IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 12, 2018, accepted June 13, 2018, date of publication July 20, 2018, date of current version September 21, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2857843

Airborne Cognitive Networking: Design, Development, and Deployment

GEORGE SKLIVANITIS[1], (Member, IEEE), ADAM GANNON[2], (Student Member, IEEE), KONSTANTINOS TOUNTAS[1], (Student Member, IEEE), DIMITRIS A. PADOS[1], (Senior Member, IEEE), STELLA N. BATALAMA[3], (Senior Member, IEEE), STEPHEN REICHHART[4], MICHAEL MEDLEY[4], (Senior Member, IEEE), NGWE THAWDAR[4], (Member, IEEE), ULYSSES LEE[4], JOHN D. MATYJAS[4], (Member, IEEE), SCOTT PUDLEWSKI[4], (Member, IEEE), ANDREW DROZD[5], (Fellow, IEEE), ASHWIN AMANNA[5], FRED LATUS[5], ZACHARY GOLDSMITH[5], AND DAVID DIAZ[5]

[1]I-SENSE and the Department of Computer and Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431-0991, USA
[2]Department of Electrical Engineering, The State University of New York at Buffalo, Buffalo, NY 14260, USA
[3]College of Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431-0991, USA
[4]Air Force Research Laboratory, Information Directorate, Rome, NY 13441, USA
[5]ANDRO Computational Solutions, LLC, Rome, NY 13441, USA

Corresponding author: Dimitris A. Pados (dpados@fau.edu)

This work was supported by the U.S. Air Force Research Laboratory under Grant FA8750-14-1-0074 and in part by the National Science Foundation under Grant CNS-1704813. The work of D. A. Pados was also supported by the Schmidt Family Foundation, Boca Raton, FL.

ABSTRACT We design, develop, and experimentally validate a complete integrated software/hardware platform for airborne cognitive networking in both indoor and outdoor environments. We first present the concept of all-spectrum cognitive networking and describe a distributed algorithm for maximizing network spectral efficiency by jointly optimizing channel access code-waveforms and routes in a multi-hop network. We then discuss system design parameters and implementation details for setting up a software-defined radio (SDR) testbed that enables reconfigurability at the physical (PHY), medium-access control (MAC), and network (NET) layers of the network protocol stack, either by a user or by means of autonomous decisions. Our algorithmic developments toward spectrally-efficient cognitive networking are software optimized on heterogeneous multi-core general-purpose processor-based SDR architectures by leveraging the design of a novel software-radio framework that offers self-optimization and real-time adaptation capabilities at the PHY, MAC, and NET layers of the network protocol stack. We verify our system design approach in a large-scale testbed deployment of ten terrestrial and one airborne SDR platforms at the Stockbridge Controllable Contested Environment at the Air Force Research Laboratory, Rome, NY, USA. Proof-of-concept experimental results from both indoor and outdoor testbed deployments show that the proposed system can be used to build all-spectrum cognitive networks that withstand intentional interference at PHY and NET layers and can cognitively coexist with non-cross-layer optimized networks.

INDEX TERMS Cognitive radio, cross-layer design, distributed computing, dynamic routing, multi-core processing, software-defined radio, waveform design.

I. INTRODUCTION

Future airborne networks will be defined by the resilience to maintain wireless connectivity in dynamic communication environments and the intelligence to adapt to varying traffic loads, RF interference, and frequent network failures. The design of airborne networks is challenged by rapidly changing network dynamics, limited energy and link-bandwidth capacity between high/low-speed mobile wireless nodes that are either geographically or hierarchically dispersed [1]–[3]. Existing design approaches rely on either channel/network emulations [4]–[6] or implementation of layered wireless network protocol architectures on off-the-shelf radio devices [7], [8]. As a result, integration of autonomous radio reconfiguration functionalities to unmanned aerial vehicle (UAV)

platforms [9] and their evaluation in real-time remains quite challenging.

Wireless links in UAV networks are subject to time-varying quality due to high mobility, distance-dependent path loss, interference, multi-path, and doppler effects among others. Communication is fundamental to UAV networks. Contrary to terrestrial networks, UAV wireless networks rely on distributed network protocol architectures that have both cognitive coexistence and intra-network interference-avoidance requirements [2]. Node mobility exposes UAV wireless links to unpredictable interference levels that are time-varying and location-dependent, therefore dynamic management of link parameters is needed for being robust to network disruptions at any time and location. Continuously changing channel and interference conditions entail new network layer designs [9], where routes in multi-hop airborne networks are jointly optimized with communication link parameters at each hop. The requirements listed above are particularly important for UAV swarms because network disruptions may prevent the delivery of sensed data to the appropriate processing nodes, lead to insufficient situational awareness for effective in-field planning, and result in delayed/lost command-and-control messages, thereby creating significant deviation from the desired system behavior. At best, this will increase the time and number of assets required to complete a mission; at worst, it may result in mission failure. The ability to recover from the loss of communication to any node is necessary to resilient operation of airborne networks [3].

Airborne communication systems can significantly benefit from the integration of cognitive radio (CR)-based resource allocation schemes, which can further enhance spectrum utilization efficiency and guarantee network connectivity when airborne communication links are constrained or unavailable. A comprehensive survey regarding spectrum utilization trends and CR in aeronautical communication systems can be found in [10]. Spectrum occupancy measurements show that certain parts of the radio spectrum, particularly frequency bands allocated for air-to-ground communications, remain underutilized [11], [12]. This is partly due to the adoption of protocol architectures that rely on static resource allocation. Additionally, inherent couplings between the physical (PHY), medium-access control (MAC), and network (NET) layers of the network protocol stack entail dynamic control and management of shared networking resources that vary in frequency, time, and space. Multi-UAV mesh networks with information sharing and multi-hop routing introduce additional communication overhead and require new protocol designs to maximize network throughput performance [9]. Consequently, self-optimized, reconfigurable network protocol architectures that implement distributed, cross-layer optimized control decisions are expected to satisfy the dynamic demands and complex cross-layer interactions in future airborne mesh networks [13].

In this work we present, for the first time, a systematic analysis across the PHY, MAC, and NET layers of the network protocol stack toward the implementation and real-time validation of all-spectrum cognitive networking [14] in a software-defined radio testbed. We consider the development of an ''elastic'' testbed comprised of cross-layer cognitively optimized links that are co-located with static/baseline non-cognitively optimized links. The ''elastic'' testbed consists of terrestrial and airborne integrated software/hardware platforms. Cognitive links implement a novel distributed approach for spectral efficiency maximization by jointly designing channel-access waveforms that span the entire available spectrum (all-spectrum channelization) and selecting network routes under both intra- and inter-network interference scenarios. We first identify system design challenges regarding software, hardware, and baseband processing requirements at PHY, MAC, and NET layers to fulfill the needs of the proposed distributed algorithm for cross-layer interactions and cognitive decision-making. Additionally, we discuss signal processing details for the implementation of the cognitive framework including frame design, time and frequency synchronization, and code-waveform optimization. The modular architecture of the proposed platform allows us to abstractly define networking protocols with (or without) cross-layer interactions in a high-level description language and separate verification of upper-layer networking functionalities and physical-layer designs. Proof-of-concept experimental results from both indoor and outdoor testbed deployments show that the proposed cognitive platform can withstand intentional interference at PHY and NET layers as well as enable cognitive coexistence with non cross-layer optimized networks.

The development of the ''elastic'' testbed enables accelerated design and validation of cross-layer optimized network protocol solutions in realistic scenarios. Our main contributions can be summarized as follows:

- **Cross-layer System-Level Analysis.** We identify system design challenges and requirements at PHY, MAC, and NET layers for the implementation of all-spectrum cognitive networking.
- **Distributed Network Optimization.** We propose computationally efficient code-waveform optimization techniques and distributed network control mechanisms to handle real-time waveform and routing decisions at each cognitive network node.
- **Software-Radio Reconfigurable Framework.** We develop a complete integrated software/hardware radio reconfigurable framework to enable the first field deployment of all-spectrum cognitive networking on a hybrid ground-air software-defined radio testbed.
- **Testbed Evaluation.** We experimentally validate all-spectrum cognitive networking in realistic testbed scenarios (both indoor and outdoor) and show that cognitively optimized network nodes can withstand both PHY- and NET-layer interference and enable underlay spectrum coexistence with non cross-layer optimized network nodes.

II. RELATED WORK

Cognitive radio has emerged as a promising technology to improve spectrum utilization efficiency in wireless networks [15]–[18] while cognitively networked UAVs have attracted interest for public safety and next-generation first-responder applications [19], [20]. Implementation of cognitive decision making algorithms for optimal network control yet remains quite challenging [21]. In this section, we review state-of-the-art work in i) airborne cognitive networking, ii) cognitive radio algorithms for (joint) optimization of resource allocation parameters at either one or multiple layers of the network protocol stack, and iii) software-defined/cognitive radio testbeds.

A. AIRBORNE COGNITIVE NETWORKS

The need for cognitive spectrum sensing and sharing in aeronautical communication networks is highlighted in [22]. Cognitive radio is proposed in [23] as a solution to meet future UAV communication demands, particularly latency-sensitive control and high-throughput sensor data. Work in [24] proposes to use the position of a UAV as an additional degree of freedom to protect primary user transmissions in an underlay dynamic spectrum access scheme [17]. A survey in energy-based spectrum sensing techniques is studied in [25] for broadband VHF (B-VHF) cognitive aeronautical communication systems. Cognitive algorithms for LTE-A aerial base stations for public safety applications are discussed in [26].

B. COGNITIVE RESOURCE ALLOCATION ALGORITHMS

All-spectrum cognitive networking in [14] proposes for the first time in the literature, joint distributed power, code-waveform, and route optimization in a cognitive multi-hop network. Work in the field of all-spectrum channelization considers the design of code-waveforms for secondary (ad-hoc) wideband links that cooperate [27], or simply coexist without any form of cooperation with primary (spectrum licensees) wideband [28] and narrowband [29] stations in an underlay fashion. Since the spectrum environment in a multi-hop network varies in time and space, and interference to primary stations depends on the location of the cognitive nodes, it is important to jointly and dynamically optimize PHY-layer and routing functionalities at each network node. Transmit power and spectrum sensing parameters are optimized in [30] to minimize energy consumption in a cooperative-sensing-based code-division multiple access (CDMA) cognitive network.

Outside the framework of code-waveform based cognitive channelization altogether, interesting work in the form of joint beamforming and power allocation is reported in [31], while multi-antenna cognitive beamforming and interference avoidance algorithms are described in [32] and [33]. Hybrid overlay/underlay CR transmission systems in [34] and [35] efficiently exploit both unused and underutilized spectrum through orthogonal-frequency-division-multiplexing (OFDM) and multicarrier-CDMA.

An extensive overview of routing techniques for multi-hop cognitive radio networks can be found in [36]. A distributed algorithm for joint routing and dynamic spectrum allocation in unused spectrum bands (i.e. ''white spaces'') is proposed in [37]. Distributed routing to reduce interference to primary stations and minimize route delay is proposed in [38], while [39] describes a route selection algorithm based on location information and locally available spectrum access opportunities at each node. Two classes of routing protocols in [40] aim to reduce end-to-end latency and increase energy efficiency and throughput based on cooperative routing and spectrum aggregation techniques. On-demand based routing in [41] is conducted based on clustering of nodes according to spectrum availability, power, and node stability. Finally, work in [42] describes a joint rate adaptation, channel assignment, and routing approach to maximize network resource utilization.

C. SOFTWARE-DEFINED/COGNITIVE RADIO TESTBEDS

Indoor testbed deployments in [43]–[45] describe hardware and software requirements toward rapid evaluation of CR protocols in real-world conditions with software-defined radio (SDR) and Android OS devices. All-spectrum cognitive channelization around wideband and narrowband primary stations is evaluated for the first time in an indoor SDR testbed in [29]. Work in [46] describes an experimental prototype for multi-antenna cognitive beamforming, while over-the-air experiments in [47] and [48] evaluate a hybrid overlay/underlay CR technique. Experimental work for spectral coexistence in ''white spaces'' is presented in [49], [50]. A software-defined CR prototype based on off-the-shelf IEEE 802.11a/b/g wireless cards is developed in [51]. A distributed algorithm for joint routing and dynamic spectrum access in ''white spaces'' is implemented in SDRs in [52], while [53] evaluates the algorithm in SDRs that communicate with each other through a wireless network emulation platform. Simulations and experiments on an SDR testbed in [54] evaluate the performance of different routing protocols for CR networks. Route selection schemes based on reinforcement learning and spectrum leasing are experimentally evaluated in an SDR testbed in [55].

In the following sections, we describe a reconfigurable software/hardware framework that enables self-optimization of protocol parameters across multiple layers of the network protocol stack in response to dynamic conditions. Particularly, we discuss the requirements, design considerations, and algorithmic developments towards the first field deployment and evaluation of all-spectrum cognitive networking in a hybrid ground-air software-defined radio testbed. Finally, we present proof-of-concept results that demonstrate cognitive coexistence and interference avoidance, capabilities critical to satisfy the increasing throughput demands of unmanned airborne networks while maintaining connectivity in dynamic environments.

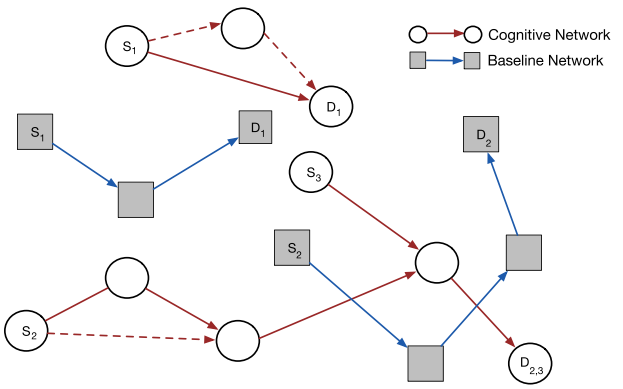


FIGURE 1. The "elastic" network: cross-layer optimized cognitive links (red-colored) are co-located with static/baseline links (blue-colored) that implement a layered network protocol stack. Solid lines depict selected routes from source s to destination d, while dashed lines depict alternate routes.

III. SYSTEM OVERVIEW

In this section, we provide a system overview of the "elastic" network. More specifically, we present the PHY-layer signal model for all-spectrum cognitive and static/baseline links, review MAC- and NET-layer protocol functionalities and definitions, and describe the proposed cross-layer optimization algorithm for code-waveform and routing adaptation.

A. SYSTEM MODEL

We consider an "elastic" network comprised of cognitively cross-layer optimized wireless links that are co-located with static/baseline, non-cognitively optimized wireless links that implement a layered network protocol stack (Fig. 1). Traffic flows for both networks are carried over multi-hop routes and traffic demands consist of unicast sessions that are characterized by a source node s and a destination node d. Static/baseline network nodes have pre-assigned unique code-waveforms and routes, while cognitive network nodes jointly optimize modulating code-waveforms and selected routes in a continuous fashion on a hop-by-hop basis.

1) PHYSICAL LAYER

We let (i, j) denote a wireless link and consider data frame transmissions between transmitter i and receiver j. The data frame is comprised of N information symbols that are drawn from a complex constellation $\mathcal{A}$ and are modulated by a waveform $\psi_{ij}(t)$ of duration T. More specifically, the transmitted signal for the link (i, j) can be written as

$$x_{ij}(t) \triangleq \sqrt{E_i} \sum_{n=0}^{N-1} b_i[n]\psi_{ij}(t - nT)e^{j(2\pi f_c t+\phi_i)} \quad (1)$$

where $E_i > 0$ denotes the transmitted energy per symbol, $b_i[n] \in \mathcal{A}$ is the n-th information symbol and ϕ_i is the carrier phase related to carrier frequency f_c at the i-th transmitter. The modulating waveform for link (i, j) is given by

$$\psi_{ij}(t) \triangleq \sum_{l=0}^{L-1} s_{ij}[l]g_{T_c}(t - lT_c) \quad (2)$$

where $s_{ij}[l] \in \left\{\pm 1/\sqrt{L}\right\}$ is the l-th bit of a length-L binary code, and $g_{T_c}(\cdot)$ is a square-root-raised-cosine (SRRC) pulse with roll-off factor α and duration T_c, so that symbol duration $T = LT_c$ and bandwidth $B = (1 + \alpha)/T_c$.

