

Experiences from the Iris testbed in dynamic spectrum access and cognitive radio experimentation

Linda E. Doyle, Paul D. Sutton, Keith E. Nolan, Jörg Lotze, Barış Özgül,
Thomas W. Rondeau*, Suhaib A. Fahmy**, Hicham Lahlou, Luiz A. DaSilva
CTVR The Telecommunications Research Group, University of Dublin, Trinity College, Ireland

*Center for Communications Research in Princeton, N.J. USA

** School of Computer Engineering Nanyang Technological University, Singapore

Abstract—The focus of this paper is an experimentation platform known as Iris that has a runtime reconfigurable software radio at its core. We have employed this platform to enable a wide variety of tests and experimentation in the fields of dynamic spectrum access and cognitive radio. The paper charts the progress of the Iris system since its inception as well as details of some of the experiments and trials conducted using this platform. We also discuss the challenges involved in the development and deployment of a software radio platform that supports runtime reconfigurability and provide recommendations for future improvements.

I. INTRODUCTION

To significantly advance research in the field of dynamic spectrum access and cognitive networking, it is essential that ideas be translated from paper to working prototype. This is a challenging and more often than not slow process. Some of the bold ideas of new spectrum usage regimes are met with skepticism by some regulators, broadcasters, and wireless industry, making the prototyping process all the more important. While the field has grown and matured, it is not yet obvious that we have found a real home for cognitive radio beyond the much discussed TV white spaces and there are many who still doubt the potential of cognitive radio and its role in the future.

The purpose of this paper is to discuss our ongoing effort focusing on cognitive radio experimentation and reconfigurability. This effort centres around the use of the Iris runtime reconfigurable software radio platform, a platform designed in CTVR in Trinity College Dublin.

Iris is introduced in Section II. The basic system has grown and advanced over the years and these advances are described in Section III. An Iris radio consists of a chain of components and some typical Iris components for dynamic spectrum access experimentation are presented in Section IV. Section V is a key section and summarises the major demonstrations which were based on Iris and highlights three examples in particular. Section VI gives some brief details of an innovative test and trial scheme run by the Irish regulator Comreg, that aids experimentation. Section VII shows how Iris has commercial potential and section VIII introduces two European research projects that feature Iris. Section IX discusses both the specific issues relating to experimentation with Iris as well as the

broader challenges of experimentation in general. Section X concludes.

II. EARLY HISTORY

Iris was created as a runtime reconfigurable software radio for general-purpose processors (GPP) using a Windows operating environment [1]. It was inspired by the SpectrumWare work at MIT [2]¹. Iris is a component-based system. Discrete signal processing functions such as a digital filter or modulator are implemented as components with generic interfaces for lifecycle control, data passing and reconfiguration. These components are linked together to build radio transmit and receive chains. An XML file is used to describe the structure of the components. Both the Iris runtime system and the Iris components are implemented completely in portable C++.

In contrast with software architectures such as GNU Radio and the Software Communication Architecture (SCA), Iris was designed specifically to support maximum reconfigurability while the radio is running. Iris nodes can therefore carry out reconfigurations seamlessly in response to observed changes in the operating environment. Reconfigurability is realised through a number of different mechanisms. When implementing an Iris component, the radio designer can choose to expose a number of parameters. While the radio is running, these parameters can be dynamically reconfigured to adjust the operation of the component. Alternatively, individual components can be changed at runtime, as can whole configurations of components.

The emphasis from the beginning was on ‘ease of design for developers, the ability to rapidly prototype and tailor a transceiver implementation, and a GUI that provided a suite of software radio components and control entities for use in a design.

It had always been intended that Iris connect to a RF frontend that had minimum functionality, i.e. that the bulk of the processing take place in Iris. However, initially identifying an appropriate RF frontend proved extremely difficult. The WaveRunner Plus 253 PMC transceiver model from Red Rapids, a 70MHz IF radio module, was used and much of its unnecessary functionality (from our perspective) was bypassed.

¹The SpectrumWare work was the basis for the formation of the company Vanu.

The emergence of the Universal Software Radio Peripheral (USRP) from Ettus Research LLC (<http://www.ettus.com>) was a welcome development. This RF frontend had a large impact on the Iris platform. The on-board functionality of the USRP is minimal, limited to interpolation, decimation, up/down conversion and power amplification. The USRP therefore provides developers with the freedom to transmit and receive bespoke (custom-made) signal waveforms. In the USRP 1.0 case, a USB 2.0 interface is used. The second generation of the USRP uses GigE technology and can handle up to 25MHz signal bandwidth.

