Design of a software defined radio-based tactical DSA network

Susan Watson
DRDC – Ottawa Research Centre

Jean-Benoit Larouche
Nutaq

IEEE; Military Communications Conference (MILCOM) 2017

Baltimore, MD, USA

Date of Publication from Ext Publisher: October 2017

Defence Research and Development Canada

External Literature (P)
DRDC-RDDC-2018-P011
February 2018

Design of a Software Defined Radio-Based Tactical DSA Network

Susan Watson
Defence R&D Canada – Ottawa Research Centre
3701 Carling Avenue, Ottawa, ON K1A0Z4
Email: susan.watson@drdc-rddc.gc.ca

Jean-Benoit Larouche
Nutaq
2150 Rue Cyrille-Duquet #100
Ville de Québec, QC G1N2G3
Email: djibylarouche@hotmail.com

***Abstract*—This paper presents the design for a prototype tactical dynamic spectrum access (DSA) mobile ad hoc network, where the network is organized into clusters operating on a single frequency. The frequency may be changed autonomously by the network in response to jamming or interference, after some frequency switching delay. The network node design is implementable on a software defined radio (SDR), such as the Nutaq PicoSDR. The contribution of this work is the prototype design and mapping of the design to the hardware and software architecture of the SDR, with a focus on the estimated delay in switching to a new frequency channel. The frequency switching delay is the result of systematic delays arising from the SDR platform, compounded by delays in the protocols enabling DSA, the gathering of sensing data from network nodes, and the dissemination of the frequency change command throughout the network.**

I. Introduction

In military wireless networks, spectrum is a scarce commodity, and there is a need to automate the management of spectrum to facilitate re-use. Dynamic spectrum access (DSA) technology addresses this problem.

Prototyping of a DSA radio network gives insight into the challenges of achieving DSA functionality, in a way that is grounded in the implications of using real hardware [1]. Radio network prototypes may be realized with commercial software defined radios (SDRs) such as the Nutaq PicoSDR [2] or the National Instruments univeral software radio peripheral (USRP) [3]. Over the years, prototype DSA and cognitive radio (CR) networks based on SDR platforms have been built and provided for use by the research community [4][5]. However, these implementations lack characteristics of interest for DSA in a military context. For example, in a tactical network, frequency allocation decisions are likely to be geared towards maximizing co-existence with allied nations (co-primary users) and avoiding extensive pre-mission spectrum planning [6], rather than towards avoiding primary users of the spectrum as in civilian DSA. The requirement for a civilian DSA network to avoid primary users impacts its design significantly. In a civilian DSA network, interference with a primary user may be avoided by evacuating the channel when the energy level in the channel reaches a threshold; for tactical DSA networks in a contested environment, interference may be tolerated until it threatens to impair the network's ability to support mission communication requirements.

In addition to easing the burden of pre-mission spectrum planning, tactical DSA networks may be able to recover from a jamming attack by switching frequency channels, subject to some frequency switching delay. The frequency switching delay will depend on the hardware and software implementation, as well as on the particular sensing, DSA and control protocols.

The goal of this paper is to present the design of an SDR-based DSA-capable network testbed, and to provide a discussion and estimate of the delay in switching frequencies for an implementation of this design. Two factors compound each other to limit the frequency switching delay of the SDR-based network: latency arising from the SDR platform architecture, and latency in the MAC protocol.

Latency in the SDR platform arises from its architecture; the traditional SDR platform architecture, as exemplified by the USRP and PicoSDR, consists of a host processor connected by a bus interface to radio hardware. SDR frameworks (e.g. GNURadio [7]) allow signal processing to be performed by host processor and to be shared with the radio hardware. It has long been recognized that the SDR architecture and frameworks introduce delays due to bus transfers, host processing, and radio re-configuration [4]. Latest-generation hardware and faster buses, i.e. PCIe, have alleviated some of the platform latency. Researchers have also developed "split-MAC" approaches for dealing with the problem [8]. In these approaches, MAC functionality is split into modules, and those modules containing time-critical functionality are placed closer to the radio hardware in a field-programmable gate array (FPGA), where fine timing control is possible. A split-MAC approach is taken in the proposed design. However, the frequency change decision, the "cognitive" aspect of the radio, is not deployed to the hardware; it resides in the host domain where it can more easily be re-programmed, and accommodate flexible inputs such as changing policies and spectrum allocations.