Transmitted signals are considered to propagate over Rayleigh fading channels with M resolvable paths and experience multi-user interference and complex additive white Gaussian noise (AWGN) at the receiver. Multipath fading is modeled by a linear tapped-delay line with taps that are spaced at intervals of T_c and are weighted by independent fading coefficients. The received signal after carrier down-conversion at $f_c + \Delta f_{ij}$ at the j-th cognitive receiver is written as

$$y_{ij}(t) \triangleq \sum_{m=0}^{M-1} \tilde{h}_{ij}[m] \sum_{n=0}^{N-1} b_i[n]\psi_{ij}(t - (m + nL)T_c) \cdot e^{-j2\pi \Delta f_{ij} t} + i(t) + z(t) \quad (3)$$

where Δf_{ij} is the carrier frequency offset (CFO) due to manufacturing imperfections between the local oscillators of the i-th transmitter and j-th receiver and $\tilde{h}_{ij}[m] = \sqrt{E_i}h_{ij}[m]e^{-j(2\pi f_c m T_c-\phi_i)}$ denotes the complex baseband channel coefficients for the m-th path. We consider block flat-fading channels, where $\{h_{ij}[m]\}_{m=0}^{M-1}$ are independent zero-mean complex Gaussian random variables that model the fading phenomena and are assumed to remain constant over a coherence time interval $T_d = N \cdot T$. Additive white Gaussian noise and interference from co-located baseline and cognitive transmitters is denoted by $z(t)$ and $i(t)$, respectively.

2) MEDIUM ACCESS CONTROL LAYER

Both baseline and cognitive networks implement a CDMA-based random access MAC protocol, whereby a network node accesses the channel as soon as there are packets in its queue. However, non-zero cross-correlation between different code-waveforms at co-located wireless links may result in multiple-access interference (MAI). Additionally, channelization of the available bandwidth may be different at each hop in a multi-hop path, while at the same time network traffic dynamics and routing of traffic flows may frequently change. Therefore, controlling the interaction between routing and code-waveform design at the physical/link layer is of fundamental importance.

3) NETWORK LAYER

We consider that each network node maintains a separate queue for each session d for which it is either a source or an intermediate node. The queue size at the i-th network node is

updated as follows

$$Q_i^d(t+1) \triangleq \left[Q_i^d(t) + \sum_{k,k\neq i} R_{ki}^d(t) - \sum_{j,j\neq i} R_{ij}^d(t) + \mu_i^d(t) \right]^+ \tag{4}$$

where $Q_i^d(t)$ is the number of queued frames of session d waiting for transmission, $\mu_i^d(t)$ is the endogenous traffic arrival rate of session d, and $R_{ij}^d(t)$ is the transmission rate (in packets/s) on link (i, j) for session d at time t.

B. CROSS-LAYER COGNITIVE ADAPTATION

1) CENTRALIZED APPROACH

Ideally, at each hop of the cognitive network, a network-throughput optimal controller, for each decision period, should maximize a weighted-sum of differential backlogs

$$\sum_i \sum_{j\neq i} C_{ij}(\mathbf{s}_{ij}, E_i) \cdot \Delta Q_{ij}(t) \tag{5}$$

where $\Delta Q_{ij}(t) \triangleq \max_d \left[Q_i^d(t) - Q_j^d(t) \right]^+$ and the optimal weights are given by the link data rates

$$R_{ij}^d(t) \triangleq \begin{cases} C_{ij}(\mathbf{s}_{ij}, E_i), & \text{if } d = d_{ij}^*(t) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $C_{ij} \triangleq B \log_2 \left(1 + \text{SINR}_{ij}\right)$ is the channel capacity for bandwidth B and an instantaneous value of SINR_{ij} and $d_{ij}^*(t) = \arg\max_d \{Q_i^d(t) - Q_j^d(t)\}$.

The objective function in (5) is defined based on the principle of dynamic back-pressure, first introduced in [56], and provably achieves optimal network throughput by jointly optimizing resources at the physical/link and routing layers [57]. However, in practice, real-time implementation of the above ideal throughput-optimal resource allocation policy requires global knowledge of all feasible resource allocations as well as a centralized algorithm to solve a non-linear problem [14] in a time-slot basis.

2) DISTRIBUTED/UNCOORDINATED ALGORITHM

In this work, we propose to implement a distributed algorithm for joint waveform and route adaptation that aims to maximize aggregate network throughput based on real-time distributed decisions that are driven by locally collected information at the cognitive network nodes [14]. Assuming fixed network topology and transmit energy E_i, the i-th backlogged node first maximizes a local utility function $U_{ij} = C_{ij} \cdot \Delta Q'_{ij}(t)$, where

$$\Delta Q'_{ij}(t) \triangleq \begin{cases} \sum_d Q_i^d(t) - \sum_d Q_j^d(t), & \text{if } \sum_d Q_i^d(t) \neq \sum_d Q_j^d(t) \\ 1, & \text{otherwise} \end{cases} \tag{7}$$

over all feasible next hops j by optimizing modulating waveforms $\mathbf{s}_{ij}$ based on locally collected spectrum information.

Cognitive Code-Waveform and Routing Algorithm

1: At backlogged node i
2: **for each** backlogged session d **do**
3: **for each** candidate next hop j **do**
4: Calculate link capacity C_{ij} by optimizing $\mathbf{s}_{ij}$
5: **end for**
6: **end for**
7: Select next hop $j^* = \arg\max_j C_{ij} \cdot \Delta Q'_{ij}$
8: Return $\left[\mathbf{s}_{ij}^*, j^*\right]$

FIGURE 2. Pseudocode for the cognitive code-waveform and routing algorithm.

Then, each node will access the channel by selecting the waveform that optimizes its local utility. Particularly, the i-th backlogged node, at each decision period, implements the following algorithm:

1) Find the set of feasible next hops for each session d that are neighbors to node i.
2) Maximize link-capacity C_{ij} by optimizing code-waveforms $\mathbf{s}_{ij}$ for each of the feasible next hops.
3) Select as next hop j^* the one with maximal utility U_{ij}.

The algorithm returns the selected next hop j^* according to the queue size and capacity information reflected in the utility function U_{ij}. The optimization is carried out at each successive next hop to form a multi-hop path terminating at the destination. Fig. 2 depicts the steps of the distributed algorithm. Low computational complexity algorithms for solving the waveform optimization problem in the second step of the algorithm are proposed in section IV-D. At the last step of the algorithm, each cognitive network (source or intermediate) node calculates utility values U_{ij} for all candidate hops and selects the hop with maximum utility.

IV. SYSTEM DESIGN CONSIDERATIONS

In this section, we describe the system design considerations and challenges related to the implementation and deployment of the proposed distributed algorithm in a software-defined radio testbed. Specifically, we summarize our developments with respect to modulation and error-correction encoding schemes, frame detection and synchronization, maximum-SINR filtering, code-waveform optimization, medium access control, network, and application layer protocols.

A. MODULATION & ERROR CORRECTION ENCODING

Both networks may utilize phase-shift-keying (PSK) and quadrature-amplitude-modulation (QAM) schemes, such as BPSK, 4/8/16/32/64-PSK/QAM and select convolutional error-correcting codes that are punctured according to the coding rate.

B. FRAME DETECTION & SYNCHRONIZATION

The byte format of the transmitted data frame is depicted in Fig. 3. Data frames are prefixed with NET-layer headers that contain source, next hop, and destination

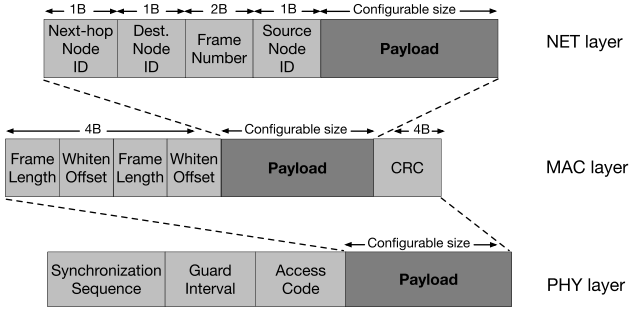


FIGURE 3. Data frame structure.

TABLE 1. Data frame parameters.

Parameter	Variable	Value
Modulation schemes	$\mathcal{A}$	PSK/QAM
Roll-off factor (SRRC)	α	0.35
Sampling frequency	F_s	1 MSps
Guard interval (in samples)	N_g	100
Code-waveform length	L	8
Code-waveform duration	T_c	4 μs
Num. of samples per code	N_{spc}	4
Num. of payload symbols	N_p	1024
Num. of access-code symbols	N_{ac}	32

MAC addresses, as well as a frame number. Subsequently, the frames are appended with a 32-bit cyclic-redundancy check (CRC) for error detection and a repetitive pattern of 4-bytes that contain information regarding the whitener offset and frame length. Whitener offset information is utilized by a randomizer block at the PHY layer to XOR payload bytes with a pseudo-noise (PN) sequence to evenly distribute power across the available bandwidth. Finally, at the PHY layer, each payload data frame is appended with a randomly-generated access code training sequence unique to each (i, j)-th link (used for fine time-synchronization and channel estimation), a guard-band interval, and a frame synchronization/detection sequence. Table 1 presents the PHY-layer parameters of the data link.

We utilize a synchronization/detection sequence based on chirp sequence keying (CSK) that is known at the receiver a priori. Chirp sequences are selected due to their superior correlation characteristics in low SINR and multipath environments [58], [59]. We generate frame detection sequences consisting of four sub-chirps [59]. Every sub-chirp is a frequency ramp with either increasing (up-chirp) or decreasing (down-chirp) frequency in the upper or lower sideband of the channel. Table 2 depicts the chirp synchronization sequence parameters.

1) PRESENCE OF A FRAME

The beginning of a data frame is detected if the peak value of the normalized cross-correlation between the received signal and the chirp synchronization sequence exceeds a predefined threshold value. Fig. 4 illustrates the sub-chirp pattern

TABLE 2. Chirp synchronization sequence parameters.

Parameter	Variable	Value
Initial frequency	f_0	$\pm 3.15 \cdot 10^6$ Hz
Sub-chirp sampling frequency	F_{sub}	32 MSps
Frequency sweeping ratio	μ	$\pm 14.62 \cdot 10^{12}$
Num. of samples per sub-chirp	N_s	38

for link (i, j) and the output of the frame correlator with the chirp synchronization sequence from over-the-air indoor SDR measurements. After acquiring the beginning of a data frame, the received signal (which includes access code and payload samples) is pulse-matched filtered, sampled over the multipath-extended duration of $(NL + M - 1)T_c$ seconds

$$\mathbf{y}_{ij}[n] \triangleq y_{ij}(kT_c), \quad k = nL, \ldots, (nL + L_M - 1), \quad \text{for} \quad n = 0, \ldots, N - 1. \quad (8)$$

with $L_M \triangleq L + M - 1$. In the rest of this section we consider frequency synchronized radio transceivers, thus Δf_{ij} is considered to be negligible. The chirp synchronization sequence alone cannot accurately determine timing. Therefore, upon the detection of a frame synchronization peak, fine synchronization at the waveform-level is done as a separate step. Fine timing synchronization is performed with a matched filter that cross-correlates the received symbols with the access code training sequence unique to each link and known at the receiver a priori. The same access code sequence is used to estimate multipath channel coefficients.

2) ABSENCE OF A FRAME

If we do not detect a distinct peak at the output of the frame correlator, we assume that the received signal contains only

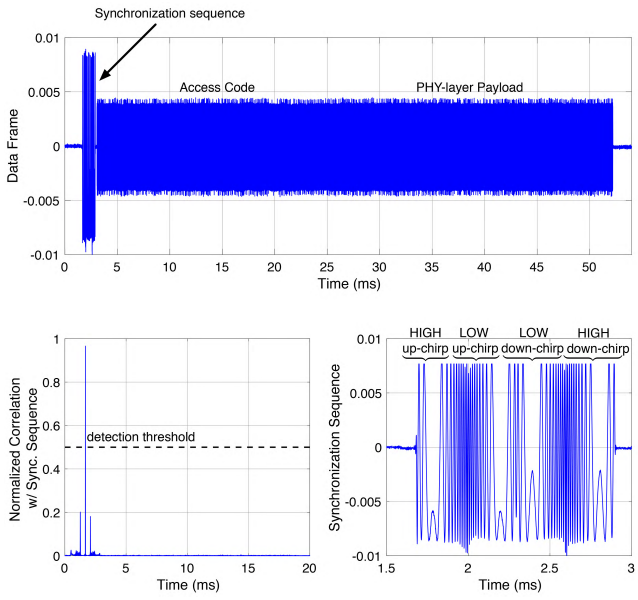


FIGURE 4. Received data frame (top). Normalized cross-correlation of the received data frame with the chirp synchronization sequence (bottom).

noise and interference (''disturbance''). Disturbance samples as measured at the j-th receiver node are similarly pulse-matched filtered and sampled over the multipath-extended duration of $(NL + M - 1)T_c$ seconds

$$\begin{aligned}\mathbf{d}[n] &\triangleq i(kT_c) + z(kT_c) \\ &= \mathbf{i}[n] + \mathbf{z}[n], \quad k = nL, \ldots, (nL + L_M - 1) \quad (9)\end{aligned}$$

where $\mathbf{i}[n] \in \mathbb{C}^{L_M}$ models colored interference with autocorrelation matrix $\mathbf{R}_{\mathrm{I}} \triangleq \mathrm{E}\{\mathbf{i}[n]\mathbf{i}^H[n]\} \in \mathbb{C}^{L_M \times L_M}$ and $\mathbf{z}[n] \in \mathbb{C}^{L_M}$ denotes zero-mean AWGN with covariance $\sigma^2\mathbf{I}_{L_M}$. The autocorrelation matrix of $\mathbf{d}[n]$ is defined as

$$\mathbf{R}_{\mathrm{I+N}} \triangleq \mathrm{E}\left\{\mathbf{d}[n]\mathbf{d}^H[n]\right\} = \mathbf{R}_{\mathrm{I}} + \sigma^2\mathbf{I}_{L_M}. \quad (10)$$

In practice, we use $\mathbf{d}[n]$ to recursively update an estimate of the inverse autocorrelation matrix $\widehat{\mathbf{R}}^{-1}_{\mathrm{I+N}}$ as

$$\widehat{\mathbf{R}}^{-1}_{\mathrm{I+N}}[n] \triangleq \frac{1}{\beta}\left[\widehat{\mathbf{R}}_{\mathrm{I+N}}[n-1] - \frac{\widehat{\mathbf{R}}^{-1}_{\mathrm{I+N}}[n-1]\mathbf{d}[n]\mathbf{d}^H[n]\widehat{\mathbf{R}}_{\mathrm{I+N}}[n-1]}{\beta + \mathbf{d}^H[n]\widehat{\mathbf{R}}^{-1}_{\mathrm{I+N}}[n-1]\mathbf{d}[n]}\right] \quad (11)$$

where $\widehat{\mathbf{R}}^{-1}_{\mathrm{I+N}}[0] = c\mathbf{I}_{L_M}$, $c \ll 1$ and $\beta \in [0, 1]$ denotes a ''forgetting factor''.