The USRP provided the ideal frontend for Iris. It contained the necessary level of functionality. It enabled a degree of frequency agility that had not been available in previous platforms and it was affordable. Hence the Iris experimental platform really began its life as a Windows-based software radio plus USRP.

In the early days it was challenging to get a realtime FM radio operating on Iris (i.e., on the general purpose processor) but advances in processing capabilities have meant that sophisticated radio configurations are ever more possible. A wide range of Iris components have been built to take advantage of this. Iris today is particularly focused on multicarrier communication systems and has many OFDM and MC-CDMA components. Iris can contain Matlab components as a special wrapper and has been designed to cater for this. In this case, realtime operation is not possible but the ability to include Matlab components is a useful feature for testing ideas before committing to C++ implementations. While the USRP remains the main RF frontend used with Iris, it is possible to use any RF frontend by simply creating Iris components that write to and read from the hardware in question. For example Iris has been used with the Maynooth Adaptable Radio System from the National University of Ireland, Maynooth [3].

III. IRIS ADVANCEMENTS

Though Iris started out life as a Windows-based general purpose processor entity, it was extended to use different operating systems as well as run on different platforms. The experiences of using Iris in its basic form and on other platforms subsequently gave rise to a rewrite of the system and the creation of a new version of Iris. The following sections detail these advances.

A. Platform and Operating System Advances

In 2007 a joint project with Xilinx Research Labs in Ireland provided the motivation to create an FPGA-based cognitive radio platform and hence Iris was reconstituted for an FPGA platform [4], [5].

In very basic terms the FPGA-based system can be described as follows. The Iris runtime engine sits on the PowerPC. Iris components can be implemented BOTH in *hardware* and *software*. Hence it is possible, for example, to have components which involve significant processing (e.g. turbo coding) on the FPGA logic and other components in pure software. The runtime system manages access to the

components at an abstracted level and therefore can combine software and hardware components together with ease. Here again the radio configuration is described in an XML file.

As mentioned already, a designer can choose to implement an Iris component on the FPGA logic. This of course requires specialised hardware knowledge. However once a hardware component exists, other designers can seamlessly incorporate this component into their radio design. Compile-time tools partition and prepare the necessary FPGA configurations for use by the runtime system.

In parallel with the FPGA-based work Iris was also ported to the CellBE. The CellBE approach is similar in nature to the FPGA approach in that the Iris runtime engine sits on a PowerPC while the different Iris components are implemented on the specialised signal processing elements of the CellBE. Both the use of the FPGA and the CellBE reflected the growing need to include systems that were capable of parallel processing.

As a result of working on the PowerPC the basic general purpose processor version of Iris was ported to Linux. All Iris code is written in portable C++ and can be compiled for the Windows, Linux, and Mac OS X operating systems. Supported and tested CPU architectures include x86 (32- and 64-bit), PowerPC, and ARM, and other architectures are likely to work as well.

B. Design Advances

The experience of developing a number of demonstration systems and the use of a range of different platforms led to many insights regarding the design of the Iris system. In particular, we identified three key challenges:

- Dynamic memory allocation;
- Parallelism support; and
- Modular design for heterogeneous platforms.

1) *Dynamic memory allocation*: While the intention of Iris was always to maximise the runtime reconfiguration possibilities, one drawback of the original system design was the use of static memory allocation to support data passing between components. At load time, the quantity of data produced by each component in a single execution cycle is calculated. On the basis of this calculation, memory buffers are allocated and remain in place for the lifecycle of the radio. This approach provides a high level of efficiency. However, in highly reconfigurable radio designs, the amount of data produced by a component during a given execution cycle can vary according to the configuration of that component. One example of such a component would be a reconfigurable Quadrature Amplitude Modulation (QAM) symbol mapper. When configured to perform 16-QAM symbol mapping, such a component produces a single symbol for every 4 bits of input data. However, upon reconfiguration to perform 64-QAM symbol mapping, the same component produces a symbol for every 6 bits of input data. Designing such a component with the original Iris system required mechanisms to trigger a reallocation of memory throughout the radio each time a reconfiguration occurred.

2) *Parallelism support*: A second motivation to advance the design of the Iris system was the emergence of multi-core GPPs. As processor manufacturers attempt to improve GPP performance by increasing the operating frequency of individual cores, gains are diminished by a number of factors. These include an increasing gap between the speeds of the processor and memory, exponentially increasing power consumption, and increased difficulty in finding enough parallelism in a single instruction stream to fully utilize a high performance processor core. Thus, rather than increasing operating frequencies, manufacturers have instead turned to processing systems comprising multiple cores.

