

Cognitive Computing Resource Management for a Ubiquitous Wireless Access*

Vuk Marojevic, Nemanja Vucevic, Xavier Revés, and Antoni Gelonch

Dept. of Signal Theory and Communications, Universitat Politècnica de Catalunya,
C/ Jordi Girona 1-3, Modul D4,
08034 Barcelona, Spain
{marojevic,vucevic,xavier.reves,antoni}@tsc.upc.edu

Abstract. In the near future many radio access technologies (RATs) will coexist to offer a vast variety of user services at different quality levels. Software-defined radio (SDR) facilitates to flexibly choose the RAT by means of a software-reconfiguration of the SDR terminal. Cognitive radio automates this process, being an ambient aware, intelligent radio resource management system. We extend its scope in that we introduce the cognitive computing resource management. Our cognitive radio system scans the computing environment, decides whether to reconfigure a candidate SDR mobile terminal or not, and learns from the effects of these decisions. The simulations, which model a simplified though realistic case study, show that two simple computing resource management algorithms outperform an algorithm that is unaware of the present computing environment. The paper concludes that a cognitive computing resource management is necessary for providing a ubiquitous wireless access where the user demands are satisfied as a function of the currently available computing resources of their terminals.

1 Introduction

Today's wireless communications can be described as a heterogeneous mix of radio access technologies (RATs). The emerging 3rd generation (3G) systems, for example, neither pretend to substitute the worldwide available access to 2nd generation (2G) services nor directly compete with wireless local area networks (WLANs). These RATs will rather coexist and complement each other: 2G systems offer worldwide roaming for carrying voice traffic and low-volume data. 3G systems will offer higher data rates for many different classes of multimedia services. WLANs locally offer very high data rates at lower cost. This view of cooperative rather than competitive networks facilitates the ubiquitous wireless access.

An intelligent (radio) resource management system is necessary to fully exploit the heterogeneous and dynamic wireless access networks of today and tomorrow. Such a system should observe the environment and react intelligently upon its changes. This

* This work has been supported by the DURSI of the Catalanian Government and the CYCIT (Spanish National Science Council) under grants TIC2003-08609 and TEC2006-09109, which is partially financed from the European Community through the FEDER program.

idea is known as *cognitive radio* [1] [2]. Cognitive radio is an ambient aware system that automates the joint radio resource management (JRRM). JRRM refers to the coordination of the spectrum management of the different RATs that are accessible within a certain area.

The cognitive radio concept is an extension of software radio or software-defined radio (SDR) [3]–[5], which was coined in the early 90s. Software radio has led to a conceptual change in radio communications, from inflexible and hardware-centric to software-reconfigurable mobile terminals and network elements. In particular, the signal processing chains for radio communication of SDR platforms are to some extent defined in software and executed on programmable hardware. Then, a simple software switch can change the operational mode of an SDR platform to use a particular radio access technology for establishing a radio link between the communicating entities.

Future SDR platforms will contain some number of programmable processors, such as general-purpose processors (GPPs), digital signal processors (DSPs), or field-programmable gate arrays (FPGAs) [6]. An SDR mobile terminal will comprise a few, an SDR base station many of these processors to provide sufficient processing power for the execution of SDR applications. An SDR base station will execute many SDR applications in parallel, one for each user. Although this paper deals with SDR mobile terminals, the introduced concepts and conclusions can be directly applied to SDR base stations.

In future, the software-defined signal processing chains for many RATs will be available on servers. It is not envisaged that more than one RAT-specific signal processing chain, the active one, will be locally available on an SDR mobile terminal, because of its memory constraints and the steady evolution of RATs. Therefore, before the reconfiguration of an SDR mobile terminal, the new RAT-software is downloaded from the network using a radio link (over the air, or OTA, download) [7].

We consider software radio as an application of general-purpose computing, where the software, or application, defines the hardware functionality: An SDR application represents the signal processing chain that defines the radio functionality of an SDR platform. Hence, software-defined radio is not only about radio but also about computing [8].

In this paper we extend this vision of computing resource management to cognitive radio systems. Our cognitive radio system analyzes the computing environment and takes intelligent decisions from its observations and the accumulated knowledge (section 2). We simulate a real-world scenario to demonstrate the importance of cognitive computing resource management for a ubiquitous wireless access (sections 3–5).

