A SDN Approach to Spectrum Brokerage in Infrastructure-based Cognitive Radio Networks

Anatolij Zubow, Michael Döring, Mikolaj Chwalisz and Adam Wolisz
{zubow, doering, chwalisz, wolisz}@tkn.tu-berlin.de

***Abstract*—Cognitive Radio (CR) is a promising approach to overcome the *spectrum crunch* faced by today's enterprise and residential WiFi (IEEE 802.11) due to rapid growth of wireless devices and traffic load. However, the expected high-density CR Networks (CRN) will suffer from similar problems as we see today with WiFi, i.e. any uncoordinated spectrum access will inevitably result in interference between Secondary Users and hence in a low spectral efficiency.**

In this paper we take advantages of the ideas of Software-Defined Networking (SDN) and cloud computing technology to manage interference in CRN deployments in residential areas. Specifically, we propose a flexible SDN-based CR architecture where a cloud-based centralized controller, the Spectrum Broker (SB), takes control over the spectrum assignment for the CR Base Stations (CR-BS). To enable that, the CR-BSs under control report aggregated wireless statistics to the SB. Moreover, by configuring proper rules in OpenFlow-enabled CR-BSs, the SB controller can get up-to-date information about the network traffic condition in the CRN. With this information the SB can perform a very fine-grained topology-, traffic- and channel-aware spectrum allocation.

Our architecture, as well as the proposed spectrum allocation scheme, were analyzed by means of emulation within Mininet. Results demonstrate a gain of up to $5\times$ as compared to a static spectrum allocation scheme.

***Index Terms*—Cognitive Radio, Wireless Networks, Software-Defined Network, OpenFlow**

I. Introduction

In recent years we have seen a rapid growth in the use of wireless devices such as laptops, tablets and smart phones in all environments e.g., enterprise and homes. Especially dense wireless deployments technologies, like WiFi (IEEE 802.11), suffer performance issues due to insufficient free radio spectrum resulting in high contention and interference from WiFi and non-WiFi sources.

Cognitive Radio (CR) is a promising approach to overcome such *spectrum crunch* by exploiting the fact that despite a static spectrum allocation, where governmental agencies assign wireless spectrum to license holders (Primary Users (PU)) on a long-term basis for large geographical regions, a large number of frequency bands have considerable, temporary, dormant time intervals (e.g. TV bands), leading to underutilization of a significant amount of spectrum [1]. CR aims to solve such spectrum inefficiency by allowing secondary spectrum usage by Secondary Users (SU) based on Opportunistic Spectrum Access (OSA) which allows to share the wireless channel with PUs in an opportunistic manner.

So far the focus in CR research was mainly on co-existence issues between PUs and SUs [2], [3]. However, the expected

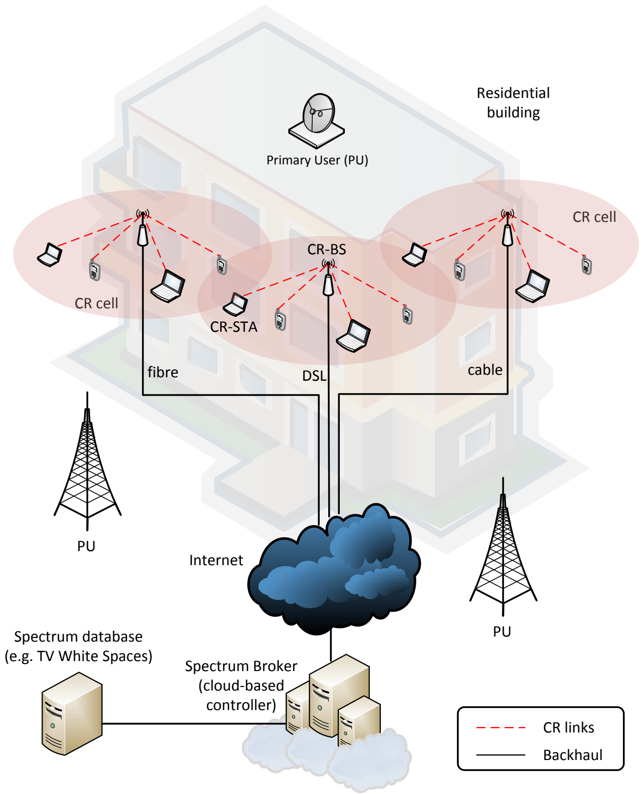


Fig. 1. Deployment scenario for the envisioned Cognitive Radio Network.

high-density CR Networks (CRN) will suffer from similar problems as we see today with WiFi. Any uncoordinated spectrum access will inevitably result in interference between SUs and hence in a low spectral efficiency. Therefore, the focus of this paper is on SU co-existence schemes.

Software-Defined Networking (SDN) and OpenFlow concepts aim to simplify network configuration processes by decoupling the control and data forwarding plane. The network intelligence is logically centralized in software-based controllers and the network devices then become simple forwarding devices [4]. Such architecture enables simple programmatic control of the network data-path and allows much more innovation in the control logic.

We present an architecture for a cloud-based centralized approach to spectrum brokerage in CRNs using SDN concepts. The centralized entity collects spectrum information as well as traffic requirements and decides on the spectrum assignments for the CRN. In summary, SDN concepts are applied to the

978-1-4799-7452-8/15/$31.00 ©2015 IEEE

SU spectrum assignment problem, where a central spectrum broker controls several, now simplified, CR devices. We have build a prototype of the system within Mininet [5], a framework that runs on a single computer and is able to simulate SDN topologies with respect to their conditions. We compared the performance of our proposed spectrum allocation scheme for CRN with state-of-the-art. Results demonstrate a gain of up to $5\times$ as compared to a static spectrum allocation scheme.

Contributions: In this paper we propose an architecture for a cloud-based centralized approach to spectrum brokerage in CRNs using SDN which allows a very fine-grained spectrum allocation in CRNs taking into account network topology information, dynamically changing traffic as well as channel conditions. Specifically, a concrete spectrum assignment algorithm is presented and compared with state-of-the-art.

The rest of the paper is organized as follows. In Sec. II the CRN system model is introduced and the spectrum allocation problem in a CRN is formulated as an optimization problem. Sec. III-A describes the envisioned cloud-based architecture for CRNs. In Sec. III-B the proposed spectrum assignment algorithm is presented whereas its performance is evaluated in Sec. IV. In Sec. V related research is discussed and finally, Sec. VI summarizes our main findings and concludes the paper.

II. Problem Statement

A. System Model

The deployment scenario for the envisioned Cognitive Radio Network (CRN) is depicted in Fig. 1. We focus on CRN deployments in residential areas where the Cognitive Radio-Base Stations (CR-BS) are located in, e.g., apartment buildings. Each CR-BS serves a set of end-user terminals, here called CR-Stations (CR-STAs). Due to the high density of nearby CR-BSs we expect that the network cells will overlap. Each CR-BS is connected to the Internet via a wired technology like DSL, cable or fiber. In the envisioned system the assignment of secondary spectrum to CR-BSs is performed by a cloud-based controller, the Spectrum Broker (SB), running in, e.g., a data center. Note, that in contrast to enterprise environments we have to deal with higher latencies (in the order of tens of milliseconds) between the controller and the CR-BSs which allows loosely coupled control only. Hence the SB is able to account only for medium- and long-term statistics when calculating the spectrum allocation. Finally, from the scalability point of view it is meaningful if only a limited number of co-located apartment buildings is controlled by a single controller.