In order to leverage such multi-core processors, a software radio system requires powerful mechanisms to capture parallelism within a executing radio. The original Iris system provided support for parallel execution of two or more signal chains. However, capturing parallelism inherent in a single chain was not possible.

3) *Modular design for heterogeneous platforms*: A third challenge associated with the original Iris system design concerned the use of heterogeneous platforms. The approach taken to support non-GPP processing platforms such as FPGAs and CellBE SPEs was to use a software wrapper component [5]. This wrapper encapsulated all code executing upon the platform and presented a generic interface to the Iris system. While this approach was successful, it led to the use of monolithic code design for these platforms, limiting reusability. As our use of these platforms increased, the need for a modular design approach became more pressing.

4) *Addressing the issues*: To address these challenges, a major redesign of the Iris system was undertaken in 2008. A central feature of this redesign was the introduction of the *Iris Engine*. An Iris Engine encapsulates one or more components of the overall dataflow graph of a radio and defines the data-passing, execution and reconfiguration semantics for those components. Engines present a generic interface to the overall Iris system and a radio may comprise one or more of these engines.

The ability to create different engines with different properties provides a powerful solution to each of the challenges outlined above. To address the challenge of dynamic memory allocation, a flexible PHY engine (fPHY) was designed. The fPHY engine supports transparent reallocation of memory buffers between components, enabling a high level of reconfigurability. While a number of components in a radio design may require this level of reconfigurability, many may not. For example, the overhead associated with such reconfigurability is not required for a digital filter component with fixed taps. To reduce this overhead and maximise performance for more static components, we designed a scheduled PHY engine (sPHY). The sPHY engine makes use of a static schedule, calculated at load-time, to minimise overhead and increase data throughput. Thus, by using a mixture of fPHY and sPHY engines, a radio designer can choose an appropriate tradeoff between reconfigurability and performance.

Iris Engines also provide a powerful mechanism for captur-

ing parallelism within a software radio design. A single engine encapsulates a single thread of execution and is responsible for the execution of one or more components. Thus, a radio design comprising three components may execute using one, two or three engines. In this way, between one and three parallel threads of execution can be employed.

The third challenge of modular design for heterogeneous platforms can also be overcome through the use of Iris Engines. An engine completely defines the data-passing, execution and reconfiguration semantics for one or more components. Thus, an engine is well-suited to encapsulate a system of components executing upon a particular processing platform. For example, an FPGA engine could support the use of FPGA components, each comprising a section of VHDL code. At load-time, the engine could combine the specified components into a single FPGA configuration and load it into the FPGA fabric. At run-time, the engine could manage all communication with the FPGA hardware and could also reconfigure it when required. In this way, the engine enables the use of modular code design for an FPGA platform.

IV. IRIS BUILDING BLOCKS FOR DYNAMIC SPECTRUM ACCESS

The paper thus far has focused on the general details of Iris, the platforms on which it operates and advances that have been made over the years. In this section we look in more detail at some of the Iris components and concentrate in particular on components that are relevant to dynamic spectrum access experimentation.

There are a variety of challenges in dynamic spectrum access system design. Three key challenges are (1) the reliable identification of unoccupied spectrum, (2) the creation and transmission of an appropriate waveform that makes best use of the unoccupied spectrum while not interfering with primary users and (3) the enabling of rendezvous between transmitter and receiver whenever the spectrum band in use changes. Iris components have been designed for these purposes and a significant amount of Iris experimentation has been based around these three challenges.

A. Sensing

There are many different ways to sense the existence of a primary user and Iris contains a range of components for this purpose. Like most systems, it supports simple energy detectors. Additionally, Iris contains components based on cyclostationary feature detection, as well as components that use filterbank approaches. The sensing components have been designed to target different application areas. For example, a component exists for the detection of DVB-T and DVB-T2 signals for TV white space applications. Another component exists for detection of LTE signals for next generation femto-cell systems [6].

B. Shaping

OFDM has tended to be the preferred technology for dynamic spectrum access and cognitive radio systems. The ability to manipulate the signal bandwidth and throughput through

changing numbers of subcarriers and modulation techniques, for example, provides a flexible means for fitting a signal into unoccupied spectrum. However, OFDM waveforms tend to have high sidelobes, which can cause unacceptable levels of interference to systems on adjacent channels. Therefore there is broad interest in signal processing techniques which reduce the strength of the sidelobes. Iris provides a range components for shaping OFDM signals. These components use different windowing techniques.