C. MAXIMUM SINR FILTERING & BIT RECOVERY

The received baseband signal vector in (8) can be rewritten as

$$\mathbf{y}_{ij}[n] = \mathbf{H}_{ij}\mathbf{s}_{ij}b_i[n] + \mathbf{d}[n], \quad n = 0, \cdots, N-1 \quad (12)$$

where $\mathbf{H}_{ij}$ is the multipath channel fading matrix between transmitter i and receiver j given by

$$\mathbf{H}_{ij} \triangleq \sum_{m=0}^{M-1} \tilde{h}_{ij}[m]\begin{bmatrix}\mathbf{0}_{m\times L}\\ \mathbf{I}_{L\times L}\\ \mathbf{0}_{(M-m-1)\times L}\end{bmatrix} \in \mathbb{C}^{L_M \times L}. \quad (13)$$

Assuming knowledge of the true autocorrelation matrix $\mathbf{R}_{\mathrm{I+N}}$ and multipath channel coefficients $\mathbf{H}_{ij}$, the linear filter at the receiver that exhibits maximum output SINR [60] is given by

$$\mathbf{w}_{\max-\mathrm{SINR}}(\mathbf{s}_{ij}) \triangleq \frac{\mathbf{R}^{-1}_{\mathrm{I+N}}\mathbf{H}_{ij}\mathbf{s}_{ij}}{\mathbf{s}^T_{ij}\mathbf{H}^H_{ij}\mathbf{R}^{-1}_{\mathrm{I+N}}\mathbf{H}_{ij}\mathbf{s}_{ij}} \quad (14)$$

and will attain an SINR at the output of the filter equal to

$$\mathrm{SINR}_{ij}(\mathbf{s}_{ij}) \triangleq \mathbf{s}^T_{ij}\mathbf{H}^H_{ij}\mathbf{R}^{-1}_{\mathrm{I+N}}\mathbf{H}_{ij}\mathbf{s}_{ij} = \mathbf{s}^T_{ij}\mathbf{P}_{ij}\mathbf{s}_{ij} \quad (15)$$

where $\mathbf{P}_{ij} \triangleq \mathbf{H}^H_{ij}\mathbf{R}^{-1}_{\mathrm{I+N}}\mathbf{H}_{ij} \succ 0$. We utilize the knowledge of access code training symbols at the receiver to estimate the SINR at the output of the maximum-SINR filter. More specifically, we use the error vector magnitude (EVM) of the soft-decoded received symbols given by

$$\mathrm{EVM}_{ij} = \sqrt{\frac{\frac{1}{N_{ac}}\sum_{n=0}^{N_{ac}-1}\left|\mathbf{w}^H_{\max-\mathrm{SINR}}(\mathbf{s}_{ij})\mathbf{y}_{ij}[n] - b_i[n]\right|^2}{\frac{1}{N_{ac}}\sum_{n=0}^{N_{ac}-1}|b_i[n]|^2}}. \quad (16)$$

Consequently, a quality estimate of SINR for link (i, j) is given by $\widehat{\mathrm{SINR}}_{ij} \triangleq 1/\mathrm{EVM}^2_{ij}$. Finally, the receiver detects the k-th transmitted symbol by minimizing the Euclidean distance between the maximum-SINR filtered signal and the transmitted symbols as follows

$$\widehat{b}_i[n] \triangleq \underset{b_i[n]\in\mathcal{A}}{\mathrm{argmin}}\left|\mathbf{w}^H_{\max-\mathrm{SINR}}(\mathbf{s}_{ij})\mathbf{y}_{ij}[n] - b_i[n]\right|^2 \quad \text{for} \quad n = 0, \ldots, N-1. \quad (17)$$

D. WAVEFORM OPTIMIZATION

At the j-th receiver of each cognitive link we focus on the design of a binary waveform $\mathbf{s}$ that maximizes SINR($\mathbf{s}$) at the output of the maximum-SINR filter

$$\mathbf{s}_{opt} \triangleq \underset{\mathbf{s}_{ij}\in\left\{\pm\frac{1}{\sqrt{L}}\right\}^L}{\mathrm{argmax}}\ \mathrm{SINR}_{ij}(\mathbf{s}_{ij}) = \underset{\mathbf{s}_{ij}\in\left\{\pm\frac{1}{\sqrt{L}}\right\}^L}{\mathrm{argmax}}\ \mathbf{s}^T_{ij}\mathbf{P}_{ij}\mathbf{s}_{ij}. \quad (18)$$

The problem of obtaining the optimal waveform for link (i, j) is NP-hard [61], [62] and can be solved through exhaustive search over all possible L-waveform-bit combinations. In the next two subsections we present two computationally efficient methods for optimizing waveform $\mathbf{s}$ for link (i, j).

1) RANK-1 WAVEFORM DESIGN

Given the eigendecomposition of $\mathbf{P}_{ij}$, where $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_L$ are the eigenvectors, and $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_L > 0$ are the corresponding eigenvalues of $\mathbf{P}_{ij}$, the optimization problem in (18) is rewritten as

$$\mathbf{s}_{opt} \triangleq \underset{\mathbf{s}_{ij}\in\left\{\pm\frac{1}{\sqrt{L}}\right\}^L}{\mathrm{argmax}}\left\{\sum_{i=1}^{L}\lambda_i\left\|\mathbf{s}_{ij}{}^T\mathbf{q}_i\right\|^2\right\} \quad (19)$$

where $0 \leq \left\|\mathbf{s}_{ij}{}^T\mathbf{q}_i\right\|^2 \leq 1$, for $i = 1, 2, \cdots, L$. If we simplify the problem by keeping only the strongest term $\lambda_1\left\|\mathbf{s}^T_{ij}\mathbf{q}_1\right\|^2$, we obtain the rank-1 optimized waveform

$$\widehat{\mathbf{s}}_{opt} \triangleq \underset{\mathbf{s}_{ij}\in\left\{\pm\frac{1}{\sqrt{L}}\right\}^L}{\mathrm{argmax}}\left\{\left\|\mathbf{s}^T_{ij}\mathbf{q}_1\right\|^2\right\} = \pm\frac{1}{\sqrt{L}}\mathrm{sgn}\left(\Re\left\{\mathbf{q}_1\right\}\right) \quad (20)$$

where sgn$(\cdot)$ denotes the sign operator. Hence, a rank-1 optimal waveform is acquired by first relaxing the binary constraint in the waveform optimization problem in (18) and solving optimally the relaxed problem. Finally, we simply map (quantize) the minimum-eigenvalue eigenvector of $\mathbf{P}_{ij}$ to the binary field, by taking the sign of its components [29], [63].

2) SINGLE-BIT FLIPPING (SBF) WAVEFORM DESIGN

A second computationally efficient algorithm for finding a near-optimal solution to the binary code-waveform that maximizes SINR at the output of the maximum-SINR filter is based on iterative single-bit flipping (SBF). Similar algorithms have been studied in L_1-subspace decomposition [64] and channel coding [65] literature. The steps of the SBF algorithm are described in detail in Fig. 5.

Single-bit Flipping (SBF) Waveform Design Algorithm

1: $p := 0$; input $\mathbf{P}_{ij}$ and $\mathbf{s}_{ij}^{(0)} \in \left\{\pm 1/\sqrt{L}\right\}^L$
2: $p := p + 1$;
3: $\alpha^{(p)}[k] \leftarrow s_{ij}^{(p)}[k]\left(\sum_{l\neq k} s_{ij}^{(p)}[l]\Re\left\{[\mathbf{P}_{ij}]_{k,l}\right\}\right)$
4: $(v, q) \leftarrow \min(\boldsymbol{\alpha}^{(p)})$
5: **if** $v < 0$ **then**
6: $s_{ij}^{(p)}[q] \leftarrow -s_{ij}^{(p)}[q]$
7: **else**
8: return $\widehat{\mathbf{s}}_{opt} = \mathbf{s}_{ij}^{(p)}$

FIGURE 5. Single-bit flipping (SBF) waveform design algorithm.

More specifically, SINR at the output of the maximum-SINR filter can be written as

$$\begin{aligned}\mathbf{s}_{ij}^{(p)^T}\mathbf{P}_{ij}\mathbf{s}_{ij}^{(p)} &= \mathbf{s}_{ij}^{(p)^T}\left(\Re\left\{\mathbf{P}_{ij}\right\} + \mathrm{j}\Im\left\{\mathbf{P}_{ij}\right\}\right)\mathbf{s}_{ij}^{(p)} \\ &= \mathbf{s}_{ij}^{(p)^T}\Re\left\{\mathbf{P}_{ij}\right\}\mathbf{s}_{ij}^{(p)} = \frac{1}{L}\mathrm{Tr}\left(\Re\left\{\mathbf{P}_{ij}\right\}\right) \\ &\quad + \sum_k 2s_{ij}^{(p)}[k]\left(\sum_{l>k} s_{ij}^{(p)}[l]\Re\left\{[\mathbf{P}_{ij}]_{k,l}\right\}\right) \quad (21)\end{aligned}$$

where $\Re\{\cdot\}$ and $\Im\{\cdot\}$ denote the real and complex part of a vector or matrix, respectively. By changing the sign of the k-th waveform bit $s_{ij}^{(p)}[k]$ in the p-th iteration of the SBF algorithm, the post-filtering SINR value will change by

$$\alpha^{(p)}[k] \triangleq \pm 4 s_{ij}^{(p)}[k]\left(\sum_{l\neq k} s_{ij}^{(p)}[l]\Re\left\{[\mathbf{P}_{ij}]_{k,l}\right\}\right). \quad (22)$$

Consequently, if the result in (22) is negative, then flipping the k-th code-waveform-bit from $s_{ij}^{(p)}[k]$ to $-s_{ij}^{(p)}[k]$ will increase the post-filtering SINR value in (21). Obviously, flipping the waveform-bit with the highest negative contribution will offer the biggest SINR increase. On the other hand, if there is no bit flip that will increase the SINR value in (21), then the SBF algorithm terminates.

Fig. 6 compares the post-filtering SINR loss of a randomly generated binary code-waveform to a rank-1 optimal and an SBF-optimized binary waveform as a function of the number of interferers. Particularly, SINR loss is evaluated with respect to the SINR of the optimal binary waveform $\mathbf{s}_{opt}$ that is obtained through exhaustive search over all possible L waveform-bit combinations. Additionally, bit error rate (BER) versus SNR for the user of interest is depicted for the different waveform optimization techniques and fixed number of interferers equal to the code-waveform length $L = 8$. In both simulations, the SNR for the user of interest is fixed to 8 dB, while the SNRs of the interferers are uniformly spaced between 8 and 11 dB.

E. MEDIUM ACCESS CONTROL

Each node accesses the channel using a random-access ALOHA-like CDMA-based MAC scheme for each data frame. Waveforms at each cognitive link are dynamically optimized on a hop-by-hop basis based on the algorithm described in section IV-D. Baseline network links are assigned to unique code-waveforms that remain fixed. The proposed protocol is efficient both in environments where carrier sensing cannot resolve the hidden/exposed terminal problem and in networks with long-distance communication, high data rates, and short frames. Additionally, no handshaking mechanism for virtual carrier sensing is utilized, thus powerful MAI from asynchronous transmissions is resolved with retransmissions of data frames that are not successfully acknowledged by the receiver.

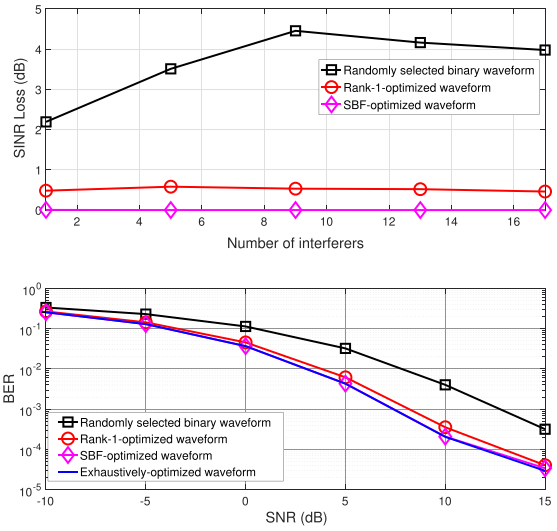


FIGURE 6. Post-filtering SINR loss of a randomly selected, a rank-1, and a SBF-optimized waveform versus the number of interferers (top). BER versus SNR of the user of interest for different waveform optimization techniques (bottom).

At the MAC layer, a data frame is received successfully if the number of errors in the decoded information bits is less than or equal to the maximum number of (correctable) bit errors allowed by the forward-error-correction (FEC) module. If no FEC is present, then the data frame is successfully received when all information bits are detected correctly. Subsequently, if the decoded data frame is detected correctly, an acknowledgment (ACK) frame is sent to the corresponding transmitter. Alternatively, if the received data frame contains errors, the transmitter is notified for a retransmission with a negative acknowledgment (NACK) frame. Both ACK and NACK frames contain queue size information to allow transmitting nodes to make routing decisions. Fig. 7 describes the structure of the ACK/NACK frames and the size of each data field in bytes.

F. NETWORKING & APPLICATION LAYERS

Each network node continuously checks the queue that is maintained for each session d at the NET layer for either existing (if it is a source node) or new incoming (if it is an intermediate node) frames. Subsequently, each transmitting

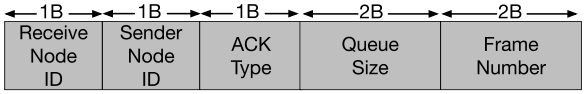


FIGURE 7. Structure of acknowledgment (ACK/NACK) frames.

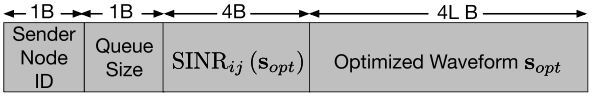


FIGURE 8. Feedback/control frame structure.

(source or intermediate) node in the cognitive network calculates utility values U_{ij} for each feasible next hop based on queuing information and spectrum dynamics that are received from acknowledgment and feedback/control packets (discussed in section IV-G). Cognitive network nodes with higher utility U_{ij} will have higher probability to be selected for transmission. If the destination is in the transmission range of the transmitting (either source or intermediate) cognitive network node, the differential backlog between the transmitter and the destination is no less than the differential backlogs between the transmitter and any other nodes, because the queue length of the destination is zero. As a result, lightly backlogged nodes (with smaller differential backlog) and higher link capacity are selected as intermediate relays and receive most of the network traffic. On the other hand, network nodes in the baseline network consider fixed route and waveform allocation.

The proposed network node architecture can accommodate end-to-end multimedia sessions, such as digital audio and video sessions from source s to destination d. Destination network nodes implement a double-buffering architecture that enables reordering of chunks of received data frames. Consequently, the nodes compensate for out-of-order frame arrivals that may appear due to dynamically selected multipath routes toward the destination network node.

G. FEEDBACK CHANNEL

Cross-layer dynamic waveform and route adaptation is implemented in practice based on a closed-loop feedback/control link that carries the optimized waveform and post-filtering link SINRs to the transmitting (source or intermediate) cognitive network nodes. The set of feasible next hops at each cognitive network node can be obtained by either neighbor discovery protocols or from position information (e.g. GPS) of each node. Receiving (intermediate or destination) nodes, that are candidates to be selected as next hops, send feedback/control frames to an associated transmitting node at time intervals that are either preprogrammed by the user or calculated in an autonomous fashion. The structure of the feedback link frames is illustrated in Fig. 8. Transmitting nodes can take decisions on the modulating waveform and next hop based on locally collected utility information from neighboring nodes according to the algorithm described in Fig. 2. Finally, waveform and routing decisions taken at each transmitting node are broadcasted to the rest of the network. A flow chart representation of the process of handling data and feedback frames in cognitive network nodes is given in Fig. 9.

V. HARDWARE AND SOFTWARE PROCESSING

In section V we discuss hardware and software processing challenges for the implementation and deployment of our algorithmic developments in a large-scale cross-layer-reconfigurable software-defined radio testbed.

A. COGNITIVE RADIO HARDWARE ARCHITECTURE

Software-defined radio technology proposes a paradigm shift from inherently inflexible hardware-radio platforms by combining analog static (front-end) circuits and digital hardware (back-end) that are easily reconfigurable via software updates. SDR-based architectures are therefore ideal for rapid prototyping and testing of new applications and commercial standards.

A generic SDR front-end contains analog circuitry for signal up/down-conversion from baseband to a desired center or intermediate frequency (IF) as well as radio-frequency (RF) amplifiers and passband filters for signal conditioning. Baseband signal processing is performed digitally at the SDR back-end [1]. To leverage the functionalities of a typical heterogeneous hardware architecture, execution of different algorithms may be split between a field-programmable gate array (FPGA) and a general-purpose processor (GPP), generally connected through a high-speed data bus.

