congress*, pages 71–78. International Teletraffic Congress, 2011.

[24] Adrian Popescu, Haesik Kim, Franco R Davoli, Raul Dopico Lopez, Somsai Thao, Javier Del Ser, and Gerhard Wunder. ENVIRAN: Energy efficient virtual radio access networks. In *International Symposium on Wireless Personal Multimedia Communications (WPMC), 2013 16th*, pages 1–5. IEEE, 2013.

[25] S. Costanzo, L. Galluccio, G. Morabito, and S. Palazzo. Software defined wireless networks: Unbridling sdns. In *EWSDN*, pages 1–6, 2012.

[26] Tie Luo, Hwee-Pink Tan, and T.Q.S. Quek. Sensor openflow: Enabling software-defined wireless sensor networks. *IEEE Communications Letters*, 16(11):1896–1899, 2012.

[27] L.E. Li, Z.M. Mao, and J. Rexford. Toward software-defined cellular networks. In *EWSDN*, pages 7–12, 2012.

[28] A. Dutta, D. Saha, D. Grunwald, and D. Sicker. An architecture for software defined cognitive radio. In *ACM/IEEE ANCS*, pages 1–12, 2010.

[29] Shamik Sengupta, Swastik Brahma, Mainak Chatterjee, and N Sai Shankar. Self-coexistence among interference-aware IEEE 802.22 networks with enhanced air-interface. *Pervasive and Mobile Computing*, 9(4):454–471, 2013.

[30] Raed Al-Zubi, Mohammad Z Siam, and Marwan Krunz. Coexistence problem in IEEE 802.22 wireless regional area networks. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–6. IEEE, 2009.

[31] Kaigui Bian and Jung-Min Park. A coexistence-aware spectrum sharing protocol for 802.22 WRANs. In *Proceedings of 18th Internatonal Conference on Computer Communications and Networks, 2009. ICCCN 2009.*, pages 1–6. IEEE, 2009.

[32] Wendong Hu, Daniel Willkomm, Murad Abusubaih, James Gross, George Vlantis, Mario Gerla, and Adam Wolisz. Cognitive radios for dynamic spectrum access-dynamic frequency hopping communities for efficient IEEE 802.22 operation. *Communications Magazine, IEEE*, 45(5):80–87, 2007.

[33] P Camarda, C Cormio, and C Passiatore. An exclusive self-coexistence (ESC) resource sharing algorithm for cognitive 802.22 networks. In *International Symposium on Wireless Pervasive Computing (ISWPC), 2010 5th IEEE*, pages 128–133. IEEE, 2010.

[34] Minyoung Yoo and Seonhwan Hwang. A self-coexistence method for the ieee 802. 22 cognitive WRAN. In *Recent Researches in Automatic Control, Systems Science and Communications*, 2012.