The latency in the MAC protocol also limits the channel switching speed. The MAC protocol dictates the sensing schedule for all nodes in the network, as well as the means by which nodes communicate their sensing data to other nodes, be it for distributed sensing or for centralized sensing. Once a spectrum allocation decision has been made, it must also be communicated to neighbours successfully before the new channel can be acquired.

978-1-5386-0595-0/17/$31.00 ©2017 IEEE

Similar network designs to the one that is presented here have been proposed in the literature [9][10], but their implications for SDR platform implementation have not been evaluated. In [9], simulations of a cluster-based cognitive radio mobile ad hoc network (MANET) under frequency-hopping jamming indicate that frequent sensing and reporting, similar to the hop rate of the jammer, supports the network in changing frequencies to maintain network command and control functionality in the presence of jamming. Despite the frequency switching delay being an important characteristic in SDR-based DSA network implementation work, reported frequency switching delays can be very long [11], or delays are reported with minimal discussion [12][13].

II. Network description

Multi-hop mobile ad hoc networks (MANETs) are a good choice for tactical networks because they are self-organizing and self-healing. Proposed DSA MANET designs tend to fall into one of two broad categories depending on how they approach the network scalability problem. On one hand, a MANET could be built using opportunistic link-based DSA in which each link in the MANET is realized on a different frequency channel (even for links emanating from the same node). An example of such a MANET is the Wireless Network after Next [14]. These networks achieve high frequency reuse, but require many RF tuners per radio. On the other hand, a MANET could also be built with clusters of nodes. Within a cluster, nodes communicate on a single frequency channel, and a clusterhead (CH) node provides control [11]. In order to communicate between clusters, gateway nodes relay traffic across to a neighbouring cluster, which operates on a different frequency (see Figure 1). Cluster-based networks may be implemented with single- or dual-transceiver radios.

Tactical MANETs need to support several real-time communication channels. In order for real-time communication to be set up quickly, routes need to be pre-established and constantly maintained. Single-frequency clusters allow for spatial re-use of the spectrum as well as simplified multi-hop routing within a cluster, and simplify the network control. Another reason that clusters work well is that in the military command structure, small groups (sections) of soldiers work together and are in constant communication, whereas inter-cluster communication is less frequent.

The proposed prototype tactical network is a cluster-based ad hoc network. Communication from clusterhead (CH) to regular node (RN) and back is on a time division duplex (TDD) basis, within a time division multiple access (TDMA) frame. TDMA is typically favoured as a multiple access strategy for tactical MANETs, as it offers fixed latency which benefits voice connections. In the SDR testbed, synchronization relies on GPS as a timing reference, which simplifies the design.

A TDMA frame consists of a CH beacon, and one data slot for each node (including the CH). The data slot is used for traffic including data and a sensing report. See Figure 2 for a frame for a network with one CH and 8 RNs. The beacon includes a cluster ID, a frequency change notification (if required), time slot allocations for all nodes for the following frame, and inter-cluster control data.

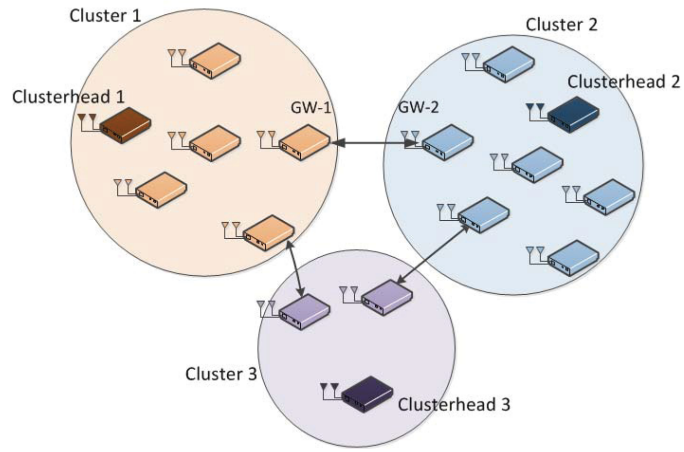

Fig. 1: Cluster-based mobile ad hoc network with gateways.

