# Competing Cognitive Resilient Networks

Siamak Dastangoo, *Member, IEEE*, Carl E. Fossa, *Senior Member, IEEE*,
Youngjune L. Gwon, *Member, IEEE*, and Hsiang-Tsung Kung

***Abstract*—We introduce competing cognitive resilient network (CCRN) of mobile radios challenged to optimize data throughput and networking efficiency under dynamic spectrum access and adversarial threats (e.g., jamming). Unlike the conventional approaches, CCRN features both communicator and jamming nodes in a friendly coalition to take joint actions against hostile networking entities. In particular, this paper showcases hypothetical blue force and red force CCRNs and their competition for open spectrum resources. We present state-agnostic and stateful solution approaches based on the decision theoretic framework. The state-agnostic approach builds on multiarmed bandit to develop an optimal strategy that enables the exploratory-exploitative actions from sequential sampling of channel rewards. The stateful approach makes an explicit model of states and actions from an underlying Markov decision process and uses multiagent *Q*-learning to compute optimal node actions. We provide a theoretical framework for CCRN and propose new algorithms for both approaches. Simulation results indicate that the proposed algorithms outperform some of the most important algorithms known to date.**

***Index Terms*—Cognitive radio, strategy, multi-armed bandit (MAB), reinforcement learning.**

## I. Introduction

COGNITIVE radios have arisen commercially over the last decade, enabling a new means to share radio spectrum. Dynamic spectrum access (DSA) [1] is a compelling usage scenario for cognitive radio systems. DSA aims to relieve shortages of radio spectrum, which is the scarcest—hence, the most expensive—resource to build a wireless network. Much of contemporary research has considered cognitive radios as the secondary user of a licensed spectrum and focused on the development of a flexible mechanism to opportunistically access the licensed channel to its maximal spectral efficiency.

We envision the use of cognitive radio technology for tactical wireless networks operating in an environment where malicious jammers and other adversarial threats exist in addition to scarce spectral resources. The past approaches [2]–[6] have concentrated on a defense mechanism against the adversarial jamming attacks (i.e., antijamming strategy). In contrast, our approach in this paper is to optimize antijamming and jamming actions jointly.

This paper introduces Competing Cognitive Resilient Network (CCRN), where a network of communicator (comm) nodes and jammers attempts to dominate the access to an open spectrum *competitively* against a hostile opponent, which is possibly another cognitive radio network of similar capabilities. An antijamming-jamming strategy is a critical requirement for CCRNs, and we propose two different approaches in computing the optimal CCRN strategy, namely state-agnostic and stateful.

The state-agnostic approach is based on Multi-armed Bandit (MAB) problems [7] that address the exploration-exploitation dilemma for allocating resources on sequential reward sampling. Lai *et al.* [8], for example, discusses the primary-secondary usage framework in the context of MAB for cognitive radios in the DSA paradigm. In our work, however, we have devised a randomized algorithm for the CCRN nodes taking actions (i.e., communicate or jam) on a block of multi-channel spectrum that works statelessly and is guided only by channel reward sampling. The state-agnostic algorithm essentially runs Thompson sampling [9], an old probability matching heuristic, under which we set up the optimal Bayesian conjugate prior from an extreme-valued likelihood. We will explain the detailed rationale and present its superior performance over some of the most important MAB algorithms applied to CCRNs in later sections of this paper.

The stateful approach, on the other hand, is based on game theory [10]. It requires explicit modeling of CCRN states and actions from an underlying Markov Decision Process (MDP). We have formulated the two-network CCRN game and decompose it to antijamming and jamming subgames. Unlike the existing game-theoretic framework for cognitive radios, we solve the antijamming and jamming subgames jointly for an optimal strategy, applying multi-agent Q-learning [11]. Given perfect sensing at the lower layer, we will show that Q-learning can result in channel access decisions that converge to the best cumulative average reward in the steady state.

We summarize our main contributions:

- This paper develops a theoretical framework for cognitive radio networks under competition that simultaneously access (communicate) and suppress (jam) for spectral dominance in tactical setting;

Manuscript received October 23, 2015; revised March 8, 2016; accepted April 18, 2016. This material is based upon work supported by the Office of the Secretary of Defense under Air Force Contract FA8721-05-C-0002 and/or FA8702-15-D-0001. The associate editor coordinating the review of this paper and approving it for publication was C. Clancy.

S. Dastangoo and C. E. Fossa are with the Tactical Networks Group, MIT Lincoln Laboratory, Lexington, MA 02421 USA (e-mail: sia@ll.mit.edu; cfossa@ll.mit.edu).