Additionally, we assume that interference management within each cell is performed by the corresponding CR-BS and hence no coordination between neighboring CR-BSs is required as the centralized SB ensures that all CR-BSs in interference range are separated in frequency. In particular, the available radio resources within a cell are assigned to associated CR-STAs for both up- and downlink by a scheduler residing in the CR-BS.

The following CR model is used. We assume that a range of spectral frequencies from $F_{\min}$ to $F_{\max}$ can be potentially used by the CRN if the PUs are not present. It exists a CR spectrum database storing information about spectrum fragments available for secondary usage in a given time interval over a given spatial area. Moreover, the allowance for secondary usage can be revoked on short notice. Our SB cloud controller has therefore access to this CR spectrum database allowing him to exclude spectrum, that is used by PUs, from allocation. PUs are assumed to be totally unaware of secondary spectrum usage and must be protected by any means.

We assume a physical layer which allows a flexible spectrum shaping for PU protection and SU co-existence and efficient access to even very fragmented spectrum (e.g. Non Continuous-OFDM (NC-OFDM) or Filter Bank MultiCarrier (FBMC) [6]). In particular, the total spectrum, i.e. $F_{\min}$ to $F_{\max}$, is divided into NSC subcarriers, which equals the size of the FFT. Adjacent subcarriers are grouped into physical subchannels resulting in a total number of SCH subchannels.

The wireless communication between neighboring CR-BSs is limited to the in-band exchange of beacons which is used for the neighbor discovery.

B. Problem Description

The spectrum allocation problem to be solved by the Spectrum Broker (SB) is to find an optimal, interference free [1] (w.r.t. other CR-BSs) assignment of spectrum to CR-BS cells which is not claimed by PUs such that the data rate (throughput) of all active flows is maximized. In particular the SB performs the spectrum allocation based on the following available information: i) information on the network topology, ii) network traffic conditions in each cell and iii) the average link quality in each cell.

This optimization problem can be formulated as follows:

Instance: A set of $\mathcal{V}$ CR-BS nodes and for each CR-BS $v \in \mathcal{V}$ a set of associated CR-STAs $\mathcal{W}_v$. For each $v \in \mathcal{V}$ a set of subchannels $\mathcal{S}_v$ which are available for secondary usage at node v, i.e. not used by any PU in vicinity. Moreover, a graph representing the interference between CR-BSs, $I = (\mathcal{V}, \mathcal{E} = \mathcal{V} \times \mathcal{V})$. Finally, for each wireless link between a CR-BS, $v \in \mathcal{V}$ and its corresponding CR-STA, $w \in \mathcal{W}_v$ the long-term average SNR per subchannel, $\tilde{\gamma}^s_{v,w}$ of a particular subchannel, $s \in \mathrm{S}$ and the number of active flows, $f_{v,w} = [0; \infty[$ is given.

Objective: The goal is to find an optimal assignment for all subchannels to CR-BS nodes (cells):

$$A_{v,s} = \begin{cases} 1, & \text{if subchannel } s \text{ is assigned to node } v \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

such that the minimum data rate over all data flows over all cells is maximized (ref. to **max-min fairness** [7]):

$$\arg\max_{A} \min_{\substack{v\in\mathcal{V},\\ w\in\mathcal{W}_v}} \left(\frac{\tau_{v,w}}{f_{v,w}} \times \sum_{s\in\mathcal{S}} A_{v,s} \times \tilde{R}^s_{v,w} \right) \tag{2}$$

[1] This is a simplification made in this paper. A strict interference avoidance is not spectrally efficient.

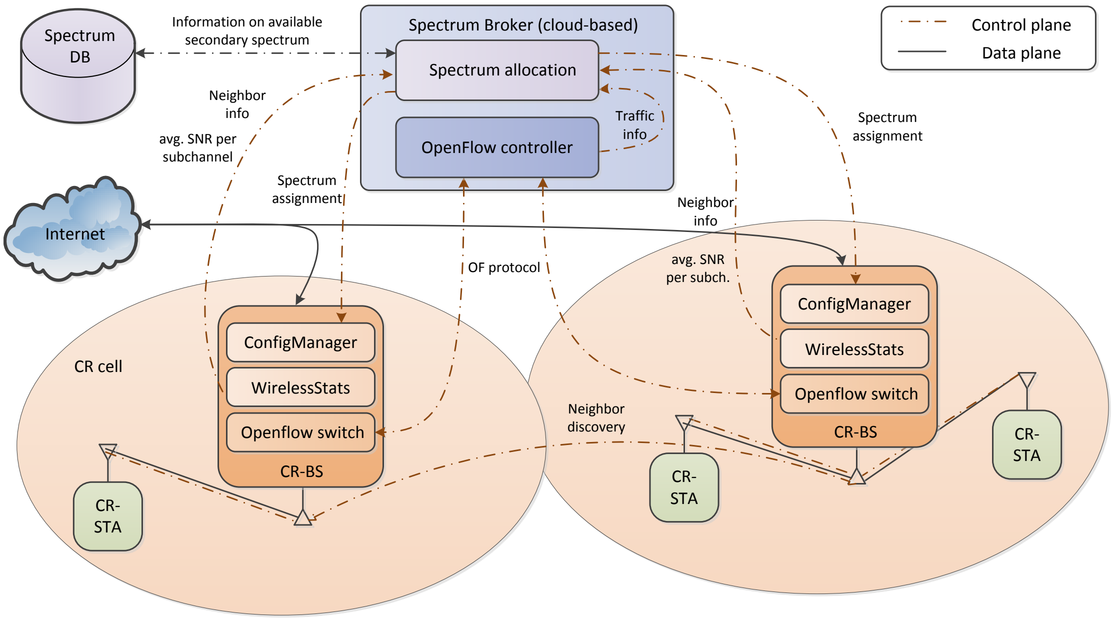


Fig. 2. Illustration of the proposed cloud-based centralized CR spectrum broker architecture.

with the average bitrate per subchannel:

$$\tilde{R}^s_{v,w} = \text{BW} \times \log_2\left(1 + \tilde{\gamma}^s_{v,w}\right) \tag{3}$$

where $\tau_{v,w}$ represents the time share assigned to a given CR-STA w within the corresponding CR-BS cell v. $f_{v,w}$ is the number of active flows sharing a particular link, BW is the subchannel's bandwidth and $\tilde{\gamma}^s_{v,w}$ is the average per subchannel SNR of a given link.

Subject to:

(I) **Interference avoidance:** assignment of subchannel s to node v does not cause interference to any CR-BS node in interference range, i.e. if $A_{v,s} = 1$ then $\forall u \in \mathcal{V} : (v,u) \in \mathcal{E} \rightarrow A_{u,s} = 0$.
(II) **PU protection:** assignment of subchannel s to node v does not cause interference towards any PU, i.e. if $A_{v,s} = 1$ then $s \in \mathcal{S}_v$.
(III) **Time sharing constraint:** the available radio resources within a cell need to be shared among active CR-STAs, i.e. $\forall v \in \mathcal{V} : \sum_{w \in \mathcal{W}_v} \tau_{v,w} = 1$.