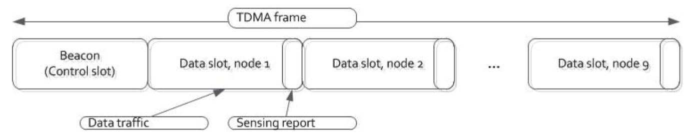

Fig. 2: TDMA frame serving a cluster of 9 nodes.

In support of the spectrum awareness and DSA requirements, each regular node (RN) collects spectrum sensing measurements and reports them back to their CH node along with their current location. The CH is responsible for making the spectrum allocation decisions based upon the following foundational algorithms, described in detail in [15]:

- Spectrum sensing algorithm - a rule set for which channels to sense and when,
- Sensor reporting criteria - rules for what data to report and when, e.g., energy in band, link quality, and
- Spectrum access algorithm - a rule set for channel selection based on the sensing outcome.

Dual-transceiver radios are proposed for all nodes in the SDR network testbed, where the second transceiver is a dedicated spectrum sensor. The spectrum sensor scans other available frequency channels for spectrum opportunities according to the sensing schedule defined by the CH in the beacon; a scan is also triggered in the event that communication with the CH is lost so that the node may rendezvous with its cluster.

For spectrum de-confliction between clusters, a control channel is established between neighboring CHs using the beacon. The control channel allows sharing of spectrum usage messages and communication of spectrum opportunities. Through coordination with adjacent clusters, some access prioritization might be achieved, which is operationally important in a multi-national scenario.

III. Software architecture

Each node in the proposed testbed consists of a host processor (the software platform), and a Nutaq PicoSDR (the hardware platform). Figure 3 depicts the proposed software architecture. The node functionality is mapped to the host processor and SDR such that, with the exception of time slot synchronization functionality, the MAC software runs on the host processor. With the MAC on the host, there is less re-programming effort when modifying components of the MAC such as the algorithms governing spectrum sensing (which channels to sense, and when) and spectrum access.

From an application point of view, it is desirable to use native Linux communications applications such as the command line application 'ping', or 'iPerf' to communicate over the SDR testbed network. These communications applications pass data through the Linux networking stack residing in the kernel. Thus, a Linux kernel module, or driver, must be created for the embedded Linux device, implementing the MAC layer functionality of the radio node.

Implementing the MAC software as a kernel module approach somewhat complicates the proposed design. Typically, SDR software applications are built upon the drivers supplied by the SDR vendor, and send data to and from the SDR via user-space application programming interfaces (APIs), such as Nutaq's real-time data exchange (RTDEx) API for Host-FPGA communications. The API residing in the user-space, while the MAC driver resides in the kernel, implies that a mechanism is required to pipe data quickly back and forth across the border between the user-space memory and the kernel memory - a cross-border pipeline.

To test the feasibility of a cross-border pipeline, one was implemented based on Netlink sockets, which are used for inter-process communications between the kernel and the user-space. It was determined that the pipeline would add a small delay to a node's data path. To test the data rate, frames of 1472 B, the maximum size for the Ethernet bus, were sent through the pipeline and the speed was increased until packet drops occurred. The pipeline was determined to carry roughly 100 MBps in either direction, and a small latency (5-10 μs) was measured.

Between the pipeline and the PicoSDR, an external API transports data over a Gigabit Ethernet (GigE) link; a PCIe interface is also available. In the design, this data is formatted according to a custom packet format defined for host-FPGA communications. Data is designated in the packet header as being transmit (TX), receive (RX) or control (CTRL) data. Control packets are not for network communications - they are used by the host for configuration of the hardware platform, specifically the physical layer time/frequency behaviour in the coming frame.