C. Rendezvous

Frequency rendezvous is the process by which two radios meet on a given frequency and begin the communication process. In a dynamic spectrum access environment, spectrum is opportunistically available and hence a cognitive radio must somehow inform the cognitive radio with which it wants to communicate the frequency on which it will operate. One such mechanism for doing this is to use some kind of control channel. A control channel can be problematic in a dynamic spectrum access scenario, as it is not an easy task to ‘set aside’ spectrum for this function. Iris has components for managing frequency rendezvous without the use of a control channel.

These components make use of what is known as *cyclostationary signatures*. Full details of these are given in [7] and [8]. In short these signatures are intentionally embedded features in an OFDM signal that give rise to strong and easily detectable cyclostationary features (as opposed to naturally occurring features that arise as a byproduct of normal processing of the signal, such as modulation). The trick is that the embedded signature also provides information about the frequency of operation. Hence a cognitive radio can follow the varying transmissions of another cognitive radio simply by scanning for the embedded signatures.

V. EXPERIMENTATION

The components described in the previous section have allowed us to carry out a variety of experiments relating to dynamic spectrum access. Table V summarises the experiments that have been presented to wider audiences at international conferences such as DySPAN, which have afforded great opportunities for disseminating experimental work conducted using Iris.

Next, we describe three example experiments conducted using the Iris platform.

A. Example 1: Sensing for Femtocells

Figure 1 gives an overview of the setup for an experiment and demo based on the sensing of LTE femtocells. The scenario under test is one in which a LTE femtocell is operating and two others subsequently come on the scene and avoid the bands that are already used by the existing femtocell. The femtocell network underlays an existing macrocell network. For example, it is likely that LTE femtocells might coexist with a GSM macrocell. The sensing technique in use is described in [6] and it is possible to detect weak signals with this technique, to guarantee interference-free operation.

Conference	Demonstration System
DySPAN 2007	Cyclostationary signatures for detection, identification and carrier frequency estimation of OFDM-based waveforms.
DySPAN 2007	Coexistence of non-contiguous OFDM-based waveforms with co-channel single-carrier waveforms.
DySPAN 2008	A reconfigurable FPGA-based system using a single carrier waveform, employing sensing for carrier-frequency estimation.
DySPAN 2008	A DSA network using a bandwidth-adaptive OFDM waveform with embedded cyclostationary signatures for blind bandwidth estimation and network coordination.
DySPAN 2008	A real-time cyclostationary analyzer using the Cell Broadband Engine.
FCCM 2009	A reconfigurable FPGA-based system using a single-carrier waveform with adaptive coding.
SIGCOMM 2009	A reconfigurable FPGA-based system employing sensing for carrier-frequency estimation and adaptive coding.
DySPAN 2010	A DSA network using reconfigurable pulse-shaped waveforms to control out-of-band (OOB) emissions.

TABLE I

DEMONSTRATION SYSTEMS BUILT USING THE IRIS ARCHITECTURE.

There is one key point worth noting here. Iris is running on more than one platform. It is running on the PC (i.e. a GPP) AND on an FPGA platform, namely the Xilinx ML507 development board [9], which hosts a Xilinx Virtex-5 XC5VFX70 FPGA. Though not mentioned before in this paper, Iris does not need to be confined to any one platform and a number of platforms can be strung together to create a ‘distributed’ implementation.

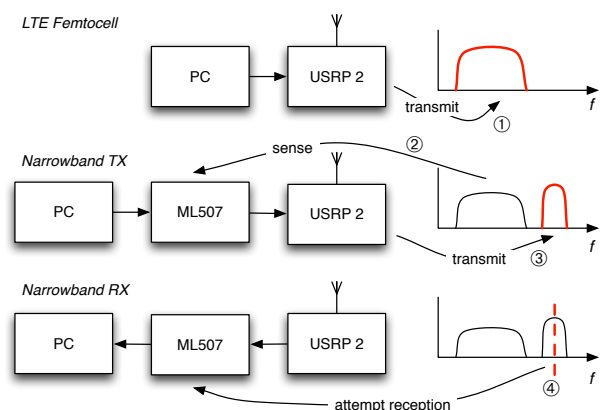


Fig. 1. LTE sensing for femtocell autoconfiguration

The demonstration consists of three radio nodes as illustrated in Figure 1. The first broadcasts the pre-recorded LTE-like signal and acts as the interfering femtocell (1). The second node is the transmitter, which searches the spectrum to find spare channels (2), then begins to transmit streaming video