From Eq. 2 we can see that a larger amount of spectrum need to be assigned to cells having a large number of flows and/or low quality wireless links. Note, this optimization problem is a binary integer linear programming problem which is known to be NP-complete ($\mathcal{NPC}$).

III. Proposed Approach

This section is divided into two parts. First, our envisioned cloud-based architecture is described in detail with a focus on the API provided by the CR-BSs towards the SB. Second, we present a topology-, traffic- and channel-aware spectrum assignment algorithm.

A. Architecture Details

The components of the envisioned architecture for a cloud-based centralized approach to spectrum brokerage in CRNs are shown in Fig. 2. The integral part of the proposed architecture is the cloud-based centralized controller, the Spectrum Broker (SB), which takes control over the medium to long-term spectrum assignment for the CR-BSs. Therefore, the CR-BSs under control report aggregated wireless statistics to the SB using a well-defined API. Moreover, by configuring proper rules in OpenFlow-enabled CR-BSs, the SB controller can get up-to-date information about the network traffic condition in the CRN.

1) CR-Base Station (CR-BS): The CR-BS serves a set of end-user CR-Stations (CR-STA). It implements a *ConfigManager* and a *WirelessStats* module. The *ConfigManager* is responsible for receiving configuration commands from the SB and executes them on the CR-BS. In particular, it configures the spectrum allocation to be used by a CR-BS and its associated CR-STAs as computed by the SB (*setSpectrumAllocation()*). The CR-BS is then responsible for short term resource allocation for its connected CR-STAs. The *WirelessStats* module reports wireless statistics like detected neighboring CR-BS nodes (*getNeighborInfo()*) and information about associated CR-STAs including the link quality (*getClientInfo()*) to the SB. Finally, each CR-BS is an OpenFlow (OF) compliant switch which allows the SB to monitor traffic on each CR link in each cell using the standardized OF protocol.

Although the SB makes medium to long-term spectrum assignments, it is however possible to detect PU activity locally and vacate the spectrum in short time to protect PUs. The following solutions to get a new spectrum assignment from the SB are possible: event based notification from CR-

TABLE I
CR-BS API DESCRIPTION.

ConfigManager	CR-BS receives configuration commands from the Spectrum Broker (SB) and executes them.
	setSpectrumAllocation(set of subchannels): configures the CR-BS to use the particular spectrum allocation, i.e. the set of subchannels.
WirelessStats	CR-BS reports wireless statistics to the SB.
	getNeighborInfo(): Scans the whole spectrum for neighboring CR-BS beacons and reports their MAC address and average signal strength on each subchannel. *getClientInfo()*: Reports information about associated CR-STAs, i.e. MAC address and average link SNR per subchannel.
OpenFlowSwitch	Each CR-BS is an OpenFlow (OF) enabled switch which is used by the SB for traffic monitoring within a cell. The OF protocol is used to add an OF matching rule for each new flow (*packet_in* operation) and to count the number of flow entries which allows the SB to estimate the number of active flows on each CR link in each CR-BS cell.

BS to the SB about the spectrum being used by the PU or excluding spectrum used by the PU from spectrum allocation at the CR-BS and passively waiting for new spectrum updates from the SB.

In summary, the API provided by each CR-BS is shown in Table I which is used by the SB to receive wireless statistics as well as to configure the spectrum allocation in each CR-BS node.

2) Spectrum Broker (SB): The SB is a centralized cloud-based controller which performs spectrum assignment to the cell under CR-BS control. Its decisions are based on wireless statistics he receives from each CR-BS under control (see previous Sec. III-A1). Moreover, it acts as an OpenFlow controller which configures the OF switch component in each CR-BS in such a way, that it is able to obtain network traffic statistics. In particular, the SB is interested in the number of active flows on each CR link in each cell. Flows can be identified based on fields in the packet header, such as source and destination IP address, protocol type, as well as source and destination port number (see Listing 1). If a packet does not match any of the existing rules in the CR-BS OF Switch (OFS), OpenFlow's default policy is to send a copy of that packet up to the controller which is our SB (*packet_in* operation). Our controller then adds the following OF forwarding/ matching rule to the OF table of the OFS of the corresponding CR-BS so that subsequent packets of the same flow will find a matching rule:

Listing 1. OF matching rule for TCP flows (similar rule need to be added for UDP flows)

```
OFPMatch( in_port=in_port, eth_type=eth_type,
  ipv4_src=src_ip, ipv4_dst=dst_ip, ip_proto=ip_proto,
  tcp_src=src_port, tcp_dst=dst_port, eth_dst=dst, eth_src=src )
```

The priority is set to be higher than the priority of the default rule. The timeout for the OF rule entry is set to ten seconds to ensure that inactive flow rules are removed from OFS.

Finally, the OpenFlow controller in the SB is configured to periodically poll the OpenFlow switches in each CR-BS to get updated information about the number of entries in the flow table. All flow entries having the same source MAC address identify the flows belonging to the same CR-STA. Hence, the SB is able to calculate the number of active flows on each wireless link in the CRN.

B. Spectrum Allocation Algorithm

1) Overview: In Sec. II-B the spectrum allocation is formulated as an optimization problem having a high computational complexity. A practical solution faces the following challenges. First, it needs to adapt to the dynamically changing secondary spectrum (i.e. due to appearance and disappearance of PUs). Second, in order to achieve the fairness among all flows it needs to adapt to the changing network traffic and channel conditions. Third, from the practical point of view a low-complexity algorithm (heuristic) is desirable.

Due to the Internet scale latencies between CR-BSs and the SB it is only possible to loosely couple the control and data plane. Hence, the envisioned spectrum broker performs spectrum assignment to CR-BSs based on coarse traffic monitoring (counting number of active flows on each wireless link), long-term link qualities (average SNR), as well as long-term network topology information (neighboring CR-BSs).

The basic objective of our proposed spectrum allocation algorithm is to assign available spectrum depended on the number of active flows per cell and station. Therefore, the obtainable (spectrum) resources are optimally shared between cells and their flows.

2) Detailed Description: In the following we give a detailed description of the proposed centralized spectrum allocation algorithm. The steps involved to calculate the spectrum allocation, i.e. the set of subchannels, for each CR-BS node (cell) are as follows.