The RTDEx API function call to send data over the GigE link requires approximately 1-2 ms, which dominates the data transfer delay. The transmission of one time slot of data, to or from a Linux Host using frames of 1472 B is accomplished in 13 μs (throughput of the RTDEx bus over GigE is 114.0 MBps). This delay could be further reduced to a few μs using a PCIe bus interface.

A global synchronization strategy is specified for the network. Each host runs a network timing protocol (NTP) server, each using an attached GPS receiver as a stratum-1 timing reference, allowing the host to synchronize its system time to coordinated universal time (UTC) to within several microseconds. Then, the host pre-configures the physical layer on the SDR, which is also synchronized with the GPS receiver, for transmission and reception of the next frame, while the current frame elapses.

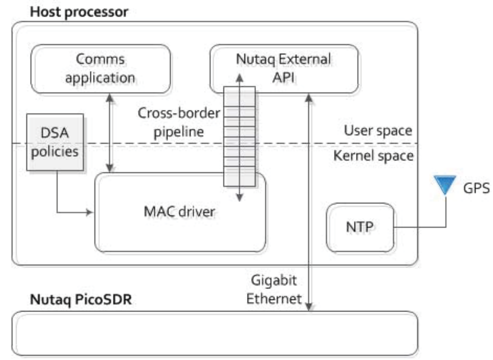

Fig. 3: SDR testbed node software architecture.

IV. Hardware Architecture

Commercial SDRs such as the Nutaq PicoSDR consist primarily of an FPGA carrier board with one or more FPGA mezzanine cards (FMCs) bearing analog-to-digital and digital-to-analog converters and the radio-frequency (RF) interfaces. In the proposed implementation, the physical layer functionality of a node, including transmission, reception and sensing, is implemented entirely in the FPGA, as the FPGA is better suited than the host to realize time-critical physical layer functionality. See Figure 4 for the SDR testbed node hardware architecture, which includes a Perseus 601X carrier card with Xilinx Virtex-6 FPGA, and two radio cards: the Radio420X FMCs.

Based upon the MAC schedule, the host configures the physical layer for the next frame, including sending any transmit data to the FGPA. Transmit data is then buffered in the FPGA, until it is sent over the RF interface in the appropriate slot.

To configure the physical layer, the host sets up several custom registers in the FPGA. Configuration is done via an RTDEx function call which sends a message to a small footprint PetaLinux operating system running in the FPGA, known as the Microblaze; the Microblaze populates the custom registers. The custom registers are set up with the time slots within a frame, and frequencies in which the node performs radio operations: data transmission, data reception and spectrum sensing. The configuration is done one frame in advance of a coming TDMA frame due to the node re-configuration time; node re-configuration requires 1-2 ms for the RTDEx function call.

Based upon the custom register configuration, the FPGA is responsible for time slot control logic, which synchronizes the radio operations to time slot boundaries based on fine timing

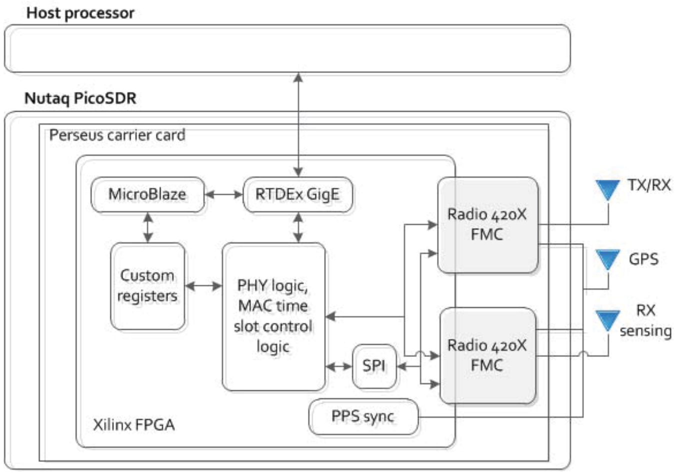

Fig. 4: SDR testbed node hardware architecture.

signals derived from each node's GPS reference clock. As each time slot elapses in the current frame, the physical layer re-tunes the radios according to the configuration from the host.