High parallelism offered by the FPGA enables acceleration of computationally intensive signal processing functionalities, such as filtering. On the other hand, development

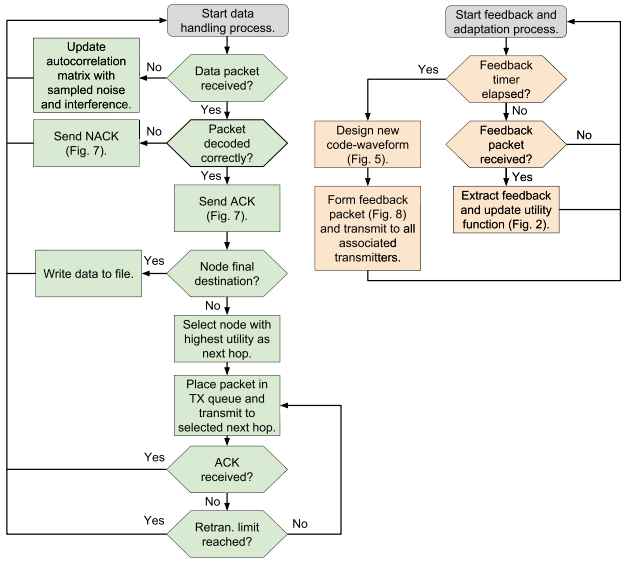


FIGURE 9. Flow chart showing the process of data and feedback reception and cognitive functionalities executed at each receiver node.

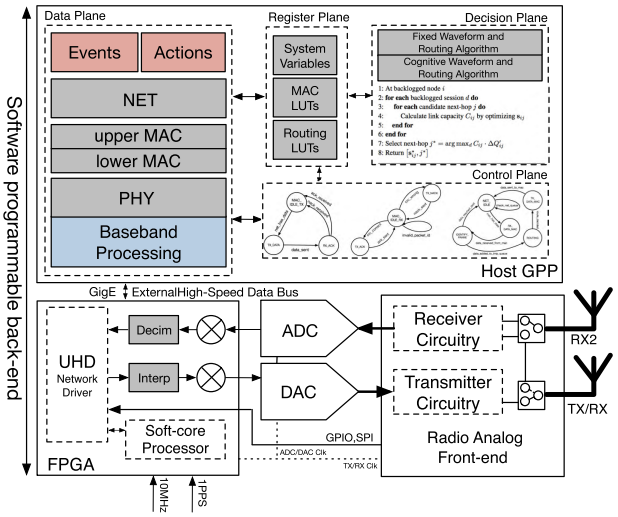


FIGURE 10. Hardware and software architecture of the proposed cognitive software-radio platform.

and implementation of signal processing operations on GPPs generally involves shorter development cycles. GPPs are also well-suited for the implementation of highly branching algorithms. We facilitate rapid development, prototyping, and testing of the proposed cognitive channelization and routing algorithm by implementing signal processing and upper-layer networking in a GPP-based SDR. Basic digital upsampling, downsampling, and signal filtering at IF are implemented in the FPGA (as depicted in Fig. 10).

B. SOFTWARE RADIO RECONFIGURABLE FRAMEWORK

Software implementation of our algorithmic developments described in the sections above are hosted by a software-radio framework that runs entirely in a GPP and enables self-optimization and real-time reconfiguration at PHY, MAC, and NET layers of the network protocol stack. In its current instantiation, the framework is implemented in a host-PC and takes advantage of the modularity and flexibility of the GNU Radio open-source framework and the multi-core processing libraries supported by Python.

1) PHY-LAYER BASEBAND PROCESSING

We utilize the GNU Radio open-source software application-programming interface (API) to implement new signal processing blocks and applications in C++ and Python. A collection of connected blocks with a particular flow of samples from a source towards a sink is called flowgraph. A great advantage of GNU Radio is that apart from real-time signal processing of samples from an SDR, it can also be used for PHY-layer simulations. This is easily possible by looping back the generated sample stream into the receiver without interfacing actual radio hardware. The GNU Radio framework handles tasks such as memory allocation and sample transfer between signal processing blocks in a stream-based fashion. In this work, we rely on GNU Radio mechanisms such as stream tags and message ports to enable packet-based processing at each network node.

Stream tags offer the ability to align metadata with specific samples in a stream to denote information such as packet boundaries. A more generic asynchronous message passing mechanism uses protocol data units (PDUs) to transfer an arbitrarily large chunk of data and metadata between any set of GNU Radio modular blocks with the use of message ports. Message ports follow a publisher/subscriber model where receive queues exist at message input ports.

Fig. 11 provides an abstract illustration of the transmit and receive flowgraphs in GNU Radio. Multiple instances of transmit and receive flowgraphs may run in parallel at each network node. User datagram protocol (UDP)-type message ports are utilized to exchange PHY-layer related information with the protocol execution logic of MAC and NET-layer functionalities. More specifically, upstream information to the MAC layer is comprised of payload bytes, post-filtering SINR estimates, and optimized code-waveform designs at the receiver flowgraph. Payload bytes, waveform updates, and user-driven decisions for waveform adaptation are exchanged downstream from MAC to PHY-layer blocks at the transmitter and receiver flowgraphs.

NET-layer data frames are first inserted to the transmitter flowgraph in the form of PDUs. Subsequently, data frames are converted into stream-based samples and are appended with MAC and PHY-layer headers as described in section IV-B. Prior to streaming samples to the digital-to-analog converter (DAC) and the SDR front-end device, the samples undergo continuous functions such as waveform modulation and pulse-shape filtering. Stream tags are used to denote the beginning and the end of a data frame.

The receiver implements frame detection as a continuous streaming block that operates on the stream of the incoming samples from the analog-to-digital converter (ADC) of the SDR. Detected frames are formatted as PDUs and are passed through waveform-level synchronization, channel estimation, maximum-SINR filtering, and minimum-distance detection operations. If no data frame is detected, compound disturbance (noise and interference) samples are used to maintain an up-to-date estimate of the inverse autocorrelation matrix (as discussed in section IV-B). At regular time intervals a (user-defined or autonomous) decision mechanism at the NET layer triggers the waveform optimization process. The optimized waveform and post-filtering SINR values are communicated to the upper layers to create the feedback/control link frame. The optimized waveform is fed back to both the transmitter and receiver PHY-layer in the form of a PDU to update waveform-dependent PHY-layer signal processing blocks in GNU Radio.

2) CROSS-LAYER PROCESSING

The software architecture of each network node is divided into four interacting planes: *decision*, *control*, *data*, and *register* [66] that enable separation of data processing, logical control, and decision-making mechanisms from the execution

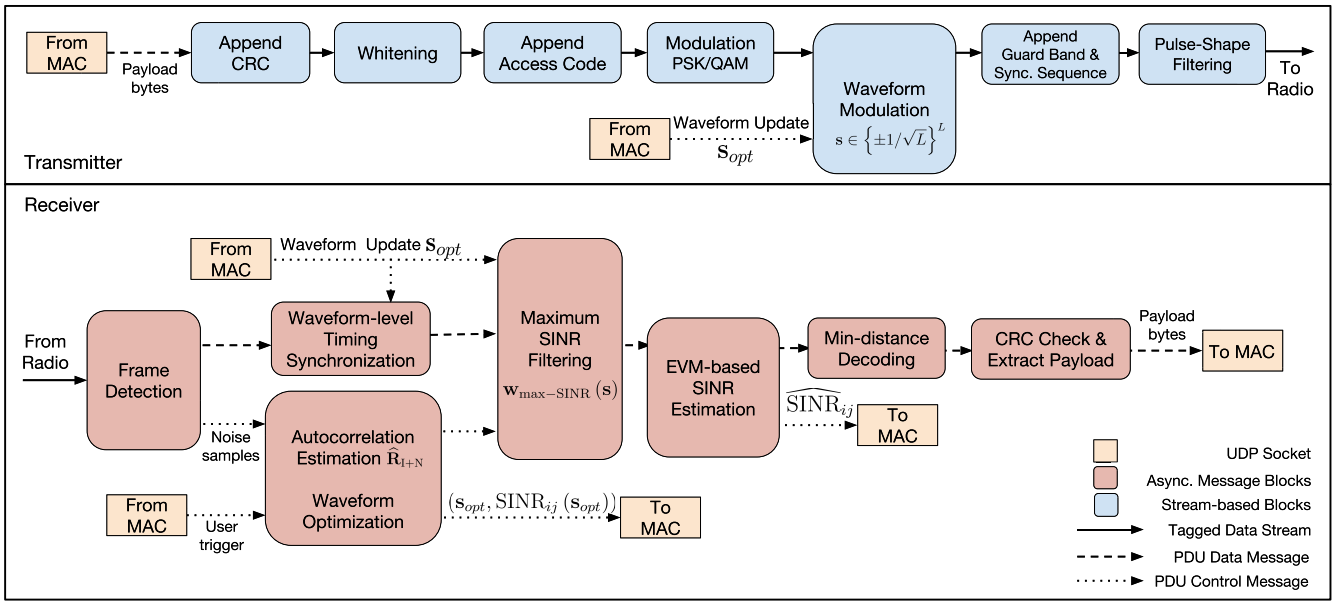


FIGURE 11. PHY- and lower-MAC (transmitter and receiver) signal processing blocks in GNU Radio. Both blocks and connections to radio front-end(s) and upper networking layers (i.e. MAC) are the same for both static and cognitive network nodes.

of the protocol stack. Fig. 10 depicts the software/hardware architecture of the proposed cognitive radio platform. Data processing and handling of data and control queues takes place in the *data* plane. The execution logic of data processing in the *data* plane is controlled by three execution engines that are defined in the *control* plane. Execution engines contain finite state machines (FSMs) that implement the execution logic of the transmit and receive MAC and NET-layer protocols. Fig. 12 depicts the three FSM designs comprised of symbolic states and extended state transitions that are defined as a triplet of events, conditions, and actions [66]. Actions are designed to trigger *data* plane functionalities, while events occur as a result of *data* plane actions. Actions and events (as depicted in the FSM state-transitions of Fig. 12) are primitive building blocks that can be used to decompose a wide range of wireless networking protocols. As a result, the execution of the protocol stack in the *data* plane can be easily reconfigured by simply changing the order of execution of the blocks of actions and events. The *register* plane stores and manages access to system parameters and state variables at different network layers, look-up-tables (LUTs) for the implementation of different MAC and routing protocols, and control messages such as queue updates from next-hop network nodes.

The proposed software-radio framework is implemented in a host GPP using Python's multi-processing library to allow multiple processes to run in parallel by distributing processing across multiple computing resources. Queues are used to buffer and transfer data bytes between processes in a thread-safe manner. The framework is programmed in a modular manner such that multiple instances of each plane can be generated and run in parallel to handle data that may arrive from multiple network nodes. Intermediate network nodes must be capable of processing frames that may arrive from more than one neighbor network nodes at the same time. In GNU Radio separate instances of transmit and receive signal processing flowgraphs are initialized for communication with candidate next-hop neighbor nodes. Similarly, multiple instances of the *decision*, *control*, *data*, and *register* planes are initialized and process frames received from different sources in an independent fashion. In the *data* plane a dedicated handler is initialized for each UDP socket to control communication between lower-layer baseband signal processing blocks in GNU Radio and the rest of the *data* plane blocks. Streams of transmitted/received bytes are placed in a queue and handled by *data* plane/GNU Radio processes. In the same way, asynchronous waveform updates and post-filtering SINR messages are communicated to the upper-layers of the network protocol stack via dedicated handlers for UDP sockets.

3) DISTRIBUTED DECISION MAKING

The proposed software-radio framework incorporates a *decision* plane that enables on-the-fly reconfiguration of user-defined decision algorithms without affecting the execution of the network protocol stack. The *decision* plane is interfaced with the *control* and *data* planes through the *register* plane and contains a set of decision algorithms for fixed and cognitive waveform and routing allocation that can be executed in parallel. Algorithms can be executed in a synchronous or asynchronous fashion and modify parameters across PHY, MAC, and NET layers of the network protocol stack without influencing the on-going protocol execution logic. Both *control* and *data* planes can reconfigure their

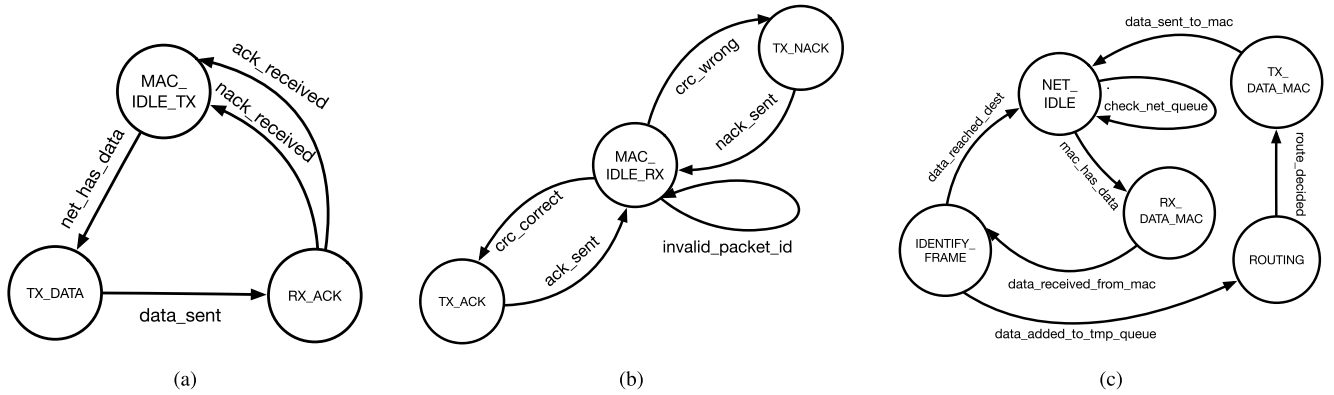


FIGURE 12. Upper MAC- and NET-layer FSM architectures executed at the each network node. State transitions are executed based on the definition of events that are depicted in the arches of the FSMs. (a) Upper MAC-layer transmit FSM. (b) Upper MAC-layer receive FSM. (c) NET-layer FSM.

execution logic by checking the updated parameters in the *register* plane. The *decision* plane is optional upon design therefore, it may be enabled or disabled to accommodate centralized or distributed control of cross-layer optimized parameters.

4) COMPUTATIONAL PERFORMANCE

Improved computational performance through threaded operation and optimized instructions is essential for the software implementation of the proposed cross-layer optimized transceiver that is entirely hosted by a GPP. Before we started this project, there was no transmitter/receiver design available in GNU Radio that implements adaptive code-waveform-based multiple access. Therefore, it was unclear whether it was possible to realize and execute the proposed cross-layer optimized transceiver design in real-time on a normal laptop or desktop PC or if splitting functionalities between the GPP and the FPGA would be required.

For small signal bandwidths, we are able to deal with transmitted/received streams of data in real-time and exploit the inherent capabilities of GNU Radio signal processing blocks to distribute the load between the cores of a GPP. Real-time processing implies that the average processing time per sample is smaller than the sample duration, thus the transceiver does not drop samples, which may lead to wrong interpretations of measurement results. To further speed up computations, the individual blocks exploit vectorized instructions that multicore CPUs provide through single-input multiple-data (SIMD) extensions like MMX, SSE, and AVX. With GNU Radio, these instructions are accessed through the vector-optimized library of kernels (VOLK) [1], [67], which provides optimized implementations of common signal processing functions.

5) SCALABILITY/INTEROPERABILITY

The architectural components of the proposed software-radio framework enable either network-level simulations by re-directing the generated NET and MAC-layer frames to a UDP/TCP socket instead of GNU Radio transmitter/receiver or PHY-level simulations by looping back the generated sample stream in GNU Radio into the receiver without interfacing an actual SDR hardware. GNU Radio aids simulation setup by providing models for hardware impairments like phase noise and clock drift, as well as propagation environments like AWGN, Rayleigh, and multipath fading. Interoperability tests are conducted with different SDR platforms and GPP architectures (such as Intel i5/i7, and ARM Cortex-A15). Table 3 summarizes the SDR platforms and host-PC models and features that are utilized for both simulation and experimental testing.