Y. L. Gwon was with Computer Science at Harvard John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 USA. He is now with MIT Lincoln Laboratory, Lexington, MA 02421 USA (e-mail: gyj@ll.mit.edu).

H.-T. Kung is the William H. Gates Professor of Computer Science and Electrical Engineering at Harvard John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 USA (e-mail: kung@harvard.edu).

Digital Object Identifier 10.1109/TCCN.2016.2570798

- We propose new algorithmic approaches to compute a joint antijamming-jamming strategy in-network;
- We evaluate the empirical performance of the proposed algorithms and show that they outperform the key classical approaches known to date.

This paper is an extended version of our two previous publications in tactical cognitive radio networks [12], [13]. The paper provides complete system and architectural considerations for an intelligent tactical network. In particular, we describe a detailed architecture under which cognitive radio nodes collect information by sensing and assess the performance of learning-based algorithms for dyanamic spectrum access. Furthermore, we validate the optimality of our algorithmic approaches and their equivalence to analytical models, which are generally known intractable, using probabilistic sampling (i.e., by Markov Chaning Monte Carlo method) on the derived transition probability functions. This paper also presents the complete reward performance results by iterating all comm and jamming probabilities of cognitive radio nodes.

The rest of the paper is organized as follows. In Section II, we describe the CCRN system architecture. Section III presents our mathematical formulation. Section IV develops algorithmic approaches to find an optimal CCRN strategy. We describe our adaptation of classical algorithms under the state-agnostic and stateful approaches and propose two new algorithms. In Section V, we numerically evaluate the performance of the proposed algorithms. Section VI discusses related work, and Section VI concludes the paper.

## II. Competing Cognitive Resilient Network (CCRN): System Architecture

### A. Overview

For clarity of discussion, imagine two networks of cognitive radios, Blue Force (BF or the *ally*) and Red Force (RF or the *enemy*). Each network consists of two types of nodes: communicator (comm) and jammer. In their field operations, the BF and RF networks face each other in a competition to achieve higher comm data throughput and suppress the opponent's comm activities by jamming, all trying to dominate an open spectrum.

Accessing channels by a comm node is determined upon sensing and cognizance of vacant spectrum blocks. The primary-secondary user dichotomy popular in the existing cognitive radio networking literature is mostly invalid for our purpose. We use the term Competing Cognitive Resilient Network (CCRN) to designate our BF and RF networks instead.

In tactical ad hoc networks, node geometry in the field can be critical for effective antijamming and jamming capabilities. For simplicity, we assume that the internal layout of the nodes for BF or RF networks does not play a significant role. However, we use two different network control mechanisms, namely centralized and distributed. The centralized control for CCRN designates a central entity that collects channel sensing (performed by each node) and other network information to make a coherent, network-wide decision to access or suppress a channel. For each time slot, the centralized control chooses one of the channels to broadcast control information.

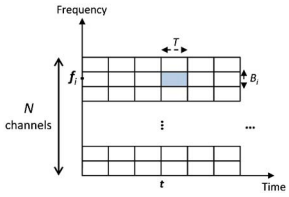



Fig. 1. Transmission opportunity $\langle f_i, B_i, t, T\rangle$ (shaded region).

The information on control channel selection is transmitted through the previous slot's control channel. Under the distributed control, each node in a CCRN acts as a decision maker for its own action after exchanging the information with other nodes. Due to staleness of information at each node, however, distributed decision making may result in suboptimal actions for the network as a whole.

### B. Communications Model

Spectrum for open access is partitioned in time and frequency. There are $N$ non-overlapping channels located at the center frequency $f_i$ (MHz) with bandwidth $B_i$ (Hz) for $i = 1, \ldots, N$. A transmission opportunity is represented by a tuple $\langle f_i, B_i, t, T\rangle$, which designates a time-frequency slot at channel $i$ and time $t$ with time duration $T$ (msec) as depicted in Fig. 1. We assume a simple CSMA in which comm nodes first sense before transmitting in a slot of opportunity.

In order to coordinate a non-conflicting spectrum access and jamming strategy network-wide, we assume that the nodes (both comm and jammers) exchange necessary information via control messages. We call the channels used to exchange control messages 'control channels.' On the contrary, 'data channels' are used to transport regular data packets. We follow the DSA approach [2] that control or data channels are dynamically allocated. Due to cryptographic randomization, guessing the correct control channel by the enemy network would be a hard problem. However, if the control channel happens to be jammed, say at time $t$, the spectrum access at time $t + 1$ will be uncoordinated. For the case of blocked control channels, the CCRN performance will be suboptimal.