Re-tuning can be accomplished within the guard time at the beginning of a slot; the time required to change the operating frequencies of the radio-frequency integrated circuits (RFICs) on the Radio420X daughterboards is experimentally determined to be approximately 60 μs. The delay includes the time to re-program a set of internal registers in the Lime Microsystems LMS6002D radio frequency integrated circuit (RFIC) on the Radio402X, as well as time for the local oscillators to re-settle. It is assumed that there are a fixed number of frequencies which are previously known, such that the register write operations may be prepared in advance, with the register addresses and values in read-only memory.

V. Physical and MAC layer implementation

The physical layer in the proposed implementation is based upon the OFDM reference design that is included with the Nutaq PicoSDR, with some additions including interleaving and fine carrier frequency offset estimation. The radio operates in the unlicensed band at 2.4 GHz, using an OFDM waveform with a bandwidth of 12.8 MHz. For robustness, rate 1/2-coded BPSK is used as the subcarrier modulation. See Table I for more physical layer characteristics. Spectrum sensing is performed at the physical layer, occurring at prescribed time slots and frequencies according to the frame configuration dictated by the CH. The testbed spectrum sensing will initially be energy detection over the band being measured, over the period of a time slot.

Simple round-robin TDMA scheduling is proposed for the testbed. Each node is serviced at least once per frame. Reliability mechanisms such as acknowledgements and re-transmissions are planned for the future.

The TDMA time slot length is selected to be 2 ms, which provides enough time to transmit 255 OFDM symbols, and includes a guard time of 406.25 μs. This is a long guard time, which allows for timing errors between nodes as well as time

TABLE I: Physical layer parameters.

Modulation	1/2 BPSK
Used subcarriers	52
Pilot subcarriers	4
Data subcarriers	48
Virtual subcarriers	12 (DC and near edges)
IFFT/FFT sampling rate	12.8 MHz
Subcarrier spacing	0.2 MHz (12.8 MHz/64)
OFDM symbol length	5 μs (1/0.200 MHz)
Cyclic prefix	1.25 μs
Total OFDM symbol length	6.25 μs

to re-tune a transceiver at each slot. Within a time slot, 255 OFDM symbols are transmitted at 24 data bits per OFDM symbol, or 6120 bits in a 1.6 ms period. In this design, with 9 nodes the typical per-node data rate is 306 kbps, given that one slot is guaranteed to be available for transmission in a 10-slot frame. This rate supports light data traffic and command and control, which is suitable for DSA experimentation.

VI. Cluster frequency switching delay estimation

In this section, an approximate cluster frequency switching delay will be estimated. In the discussion of the delay estimate, it is demonstrated that there is a dependence between the delays inherent in the SDR and host platforms, and the delays in communication over the network.

The time to switch (T_{switch}) can be defined as the time elapsed from the start of the interference or jamming in the channel of operation, to the time when the cluster as a whole begins communicating on a new frequency channel, starting with a beacon from the CH. The components of this time (some of which are labelled in Figure 5) are:

- T_{sense} - the time from the start of the interference event to the time the CH has received sensing reports from its regular nodes;
- $T_{transfer}$ - the time to transfer a report or beacon between the CH host and the SDR, or vice versa, including the bus, API call and pipeline delays;
- $T_{process}$ - the processing time required for the CH host to make a frequency change decision;
- $T_{disseminate}$ - the time required for the CH to transmit the beacon to all RNs; and
- T_{config} - the time required for each RN to configure its radio and execute the frequency change.

The values of these delay components are dependent on factors such as the number of nodes in the network, the node implementation platform, and the specific MAC protocol. For example, the beacon dissemination time (which carries the frequency change command) could be quite long in practice as it is influenced by the effects of the wireless channel; MAC-level re-transmissions require time, which is not considered here.