Step 1: With the help of the information about the set of active flows we are calculating the average flow rate in each cell. Here we assume that the total secondary spectrum is available in each cell, i.e. all cells are isolated. Moreover, the algorithm requires information about the radio resource allocation algorithm used by the CR-BSs, e.g. proportional-fair, round-robin or maxmin-fair-scheduler. As discussed for Eq. 2, in case of maxmin-fair-scheduling we can compute the average flow rate in each cell, $v \in \mathcal{V}$, by solving the following system of linear equations:

$$A_v\, \boldsymbol{x}_v = \boldsymbol{b}_v\,, \tag{4}$$

where $A_v \in M(n \times n)$ and $\boldsymbol{b}_v \in \mathbb{R}^n$ are given, n is the total number of flows in cell v, i.e. $n = \sum_{w \in \mathcal{W}_v} f_{v,w}$ and $\boldsymbol{x}_v = (x_{v,1}, \ldots, x_{v,n})^T$ is the vector of unknowns which represents the optimal relative share of radio resources for each flow in cell v. A_v and $\boldsymbol{b}_v$ are constructed as follows:

$$A_v = \begin{pmatrix} \tilde{R}^*_{v,1} & -\tilde{R}^*_{v,2} & 0 & 0 & \dots & 0 \\ 0 & \tilde{R}^*_{v,2} & -\tilde{R}^*_{v,3} & 0 & \dots & 0 \\ \vdots & & \ddots & & & \vdots \\ 0 & \dots & & 0 & \tilde{R}^*_{v,n-1} & \tilde{R}^*_{v,n} \\ 1 & 1 & 1 & 1 & \dots & 1 \end{pmatrix}$$

$$\boldsymbol{b}_v = \begin{pmatrix} 0 & \dots & 0 & 1 \end{pmatrix}^T$$

where the vector $\tilde{R}^*_v$ is constructed as follows:

$$\tilde{R}^*_v = \big(\underbrace{\tilde{R}_{v,1} \;\cdots\; \tilde{R}_{v,1}}_{f_{v,1} \text{ times}} \;\cdots\; \underbrace{\tilde{R}_{v,t} \;\cdots\; \tilde{R}_{v,t}}_{f_{v,t} \text{ times}}\big)$$

where $1 \dots t \in \mathcal{W}_v$ and $\tilde{R}_{v,w}$ represents the average bitrate of link $v \rightarrow w$ per subchannel:

$$\tilde{R}_{v,w} = \text{BW} \times \sum_{s \in S} \log_2 \left(1 + \tilde{\gamma}^s_{v,w}\right) \tag{5}$$

where S is the total available secondary spectrum.
Now we are able to calculate the average flow rate in each cell as:

$$\tilde{r}_v = \text{mean}(\boldsymbol{x}_v \circ \tilde{R}^*_v) \tag{6}$$

where $\circ$ is the element-wise product of vectors.

Step 2: In order to avoid inter-cell interference different parts of the spectrum need to be assigned to co-located cells. Therefore, we have to find which cells are overlapping. In particular, the maximal cliques of CR-BS nodes[2] given a graph's boolean adjacency matrix Z are computed[3]:

$$C = \text{maxCliques}(Z) \tag{7}$$

Next we compute for each cell v in each clique c the optimal relative share of spectrum $x_{c,v} \in [0, 1]$ to be used. Our objective is to find a spectrum share such that each cell within a clique gets the same effective average cell flow rate, i.e. $\forall v1, v2 : \tilde{r}_{v1} \times x_{c,v1} = \tilde{r}_{v2} \times x_{c,v2}$. In particular, for any $c \in C$ we have to solve a system of linear equations, i.e.

$$A_c \, \boldsymbol{x}_c = \boldsymbol{b}_c \,, \tag{8}$$

where $A_c \in M(m \times m)$ and $\boldsymbol{b}_c \in \mathbb{R}^m$ are given, and $\boldsymbol{x}_c$ is the vector of unknowns which represents the relative spectrum share each cell in clique c will get.

$$A_c = \begin{pmatrix} \tilde{r}_1 & -\tilde{r}_2 & 0 & 0 & \dots & 0 \\ 0 & \tilde{r}_2 & -\tilde{r}_3 & 0 & \dots & 0 \\ \vdots & & \ddots & & & \vdots \\ 0 & \dots & & 0 & \tilde{r}_{n-1} & \tilde{r}_n \\ 1 & 1 & 1 & 1 & \dots & 1 \end{pmatrix}, \quad \boldsymbol{b}_c = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$

where n is the number of cells (CR-BSs) in clique c and $\tilde{r}_i$ are the average per cell flow rates as computed in step 1, Eq. 6.

[2]Can be efficiently done using the Bron-Kerbosch algorithm [8].
[3]Z can be easily computed from the learned network topology.

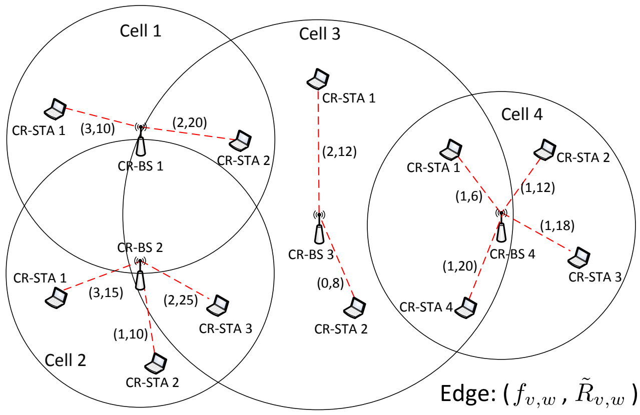


Fig. 3. Illustrative example network.

For any node which is member of more than one clique the spectrum share is set to the minimum, i.e. $\tilde{x}_v = \min(x_{c,v}), \forall c \in C$. Moreover, cells without any active flow get a minimum spectrum share τ.

Step 3: In the final step we have to map the computed spectrum share to the actual spectrum, i.e. the set of subchannels to be used. Our objective is to make sure that spectrum is assigned in such a way that there is no interference between co-located cliques of cells, here called inter-clique interference. Furthermore, to keep the computational complexity low, we limit ourselves to contiguous spectrum assignment per CR-BS. At first, the cliques are sorted according to their size in descending order. Hence, we start allocating spectrum to cells contained in the largest cliques. Unfortunately, the order in which the spectrum is assigned to cells in each clique plays an important role. An unfavorable order can result in suboptimal spectrum assignment. Therefore, we use an exhaustive, but computationally inexpensive search over all possible permutations, P_n, and select the solution with the minimum error between the computed spectrum share, $\tilde{\boldsymbol{x}}$, and the actual share, $\hat{\boldsymbol{x}}$, i.e.:

$$\arg\min_{p \in P_n} \max |\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}| \tag{9}$$

3) Example: We finish the description by giving an illustrative numerical example. Fig. 3 shows an example network of four cells. The tuple on every link represents the number of active flows $f_{v,w}$ on that link as well as the average link bitrate $\tilde{R}_{v,w}$.

The result from *step 1* for cell 1 is:
Given $\tilde{R}_{1,1} = 10$ and $\tilde{R}_{1,2} = 20$, $\tilde{R}^*_1$ is constructed to $\tilde{R}^*_1 = \begin{pmatrix} 10 & 10 & 10 & 20 & 20 \end{pmatrix}$ and therefore

$$A_1 = \begin{pmatrix} 10 & -10 & 0 & 0 & 0 \\ 0 & 10 & -10 & 0 & 0 \\ 0 & 0 & 10 & -20 & 0 \\ 0 & 0 & 0 & 20 & -20 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\boldsymbol{b}_1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \end{pmatrix}^T$$

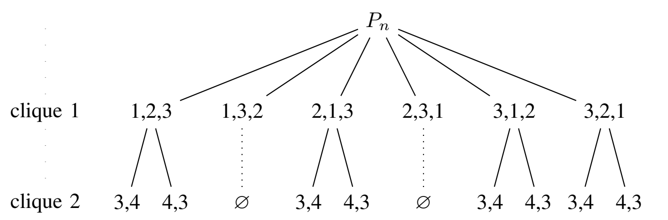


Fig. 4. All possible permutations of the given example. For clique 1 all $n!$ permutations are valid, while for clique 2 some combinations are prohibited due to constraints (no interference between co-located cliques).

which allows us to calculate the average flow rate for cell 1 as $\tilde{r}_1 = 2.5$. In the same way, we calculate the values for the remaining cells $\tilde{r}_2 = 2.6316, \tilde{r}_3 = 6.0$ and $\tilde{r}_4 = 2.8125$.