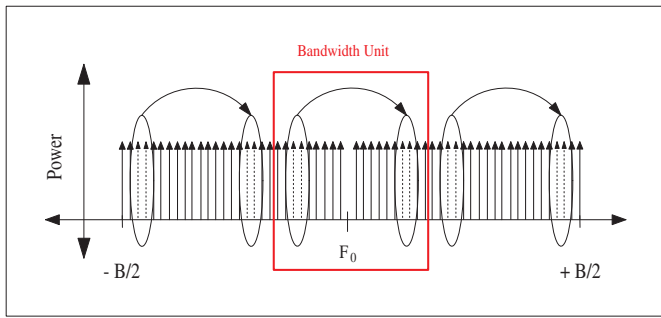


Fig. 2. A frequency and bandwidth adaptive system based on cyclostationary signatures

using coded DQPSK modulation (3). The final node is the receiver, which attempts to receive on each of the available channels, locks onto the transmission, and decodes the video (4). The playback of the video happens on the PC while the sensing and decoding of the signals occurs in hardware on the FPGA platform.

B. Example 2: bandwidth-adaptive OFDM

A second example is one that uses rendezvous components but does so to not alone achieve frequency rendezvous but also to signal the amount of bandwidth in use. The scenario under test here is one in which a cognitive basestation gets access to different frequency bands of different bandwidths over time. Rather than signalling this to the cognitive radios in its range through a control channel, the basestation employs cyclostationary signatures for this purpose. The signatures are embedded by correlating data on subcarriers. A bandwidth unit has been defined and the signatures are repeated for each bandwidth unit. The receiver determines the frequency of operation using the signatures and determines the bandwidth through counting the number of signatures. Figure 2 shows how the signatures are created through intentionally correlating data in selected subcarriers. The particular subcarriers chosen reflect the frequency of operation.

An indepth description of the experiment is given in [10]. The work was demonstrated at DySPAN 2008. The great strength of this demonstration lay in its ability to substantiate a theoretical idea that had up to that point merely been investigated in Matlab. The interference from other systems in the room in which the demonstration took place did not affect the system and further illustrated its robustness.

C. Example 3: Shaping for Coexistence

The final example in this paper focuses on a demonstration that is taking place at DySPAN 2010 and hence provides an opportunity for interested parties to see Iris in action. In this demonstration we present a dynamic spectrum access network which employs a reconfigurable orthogonal frequency-division multiplexing (OFDM) based waveform. In order to avoid the creation of harmful interference, the out-of-band (OOB) emissions of the waveform are dynamically tailored to the

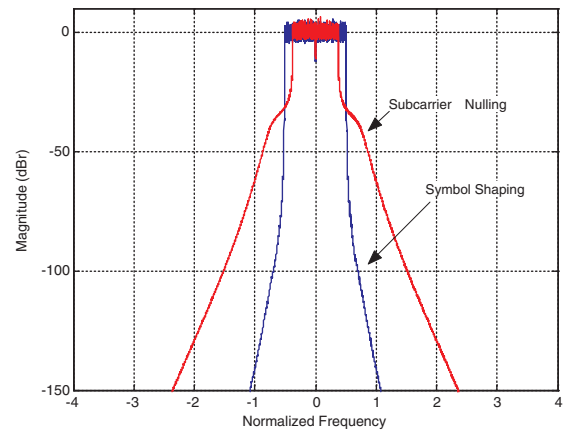


Fig. 3. The effect of shaping on a signal

properties of spectrum neighbours through the use of OFDM pulse shaping.

In this experiment two neighbouring systems will exist peacefully when shaping is turned on but have problems when it is not. Figure 3 illustrates the effect of shaping on the OFDM signal. The demonstration is therefore all about coexistence.

VI. COMREG TEST & TRIAL LICENSES - AN ENABLER

The experimentation described in the previous section has taken place in many different venues. When at home in Ireland we have the advantage of making use of various different frequency bands for experimentation, thanks to a Test & Trial scheme, driven by the national regulator authority. The Irish spectrum regulator, Comreg, enables cognitive radio and dynamic spectrum access activities to take place under a wireless test and trial licensing scheme. The regulators promise that licence applications will be turned around as soon as possible, typically within 10 days after receipt of a completed application. And more importantly they state that all available radio frequency bands will be considered for a test or trial licence, including the UHF band; radio spectrum's 'sweetpot'.

We have availed ourselves of this favourable spectrum licensing regime in Ireland to conduct tests and trials of Iris-based cognitive radio and dynamic spectrum access systems outside of the lab and across Ireland.

In 2007, the first multi-party collaborative trial of DSA and cognitive radio technologies was carried out during the IEEE DySPAN symposium in Dublin, using a wireless trial licence. This event enabled a mix of industrial and academic groups to attempt to communicate, co-exist, and interfere with one another to gain insight into the practical real-world issues following years of research largely limited to theoretical and simulation-based activities. The concept was proven; multiple heterogeneous DSA networks could co-exist in the same spectrum segments. However, the event also demonstrated that the state of the art was limited to infrastructure development e.g. RF hardware, processing platforms, sensing and decision-making, and not yet at the communications applications level.