Following the start of the jamming or interference which affects network throughput but does not completely suppress

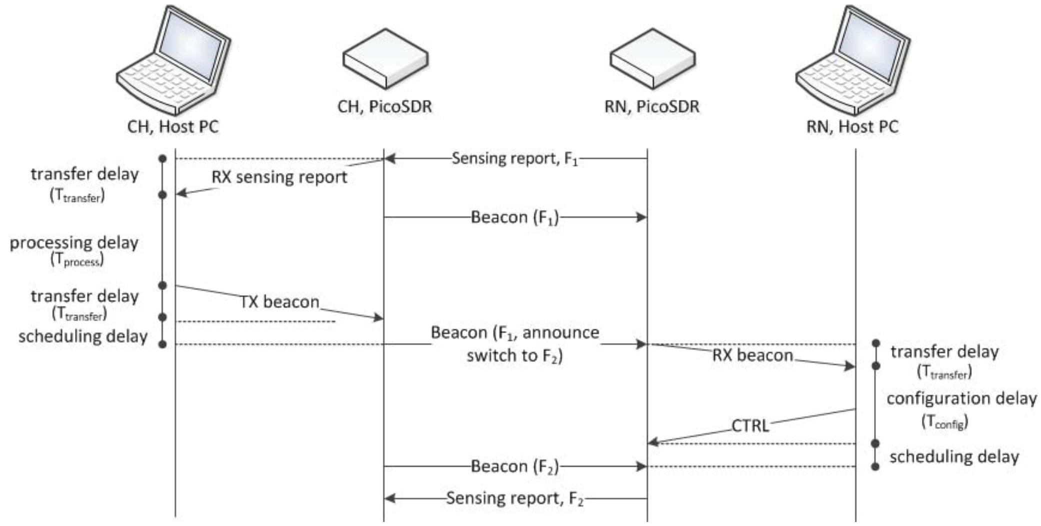

Fig. 5: Hardware and software delays in frequency switching. The scheduling delay illustrates the delay before the beacons are sent and received according to the regular TDMA schedule.

communications, RNs detect it and report back to their CH in the next sensing data reporting period. The sensing data reporting period may not require each node to report every slot (Figure 2). Rather, the reporting period may be every η data slots allocated, to reduce reporting overhead. While the spectrum allocation decision may be based on any number of sensing results from the RNs, it is assumed here for simplicity that a decision is made after at least one report has been collected from each of the nodes, and that the cumulative sensing evidence from the RNs indicates a frequency channel change should be made. If there are N nodes in the network and data slots are allocated in a round-robin fashion, the number of data slots required to receive a report from every node is $N\eta$. If there are M available data slots per frame, the time for the CH to receive all sensing reports is given by:

$$T_{sense} = \left\lceil \frac{N\eta}{M} \right\rceil \times T_{frame}. \quad (1)$$

where T_{frame} is the duration of a frame, and the upper bound indicates the number of frames.

At the CH node, the last report must be transferred to the host, from the radio. As discussed in the previous section, the data transfer delay $T_{transfer}$ = 2 ms, primarily for the RTDEx function call. Once the CH host has received the last of the required reports at the SDR, it must interpret the sensing reports and make a decision. In this design, the DSA algorithm is procedural, based upon the sensing data from the regular nodes in the cluster, and given the pre-existing spectrum access policy. A representative algorithm is estimated to require tens of μs, based upon a benchmark test (Ubuntu 14.04, Intel Core i7, 3.0 GHz), in which a representative kernel module function call was timed 300 times, with the greatest value being 49 μs.