In *step 2* we estimate the maximal cliques as $C_A = (1, 2, 3)$ and $C_B = (3, 4)$. For clique A_A we compute:

$$A_A = \begin{pmatrix} 2.5 & -2.6316 & 0 \\ 0 & 2.6316 & -6.0 \\ 1 & 1 & 1 \end{pmatrix}$$

$$\boldsymbol{b}_A = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$$

Now, we can compute the optimal relative share of spectrum for each cell in each clique, i.e. $x_{A,1} = 0.4225, x_{A,2} = 0.4014, x_{A,3} = 0.1761, x_{B,3} = 0.3191$ and $x_{B,4} = 0.6809$. Note, that for cell 3 we have to take the minimum. The resulting spectrum share per CR-BS of each cell is $\tilde{x}_1 = 0.4225, \tilde{x}_2 = 0.4014, \tilde{x}_3 = 0.1761$ and $\tilde{x}_4 = 0.6809$. This numbers represent the percentage amount of spectrum to be allocated by each CR-BS.

Finally, as an example for *step 3*, Fig. 4 shows all possible permutations ($n!$) for mapping the computed spectrum share to the actual spectrum assignment. Some combinations are not feasible due to constraint of avoiding inter-clique interference. In order to illustrate that, in Fig. 5 three different spectrum arrangements are visualized. Fig. 5 (a) (b) shows some valid assignments for clique 1 and 2, while Fig. 5 (c) depicts an invalid combination. Due to the required alignment of cell 3 between both cliques there is not enough contiguous spectrum available for cell 4 in clique 2, i.e. the alignment of cell 3 leads to a collision with the spectrum chosen by cell 4.

IV. Evaluation

In this section we analyze the advantage of the proposed spectrum assignment algorithm as compared with the state-of-the-art by means of quantitative performance analysis. At first the methods under study are presented, followed by a description of the used evaluation methodology. Then the considered scenarios are presented and the obtained results are discussed in great detail. Finally, the communication overhead that is introduced due to the centralized control plane is analyzed.

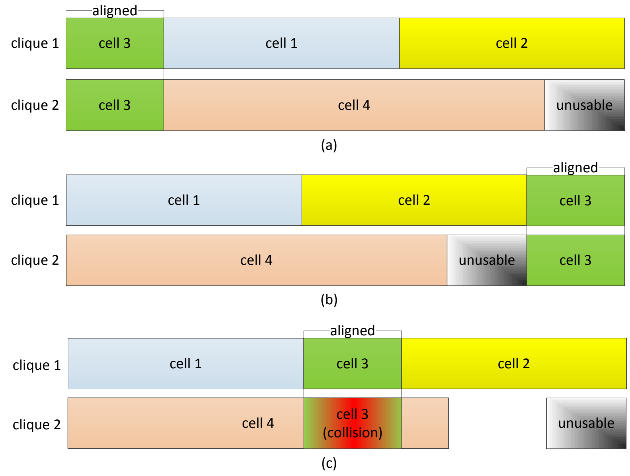


Fig. 5. Example visualization of step 3: (a) and (b) are valid arrangements, while (c) is prohibited.

A. Methods under Study

Our proposed spectrum assignment algorithm, hereinafter referred to as *proposed*, is compared against the following approaches:

- *Fair share per cell (base line)* - the simplest strategy which assigns to each cell the same share of spectrum independent from the number of CR-STAs, flows or channel qualities. It serves as a baseline.
- *Fair share per CR-STA* - the amount of spectrum assigned to a CR-BS depends on the number of associated CR-STAs. The larger the number of CR-STAs the larger the amount of spectrum assigned.
- *Fair share active CR-STAs only* - same as previous algorithm but considering only active CR-STAs, i.e. those having active flows.
- *Fair share active flows only* - the amount of spectrum assigned to each CR-BS depends on the number of active flows within this cell.

B. Methodology

The spectrum allocation algorithms under study are analyzed by means of a mix of system-level emulation and simulation. For the emulation various widely used tools are combined. We implemented a prototype of our envisioned system using Mininet [5], a container-based emulation which is able to simulate large topologies on a single computer. Mininet uses real kernel modules, as well as OpenFlow (OF) switches and applications installed on the host system. To emulate the OF controller we use the Ryu [9] framework, which is widely supported and applied in real deployments. Further, zeromq [10] is used for the exchange of wireless statistics and spectrum allocation configurations from the CR-BSs under control and the SB. Information about network traffic is obtained by the OF controller using the OF protocol [11]. The wireless channel is simulated using precalculated channel files from the ILMProp channel simulator [12]. The simulator was configured to calculate the post-processing SNR

TABLE II
PARAMETERS USED IN EVALUATION.

Parameter	Value
Center frequency	768 MHz
System bandwidth	512 MHz
PHY	NC-OFDM
No. of subcarriers	2048
No. of subchannels	64
No. of subcarriers / subch.	32
No. of data subcarriers / subch.	24
No. of pilots / subchannel	4
No. of guards / subchannel	$2 + 2 = 4$
Symbol interval	$4\,\mu s$
MAC	TDMA (proposed)
Transmit power (CR-BS/STA)	20 dBm
STA noise density (dBm/Hz)	-167 dBm/Hz
STA noise figure	6 dB
Direction	Downlink
Pathloss model	ILM prop (mix of LOS/NLOS [12])
No. of antennas	1 (SISO)
STA placement	Random disc (30-150 m)
Flow duration	10 s
PU model	Trace (Spectrum analyzer)
No. of placement seeds	500
Emulation hardware	Intel i7-5930K CPU, 32GB RAM
Emulation software	Ubuntu 12.10
	Mininet 2.2.0 [5]
	OVS on Mininet
	Ryu OFC framework [9]

per subcarrier and time sample (see Table II for details). Each sample in the channel trace file consists information about path loss, shadowing and fading components. At runtime we updated the effective bitrate of each wireless link emulated in Mininet based on the spectrum allocation by the SB and the resource scheduling within each CR-BS. Note, that we explicitly simulated the radio resource scheduling within each CR-BS' cell, i.e. a proportional fair scheduler was used.

For traffic modeling we used *iperf* to simulate TCP/IP packet flows between the CR-STAs and an imaginary gateway server.