VII. TOWARDS COMMERCIAL OPPORTUNITIES

In addition to being an enabler for research, Iris is being used under licence for real-world proof of concept prototypes and to develop commercial applications. In 2008, the European Microsoft Innovation Centre in Germany used Iris to rapidly develop new prototype wireless communications systems. In 2009, a trial was conducted in Limerick, Ireland by a pre-spin out group called reconfigr (reconfigr.com) using Iris in a UHF-band whitespaces scenario. This trial demonstrated how secondary opportunistic users could establish a network and co-exist with a digital terrestrial television (DTT) incumbent. The interference potential to subscriber receivers was minimised using spectrum sensing. This work helped validate claims that secondary opportunistic usage of whitespaces in real-world scenarios is feasible, even with strong adjacent-channel incumbents, and that spectrum sensing combined with frequency agility is a solution.

VIII. IRIS AND FP7 FRAMEWORK RESEARCH

Iris is also playing a role in two EU-funded activities. The first has just started and the second is expected to begin in September. As each will bring a new dimension to test and experimentation in cognitive radios, they are both worth mentioning briefly.

A. COGEU

The first project is entitled ‘Cognitive Radio Systems for Efficient Sharing of TV White Spaces in the European Context’ (COGEU) and is funded under the European Union’s Framework Programme 7 (FP7). The project began in January 2010, and it looks into the variety of regulatory opinions on TV white space usage across the European regulatory landscape. It also aims to create working prototypes of systems for these bands and, in doing, so explore different service/application scenarios. COGEU will not just consider unlicensed modes of operation but also look at schemes based on exclusive usage rights.

The consortium consists of the Instituto de Telecomunicações (Portugal), CTVR, Trinity College Dublin (Ireland), Thales Communications (France), Rohde & Schwarz (Germany), Portugal Telecom Inovacao (Portugal), a wireless provider, Sigint Solutions (Cyprus), Poznan University of Technology (Poland), University of the Aegean (Greece), and the Institut fuer Rundfunktechnik (Germany), a research centre that focuses on public broadcasting.

The Iris platform will be featured in a number of the working prototypes created in this project. We will conduct experiments in IMS bands as well as in licensed bands, using Ireland’s Wireless Test & Trial Licensing Programme. One of the members of the COGEU consortium, Rohde & Schwarz (R&S), manufactures device, infrastructure, protocol, and compliance testers for wireless systems. They will set up a coexistence testbed to evaluate the coexistence between DVB-T and cognitive radio devices in several different scenarios, such as the extension of WiFi and WiMax using TV white

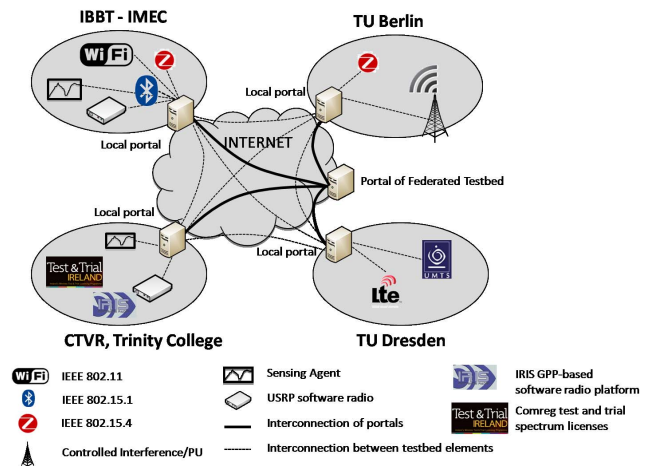


Fig. 4. The CREW federated testbed

spaces. A system supported by Iris and USRP hardware will provide some of the nodes for this testbed.

B. CREW

The second project is called Cognitive Radio Experimental World (CREW) and is again a FP7 funded project. The aim of CREW is to create a federated cognitive radio testbed in Europe that enables meaningful cognitive radio research. The consortium consists of IBBT (Belgium), IMEC (Belgium), CTVR, Trinity College of Dublin (Ireland), Technical University of Berlin (Germany), Technical University of Dresden (Germany), EADS (Germany), and Thales Communications (France).

Figure 4 is a schematic of the testbeds involved in CREW; the Iris based testbed is in the bottom left hand corner. The concept of federation can be a challenging concept in wireless testbeds. The notion of federation tends to be more readily understood in the context of optical networking, for example. To address this challenge we have defined three possible modes of interaction with the testbeds.