2 Cognitive Computing Resource Management

Cognitive radio, introduced in the late 90s, is an ambient aware system that reacts upon changes in the radio environment. It incorporates a learning mechanism and is, thus, an intelligent system that pretends to optimize the usage of radio resources [9].

We already argued for the importance of computing resource management in software-defined radio [8]. In this paper we correspondingly extend the scope of cognitive radio introducing the *cognitive computing cycle*, which observes and intelligently

manages the computing resources of SDR platforms. Some arguments for this extension are the expected commercialization of software-defined radio and the vast variety of SDR platforms.

Fig. 1 shows the cognitive computing cycle. It continuously scans the computing environment and provides any changes to the computing resource management (CRM) entity, which decides upon an appropriate action. The CRM entity is the brain of the system; it reacts upon its inputs and its current knowledge. This knowledge is accumulated during the learning process of the CRM algorithm, which observes the effects of its decisions (*learning-by-observing*). Hence, the cognitive computing cycles has to run continuously.

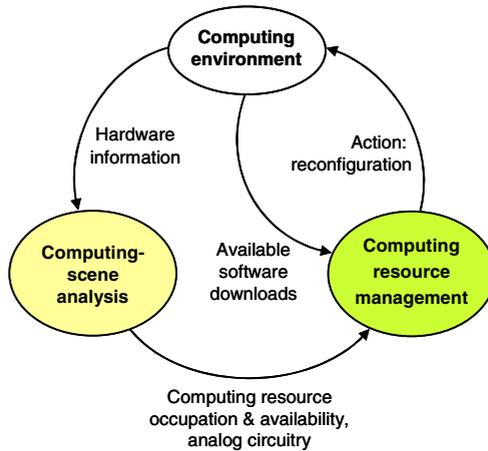


Fig. 1. The cognitive computing cycle

Let us consider a simple example: In a certain area two radio access technologies, RAT A and RAT B, are supported. If the radio resources of RAT A are overused, while those of RAT B are not (radio scene analysis, see [9]) and if the network infrastructure, including software downloads, and mobile terminals facilitate a switch to RAT B (computing scene analysis, see Fig. 1), the action could be to reconfigure some mobile terminals to access RAT B. This paper argues that such a scenario requires a cognitive computing resource management that should be able to initiate the reconfiguration of the appropriate SDR mobile terminals.

The computing resources of emerging SDR platforms are and will be limited. Even though the computing power may once grow faster than its requirement, related issues, such as energy consumption, will not become obsolete even in the far future of radio communications. It is therefore necessary to optimize the usage of SDR computing resources to facilitate an anytime, anywhere, and anyhow wireless access. In particular, we expect that many cognitive computing cycles will be globally distributed and coordinated to intelligently use the available computing resources of SDR platforms and, thus, provide a ubiquitous wireless access where the user is unaware of the RAT that satisfies its service demand.

The computing cycle of Fig. 1, which addresses the intelligent reconfiguration of SDR mobile terminals, can be easily extended to aid the reconfiguration of SDR network elements as well. Although network elements are less critical in terms of computing power, their intelligent management can greatly reduce power consumption and, thus, reduce the operational costs. The rest of the paper does not study these aspects but rather pretends to demonstrate the need for a cognitive computing resources management in future wireless communications. Therefore, we consider a typical case study and evaluate three simple CRM algorithms or policies. Without loss of generality, we focus on the cognitive management of SDR mobile terminals.

3 Case Study

The spectrum scanning at a certain time and area with 2G and 3G coverage shows that the number of GPRS (general packet radio service) sessions is elevated. Some of these terminals should be reconfigured to operate in another mode to increase the network capacity and, thus, potentially satisfy more users. We assume that the UMTS (universal mobile telecommunications system) radio resources are underused and that the necessary infrastructure is available. A reconfiguration of some of the SDR-GPRS terminals to access the UMTS RAT would balance the traffic load between the two standards so that penetrating 2G mobile terminals, as opposed to SDR mobile terminals, could still initiate their GPRS sessions.

This case study considers 15 000 SDR-GPRS mobile terminals of different architectures and computing loads. The objective is to reconfigure any SDR-GPRS mobile terminal that provides the necessary computing capacity for executing the SDR-UMTS signal processing chain instead of the SDR-GPRS application. We assume that additional, non-RAT-specific applications may concurrently run on an SDR mobile terminal. Hence, part of a platform's computing resources may momentarily not be available for the SDR-UMTS application.