VI. THE "ELASTIC" TESTBED IMPLEMENTATION

In this section, we identify and discuss system implementation details for the deployment of an "elastic" network testbed comprising of co-located terrestrial cognitive and baseline network nodes that operate in the presence of an airborne RF interferer. Fig. 13(a) presents a testbed schematic that depicts data command-and-control and clock-synchronization connections in the software-defined radio testbed. Fig. 13(b) provides illustrations of the testbed hardware units, which include terrestrial and airborne modular SDR, clock generation/distribution, and host-based visualization platforms. Table 3 lists the selected hardware platforms with further details.

A. SDR PLATFORMS

Both cognitive and baseline network nodes in the "elastic" network are based on the Ettus Research family of Universal Software Radio Peripherals (USRPs). USRP SDRs are connected to a Linux-based host-PC with either an external high-speed Gigabit Ethernet (GigE) or USB 3.0 data interface. All models of the USRP family can interface to the proposed cognitive radio framework through the USRP Hardware Driver (UHD) software API that acts as a host communication driver for controlling the SDRs.

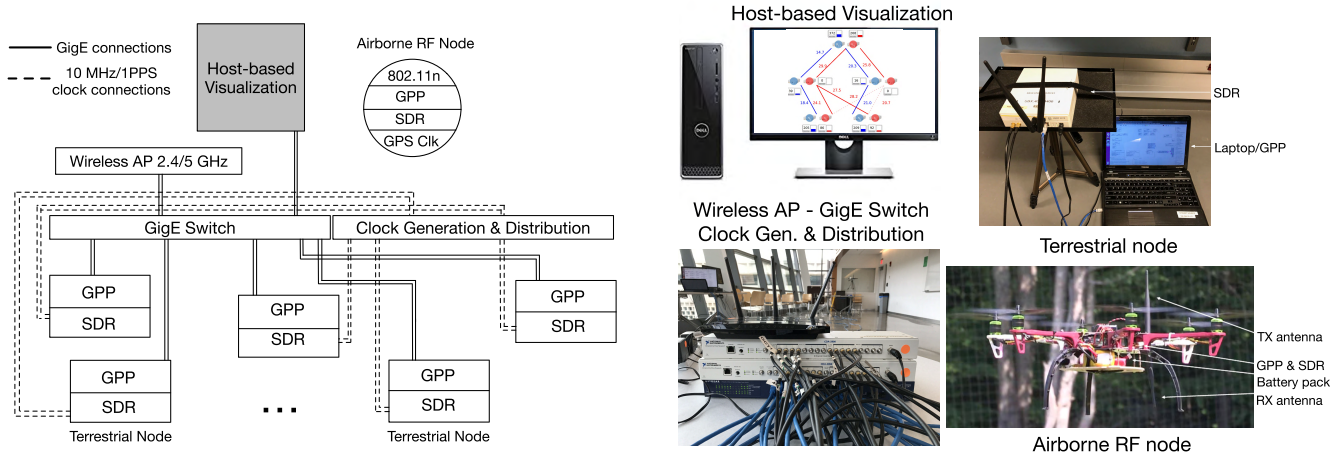


FIGURE 13. Testbed implementation overview: (a) Data command-and-control GigE links and clock distribution connections for both terrestrial and airborne SDR platforms. (b) Assembled off-the-shelf airborne and terrestrial SDRs with data command-and-control, clock generation/distribution, and host-based visualization units.

TABLE 3. Selected hardware components.

Type	Model	Features
Terrestrial SDR	Ettus USRP N210 & SBX-40	1 Xilinx Spartan-3A-DSP FPGA 100 MSps ADC, 400 MSps DAC GigE connectivity 1 TX/RX and 1 RX RF front-ends 40 MHz bandwidth tunable from 400 MHz to 4400 MHz
Host Computer	Toshiba Satellite/Dell Optiplex	Laptop/Desktop PCs with Intel i5/i7 Processor 4 up to 8 core CPUs, 8 up to 16 GB RAM Software configuration: Ubuntu 16.04, GNU Radio 3.7.10, UHD 3.11
Airborne SDR	Ettus USRP B210	1 Xilinx Spartan-6 XC6SLX150 FPGA 61.44 MSps ADC/DAC USB 3.0 connectivity 2 TX and 2 RX RF front ends 56 MHz RF bandwidth tunable from 70 MHz to 6.0 GHz
Airborne Host Computer	Hardkernel ODROID-XU4	ARM Cortex-A15, 2 GHz & A7 Octacore CPUs 2 GB LPDDR3 RAM PoP stacked 16 GB eMMC5.0 HS400 flash storage Software configuration: Ubuntu 16.04, GNU Radio 3.7.10, UHD 3.11
Ref. Clock Generation & Distribution	Ettus OctoClock-G CDA-2990	GPS-disciplined fully integrated timing source 8-way distribution (10 MHz/1 PPS)
Ref. Clock Distribution	Ettus OctoClock CDA-2990	8-channel time and frequency distribution circuit Requires external 10 MHz/1 PPS reference source
Switch	Netgear	24-port Gigabit Ethernet Switch
Wireless Access Point	TP-Link Panda	Dual-band 2.4/5 GHz TP-Link AC 1200 access point Dual-band 2.4/5 GHz Panda N 600 USB dongles
Airborne Vehicle Platform	Air Force Research Lab (Rome, NY)	Custom-built hex-rotor with 550 frame Requires 11200 mAh battery pack Flight time: 18 min with 2 kg payload

1) TERRESTRIAL NODES

Terrestrial nodes are based on the USRP N-series SDRs. Each SDR contains a 14-bit dual 100 MSps ADC, a 16-bit dual 400 MSps DAC, a Xilinx 3A-DSP FPGA-based motherboard, and a 40 MHz bandwidth daughterboard that supports a variable center frequency from 400 MHz to 4.4 GHz and TX power of 20 dBm. Received baseband samples at the SDR are sent to the host PC via a GigE data bus in the format

of 4 bytes per complex sample, therefore the maximum data rate over GigE is $\frac{125\ \text{MB/s}}{4\ \text{B/sample}} = 31.25$ MSps (reduced to 25 MSps due to overhead [1]).

2) AIRBORNE NODES

Airborne network nodes require a more compact and lightweight SDR platform. Therefore, airborne nodes are equipped with USRP B-series SDRs that offer a small-form-factor, fully integrated, two-channel radio with continuous RF coverage from 70 MHz to 6 GHz. Baseband samples from the SDR are transferred to an ARM-based single-board computer through a USB 3.0 connection. Both the radio and host-PC offer a combined payload of less than 2 kg that can be easily mounted to the airborne vehicle platform (Table 3) and powered by a compact battery pack.

B. TESTBED OPERATION

1) COMMAND & CONTROL

Portable laptop PCs are used to control terrestrial SDRs. Laptops are positioned in a central location (i.e. command-and-control center) and control the SDRs through a GigE switch. The GigE switch is used to route ACK/NACK frames and feedback/control frames between network nodes. A dedicated desktop PC is utilized to store and display status-update messages from the ''elastic'' network nodes which are transmitted periodically to the desktop PC through the GigE switch. During network deployment and testing it is also important to remotely control the SDR on the airborne interfering node. Particularly, we should be able to start, stop, and reconfigure PHY-layer parameters such as the code-waveform and transmission power during flight. Command and control of the SDR on the airborne node is achieved by remote access to the airborne host-PC over an 802.11n-based wireless control link. The wireless control link is established with a wireless access point on the ground and a USB-dongle on the airborne platform. We select to operate the wireless control link at 5 GHz to avoid interference with the flight controller.

2) HOST-BASED VISUALIZATION

Cognitive and baseline network nodes can periodically send status-update messages to a desktop PC to report modifications in parameters such as post-filtering link SINR, frame queue sizes, optimized code-waveforms, and selected routes. Status-update messages are displayed in real-time on a host-based visualization that is built in Python. Display messages are time-stamped and stored at the desktop PC for post-processing offline analysis. A screenshot of the host-based visualization for the ''elastic'' network testbed in an outdoor environment is depicted in Fig. 18(c). A bar graph next to each network node in the visualization demonstrates the number of data frames waiting for transmission in the network node queue. On the other hand, as the destination node does not have a network queue, the bar graph depicts the total number of packets that have been correctly received from the source of each session. Candidate routes in the cognitive network are depicted as dashed lines, while active routes for each session are displayed as solid lines.

C. SYNCHRONIZATION

To compensate for time and frequency clock drifts between terrestrial SDRs we utilize a GPS-disciplined clock generation and distribution circuit. Particularly, one GPS-disciplined Ettus Research OctoClock module (as depicted in Fig. 13) generates and distributes high accuracy 10 MHz and 1 PPS signals to seven SDR nodes using SMA cables. The eighth output of the OctoClock is utilized as an external reference to a second OctoClock unit which can distribute clock reference signals to eight additional SDRs. Reference clock signals can discipline SDRs to a frequency stability of 0.01 ppm. Airborne SDR nodes can optionally be equipped with on-board mounted GPS-disciplined clock units that offer 10 MHz clock and 1 PPS reference signals and can achieve a global timing alignment of 50 ns in locked condition. However in the context of this work, the airborne interfering node operates without any requirement of either time or frequency synchronization to the rest of the terrestrial network nodes.

VII. EXPERIMENTAL PROOF-OF-CONCEPT RESULTS

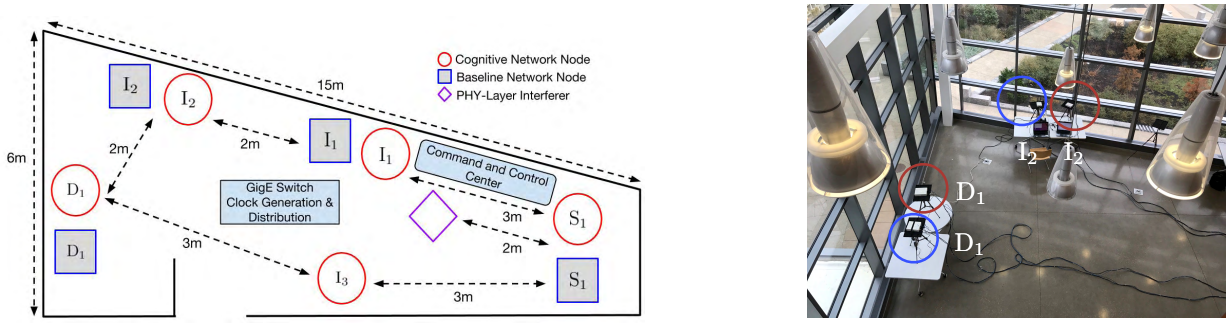
In this section we describe the design details of three experimental scenarios toward the validation of the ''elastic'' network concept in both indoor and outdoor environments. Each scenario addresses a challenging aspect of communication in UAV networks, as identified in the literature (see section I) but is also applicable to cognitive radio networks. We present experimental proof-of-concept results for both testbed deployments and evaluate the ''elastic'' testbed in terms of end-to-end network throughput performance, average post-filtering SINR at selected links, average aggregate queue size at the intermediate nodes, and selected routes.

A. SCENARIOS

In each scenario, we build two independent but co-located networks of terrestrial nodes. The first network operates on statically assigned waveforms and fixed routes. The second network consists of cognitively cross-layer optimized wireless links that are capable of dynamically adapting code-waveforms and routes, as discussed in section III-B. Multimedia traffic flows for both cognitive and baseline networks are carried over unicast sessions from a source node s to a destination node d. In both networks, we consider uncoded transmissions of multimedia data that are 4-QAM modulated. The rest of the data-link parameters can be found in Table 1.

1) NETWORK COEXISTENCE

The first testbed scenario evaluates the simultaneous operation of two independent wireless networks coexisting at



(a) (b)

FIGURE 14. (a) Indoor testbed deployment at the Department of Electrical Engineering, State University of New York at Buffalo. (b) One group of intermediate (I_2) and destination (D_1) nodes from baseline and cognitive networks.

the same location, time, and frequency. Lack of network-wide backhaul links and centralized common control in UAV networks [2] leads to distributed self-reconfigurable solutions to avoid multiple-access interference. We consider fixed initialization of baseline network nodes on a static set of code-waveforms and routes. Digital audio or video traffic sessions are carried over a layered network protocol stack. Baseline network operation is then disrupted by the deployment of a second network that is co-located in space, time, and frequency with the baseline network and is initialized to the same code-waveforms and routes. Multiple-access interference between the co-located wireless links leads to increased error rates in the received data frames, thus increases queue backlogs at both the source and intermediate network nodes. Data frames may be dropped completely if the maximum number of re-transmissions is reached by the transmitting nodes.

Our objective in this scenario is to eliminate destructive interference and improve aggregate network throughput performance by jointly optimizing code-waveforms and routes in the cognitive network. Particularly, the *decision* plane in the cognitive network nodes dynamically switches to the execution of the cross-layer optimization algorithm (as described in Fig. 2) upon the arrival of an external trigger (coexist mode).

2) NET-LAYER INTERFERENCE

The second testbed scenario considers network congestion created by intentionally flooding the queue of an intermediate network node in both cognitive and baseline networks. Resilient airborne network operation is achieved by autonomous route reconfiguration to bypass network disruption and successfully deliver information messages to the intended destination [3]. Random data frames, which mimic the frame format of cognitive and baseline data links, are generated but will not be correctly received and decoded. The random frames are injected in both networks at the same time. Cross-layer adaptation in the cognitive network is expected to balance traffic loads at the intermediate network nodes by jointly optimizing code-waveforms and selecting routes that maximize local utility values U_{ij} between transmitter i and receiver j. As a result, multimedia sessions in the cognitive network are expected to withstand NET-layer interference by selecting to reroute traffic around the heavily backlogged intermediate nodes. However, the same decision logic is not adopted by the baseline network nodes, therefore network throughput performance is expected to degrade significantly.

3) PHY-LAYER INTERFERENCE

Finally, the third testbed scenario considers disruption of baseline and cognitive network operation due to RF interference from either a mobile airborne or terrestrial node. UAV networks encounter unpredictable location-dependent interference [13] and should be capable of autonomously adapting network parameters across multiple layers to maintain link quality and network connectivity. Particularly, we introduce a wideband RF interferer that transmits data frames modulated by the same (heavily correlated) code-waveform that is already utilized by other cognitive and baseline links in the testbed. An airborne RF interferer is employed during outdoor testing, while a terrestrial RF interferer is used for indoor tests. Based on simulation results for code-waveform optimization (presented in section IV-D), cognitive network links are expected to withstand PHY-layer interference by jointly designing maximum-SINR code-waveforms and selecting routes that maximize network throughput. Depending on the power and location configuration of the RF interferer, baseline links become heavily backlogged due to destructive RF interference, thus baseline network throughput performance is expected to degrade notably.

B. INDOOR EXPERIMENTS

Indoor tests are conducted at the Department of Electrical Engineering at the State University of New York (SUNY) at Buffalo campus. The testbed is set up in a room of approximately 65 m^2 as depicted in Fig. 14. Nine nodes are used to build two co-located baseline and cognitive networks. Video traffic is carried over a single unicast session from one source

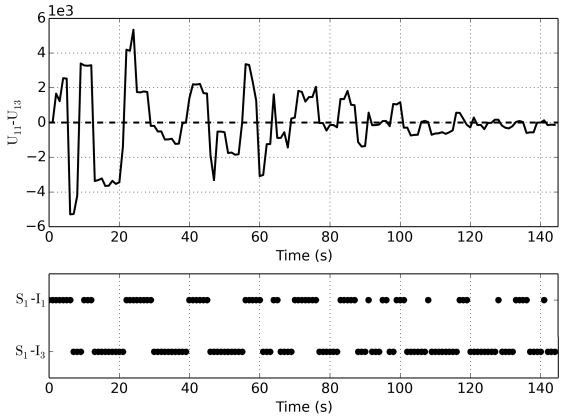


FIGURE 15. Difference of locally calculated utilities for cognitive links S_1–I_1 and S_1–I_3 (top) and corresponding routing decisions (bottom) at source node S_1 for the network coexistence scenario with coexist mode on.