Mode 1 involves the use of any one single CREW testbed. The federation activities that will support this mode include the creation of a CREW portal with clear and uniform information about access and usage of each testbed and suggested usage scenarios. In Mode 2 the user will have access to any single testbed again but the testbed will contain nodes made from elements of other testbeds and hence be highly heterogeneous in composition. Hence the federation activities will consist of hosting of nodes from one testbed in another and the creation of new nodes from combinations of CREW hardware/software components. Mode 3 is the most important. This will involve multiple sequential CREW testbed usage. Here the federation activities will involve the enabling of the capturing of data/behaviours from one CREW testbed and replay for emulation or post-processing purposes on a second CREW testbed through the definition of common data collection and storage mechanisms, data access mechanisms and through

the provision of subcomponent or subsystem access in each testbed.

This project will offer the potential for experimentation on a more extensive basis. The CREW project will organize two open calls to extend the use of the federated testbed to external researchers or research groups. The first call is envisaged for the second half of 2011, one year after the start of the project.

IX. LESSONS LEARNED & INSIGHTS GAINED

The extensive use of the Iris platform has contributed to building a great deal of experience in dynamic spectrum access experimentation. There are many lessons to be learned. Some relate to the Iris platform explicitly, while others are more general and have to do with experimentation in networks of cognitive radios.

A. Experimentation with Iris

The Iris platform has brought with it a number of distinctive advantages, including the following:

- The ability to take an idea from concept to realisation on Iris has contributed to the impact of the work that we have undertaken in the field of dynamic spectrum access. In particular with new ideas, such as the use of cyclostationary signatures, it has proven an invaluable means of proving that the concept works.
- Experimentation, particularly on both a GPP and an FPGA platform has led to improved understanding of the challenges of reconfiguration and the extent to which, dynamic reconfiguration is required in real-world applications.
- The more recent developments in the Iris platform, and especially of Iris 2.0, have led to a much more usable and accessible system, and a system that is highly targeted to runtime reconfiguration. The engine concept has been a significant enabler of the kind of dynamic reconfigurations in which we are interested, as well as being a means of more easily running Iris on different platforms.
- The use of the Iris platform has led to strong insight into the impact of real hardware on the different signal processing algorithms we design. For example the efforts to reduce the sidelobes of an OFDM signal with sophisticated shaping techniques, can in certain circumstances be undone by the ability of the USRP to deal with them.

There are of course many outstanding challenges for the Iris platform including the following:

- Due to the nature of the funding for the original development of Iris, the platform is not open-source. This has been an obstacle to its use by some of our research collaborators. At this time, we are exploring contractual instruments to facilitate the dissemination of Iris beyond our lab.
- Though Iris 2.0 has been built to handle a full stack (e.g. there is special network engine), the experimentation to date has not had a strong network flavour. In [11] we enumerate some points that highlight the need for a full protocol stack. This aspect of Iris needs to be more

significantly exploited. (This has been a prevailing issue with cognitive radio experimentation in general.)

- The ‘decision engine’ in most of the experimental work has been simple and has not pushed the boundaries of what ‘might be’ in a decision or cognitive engine of a future communication system. However, all the necessary functionality exists to support a decision/cognitive engine in that there are multiple ways to reconfigure the system on the basis of outputs from a such an entity.
- While the USRP has been true enabler of the many experiments we have carried out, it also presents challenges. The limited power has a number of impacts. It affects range and hence networking possibilities. The dynamic range of the devices means that OFDM waves will be clipped beyond certain power levels. This has an impact for example on the shaping experiments mentioned earlier. The value of some of the shaping techniques is clearer at higher powers which may not be an option of the USRP given its dynamic range.

B. Experimentation in General

Carrying out experimental work also has its more general advantages, including:

- 1) Experimentation has proven to be a strong mechanism for attracting industry collaboration. One of the reasons for this is that tangible benefits of reconfigurable radio and dynamic spectrum access can be showcased in the experimentation scenarios.
- 2) The experimental work has opened up many opportunities for dialogue with our regulators on issues relating to dynamic spectrum access and cognitive radio. Dynamic spectrum access and cognitive radio depend heavily on the right policies and regulations being put in place. The importance of taking opportunities to influence the thinking in these fields and lead by example cannot be over-emphasised.
- 3) Experimentation offers the opportunity to feed real-world insights back into theory. The general insights gained through working with the reality of the wireless world can be used to define more appropriate assumptions for theoretical work, to more appropriately set parameters in simulations and to build models of the environment which may prove useful.