The frequency decision is captured in the beacon data, which is transferred to the physical layer in the SDR, where it awaits its next scheduled beacon slot. Any nodes within range of the CH will receive the frequency change notification by the end of the beacon slot. Greater delay will be observed if the beacon data is missed by one or more nodes, or if multiple hops are required; the reliable dissemination of frequency change notification is a key problem in DSA MANETs. Following reception of the beacon data at the RN host ($T_{transfer}$ is required), the host configures the SDR to switch to the prescribed frequency on the next frame, while the current frame elapses. For the purposes of this analysis it is estimated that $T_{disseminate} + T_{transfer} + T_{config} \leq 1$ frame. This represents the simplest scenario, as it assumes that every node may be reached within one TDMA frame. If communication from the CH to one or more nodes is lost due to jamming or as a result of the frequency change, those nodes must follow a 'late-entry' procedure on the new channel once it is established. In the event the CH itself is unable to communicate prior to the frequency switch, all nodes must scan all candidate channels and rendezvous on the new channel chosen by the CH, incurring further delay.

Then, for a cluster of 9 nodes with the software and hardware architecture described in this paper, where one frame is 10 time slots of 2 ms each, the frequency switching delay T_{switch} is estimated to be roughly 3 frames, or 60 ms (summarized in Table II). This estimate is a best-case scenario, and illustrates that as a networked cluster of radios, it is difficult to quickly change frequency in response to short-term jamming or interference.

VII. Conclusion

The design of a cluster-based DSA network was proposed, and the high-level design was mapped to an architecture and implementation based on the Nutaq PicoSDR. The delay in switching frequencies was estimated; in the proposed design, a small cluster of nodes could not evade jamming of less than 60 ms duration. With significant attention given to hardware and

TABLE II: Components of estimated frequency switching delay, for N=9 nodes, reporting period η=1 (every allocated slot), M=10 time slots/frame.

T_{sense}	≤1 frame
$T_{transfer} + T_{process}$	≤1 frame
$T_{disseminate} + T_{transfer} + T_{config}$	≤1 frame
T_{switch}	≤3 frames

software partitioning of the DSA functionality, the systematic delay components arising from the SDR implementation might be reduced. Ideally DSA functionality should be designed into the software and hardware architecture from the outset, not applied as an add-on technology.

The actual frequency switching delay could be much greater when one or more nodes are out of range when the frequency change notification is transmitted. For cluster based networks, regardless of the implementation platform, the gathering of timely sensing data and the reliable dissemination of the frequency change command pose challenges for DSA implementation.

REFERENCES

[1] L. E. Doyle, P. D. Sutton, K. E. Nolan, J. Lotze, B. Ozgul, T. W. Rondeau, S. A. Fahmy, H. Lahlou, and L. A. DaSilva, "Experiences from the Iris Testbed in Dynamic Spectrum Access and Cognitive Radio Experimentation," in *2010 IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN)*, April 2010.

[2] PicoSDR Series for Wireless Multi-Standard Prototyping. Nutaq. [Online]. Available: http://www.nutaq.com/products/picosdr

[3] Ettus Research - The leader in Software Defined Radio (SDR). Ettus Research, A National Instruments Company. [Online]. Available: http://www.ettus.com/product

[4] L. A. DaSilva, A. B. MacKenzie, C. R. C. M. da Silva, and R. W. Thomas, "Requirements of an Open Platform for Cognitive Networks Experiments," in *2008 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Oct 2008.

[5] T. R. Newman, S. M. S. Hasan, D. Depoy, T. Bose, and J. H. Reed, "Designing and deploying a building-wide cognitive radio network testbed," *IEEE Communications Magazine*, vol. 48, no. 9, pp. 106–112, Sept 2010.

[6] B. Scheers, "Introduction of Dynamic Spectrum Access Technology in NATO Europe Tactical Communications," in *MILCOM 2013 - 2013 IEEE Military Communications Conference*, Nov 2013, pp. 737–742.

[7] GNURadio, The Free & Open Software Radio Ecosystem. The GNU Radio Foundation, Inc. [Online]. Available: http://gnuradio.org

[8] P. D. Francesco, S. McGettrick, U. K. Anyanwu, J. C. O'Sullivan, A. B. MacKenzie, and L. A. DaSilva, "A Split MAC Approach for SDR Platforms," *IEEE Transactions on Computers*, vol. 64, no. 4, pp. 912–924, April 2015.