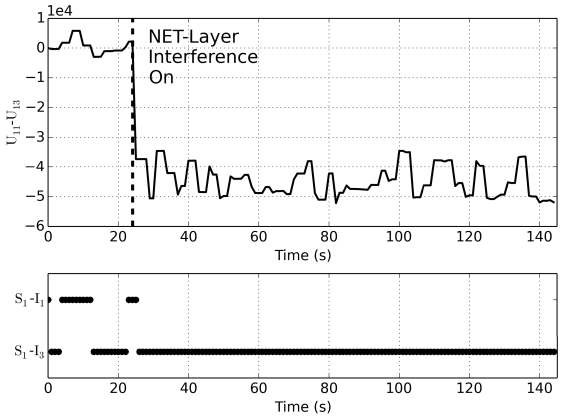


FIGURE 16. Difference of locally calculated utilities for cognitive links S_1–I_1 and S_1–I_3 (top) and corresponding routing decisions (bottom) at source node S_1 before and after NET-layer interference.

network node S_1 to a destination node D_1. Baseline network traffic can be routed only through intermediate nodes I_1 and I_2 (i.e. 2-hops from the destination). Video traffic in the cognitive network is offered two additional 1-hop routes to the destination either through intermediate node I_3 or intermediate node I_1. Consequently, the cognitive network is capable of routing video frames to the destination through multiple paths at the same time (e.g. destination D_1 can receive from either node I_1 or I_2 and I_3 at the same time). A tenth node in the testbed plays the role of an RF interferer that is preconfigured to disrupt operation of the S_1–I_1 link.

1) NETWORK COEXISTENCE

Fig. 15 depicts locally calculated utilities for cognitive links S_1–I_1 and S_1–I_3 and the corresponding routing decisions at source node S_1 for the network coexistence scenario. Utility values U_{11} and U_{13} for the cognitive network are plotted over a total time duration of 140 s and are calculated based on feedback information from the intermediate nodes. More specifically, queuing and post-filtering SINR information are received at the source node through acknowledgment and feedback/control packets that are sent by the intermediate nodes at preconfigured time intervals of 3 s. We observe that if $U_{11} - U_{13} > 0$ intermediate node I_1 receives most of the traffic from source node S_1, otherwise node I_3 is selected as the next hop. Additionally, as data frames from the source node are successfully transmitted to the next hops, the magnitude of $U_{11} - U_{13}$ decreases over time.

2) NET-LAYER INTERFERENCE

Locally calculated utility values for cognitive links S_1–I_1 and S_1–I_3 and the corresponding routing decisions at source node S_1 for the second scenario of NET-layer interference are shown in Fig. 16. Intentional flooding of the queue of cognitive network node I_1 with 1000 frames is triggered by an external trigger at approximately $t = 25$ s. We observe that the proposed distributed cross-layer optimization algorithm

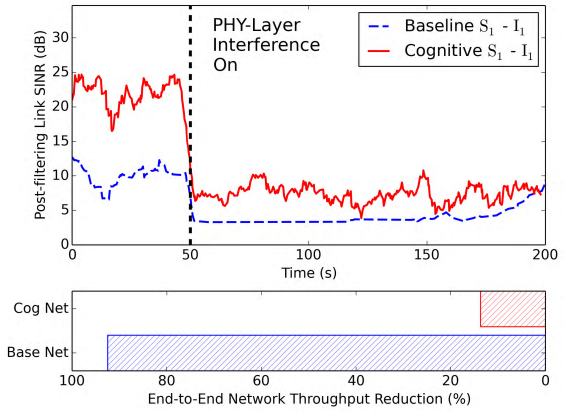


FIGURE 17. Average post-filtering SINR for the S_1–I_1 link in both cognitive and baseline networks (top) and end-to-end network throughput performance reduction (in %) for the PHY-layer interference scenario (bottom).

implemented in the cognitive network rapidly identifies the sharp drop in utility calculated for the S_1–I_1 link and decides to reroute video traffic through intermediate node I_3. While the utility of link S_1–I_1 varies slightly from changes in post-filtering SINR due to code-waveform adaptation, the large backlog prevents U_{11} from becoming larger than U_{13} and I_1 remains inactive for the remainder of the experiment.

3) PHY-LAYER INTERFERENCE

Fig. 17 presents average post-filtering SINR performance for link S_1–I_1 in both cognitive and baseline networks and evaluates end-to-end network throughput performance for the third scenario of RF interference. Only for this set of experiments, traffic flows for both networks are carried over the same fixed route S_1–I_1–D_1. An RF wideband interferer is configured to transmit at the same code-waveform with co-located baseline and cognitive links S_1–I_1. The position of the interferer is depicted in Fig. 14(a). Considering fixed routing and dynamic code-waveform adaptation at the cognitive

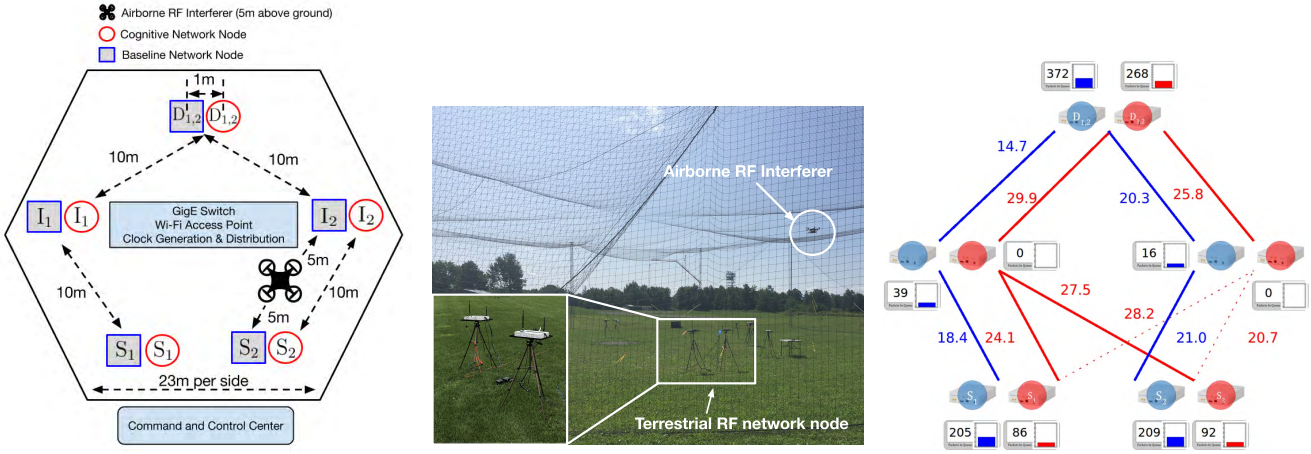


FIGURE 18. (a) Outdoor testbed deployment dimensions. (b) Terrestrial and airborne network nodes at the Stockbridge Controllable Contested Environment at the Air Force Research Laboratory in Rome, NY. (c) Host-based visualization depicts post-filtering SINRs (in dB) in real time for active links and queue information in the form of a bar-graph next to each network node. Dashed red lines illustrate feasible routes in the cognitive network. Solid blue and red lines depict selected routes in baseline and cognitive networks, respectively.

network, we observe that the post-filtering SINR for the cognitive link S_1–I_1 is approximately 8 dB higher than the SINR of the baseline link. The RF interferer operation is activated by the user at $t = 50$ s. We observe that post-filtering SINR drops significantly for both links. Although the effect of RF interference is harsher for the cognitive link SINR, code-waveform adaptation protects end-to-end network throughput, which drops only by 15% compared to the baseline network throughput that drops by 90%. The notable degradation observed in the baseline network throughput performance after $t = 50$ s is mainly due to the very low pre-detection SINR at the receiver of I_1.

C. OUTDOOR EXPERIMENTS

Outdoor field tests are conducted at the Stockbridge Controllable Contested Environment (CCE) at the Air Force Research Laboratory (AFRL) in Rome, NY. The testbed is set up inside a netted enclosure of approximately $10,000\ m^3$ as depicted in Figs. 18(a) and 18(b). Ten nodes are used to build two co-located baseline and cognitive networks and an eleventh node is mounted to an airborne vehicle platform. Digital audio and video traffic are carried over two unicast sessions from two network source nodes S_1 and S_2 to a common destination node $D_{1,2}$. Baseline network video traffic is routed through intermediate node I_1 while audio traffic is routed through intermediate node I_2. Video and audio traffic in the cognitive network is offered two additional 1-hop routes to the destination, through node I_2 and I_1, respectively. Consequently, both video and audio sessions in the cognitive network can use the same intermediate network node(s) at the same time. Additionally, backlogged intermediate nodes can offload their data queues by transmitting data to the destination at the same time or even if they are not selected as the next hop by neither of the two source nodes (e.g. Fig. 18(c)). The eleventh node is mounted on an airborne vehicle platform, which is controlled remotely to hover at a height of approximately 5 m above ground. The position of the airborne node is depicted in Fig. 18(a). The airborne node acts as a dedicated RF wideband interferer that intends to disrupt communication of both terrestrial networks.

1) NETWORK COEXISTENCE

Fig. 19 depicts average aggregate queue size at intermediate nodes I_1 and I_2 and compares the network throughput performance of both cognitive and baseline networks for the scenario of network coexistence. Dashed lines depict aggregate queue size values over 80 s when the two co-located networks utilize the same code-waveforms and routes (coexist mode off). We observe that average backlog at the intermediate nodes increases due to increased multiple-access interference.

Solid lines demonstrate the effects of cross-layer adaptation (coexist mode on) and show reduced average aggregate queue size at the intermediate nodes of both networks. In line with our expectations, when coexist mode is on, the queue size of the intermediate nodes in the cognitive network does not monotonically increase and remains low. Adaptation of waveform and routes favors the cognitive over the baseline network in terms of backlog and reduces RF interference to the baseline network. Consequently, the baseline network achieves 50% improvement in terms of network throughput. At the same time, the cognitive network throughput performance is improved by approximately 5%, therefore aggregate network throughput is increased by approximately 55%.

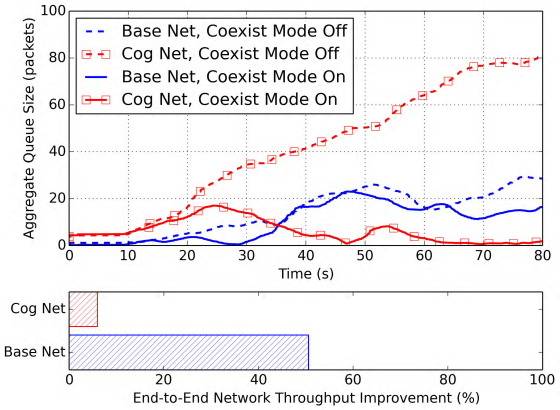


FIGURE 19. Average aggregate backlogged frames at the intermediate node queues in cognitive and baseline networks (top) and end-to-end network throughput (bottom) performance improvement (in %) for both networks before (coexist mode off) and after cross-layer adaptation (coexist mode on).

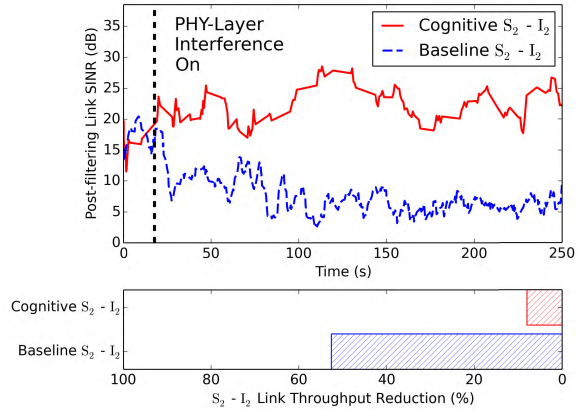


FIGURE 21. Average post-filtering SINR at the output of the maximum-SINR filter at receiver I_2 (top) and end-to-end network throughput (bottom) performance reduction (in %) for the PHY-layer interference scenario.

2) NET-LAYER INTERFERENCE

Instantaneous queue size at each intermediate network node and the end-to-end network throughput performance under NET-layer interference conditions for both networks are presented in Fig. 20. The queues of both cognitive and baseline nodes I_2 are intentionally flooded with 1600 randomly generated frames. Queue flooding is triggered by an external trigger at $t = 45$ s. Since packets are rejected by the receiver ($D_{1,2}$), neither queue decreases and both cognitive and baseline nodes I_2 remain non-operational. We observe that the queue size of baseline node I_2 continues to increase as new video frames arrive from the baseline source node S_2. On the other hand, cognitive source node S_2 dynamically reroutes video traffic through cognitive node I_1, thus the queue size at cognitive node I_2 remains constant. End-to-end network throughput performance for the baseline network is notably degraded by 60% due to network congestion, whereas cognitive network throughput performance only drops by approximately 25%.

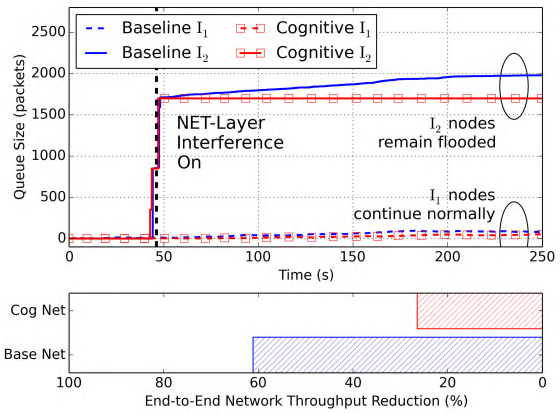


FIGURE 20. Queue size at intermediate network nodes (top) and end-to-end network throughput (bottom) performance reduction (in %) for both networks under NET-layer interference.

3) PHY-LAYER INTERFERENCE

Average post-filtering SINR and network throughput performance results for link S_2–I_2 in the presence of an airborne RF interferer are depicted in Fig. 21. An RF wideband interferer is preconfigured to operate at the same code-waveform with co-located baseline and cognitive terrestrial links S_2–I_2. The airborne source of interference is activated remotely at $t = 20$ s. We observe that the post-filtering SINR for the cognitive link S_2–I_2 remains constant after $t = 20$s, while the SINR of the baseline link drops by approximately 10 dB. Cross-layer waveform and routing adaptation at the cognitive network in the presence of the airborne interferer protects cognitive end-to-end network throughput which drops only by 5% compared to the baseline network throughput performance that drops by 50%.