The CRM algorithms, which we describe in continuation, evaluate the reconfiguration of each SDR mobile terminal separately, that is one by one; they either initiate the reconfiguration of a particular terminal or not.

3.1 Computing Resource Management Algorithms

Baseline Algorithm (Alg_0). Algorithm 0 is not aware of the SDR mobile terminals' computing capabilities and, thus, initiates the reconfiguration of any candidate SDR terminal. A terminal that does not have sufficient computing capacity to execute the SDR-UMTS signal processing chain fails in applying the requested reconfiguration and, consequently, loses its currently running session. The system chooses the 384 kbps UMTS realization because it offers the best quality of service.

CRM Algorithm 1 (Alg_1). Algorithm 1 uses the supplied hardware information. It reconfigures mobile terminals as a function their hardware capacities and the available reconfiguration information. This information is successively accumulated during the learning process: If a mobile terminal loses its session due to an unsuccessful reconfiguration (*infeasible mapping*, which means that the system's computing or

timing requirements are not met), terminals of these characteristics are not considered again for reconfiguration.

The system, furthermore, dynamically chooses the particular UMTS realization as a function of a terminal's processing capacity; it selects the UMTS realization as a function of the terminal's total processing capacity P_T :

- 384 kbps UMTS if $P_T \geq T_{384}$,
- 128 kbps UMTS if $T_{128} \leq P_T < T_{384}$,
- 64 kbps UMTS if $T_{64} \leq P_T < T_{128}$,
- no reconfiguration if $P_T < T_{64}$.

Later we specify the values of the thresholds T_{384} , T_{128} , T_{64} .

Some services may not be maintained at the same quality when implementing a 64 kbps or a 128 kbps instead of a 384 kbps UMTS receiver. Here we assume that the users are willing to accept a possible quality degradation for the sake of a ubiquitous wireless access. Typically, a user prefers a lower QoS than no service at all.

CRM Algorithm 2 (Alg. 2). Algorithm 2 is an extension of Algorithm 1. Initially, the same three thresholds are defined. The learning process is, though, slightly different: If a reconfiguration to the 384 kbps (128 kbps) realization has failed, the next terminal of the same characteristics is considered for reconfiguration to 128 kbps (64 kbps) UMTS. Unfeasibly mapping the 64 kbps UMTS processing chain means that this kind of terminal will not be considered for reconfiguration again.

3.2 SDR Platforms

Fig. 2 shows 3 SDR mobile terminal types. Fig. 2a illustrates a full-duplex architecture, where all three processors P_1 , P_2 , and P_3 are interconnected by dedicated communication lines. The half-duplex architecture of platform II permits a more flexible bandwidth assignment between any two processors. Finally, the bus system of Fig. 2c permits a flexible assignment of the platform's bus bandwidth. The application mapping process (subsection 3.4) implicitly handles the necessary bus scheduling in case of architectures II and III.

These platform types represent a small, though representative, excerpt of the numerous possible SDR mobile terminal architectures. P in Mops (million operations per second) and B in Mbps (mega-bits per second) define a platform's processing and bandwidth capacities after the projected deallocation of the SDR-GPRS transceiver functions. Each terminal's computing capacity then consists of the total processing capacity $P_T = 3 \cdot P$ and the total bandwidth $B_T = 6 \cdot B$.

This case study addresses 5000 SDR mobile terminals of each type. Their computing capacities are randomly obtained, where P and B are uniformly distributed in [2700, 2750, ..., 4200] and [200, 250..., 1500]. This way we simulate SDR platforms of different architectures and computing loads.

We assume that the 15 000 SDR-GPRS platforms contain the analog RF circuitry that is necessary to operate in UMTS mode. We address the digital signal processing part at the physical layer of an UMTS downlink receiver (see next subsection) and presume that additional computing resources for implementing the higher protocol layers and the UMTS uplink transmitter are sufficiently available.

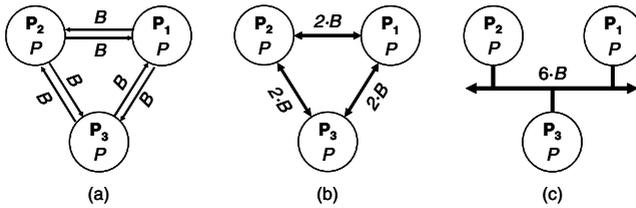


Fig. 2. SDR platform types: I full-duplex (a), II half-duplex (b), and III bus (c) architectures

3.3 SDR Applications

Figs. 3 and 4 show the computing requirements of the chip- and bit-rate processing chains of a UMTS downlink receiver supporting the raw data rates of 64, 128, and 384 kbps. We obtained the processing and bandwidth demands from [10]–[13] and from available implementations, considering that a processing requirement is the number of multiply-accumulate operations times the required processing frequency. Similarly, the bandwidth requirement is the sampling frequency f_s multiplied by the number of bits per sample.