[9] G. Szmit, J. Dołowski, and J. Łopatka, "Performance of distributed channel selection for cognitive MANET in the presence of intentional jamming," in *2016 International Conference on Military Communications and Information Systems (ICMCIS)*, May 2016.

[10] T. Chen, H. Zhang, M. Matinmikko, and M. D. Katz, "CogMesh: Cognitive Wireless Mesh Networks," in *2008 IEEE Globecom Workshops*, Nov 2008.

[11] F. Ge, A. Radhakrishnan, M. Y. ElNainay, Q. Chen, C. W. Bostian, and A. B. MacKenzie, "Software Radio-Based Decentralized Dynamic Spectrum Access Networks: A Prototype Design and Experimental Results," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, Dec 2010.

[12] H. Harada, "A Software Defined Cognitive Radio Prototype," in *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, Sept 2007.

[13] R. J. DeGroot, D. P. Gurney, K. Hutchinson, M. L. Johnson, S. Kuffner, A. Schooler, S. D. Silk, and E. Visotsky, "A cognitive-enabled experimental system," in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005.*, Nov 2005, pp. 556–561.

[14] B. Fette, "Fourteen Years of Cognitive Radio Development," in *MILCOM 2013 - 2013 IEEE Military Communications Conference*, Nov 2013, pp. 1166–1175.

[15] Y. Chen, Q. Zhao, and A. Swami, "Joint Design and Separation Principle for Opportunistic Spectrum Access," in *Signals, Systems and Computers. ACSSC '06. Fortieth Asilomar Conference on*, Oct 2006, pp. 696–700.

DOCUMENT CONTROL DATA

(Security markings for the title, abstract and indexing annotation must be entered when the document is Classified or Designated)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g., Centre sponsoring a contractor's report, or tasking agency, are entered in Section 8.) DRDC – Ottawa Research Centre Defence Research and Development Canada 3701 Carling Avenue Ottawa, Ontario K1A 0Z4 Canada	2a. SECURITY MARKING (Overall security marking of the document including special supplemental markings if applicable.) CAN UNCLASSIFIED 2b. CONTROLLED GOODS NON-CONTROLLED GOODS DMC A

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C or U) in parentheses after the title.)

Design of a software defined radio-based tactical DSA network

4. AUTHORS (last name, followed by initials – ranks, titles, etc., not to be used)

Watson, Susan; Larouche, Jean-Benoit

5. DATE OF PUBLICATION (Month and year of publication of document.)	6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.)	6b. NO. OF REFS (Total cited in document.)
February 2018	6	12

7. DESCRIPTIVE NOTES (The category of the document, e.g., technical report, technical note or memorandum. If appropriate, enter the type of report, e.g., interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

External Literature (P)

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

DRDC – Ottawa Research Centre
Defence Research and Development Canada
3701 Carling Avenue
Ottawa, Ontario K1A 0Z4
Canada

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)	9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)
10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.) DRDC-RDDC-2018-P011	10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

11a. FUTURE DISTRIBUTION (Any limitations on further dissemination of the document, other than those imposed by security classification.)

Public release

11b. FUTURE DISTRIBUTION OUTSIDE CANADA (Any limitations on further dissemination of the document, other than those imposed by security classification.)

12. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

This paper presents the design for a prototype tactical dynamic spectrum access (DSA) mobile ad hoc network, where the network is organized into clusters operating on a single frequency. The frequency may be changed autonomously by the network in response to jamming or interference, after some frequency switching delay. The network node design is implementable on a software defined radio (SDR), such as the Nutaq PicoSDR. The contribution of this work is the prototype design and mapping of the design to the hardware and software architecture of the SDR, with a focus on the estimated delay in switching to a new frequency channel. The frequency switching delay is the result of systematic delays arising from the SDR platform, compounded by delays in the protocols enabling DSA, the gathering of sensing data from network nodes, and the dissemination of the frequency change command throughout the network.

13. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g., Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

dynamic spectrum access; software defined radio; mobile ad hoc networks