Fig. 22 shows the total squared correlation (TSC) [68] between the optimized code-waveform $\mathbf{s}_{opt} \in \left\{\pm 1/\sqrt{L}\right\}^L$, $\|\mathbf{s}_{opt}\| = 1$ for cognitive link S_2–I_2 and the code-waveform of the airborne RF interferer $\mathbf{s}_I$. The code-waveform for link S_2–I_2 is optimized at preconfigured intervals of 3 s, while the code-waveform of the airborne interferer remains unchanged for the total duration of 250 s. We define TSC $\left(\mathbf{s}_{opt}, \mathbf{s}_I\right)$ as

$$\begin{aligned}\text{TSC}\left(\mathbf{s}_{opt}, \mathbf{s}_I\right) &\triangleq \left|\mathbf{s}_{opt}^T \mathbf{s}_{opt}\right|^2 + \left|\mathbf{s}_{opt}^T \mathbf{s}_I\right|^2 + \left|\mathbf{s}_I^T \mathbf{s}_{opt}\right|^2 + \left|\mathbf{s}_I^T \mathbf{s}_I\right|^2 \\ &= 2 + 2\left|\mathbf{s}_{opt}^T \mathbf{s}_I\right|^2. \end{aligned} \tag{23}$$

Consequently, if code-waveforms $\mathbf{s}_I$ and $\mathbf{s}_{opt}$ are orthogonal, $\mathbf{s}_{opt}^T \mathbf{s}_I = \mathbf{s}_I^T \mathbf{s}_{opt} = 0$ and TSC $\left(\mathbf{s}_{opt}, \mathbf{s}_I\right) = 2$. Fig. 22 demonstrates that TSC is maximum at $t = 20$ s (i.e. airborne RF interferer is activated). Max-SINR waveform optimization of the link S_2–I_2 results in code-waveforms with low TSC values. Since waveform optimization for link S_2–I_2 accounts for multiple-access interference from asynchronous transmissions in

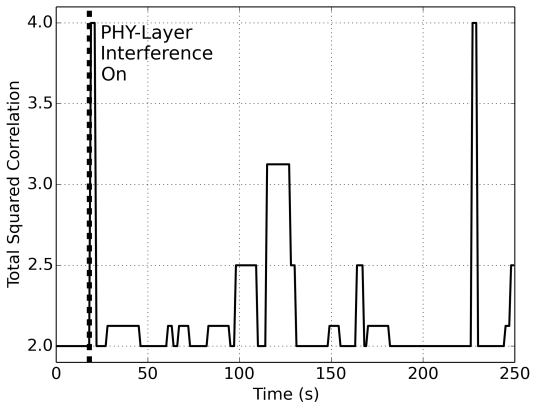


FIGURE 22. Total squared correlation between the optimized code-waveform for cognitive link S_2–I_2 and the code-waveform used by the airborne RF interferer.

co-located wireless links, we observe that TSC $\left(\mathbf{s}_{opt}, \mathbf{s}_I\right)$ is not minimized at all times.

VIII. CONCLUSION

We present a complete integrated software/hardware platform that enables the deployment of the first real-time software-defined radio testbed for prototyping spectrally efficient all-spectrum cognitive networks through cross-layer code-waveform and routing adaptation. We present algorithmic developments that span the three lowest layers of the network protocol stack (PHY/MAC/NET) and analyze corresponding software, hardware, and baseband processing requirements. We propose computationally efficient algorithms for code-waveform optimization and implement distributed control mechanisms to handle real-time waveform and routing decisions at each cognitive network node. Subsequently, we set up a programmable software/hardware radio reconfigurable framework that facilitates rapid implementation of the proposed algorithms in heterogeneous GPP-based SDR platforms. The proposed framework is optimized to run on multi-core GPP architectures that equip both terrestrial and airborne platforms. We address testbed operation challenges for the deployment of a hybrid ground-air testbed and evaluate all-spectrum cognitive networking in both indoor and outdoor environments. Finally, experimental results include post-filtering link SINR, queue size, utility measurements at the intermediate network nodes, and network-throughput performance comparison between baseline and cognitive networks that implement cognitive waveform and routing adaptation. Proof-of-concept results show that the proposed platform can be used to build cross-layer optimized all-spectrum networks that withstand interference at PHY and NET layers and maximize aggregate network throughput in underlay spectrum coexistence scenarios.

APPENDIX

PROOF OF EQUATION (21)

Consider the positive semidefinite channel-processed autocorrelation matrix $\mathbf{P}_{ij} \in \mathbb{C}^{L\times L}$. Dropping the subscript for simplicity, the matrix can be written as $\mathbf{P} = \Re\{\mathbf{P}\} + \mathrm{j}\Im\{\mathbf{P}\}$.

The imaginary part of any complex Hermitian matrix is a skew-symmetric matrix, thus $\Im\{\mathbf{P}\} = -\Im\{\mathbf{P}\}^T$. For any binary vector $\mathbf{s} \in \left\{\pm 1/\sqrt{L}\right\}^L$ it holds that

$$\mathbf{s}^T\Im\{\mathbf{P}\}\mathbf{s} = \mathbf{s}^T\Im\{\mathbf{P}\}^T\mathbf{s} = -\mathbf{s}^T\Im\{\mathbf{P}\}\mathbf{s} \tag{24}$$

therefore

$$\mathbf{s}^T\Im\{\mathbf{P}\}\mathbf{s} = 0. \tag{25}$$

The real part of any complex Hermitian matrix is a symmetric matrix. As such we can write in quadratic form

$$\mathbf{s}^T\mathbf{P}\mathbf{s} = \mathbf{s}^T\Re\{\mathbf{P}\}\mathbf{s} = \sum_{l=1}^{L}\sum_{k=1}^{L} s[l]s[k]\left[\Re\{\mathbf{P}\}\right]_{k,l} \tag{26}$$

or equivalently

$$\mathbf{s}^T\mathbf{P}\mathbf{s} = \frac{1}{L}\,\mathrm{Tr}(\Re\{\mathbf{P}\}) + \sum_{k} 2s[k]\left(\sum_{l>k} s[l]\Re\left\{[\mathbf{P}]_{k,l}\right\}\right). \tag{27}$$

ACKNOWLEDGMENT

The authors acknowledge the U.S. Government's support in the publication of this paper. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of AFRL. Approved for Public Release; Distribution Unlimited: 88ABW-2018-3669.

REFERENCES

[1] G. Sklivanitis, A. Gannon, S. N. Batalama, and D. A. Pados, "Addressing next-generation wireless challenges with commercial software-defined radio platforms," *IEEE Commun. Mag.*, vol. 54, no. 1, pp. 59–67, Jan. 2016.

[2] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May 2016.

[3] K. J. Kwak *et al.*, "Airborne network evaluation: Challenges and high fidelity emulation solution," *IEEE Commun. Mag.*, vol. 52, no. 10, pp. 30–36, Oct. 2014.

[4] J. Ahrenholz, C. Danilov, T. R. Henderson, and J. H. Kim, "CORE: A real-time network emulator," in *Proc. IEEE Military Commun. Conf. (MILCOM)*, Nov. 2008, pp. 1–7.

[5] J. Ahrenholz, T. Goff, and B. Adamson, "Integration of the CORE and EMANE network emulators," in *Proc. Military Commun. Conf. (MILCOM)*, Nov. 2011, pp. 1870–1875.

[6] B. Newton, J. Aikat, and K. Jeffay, "Simulating large-scale airborne networks with ns-3," in *Proc. Workshop NS-3 (WNS3)*, May 2015, pp. 32–39.

[7] J. Modares, N. Mastronarde, and K. Dantu, "UB-ANC emulator: An emulation framework for multi-agent drone networks," in *Proc. IEEE Int. Conf. Simulation, Modeling, Program. Auto. Robots (SIMPAR)*, Dec. 2016, pp. 252–258.

[8] T. Brown, B. Argrow, C. Dixon, and S. Doshi, "Ad hoc UAV ground network (AUGNet)," in *Proc. AIAA 3rd 'Unmanned Unlimited' Tech. Conf. Workshop Exhibit*, 2004, pp. 29–39.

[9] Y. Saleem, M. H. Rehmani, and S. Zeadally, "Integration of cognitive radio technology with unmanned aerial vehicles: Issues, opportunities, and future research challenges," *J. Netw. Comput. Appl.*, vol. 50, pp. 15–31, Apr. 2015.

[10] P. Jacob, R. P. Sirigina, A. S. Madhukumar, and V. A. Prasad, "Cognitive radio for aeronautical communications: A survey," *IEEE Access*, vol. 4, pp. 3417–3443, 2016.

[11] M. Höyhtyä *et al.*, "Spectrum occupancy measurements: A survey and use of interference maps," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 4, pp. 2386–2414, 4th Quart., 2016.
[12] T. Erpek, K. Steadman, and D. Jones, "Dublin ireland spectrum occupancy measurements collected on April 16–18, 2007," Shared Spect. Company, Vienna, VA, USA. [Online]. Available: http://www.sharedspectrum.com/wp-content/uploads/Ireland_Spectrum_Occupancy_Measurements_v2.pdf
[13] E. W. Frew and T. X. Brown, "Airborne communication networks for small unmanned aircraft systems," *Proc. IEEE*, vol. 96, no. 12, pp. 2008–2027, Dec. 2008.
[14] L. Ding, K. Gao, T. Melodia, S. N. Batalama, D. A. Pados, and J. D. Matyjas, "All-spectrum cognitive networking through joint distributed channelization and routing," *IEEE Trans. Wireless Commun.*, vol. 12, no. 11, pp. 5394–5405, Nov. 2013.
[15] I. F. Akyildiz, W. Y. Lee, and K. R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Netw.*, vol. 7, pp. 810–836, Jul. 2009.
[16] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.
[17] A. Goldsmith, S. A. Jafar, I. Maric, and S. Srinivasa, "Breaking spectrum gridlock with cognitive radios: An information theoretic perspective," *Proc. IEEE*, vol. 97, no. 5, pp. 894–914, Apr. 2009.
[18] A. Bagwari and G. S. Tomar, "Comparison between adaptive double-threshold based energy detection and cyclostationary detection technique for cognitive radio networks," in *Proc. Int. Conf. Comput. Intell. Commun. Netw.*, Sep. 2013, pp. 182–185.
[19] C. Wietfeld and K. Daniel, "Cognitive networking for UAV swarms," in *Handbook of Unmanned Aerial Vehicles*. Dordrecht, The Netherlands: Springer, 2015, pp. 749–780.
[20] G. Ding, Q. Wu, L. Zhang, Y. Lin, T. A. Tsiftsis, and Y.-D. Yao, "An amateur drone surveillance system based on the cognitive Internet of Things," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 29–35, Jan. 2018.
[21] M. E. Tanab and W. Hamouda, "Resource allocation for underlay cognitive radio networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 1249–1276, 2nd Quart., 2017.
[22] Y. Wang, "Cognitive radio for aeronautical air-ground communication," in *Proc. IEEE/AIAA 27th Digit. Avionics Syst. Conf.*, Oct. 2008, pp. 2.B.4-1–2.B.4-8.
[23] J. Kakar and V. Marojevic. (2017). "Waveform and spectrum management for unmanned aerial systems beyond 2025." [Online]. Available: http://arxiv.org/abs/1708.01664
[24] H. Ghazzai, M. B. Ghorbel, A. Kadri, M. J. Hossain, and H. Menouar, "Energy-efficient management of unmanned aerial vehicles for underlay cognitive radio systems," *IEEE Trans. Green Commun. Netw.*, vol. 1, no. 4, pp. 434–443, Dec. 2017.
[25] R. Fantacci and A. Tani, "Performance evaluation of a spectrum-sensing technique for cognitive radio applications in B-VHF communication systems," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 1722–1730, May 2009.
[26] S. Chandrasekharan *et al.*, "Designing and implementing future aerial communication networks," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 26–34, May 2016.
[27] K. Gao, S. N. Batalama, D. A. Pados, and J. D. Matyjas, "Cognitive code-division channelization," *IEEE Trans. Wireless Commun.*, vol. 10, no. 4, pp. 1090–1097, Apr. 2011.
[28] M. Li, S. N. Batalama, D. A. Pados, T. Melodia, M. J. Medley, and J. D. Matyjas, "Cognitive code-division links with blind primary-system identification," *IEEE Trans. Wireless Commun.*, vol. 10, no. 11, pp. 3743–3753, Nov. 2011.
[29] G. Sklivanitis, E. Demirors, A. M. Gannon, S. N. Batalama, D. A. Pados, and T. Melodia, "All-spectrum cognitive channelization around narrowband and wideband primary stations," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2015, pp. 1–7.
[30] N. Janatian, S. Sun, and M. Modarres-Hashemi, "Joint optimal spectrum sensing and power allocation in CDMA-based cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 9, pp. 3990–3998, Sep. 2015.
[31] Y. Xu, X. Zhao, and Y.-C. Liang, "Robust power control and beamforming in cognitive radio networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1834–1857, 4th Quart., 2015.
[32] Y. Noam and A. J. Goldsmith, "Blind null-space learning for MIMO underlay cognitive radio with primary user interference adaptation," *IEEE Trans. Wireless Commun.*, vol. 12, no. 4, pp. 1722–1734, Apr. 2013.
[33] B. Gopalakrishnan and N. D. Sidiropoulos, "Cognitive transmit beamforming from binary CSIT," *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 895–906, Feb. 2015.
[34] V. Chakravarthy *et al.*, "Novel overlay/underlay cognitive radio waveforms using SD-SMSE framework to enhance spectrum efficiency—Part I: Theoretical framework and analysis in AWGN channel," *IEEE Trans. Commun.*, vol. 57, no. 12, pp. 3794–3804, Dec. 2009.
[35] F. Jasbi and D. K. C. So, "Hybrid overlay/underlay cognitive radio network with MC-CDMA," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2038–2047, Apr. 2016.
[36] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," *Ad Hoc Netw.*, vol. 9, no. 3, pp. 228–248, May 2012.
[37] L. Ding, T. Melodia, S. N. Batalama, J. D. Matyjas, and M. J. Medley, "Cross-layer routing and dynamic spectrum allocation in cognitive radio ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1969–1979, May 2010.
[38] Q. Zhu, Z. Yuan, J. B. Song, Z. Han, and T. Basar, "Interference aware routing game for cognitive radio multi-hop networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 10, pp. 2006–2015, Nov. 2012.
[39] Y. Liu, L. X. Cai, and X. S. Shen, "Spectrum-aware opportunistic routing in multi-hop cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 10, pp. 1958–1968, Nov. 2012.
[40] S. Ping, A. Aijaz, O. Holland, and A. H. Aghvami, "SACRP: A spectrum aggregation-based cooperative routing protocol for cognitive radio ad-hoc networks," *IEEE Trans. Commun.*, vol. 63, no. 6, pp. 2015–2030, Jun. 2015.
[41] M. Zareei, E. M. Mohamed, M. H. Anisi, C. V. Rosales, K. Tsukamoto, and M. K. Khan, "On-demand hybrid routing for cognitive radio ad-hoc network," *IEEE Access*, vol. 4, pp. 8294–8302, 2016.
[42] F. Tang and J. Li, "Joint rate adaptation, channel assignment and routing to maximize social welfare in multi-hop cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 4, pp. 2097–2110, Apr. 2017.
[43] 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 Commun. Mag.*, vol. 48, no. 9, pp. 106–112, Sep. 2010.
[44] D. Raychaudhuri *et al.*, "Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols," in *Proc. IEEE Wireless Commun. Netw. Conf.*, vol. 3, Mar. 2005, pp. 1664–1669.
[45] M. Danieletto, G. Quer, R. Rao, and M. Zorzi, "CARMEN: A cognitive networking testbed on Android OS devices," *IEEE Commun. Mag.*, vol. 52, no. 9, pp. 98–107, Sep. 2014.
[46] R. Iwata, V. Va, K. Sakaguchi, and K. Araki, "Experiment on MIMO cognitive radio using Tx/Rx beamforming," in *Proc. IEEE 24th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2013, pp. 2871–2875.
[47] R. Zhou *et al.*, "Software defined radio implementation of SMSE based overlay cognitive radio," in *Proc. IEEE Symp. New Frontiers Dyn. Spectr. (DySPAN)*, Apr. 2010, pp. 1–2.
[48] P. Rose, R. Zhou, Y. Qu, V. Chakarvarthy, Z. Wu, and Z. Zhang, "Demonstration of hybrid overlay/underlay waveform generator with spectrally compliant cognitive capability via SD-SMSE framework," in *Proc. 13th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2016, pp. 258–259.
[49] G. Nychis, R. Chandra, T. Moscibroda, I. Tashev, and P. Steenkiste, "Reclaiming the white spaces: Spectrum efficient coexistence with primary users," in *Proc. 7th Conf. Emerg. Netw. Exp. Technol. (CoNEXT)*, 2011, pp. 1:1–1:12.
[50] H. Rahul, N. Kushman, D. Katabi, C. Sodini, and F. Edalat, "Learning to share: Narrowband-friendly wideband networks," in *Proc. Conf. Data Commun. (SIGCOMM)*, 2008, pp. 147–158.
[51] K. Hong, S. Sengupta, and R. Chandramouli, "SpiderRadio: A cognitive radio implementation using IEEE 802.11 components," *IEEE Trans. Mobile Comput.*, vol. 12, no. 11, pp. 2105–2118, Nov. 2013.
[52] L. Ding, P. B. Nagaraju, T. Melodia, S. N. Batalama, D. A. Pados, and D. M. John, "Software-defined joint routing and waveform selection for cognitive ad hoc networks," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Oct./Nov. 2010, pp. 1454–1459.
[53] S. Soltani, Y. Sagduyu, Y. Shi, J. Li, J. Feldman, and J. Matyjas, "Distributed cognitive radio network architecture, SDR implementation and emulation testbed," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Oct. 2015, pp. 438–443.