The three processing chains require a minimum processing capacity of 8133, 8301, and 9403 Mops. Because the total processing capacity $P_T = 3 \cdot P$ of an SDR platform takes the values 8100, 8250, ..., 9300, 9450, ..., 10 350, the CRM entity straightforwardly specifies thresholds T_{64} , T_{128} and T_{384} as 8250, 8400, and 9450 Mops. We introduce $T'_{384} = T_{384} / 3$, $T'_{128} = T_{128} / 3$, and $T'_{64} = T_{64} / 3$ for a simplified notation.

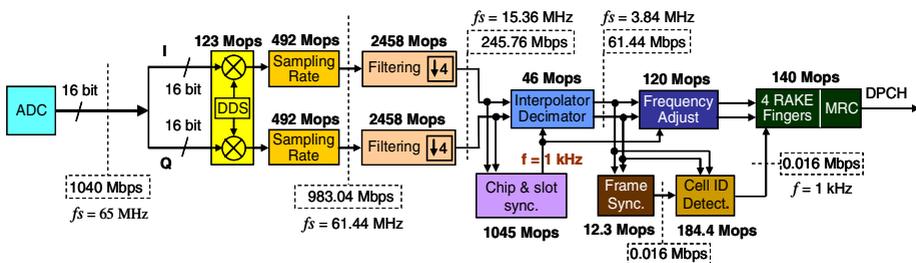


Fig. 3. Chip-rate computing requirements of a UMTS downlink receiver

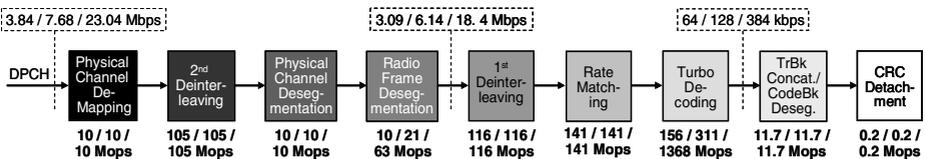


Fig. 4. Bit-rate computing requirements of a 64 / 128 / 384 kbps UMTS downlink receiver

3.4 Application Mapping

The SDR mobile terminals of this case study are equipped with the t -mapping algorithm [8], which maps the downloaded SDR-UMTS processing chain to the platform's available computing resources. The SDR functions are mapped one by one to all processors minimizing the sum between the computation and the communication costs. A bandwidth is reserved if two communicating processes are mapped to different processors; otherwise, the data is accessed via shared memory. The mapping of each SDR function to all N processors results in N different mapping options of the entire processing chain. The algorithm then chooses the least-cost solution.

The low complexity of this mapping approach greatly facilitates its implementation. Since being non-optimal, it does not always find a feasible solution. A feasible solution, or *feasible mapping*, reserves no more than 100% of any computing resource and, hence, meets the RAT-specific computing constraints. In particular, a feasible mapping can meet the timing constraints because our SDR system modeling considers time as an implicit resource [8].

4 Results

Fig. 5 shows the number of lost sessions as a function of the SDR platform index. It shows that the curves corresponding to Alg_1 and Alg_2 saturate, indicating the learning process: Alg_1 and Alg_2 learn which terminals can be feasibly reconfigured and which not. Alg_0, on the other hand, constantly loses user sessions, because lacking any hardware information.

Figs. 6a-c show the evolution of successful reconfigurations as a function of the CRM algorithm and the platform type. Fig. 7 presents the number of operating SDR terminals after the 15 000 test cases and their operational modes. Then, the sum between the remaining SDR-GPRS mobile terminals, the 64 kbps, 128 kbps, and 384 kbps UMTS sessions, and the number of lost sessions due Fig. 5 is 5000 for each platform type.