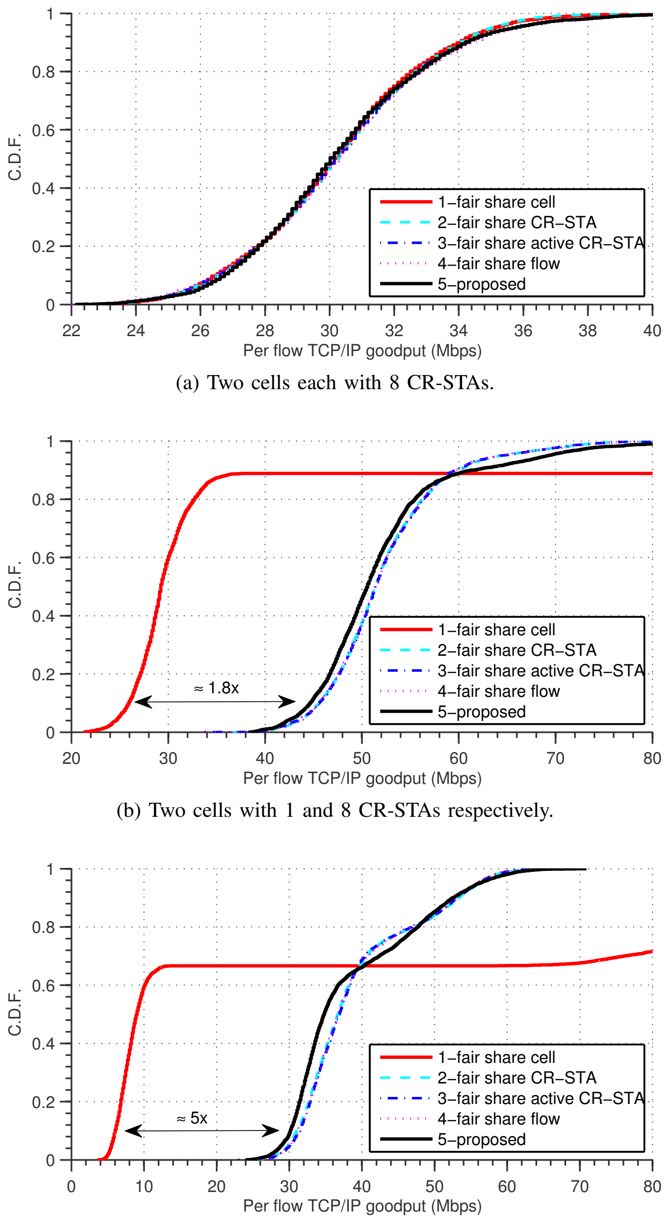
Different measurement studies show [1], [13] that PU behavior is very unsteady and depends on the used technology. Therefore, to get a realistic PU model mostly measurement studies are parametrized [14]. We followed that approach and conducted our own measurement campaign with a R&S FSV Spectrum Analyzer connected to a Multi-Polarized Ultra Wide Band antenna on the rooftop of our university building. For PU detection a simple energy detector was used with a threshold of -100 dBm. The results show that on average 55% (standard deviation of 5.6%) of the spectrum in the considered spectrum band was occupied by PUs.

The remaining most important parameters are summarized in Table II.

C. Results

This section presents the results of our evaluation. We present the per flow TCP/IP goodput (Mbps) as an CDF plot. As mentioned in Sec. II-B our objective is to maximize the minimum flow rate. Hence, the 10^{th} percentile throughput will be reported.

Experiment 1: (Impact of number of CR-STAs in each cell) To evaluate the impact from the number of CR-STAs in each cell we considered two scenarios with two and five overlapping CR-BS cells respectively. In both cases we set up a single traffic flow for each CR-STA to some host in the backhaul.



(a) Two cells each with 8 CR-STAs.

(b) Two cells with 1 and 8 CR-STAs respectively.

(c) Five cells with 1,1,1,1 and 8 CR-STAs respectively.

Fig. 6. Experiment 1.

Result 1: Fig. 6a shows the results for the scenario with two overlapping cells each with the same number of CR-STAs (here 8). Here we see that all strategies have roughly the same performance which can be explained by the homogeneity of both cells, i.e. same number of active STAs and also flows in both cells.

The situation is different if the number of CR-STAs and therefore flows in both cells is different as considered in

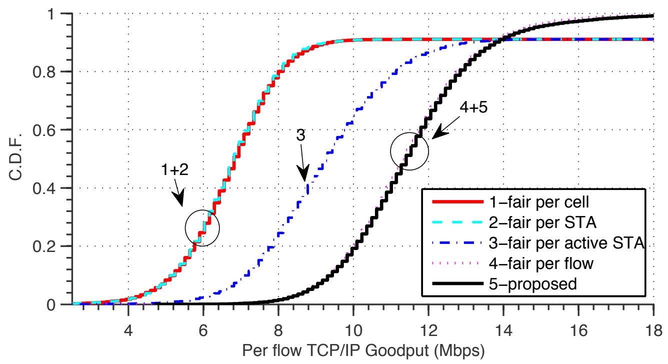


Fig. 7. Experiment 2: Two cells each with 8 CR-STAs. For each CR-STA in the left cell up to five flows with a probability of 10% each were set up. In the right cell each CR-STA had exactly 5 flows.

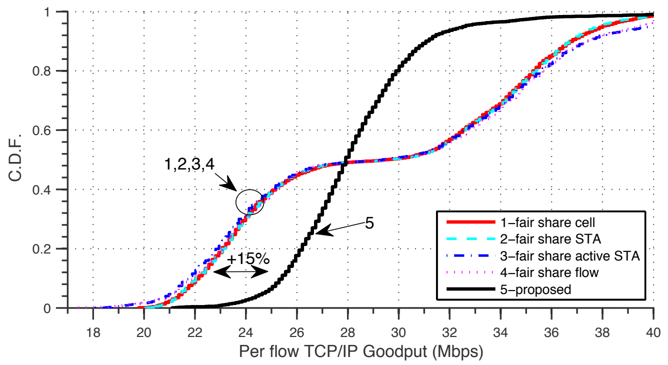


Fig. 8. Experiment 3: Two cells each with 8 CR-STAs. The CR-STAs in the first and second cell have above and below average SNR respectively.

Fig. 6b. Here, the CR-STAs are unevenly spread across the two cells, i.e. one and 8 respectively.

From the results we can see that minimum flow rate can be increased by a factor of 1.8 as compared to the simple static spectrum assignment (*fair share cell*). The advantage is even higher in the scenario with five cells with 1,1,1,1 and 8 CR-STAs respectively. From Fig. 6c we can observe an increased minimum flow throughput by a factor of around 5.

Note, that in both Fig. 6b and Fig. 6c the proposed spectrum allocation algorithm provides a slightly worse result compared with the other methods except the baseline.

Experiment 2: (Traffic-awareness) So far all CR-STAs are threaten to be active, i.e. have at least one active flow. Now we change this assumption and use following traffic model. For each CR-STA in the first cell up to five flows with a probability of 10% each were set up whereas in the second cell each CR-STA had exactly 5 flows. Hence, on average the number of flows in the second cell is ten times higher.

Result 2: From Fig. 7 we can see that the proposed method can increase the minimum flow throughput by around 78% as compared to baseline. Moreover, mode 3 (fair share active CR-STA) is unable to achieve the same performance. Hence checking whether a CR-STA is active or not is not sufficient.

Experiment 3: (channel awareness) So far we assumed the same random CR-STA placement in both cells. Now, the placement is changed in such a way that in the first cell the CR-STAs have a SNR above and in the second cell the CR-STAs have a below average SNR respectively. The number of flows per CR-STA is one.

Result 3: From Fig. 8 we can observe that only our proposed method is able to consider this by achieving a 15% higher performance.