[54] L. Sun, W. Zheng, N. Rawat, V. Sawant, and D. Koutsonikolas, "Performance comparison of routing protocols for cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 6, pp. 1272–1286, Jun. 2015.
[55] A. R. Syed, K.-L. A. Yau, J. Qadir, H. Mohamad, N. Ramli, and S. L. Keoh, "Route selection for multi-hop cognitive radio networks using reinforcement learning: An experimental study," *IEEE Access*, vol. 4, pp. 6304–6324, 2016.
[56] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.
[57] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Found. Trends Netw.*, vol. 1, no. 1, pp. 1–144, Apr. 2016.
[58] S.-W. Huang, G. Sklivanitis, D. A. Pados, and S. N. Batalama, "Underwater acoustic communications using quasi-orthogonal chirps," presented at the 51st IEEE Asilomar Conf. Signals, Syst., Comput. (ASILOMAR), Oct./Nov. 2017.
[59] F. Wunsch, H. Jakel, and F. K. Jondral, "Performance evaluation of IEEE 802.15.4 OQPSK and CSS PHY in the presence of interference," in *Proc. IEEE 82nd Veh. Technol. Conf. (VTC Fall)*, Sep. 2015, pp. 1–5.
[60] A. Kansal, S. N. Batalama, and D. A. Pados, "Adaptive maximum SINR RAKE filtering for DS-CDMA multipath fading channels," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 9, pp. 1765–1773, Dec. 1998.
[61] G. N. Karystinos and D. A. Pados, "Adaptive assignment of binary user spreading codes in DS-CDMA systems," *Proc. SPIE*, vol. 4395, pp. 137–145, Aug. 2001. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/4395/0000/Adaptive-assignment-of-binary-user-spreading-codes-in-DS-CDMA/10.1117/12.438280.full?SSO=1
[62] C. W. Sung and H. Y. Kwan, "Heuristic algorithms for binary sequence assignment in DS-CDMA systems," in *Proc. IEEE Int. Symp. Pers., Indoor Mobile Radio Commun.*, vol. 5, Sep. 2002, pp. 2327–2331.
[63] G. N. Karystinos and D. A. Pados, "Rank-2-optimal adaptive design of binary spreading codes," *IEEE Trans. Inf. Theory*, vol. 53, no. 9, pp. 3075–3080, Sep. 2007.
[64] P. P. Markopoulos, S. Kundu, S. Chamadia, and D. A. Pados, "Efficient L1-norm principal-component analysis via bit flipping," *IEEE Trans. Signal Process.*, vol. 65, no. 16, pp. 4252–4264, Aug. 2017.
[65] Z. Liu and D. A. Pados, "LDPC codes from generalized polygons," *IEEE Trans. Inf. Theory*, vol. 51, no. 11, pp. 3890–3898, Nov. 2005.
[66] E. Demirors, G. Sklivanitis, T. Melodia, and S. N. Batalama, "RcUBe: Real-time reconfigurable radio framework with self-optimization capabilities," in *Proc. 12th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2015, pp. 28–36.
[67] T. W. Rondeau, N. McCarthy, and T. O'Shea, "SIMD programming in GNU radio: Maintainable and user-friendly algorithm optimization with VOLK," in *Proc. Wireless Innov. Forum Conf. Commun. Technol. Softw. Defined Radio (WInnComm SDR)*, Jan. 2013, pp. 101–110.
[68] G. N. Karystinos and D. A. Pados, "The maximum squared correlation, sum capacity, and total asymptotic efficiency of minimum total-squared-correlation binary signature sets," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 348–355, Jan. 2005.

GEORGE SKLIVANITIS (S'11–M'18) received the Diploma degree in electronic and computer engineering from the Technical University of Crete, Greece, in 2010, and the Ph.D. degree in electrical engineering from The State University of New York at Buffalo in 2018. He is currently a Research Assistant Professor with the Department of Computer and Electrical Engineering and Computer Science, Florida Atlantic University. His research interests span the areas of signal processing, software-defined wireless communications and networking, cognitive radio, and underwater acoustic communications. In 2014, he was the first finalist and was a recipient of the 2014 Nutaq Software-Defined Radio Academic U.S. National Contest and in 2015 the 10th ACM International Conference on Underwater Networks and Systems Best Demo Award. He was also a recipient of the 2015 SUNY Buffalo Graduate Student Award for Excellence in Teaching, the 2016 SUNY Buffalo Student Entrepreneur Fellowship, and the 2017 SUNY Chancellor's Award for Student Excellence.

ADAM GANNON (S'14) received the B.S. and M.S. degrees in electrical engineering from The State University of New York at Buffalo in 2013 and 2016, respectively, where he is currently pursuing the Ph.D. degree. His research interests include signal processing, software-defined and cognitive radio, and wireless communications and networking for airborne and space environments.

KONSTANTINOS TOUNTAS (S'14) received the Diploma and M.Sc. degrees in electronic and computer engineering from the Technical University of Crete, Chania, Greece, in 2014 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Department of Computer and Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL, USA. He was with the Telecom Lab, Technical University of Crete. His research interests span the areas of signal processing and localization, software-defined wireless communications, and underwater acoustic communications.

DIMITRIS A. PADOS (M'95–SM'15) was born in Athens, Greece, in 1966. He received the Diploma degree in computer science and engineering (five-year program) from the University of Patras, Greece, in 1989, and the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA, in 1994.

From 1994 to 1997, he was an Assistant Professor with the Department of Electrical and Computer Engineering and the Center for Telecommunications Studies, University of Louisiana, Lafayette. From 1997 to 2017, he was with the Department of Electrical Engineering, The State University of New York at Buffalo, as an Assistant Professor, an Associate Professor, a Professor, and a Clifford C. Furnas Chair Professor of electrical engineering. He also served as an Associate Chair and was appointed a Chair of the Department of Electrical Engineering. He was elected as the University Faculty Senator four times and served on the Faculty Senate Executive Committee for two terms. In 2017, he joined Florida Atlantic University, Boca Raton, FL, USA, where he currently leads the University Initiative on Autonomous Systems and holds the positions of an I-SENSE Research Center Fellow and a Charles E. Schmidt Chair Professor of computer and electrical engineering and computer science. His basic research interests are in the general areas of data and signal processing and communications theory.

Dr. Pados is a member of the IEEE Communications, the IEEE Signal Processing, the IEEE Information Theory, and the IEEE Computational Intelligence Societies. He served as an Associate Editor for the IEEE Signal Processing Letters and the IEEE Transactions on Neural Networks. For articles that he co-authored with his students received the 2001 IEEE International Conference on Telecommunications Best Paper Award, the 2003 IEEE Transactions on Neural Networks Outstanding Paper Award, the 2010 IEEE International Communications Conference Best Paper Award in signal processing for communications, the 2013 International Symposium on Wireless Communication Systems Best Paper Award in physical layer communications and signal processing, the Best of IEEE GLOBECOM 2014-Top 50 Papers Distinction, and the Best Paper Selection Distinction in the 2016 IEEE International Conference on Multimedia Big Data. He was also a recipient of the 2009 SUNY-wide Chancellor's Award for Excellence in Teaching and the 2011 University at Buffalo Exceptional Scholar-Sustained Achievement Award.

STELLA N. BATALAMA (S'91–M'94–SM'15) served as the Chair of the Electrical Engineering Department, University at Buffalo, The State University of New York, from 2010 to 2017 and from 2009 to 2011 as the Associate Dean for Research of the School of Engineering and Applied Sciences. From 2003 to 2004, she was the Acting Director of the AFRL Center for Integrated Transmission and Exploitation, Rome NY, USA. She is currently the Dean of the College of Engineering and Computer Science, Florida Atlantic University. Her research interests include cognitive and cooperative communications and networks, multimedia security and data hiding, underwater signal processing, and communications and networks. She has published over 170 papers scientific journals and conference proceedings in her research field. She was a recipient of the 2015 SUNY Chancellor's Award for Excellence in Research. She was an Associate Editor for the IEEE Communications Letters (2000–2005) and the IEEE Transactions on Communications (2002–2008).

STEPHEN REICHHART received the B.S.E.E. degree from Clarkson University and the M.S.E.E. degree from the Rochester Institute of Technology. He joined the Air Force Research Laboratory (then known as the Rome Air Development Center) in 1985 and has worked in the communication technology area since 1985. He has done research in areas dealing with adaptive wideband antennas, spread spectrum communications, covert communications, RF propagation (HF up to EHF), and software reprogrammable communications.

MICHAEL MEDLEY (M'91–SM'02) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 1990, 1991, and 1995, respectively.

Since 1991, he has been a Research Engineer for the United States Air Force, Air Force Research Laboratory, Rome, NY, USA, where he has been involved in communications and signal processing research related to adaptive interference suppression, spread spectrum waveform design, spectrum management, covert messaging, terahertz communications, and airborne networking and communications links. In 2002, he joined the State University of New York Polytechnic Institute, Utica, NY, USA, where he currently serves as an Associate Professor in electrical and computer engineering.

NGWE THAWDAR received the B.Sc. degree in electrical engineering from the State University of New York Binghamton University in 2009 and the M.Sc. and Ph.D. degrees in electrical engineering from The State University of New York at Buffalo in 2011 and 2018, respectively. She has been with the Air Force Research Laboratory, Information Directorate, Rome, NY, USA, since 2012. Her research includes wireless spread spectrum communications and networking, cooperative communications, and software-defined radio implementation. Her latest research interest is on communications in emerging spectral bands such as mm-waves and terahertz band frequencies.

ULYSSES LEE received the M.S. degree in electrical engineering from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 2016. Since 2012, he has been a Research Engineer with the United States Air Force Research Laboratory, Information Directorate, Rome, NY, USA, where he has been involved in cross-layer and cooperative communications research involving unmanned aerial systems.

JOHN D. MATYJAS received the Ph.D. degree in electrical engineering from The State University of New York at Buffalo, Buffalo, NY, USA, in 2004. He has been with the Air Force Research Laboratory, Rome, NY, USA, serving as the Technical Advisor for the Computing and Communications Division and performing R&D in the areas of wireless communications and networking. His research interests include dynamic multiple-access communications and networking, spectrum mutability, statistical signal processing and optimization, and neural networks. Additionally, he serves as the U.S. National Lead for The Technical Cooperation Program, an International C3I Group. He served on the IEEE Transactions on Wireless Communications Editorial Advisory Board (2012–2014).

Dr. Matyjas is a Senior Member of the IEEE Communications, Information Theory, Computational Intelligence, and Signal Processing Societies, and a member of the Tau Beta Pi and Eta Kappa Nu, engineering honor societies. He was a recipient of the 2015 National Aerospace Technology Management Award, 2014 AFRL-wide Engineer of the Year Award, the 2013 AFRL Maj Gen Daniel C. Doubleday Award for Outstanding Technology Management, the 2012 IEEE Region 1 Technology Innovation Award, and the 2010 IEEE International Communications Conference Best Paper Award. He is the Chair of the IEEE Mohawk Valley Section Signal Processing Society.

SCOTT PUDLEWSKI (M'07) received the B.S. degree in electrical engineering from the Rochester Institute of Technology, Rochester, NY, USA, in 2008, and the M.S. and Ph.D. degrees in electrical engineering from the University at Buffalo, The State University of New York, Buffalo, NY, USA, in 2010 and 2012, respectively. From 2012 to 2015, he was a Member of the Technical Staff at the Massachusetts Institute of Technology Lincoln Laboratory, Lexington, MA, USA. He has been with the Air Force Research Laboratory, Information Directorate, Rome, NY, USA, since 2015. He has been an Adjunct Professor with the State University of New York Polytechnic Institute since 2016. He is currently managing a program related to networking and communications to enable collaborative swarm capabilities, and is leading an effort to develop new networking capabilities to enable future autonomous Air Force technologies. His research interests include network optimization, network modeling, distributed optimization, multi-objective optimization, network protocol development, and network science in general.

ANDREW DROZD (M'85–SM'96–F'02) received the bachelor's degree in physics and mathematics and the master's degree in electrical engineering (with a major in communications and signal processing) from Syracuse University in 1977 and 1982, respectively. From 1976 to 1978, he was a Technical Assistant for the USAF Rome Air Development Center. From 1978 to 1983, he was a Lead Engineer for the IIT Research Institute. He was a Lead Engineer with General Electric, Syracuse, NY, USA. From 1983 to 1984, he taught electromagnetics courses with Utica College. He was a Lead Engineer with General Electric, Syracuse, NY, USA. From 1984 to 1994, he was a Senior Scientist with Kaman Sciences Corporation. He is currently the President and a Chief Scientist of ANDRO Computational Solutions, LLC, where he is involved in systems engineering, computational electromagnetics, and the development of advanced cognitive communications and dynamic spectrum sharing technologies. He has authored over 170 technical papers, reports, and journal articles. He was the President of the IEEE EMC Society and the Chair of the IEEE EMC-S Standards Development Committee.

ASHWIN AMANNA received the B.Sc. degree in electrical engineering from the University of California at Davis, Davis, in 1991, the M.Sc. degree in electrical engineering and the Ph.D. degree from Virginia Tech in 1994 and 2012, respectively.

From 1994 to 2012, he worked in a variety of research positions with Virginia Tech specializing in intelligent transportations systems and cognitive radio. As a Senior Research Associate, he developed heuristic and experiential learning engines to optimize wireless link performance and developing techniques for predictive modeling of wireless performance. In 2012, he joined Federated Wireless, a venture capital funded startup as a first employee and the Director of engineering. Here, he focused on resource allocation algorithms for congested networks and spectrum sharing to support interaction between commercial wireless systems and incumbent DoD systems. Since 2014, he has been with ANDRO Computational Solutions, LLC, where he is currently a Senior Research Scientist and a Technical Program Manager leading advanced waveform development teams implementing tactical data links, physical layer authentication, and CDMA systems on software-defined platforms.

He is a co-inventor of four patents, co-authored six journals, and over 25 peer-reviewed conference papers. He was a recipient of the 2017 Mohawk Valley Engineering and Executive Council Engineer of the Year award.

FRED LATUS received the B.Sc. degree in electrical engineering from the Rochester Institute of Technology in 1999.

Since 2016, he has been with ANDRO Computational Solutions, LLC, where he is involved in testing and benchmarking of software-defined radio waveforms. He designs, implements, and modifies operating test procedures on existing systems that control configurable parameters and aggregate performance metrics for post statistical analysis.

ZACHARY GOLDSMITH received the B.Sc. degree in computer engineering technology and electrical engineering technology from the State University of New York Polytechnic Institute in 2016.

Since 2016, he has been with ANDRO Computational Solutions, LLC, where he started as an intern customizing OpenWRT Wi-Fi source code to achieve non-standard channel widths and frequencies through identification, modification, and recompilation of relevant files and kernel modules all implemented on a single board computer testbed. He further performed outdoor validation testing and data analysis of throughput performance under various conditions. In 2017, he became an Associate Scientist/Engineer at ANDRO, where he has performed validation testing of software defined radio waveforms addressing enhanced railroad and satellite communications. He developed an experimental test platform to benchmark performance of a DVB-S waveform observing CPU consumption, system latency, and frame error rate.

DAVID DIAZ received the B.Sc. degree in electrical and computer engineering from the State University of New York Polytechnic Institute in 2017. He is currently an Associate Scientist and Engineer with ANDRO Computational Solutions, LLC, Rome, NY, USA. His research interests include wireless communications, signal processing, cognitive radio, and waveform design and development.