Figs. 8-10 illustrates the accumulated mapping information, which is obtained during the learning process of the two CRM algorithms. (The complete information, as shown in Figs. 8-10, is only partially obtained during the first 5000 terminals of each type.) The topmost level (level 3 - 384 kbps) indicates that the corresponding (P, B) -tuples can be feasibly reconfigured to 384, 128, or 64 kbps UMTS. The second level indicates that the corresponding terminals can be feasibly reconfigured to the 128 or the 64 kbps realizations, whereas the first level means a maximum achievable bit-rate of 64 kbps. Any P - B crossing at the bottom level of Fig. 8, 9, or 10 symbolizes an insufficient computing capacity (processing or bandwidth capacity, or both) for operating in a UMTS mode due to the particular CRM algorithm and application mapping process. These figures explain the results of Figs. 5-7, which we analyze in more detail below.

Fig. 5 shows that the number of lost sessions is the highest for the SDR mobile terminals of type I and the lowest for those of type III. Correspondingly, the number of feasible reconfigurations is the lowest for the terminals of type I and the highest for

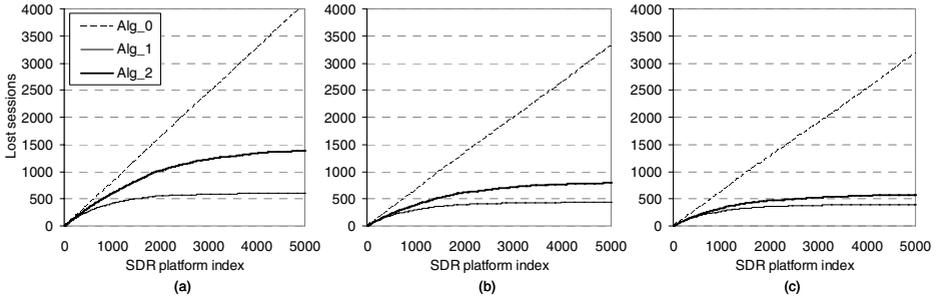


Fig. 5. Number of lost sessions as a function of the CRM algorithm and the SDR platform index for SDR platform types I (a), II (b), and III (c)

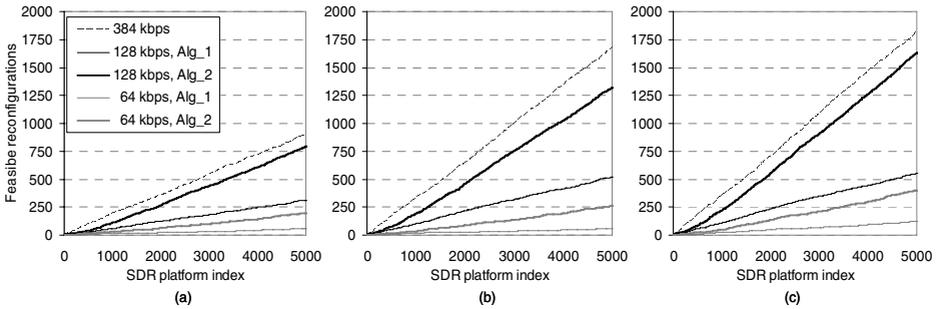


Fig. 6. Number of feasible reconfigurations as a function of the CRM algorithm and the SDR platform index for SDR platform types I (a), II (b), and III (c)

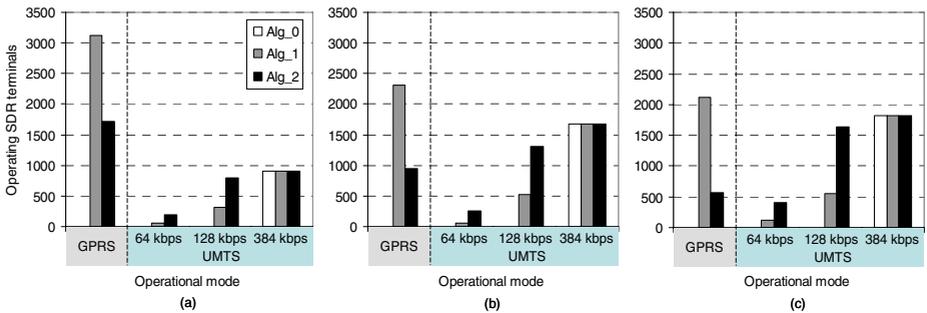


Fig. 7. Final number of non-reconfigured and feasibly reconfigured SDR mobile terminals of platform types I (a), II (b), and III (c)

those of type III (Fig. 6). The inflexibility, the partial flexibility, and the total flexibility in the bandwidth assignment of architectures I, II, and III, as discussed in section 3.2 and reflected in Figs. 8-10, explains this.