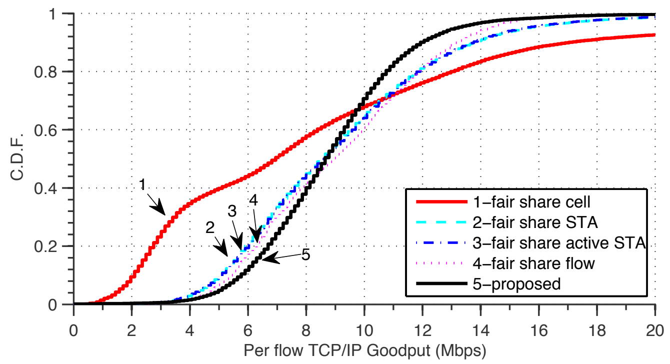


Fig. 9. Experiment 4: Five cells with 1,2,4,6 and 8 CR-STAs respectively.

Experiment 4: (Everything at a glance) Finally, we consider five cells each with 1, 2, 4, 6 and 8 CR-STAs respectively. At each CR-STA up to five flows with a probability of 50% each were set up.

Result 4: Fig. 9 confirms the results from previous experiments.

D. *Overhead Analysis*

Our centralized architecture requires the exchange of a considerable amount of control information between the Spectrum Broker (SB) and all its CR-BSs under control. For the sake of completeness, this signaling overhead is analyzed in this section.

The control overhead is highly affected by different factors: First, the reporting interval of wireless statistics from the CR-BSs under control to the SB. Second, the configuration of spectrum allocation which depends on the number of CR-BS nodes under control (as well as their associated CR-STAs) and the update rate δ. In the following analysis we have considered three different values of δ: ranging from very frequently (10 Hz, which is below channel coherence time[4]) over a moderate update rate of 1 Hz, up to an infrequent rate of 0.1 Hz, which will be enough in the case of long-term spectrum assignment. The third factor is traffic monitoring. Here, the number of new flows arriving at each CR-BS needs to considered. In OpenFlow the first packet of each unknown flow, i.e. not matching any existing flow rule, is send to the OF controller (in our case the SB) for processing (*packet_in* operation). Hence, the traffic monitoring overhead depends on the arrival rate of new flows. Flows are generated in our system by CR-STAs, which can be assumed as end user devices. It is important to note that flows like file transfer (e.g., FTP) or video streaming (e.g., RTP) generate a lot of traffic,

[4]According to [15, p.90] the channel coherence time is about 200 ms in our considered scenario.

TABLE III
PARAMETERS USED IN OVERHEAD ANALYSIS.

Parameter	Value
No. of subchannels	64
Neighbor discovery and client information update rate δ	0.1 Hz; 1 Hz; 10 Hz
getNeighborInfo()	
SNR per discovered neighbor (CR-BS)	64 Bytes (1 Byte/ subchannel)
Address of discovered neighbor	6 Bytes (IEEE MAC address)
getClientInfo()	
SNR per CR-STA	64 Bytes (1 Byte/ subchannel)
Address of CR-STA	6 Bytes (IEEE MAC address)
No. of CR-STAs per CR-BS	4; 16
setSpectrumAllocation()	
Spectrum allocation information	8 Bytes (1 Bit/ subchannel)
Address of CR-BS	6 Bytes (IEEE MAC address)
OpenFlowPacketIn()	
Packet size per *packet_in*	200 Bytes (from [17])
Flow generation rate per CR-STA	0.8 Hz (from [16])

but they are marginal in terms of control overhead, as they only sporadically generate a *packet_in* operation. Therefore, we consider mainly flows related to web browsing, as those can generate a considerable amount of control traffic. For our analysis we have used the web browsing traffic model from [16] to estimate the average number of different objects per web page, as well as the reading time of that particular web page. A web page consists of a main object and a varying number of embedded objects. Here, we assume one flow per object. Although, the mean object number per web page in [16] is 5.64, our recent analysis shows a much higher number of embedded objects for major web pages, that are using modern web technologies as, e.g. AJAX. Hence, we used an average of 25 objects in total per web page and a mean reading time of 30 seconds to estimate the flow generation rate per CR-STA. Further, according to [17], it is assumed that only 200 Bytes of the first packet are sent per *packet_in* operation to the SB. All detailed parameters are summarized in Table III.

Fig. 10 shows the results of the signaling overhead analysis for a network consisting of up to 16 overlapping cells with each 4 or 16 CR-STAs connected. First, it can be observed that the overhead for large networks (16 CR-BSs with 16 CR-STAs each) and a high refresh rate (δ = 10 Hz), which is required for short term spectrum allocation, results in a significant overhead of up to 2 Mbps. However, in the envisioned system the spectrum allocation is performed on medium to long-term basis. With this assumption, even for large CRNs, the overhead stays below 0.5 Mbps. The signaling overhead can be further reduced by lowering its reporting and update frequency δ, which however could lead to outdated wireless statistics and spectrum allocation. Another approach to reduce the control channel overhead is the reduction of the number of *packet_in* operations reported by the OFS in the CR-BSs. The basic idea is to restrict the *packet_in* operation only to long living, so called elephant flows [18]. It is not necessary to consider short live flows, since our medium to long-term spectrum assignment is not able to adapt to short-term dynamics.

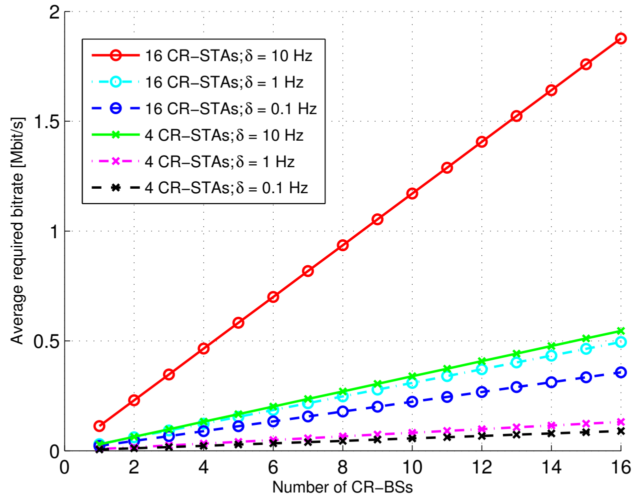


Fig. 10. Result of the overhead analysis. The overhead strongly depends on the number of CR-STAs per cell and the update rate δ of the wireless statistics.

V. RELATED WORK

There is already a wide use of SDN for centralized control and management of enterprise WiFi [19], [20], [21]. Zhao et al. [19] proposed an OpenFlow-based framework to mitigate interference among 802.11 APs by scheduling downlink packets according to corresponding rules in the flow table. Shrivastava et al. [22] proposed a framework for centralized packet scheduling to overcome the performance issues of the 802.11 DCF MAC protocol in enterprise WiFi networks. A framework to configure, control and to manage 802.11 WiFi networks in dense residential deployments using OpenFlow was proposed by Patro et al. [23]. Therefore, the authors proposed to extend the OpenFlow protocol with specific protocol extensions. One application was a centralized configuration of WiFi channels used by APs which can be seen as a special case of spectrum brokerage of assigning just a single channel. With OpenRoads [24] the OpenFlow protocol has been extended to meet the requirements of wireless networks. Specifically it allows to control and to monitor parameters on the wireless physical layer as e.g. channel frequency and transmission power using the Simple Network Management Protocol (SNMP) protocol.