However, it is true to say that experimentation has many open and varied challenges:

- 1) From an academic perspective, it still remains difficult to get credit for experimental results, in particular in traditional archival publications.
- 2) Much of the experimental work, including some of our own, is hard for others to repeat. Repeatability is an important part of the scientific process. Methods for better describing an experimental setup and capturing the experimental conditions need to be established to aid in this process. (We also note that the same should be done for simulation-based work.)

- 3) Whether in the theoretical, simulated or experimental domain, there is a pressing need for some way to benchmark the progress of the work in dynamic spectrum access and cognitive radio. In other words, there needs to be a more scientific method for claiming that, ‘my system is better than yours’. We believe that it is true to say that many comparisons made are between apples and oranges. It should be possible to take the lead in benchmarking in the experimental world and then reflect these insights back into the theoretical and simulation domains.
- 4) Mechanisms need to be found for presenting negative results, In other words, we should be able to report both what works and what does not work. The field can only truly advance if we find a systematic and scientific way for doing this.
- 5) Current cognitive radio platforms are still limited in their support of network experimentation. Among the missing pieces, we cite the lack of support for distributed sensing and classification, the practical difficulties of implementing medium access control protocols tailored to dynamic spectrum access in current experimental software defined radio platforms, the lack of a network knowledge representation language, and the lack of a cognitive specification language.
- 6) Though some databases exist, there continues to be a need for collecting measurements from experimental setups and making these measurements widely available. A common spectrum database format and repository pool is required to combine spectrum information from multiple experimental and sensor sources around the world.

X. CONCLUSIONS

The Iris platform has enabled a wide range of experimentation that is pertinent in the dynamic spectrum access and cognitive radio fields. Its key characteristic is its ability to carry out runtime reconfiguration at a parametric, component and structural level.

Moreover, Iris can function on a number of different platforms, or indeed be distributed over a number of platforms. The benefits of experimentation are many though the challenges are many more. The experimental space is ripe to be taken to the next level and put on a stronger scientific footing.

This material is based upon work supported by Science Foundation Ireland under Grant No. 03/CE3/I405 as part of CTVR, The Telecommunications Research Centre at University of Dublin, Trinity College, Ireland.

REFERENCES

- [1] P. Mackenzie, “Software and reconfigurability for software radio systems,” Ph.D dissertation, University of Dublin, Trinity College, Ireland, 2004.
- [2] V. Bose, “Design and Implementation of Software Radios Using a General Purpose Processor,” Ph.D. dissertation, Massachusetts Institute of Technology, USA, Jun. 1999.
- [3] R. Farrell, M. Sanchez, and G. Corley, “Software-defined radio demonstrators: An example and future trends,” *Hindawi International Journal of Digital Multimedia Broadcasting*, Jan. 2009.
- [4] J. Lotze, S. A. Fahmy, L. E. Doyle, and J. Noguera, “An FPGA-based Autonomous Adaptive Radio,” in *ACM SIGCOMM Conference*, Barcelona, Spain, Aug. 2009, demonstration paper.
- [5] S. A. Fahmy, J. Lotze, J. Noguera, L. E. Doyle, and R. Esser, “Generic software framework for adaptive applications on FPGAs,” in *IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM)*, Napa, CA, Apr. 2009.
- [6] J. Lotze, S. A. Fahmy, J. Noguera, B. Özgül and L. E. Doyle, “Spectrum sensing on LTE femtocells for GSM spectrum re-farming using Xilinx FPGAs,” in *Software-Defined Radio Forum Technical Conference (SDR Forum)*, Washington, DC, Dec. 2009.
- [7] P. D. Sutton, K. E. Nolan and L. E. Doyle, “Cyclostationary signatures in practical cognitive radio applications,” in *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 13–24, Jan. 2008.
- [8] P. D. Sutton, K. E. Nolan, L. E. Doyle, “Cyclostationary signatures for rendezvous in OFDM-based dynamic spectrum access networks,” in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pp. 220–231, Dublin, Ireland, Apr. 2007.
- [9] *ML505/ML506/ML507 Evaluation Platform User Guide*, Xilinx Inc., Nov. 2008. [Online]. Available: http://www.xilinx.com/support/documentation/boards_and_kits/ug347.pdf
- [10] P. D. Sutton, B. Özgül, K. E. Nolan, and L. E. Doyle, “Bandwidth-adaptive waveforms for dynamic spectrum access networks,” in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Chicago, USA, Oct. 2008.
- [11] L. A. DaSilva, A. B. MacKenzie, C. R. C. M. da Silva, R. W. Thomas, “Requirements of an open platform for cognitive network experiments,” in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Chicago, USA, Oct. 2008.