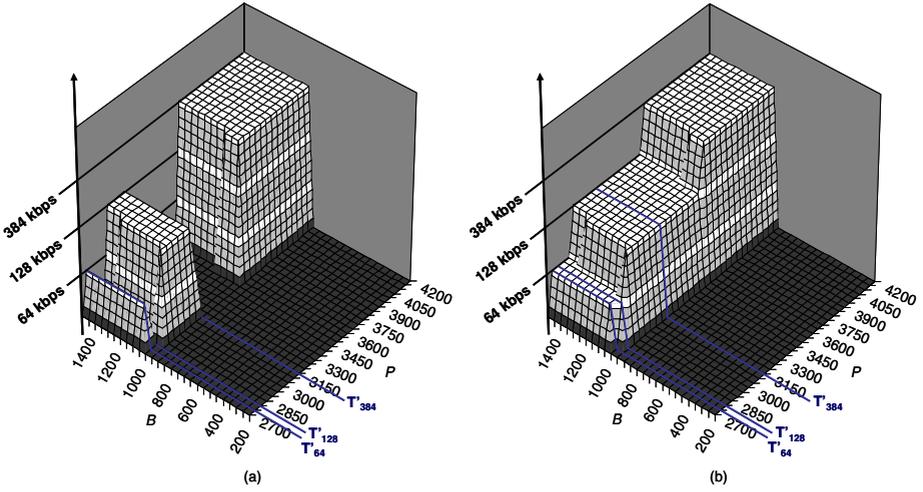


Fig. 8. Accumulated mapping information due to the learning process of Alg_1 (a) and Alg_2 (b) for SDR platform type I

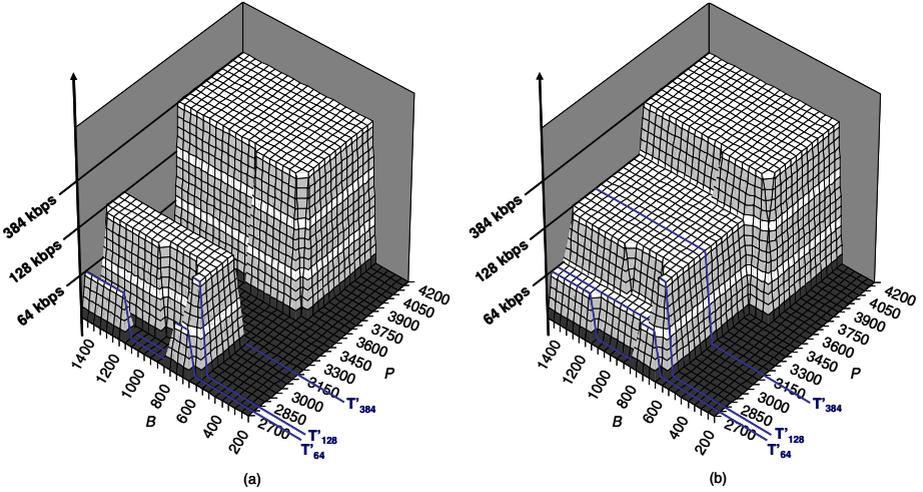


Fig. 9. Accumulated mapping information due to the learning process of Alg_1 (a) and Alg_2 (b) for SDR platform type II

Alg_0 either feasibly reconfigures an SDR-GPRS mobile terminal to the 384 kbps UMTS realization or loses the current user session. Alg_1 and Alg_2 lose much fewer sessions than Alg_0 because they either do not reconfigure those SDR-GPRS terminals that momentarily cannot operate in the 384 kbps UMTS mode or reconfigure them to a lower bit-rate UMTS realization. Alg_2 loses more sessions than Alg_1