Next, papers related to spectrum allocation schemes are discussed. Conceptually, our formulated spectrum allocation problem has similarities with dynamic multiuser subchannel allocation of OFDM systems as proposed by e.g. Rhee et al. [25]. However, they assume a single base station with multiple users and performing short term spectrum allocation. In contrast our approach is managing the spectrum allocation among multiple base stations based on medium to long-term basis. In this paper we follow a centralized approach to spectrum allocation. However, there are many distributed solutions available and are summarized in the following. Peng et al. [26] reduced the spectrum/ channel allocation problem to a graph coloring problem which is NP-hard and proposed a centralized, as well as distributed heuristic to solve it.

Cao et al. [27] suggested a distributed spectrum assignment approach based on local fair bargaining with feed poverty to ensure a theoretical lower bound for each user. This is similar to Zhao et al. [28], they suggested a distributed bargaining process to improve fairness in channel assignment. Due to the decentralized structure both approaches are using only information about locally available channels. Finally, in [29] a distributed version of the spectrum allocation algorithm proposed in this paper is discussed and deeply evaluated.

VI. Conclusions

In this paper we take advantages of the ideas of SDN and cloud computing technology to manage interference in CRN deployments in residential areas by assigning different parts of the spectrum to co-located CR-BSs. The proposed architecture allows a very fine-grained spectrum allocation in the CRN taking into account network topology information as well as dynamically changing traffic and channel conditions. A concrete centralized spectrum allocation algorithm is proposed, evaluated and compared with state-of-the-art.

As future work we are planning to implement our solution using a software defined radio platform to conduct real world experiments. It is also possible to extend the proposed architecture to take into account additional constraints for optimization, e.g. to allocate spectrum for PU detection by means of sensing in each CR-BS. Our flexible architecture enables us to take additional features into account, e.g. allowing the spectrum assignment algorithm to use non-continuous spectrum for each CR-BS. This will make the algorithm more complex but increases efficiency.

Acknowledgment

This work has been supported by the COUWBAT project of the German Federal Ministry of Education and Research (BMBF 16KIS0029) and the European Union's Horizon 2020 research and innovation programme under grant agreement No 645274 (WiSHFUL project).

References

[1] V. Valenta, R. Marsalek, G. Baudoin, M. Villegas, M. Suarez, and F. Robert, "Survey on spectrum utilization in europe: Measurements, analyses and observations," in *Cognitive Radio Oriented Wireless Networks & Communications (CROWNCOM)*. IEEE, 2010, pp. 1–5.

[2] R. N. Yadav, R. Misra, U. Gupta, and S. Bhagat, "Opportunistic spectrum access in CR network in licensed and unlicensed channels," in *Proceedings of the 2015 International Conference on Distributed Computing and Networking*. ACM, 2015, p. 39.

[3] L. Duan, L. Gao, and J. Huang, "Cooperative spectrum sharing: a contract-based approach," *Transactions on Mobile Computing*, vol. 13, no. 1, pp. 174–187, 2014.

[4] B. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *Communications Surveys Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014.

[5] [Online]. Available: http://mininet.org/

[6] B. Farhang-Boroujeny, "OFDM versus filter bank multicarrier," *Signal Processing Magazine, IEEE*, vol. 28, no. 3, pp. 92–112, 2011.

[7] D. Bertsekas and R. Gallager, *Data Networks*. Prentice-Hall, 1992.

[8] C. Bron and J. Kerbosch, "Algorithm 457: finding all cliques of an undirected graph," *Communications of the ACM*, vol. 16, no. 9, pp. 575–577, 1973.

[9] [Online]. Available: http://osrg.github.io/ryu/

[10] [Online]. Available: http://zeromq.org/

[11] [Online]. Available: https://www.opennetworking.org/

[12] [Online]. Available: http://www2.tu-ilmenau.de/nt/en/ilmprop//

[13] R. Urban, T. Korinek, and P. Pechac, "Broadband spectrum survey measurements for cognitive radio applications," *Radioengineering*, vol. 21, no. 4, pp. 1101–1109, 2012.

[14] Y. Saleem and M. H. Rehmani, "Primary radio user activity models for cognitive radio networks: A survey," *Journal of Network and Computer Applications*, vol. 43, pp. 1–16, 2014.

[15] J. G. Andrews, A. Ghosh, and R. Muhamed, *Fundamentals of WiMAX: understanding broadband wireless networking*. Pearson Education, 2007.

[16] R. Srinivasan, J. Zhuang, L. Jalloul, R. Novak, and J. Park, "IEEE 802.16m evaluation methodology document (emd)," *IEEE 802.16 Task Group m (TGm) Core Document*, 2008.

[17] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, "Nox: towards an operating system for networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 3, pp. 105–110, 2008.

[18] C.-Y. Lin, C. Chen, J.-W. Chang, and Y. H. Chu, "Elephant flow detection in datacenters using openflow-based hierarchical statistics pulling," in *Global Communications Conference (GLOBECOM), 2014 IEEE*, Dec 2014, pp. 2264–2269.

[19] D. Zhao, M. Zhu, and M. Xu, "Leveraging SDN and OpenFlow to mitigate interference in enterprise WLAN," *Journal of Networks*, vol. 9, no. 6, pp. 1526–1533, 2014.

[20] Y. Grunenberger and F. Rousseau, "Virtual access points for transparent mobility in wireless LANs," in *Wireless Communications and Networking Conference (WCNC)*. IEEE, 2010, pp. 1–6.

[21] P. Dely, J. Vestin, A. Kassler, N. Bayer, H. Einsiedler, and C. Peylo, "Cloudmac: An Openflow based architecture for 802.11 MAC layer processing in the cloud," in *Globecom Workshops*. IEEE, 2012, pp. 186–191.

[22] V. Shrivastava, N. Ahmed, S. Rayanchu, S. Banerjee, S. Keshav, K. Papagiannaki, and A. Mishra, "CENTAUR: realizing the full potential of centralized WLANs through a hybrid data path," in *Proceedings of the 15th annual international conference on Mobile computing and networking*. ACM, 2009, pp. 297–308.

[23] A. Patro and S. Banerjee, "COAP: A Software-Defined approach for home WLAN management through an open API," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 18, no. 3, pp. 32–40, 2015.

[24] K.-K. Yap, M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown, "Openroads: Empowering research in mobile networks," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 1, pp. 125–126, 2010.

[25] W. Rhee and J. M. Cioffi, "Increase in capacity of multiuser ofdm system using dynamic subchannel allocation," in *Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st*, vol. 2. IEEE, 2000, pp. 1085–1089.

[26] C. Peng, H. Zheng, and B. Y. Zhao, "Utilization and fairness in spectrum assignment for opportunistic spectrum access," *Mobile Networks and Applications*, vol. 11, no. 4, pp. 555–576, 2006.

[27] L. Cao and H. Zheng, "Distributed spectrum allocation via local bargaining." in *Second Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*. IEEE, 2005, pp. 475–486.

[28] J. Zhao, H. Zheng, and G.-H. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2005, pp. 259–268.

[29] A. Zubow, M. Doering, and A. Wolisz, "Distributed Spectrum Allocation for Autonomous Cognitive Radio Networks," in *20th European Wireless Conference*. VDE, 2014, pp. 591–597.