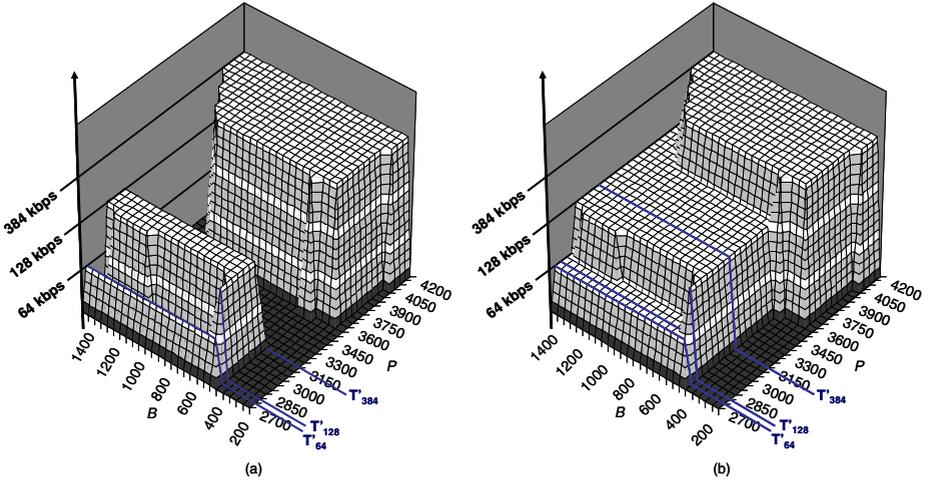


Fig. 10. Accumulated mapping information due to the learning process of Alg_1 (a) and Alg_2 (b) for SDR platform type III

(Fig. 5) but also feasibly reconfigures more terminals (Fig. 6). The shapes of the corresponding curves in Figs. 5 and 6 indicate that Alg_1 behaves better than Alg_2 for a low number of reconfiguration intents, whereas the higher their number the more efficient becomes Alg_2. The algorithm selection finally depends whether the principal objective is to lose less sessions or reconfigure more terminals; it is a trade-off between losing less sessions and reconfiguring more terminals.

Fig. 7 shows that many more of the 5000 SDR mobile terminals of each type remain in operation when applying Alg_1 or Alg_2 instead of Alg_0. In other words, our cognitive computing resource management proposal can satisfy more users in such a heterogeneous wireless scenario.

5 Conclusions

This paper has introduced the concept of computing resource management in cognitive radio. We have coined it *cognitive computing resource management* because it intelligently reconfigures SDR platforms as a function of the current state of the computing environment. A simple case study has demonstrated the importance and the potentials of our proposal in a heterogeneous radio environment.

Similarly to the insight that the problem of radio resource scarcity is actually a problem of an appropriate radio resource management, we argue for the importance of computing resource management in radio communications. In particular, we believe that a cognitive computing resource management may leverage the introduction of SDR platforms and applications, introducing flexibility in accessing a radio service for the sake of a ubiquitous wireless access.

References

1. Mitola, J., Maguire, G.Q.: Cognitive Radio: Making Software Radios More Personal. *IEEE Pers. Commun.* 6(4), 13–18 (1999)
2. Mitola, J.: Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio. Ph.D. dissertation, Royal Institute of Technology, Stockholm, Sweden (2000)
3. Mitola, J.: The Software Radio Architecture. *IEEE Commun. Mag.* 33(5), 26–38 (1995)
4. Tuttlebee, W.H.W.: Software Defined Radio: Facets of a Developing Technology. *IEEE Personal Commun.* 6(2), 38–44 (1999)
5. Buracchini, E.: The software radio concept. *IEEE Commun. Mag.* 38(9), 138–143 (2000)
6. Cummings, M., Haruyama, S.: FPGA in the Software Radio. *IEEE Commun. Mag.* 37(2), 108–112 (1999)
7. Hoffmeyer, J., Park, I.-P., Majamundar, M., Blust, S.: Radio Software Download for Commercial Wireless Reconfigurable Devices. *IEEE Commun. Mag.* 42(3), 26–32 (2004)
8. Marojevic, V., Revés, X., Gelonch, A.: Computing Resource Management for SDR Platforms. In: *Proc. 16th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Berlin, September 11–14, 2005, pp. 685–689 (2005)
9. Haykin, S.: Cognitive Radio: Brain-empowered Wireless Communications. *IEEE JSAC* 23(2), 201–220 (2005)
10. Tanner, R., Woodard, J.: *WCDMA – Requirements and Practical Design*. John Wiley and Sons Inc., Chichester (2004)
11. Faber, T., Schönle, M.: DSP-platform target report. SLATS Consortium, Project no. 27016, Deliverable D23 (1999)
12. Technical Specification Group Radio Access Network (3GPP): TS 25.213 V5.5.0 – Spreading and Modulation (FDD) (2003), <http://www.3gpp.org>
13. Technical Specification Group Radio Access Network (3GPP): TS 25.212 V6.4.0 – Multiplexing and Channel Coding (FDD) (2005), <http://www.3gpp.org>