### C. Jamming Model

Xu *et al.* [14] presents a sound taxonomy of RF jamming. A constant jammer continuously dissipates power into a selected channel by transmitting arbitrary waveforms. A deceptive jammer can instead send junk bits encapsulated in a legitimate packet to conceal its intent to disrupt comm nodes. A random jammer alternates between jamming and remaining quiet for random time intervals. A reactive jammer listens to a channel, stays quiet when the channel is idle, and starts transmitting upon sensing an activity.

The key to successful jamming is intelligence and statistical sophistication. We introduce *strategic* jamming model that extends the statistical jamming described in Pajic and Mangharam [15]. A strategic jammer can learn media access patterns of comm nodes and adapt to antijamming schemes in the same way that our comm nodes leverage sensing and cognition. Strategic jammers can remain effective for long without being detected, subsequently causing more damages than the jammers based on existing models.

TABLE I
NODE ACTIONS, OUTCOME AND RESULTING REWARD

| BF comm | BF jammer | RF comm | RF jammer | Outcome | Reward |
|---|---|---|---|---|---|
| Tx | ∅ | ∅ | ∅ | BF Tx success | $R_B \mathrel{+}= B$ |
| ∅ | Jam | Tx | ∅ | BF jamming | $R_B \mathrel{+}= B$ |
| Tx | Jam | ∅ | ∅ | BF misjamming | – |
| ∅ | ∅ | Tx | ∅ | RF Tx success | $R_R \mathrel{+}= B$ |
| Tx | ∅ | ∅ | Jam | RF jamming | $R_R \mathrel{+}= B$ |
| ∅ | ∅ | Tx | Jam | RF misjamming | – |
| Tx | ∅ | Tx | ∅ | Tx collision | – |

### D. Reward Model

We employ a reward metric measured in *bits*. When a CCRN comm node makes successful transmission of a packet containing $B$ bits of data, it receives the reward of $B$ (bits). The definition of a successful transmission follows the notion in classical wireless networking that there should be only one comm node transmission for the Tx opportunity. If there were two or more simultaneous comm transmissions (from either the same or different network), a collision occurs, and no comm node gets a reward.

Jammers do not create any reward by themselves. However, they receive a reward by suppressing an opposing comm node's otherwise successful transmission. For example, a BF jammer earns a reward $B$ by jamming the slot in which a sole RF comm node transmits. If there were no jamming, the RF comm node would have earned $B$. Also, a BF jammer can jam a BF comm mistakenly (e.g., due to faulty intra-network coordination), which we call *misjamming*.

Table I summarizes how the outcome and reward at a channel are determined given various combinations of the BF and RF node actions at a slot of transmission opportunity ('∅' means no action). Note that each CCRN keeps track of its reward cumulatively (i.e., $R_B$, $R_R$) over operated time slots.

For illustrative purposes, we provide example BF and RF actions and explain the reward computation for each CCRN. Let BF and RF networks each have two comm nodes and two jammers. At time $t$, BF comm node 1 transmits in channel 7, and BF comm node 2 in channel 3. BF jammers jam channels 1 and 5 at $t$. Similarly, RF has its comm nodes transmitting in channels 3 and 5, and its jammers jamming channels 10 and 9. Fig. 2 depicts the resulting channel-action bitmap where 1 indicates transmit or jam and 0 otherwise. BF jammer on channel 5 is successful whereas the one on channel 1 is not. There is a comm collision between BF and RF in channel 3, but BF has a successful comm transmission in channel 7. Thus, BF network receives the reward of $2B$ for one of its comm and one of its jamming actions. RF networks has no success in comm or jamming at $t$.

### E. System Model

We now describe a CCRN system in detail. A CCRN node consists of sensing, strategy, schedule, and Tx/jam components as illustrated in Fig. 3. Using local and global sensing information, a CCRN node applies a strategy to compute an action (i.e., transmit, jam, or do nothing) particular to its channel of interest. The action translates to scheduling a block of transmission/jamming opportunities. Node actions can be computed in either centralized or distributed manner.

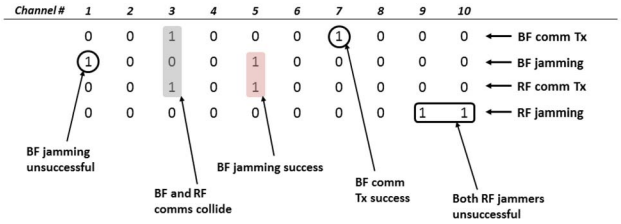



Fig. 2. Example Blue Force (BF) and Red Force (RF) CCRN node actions and resulting outcome.

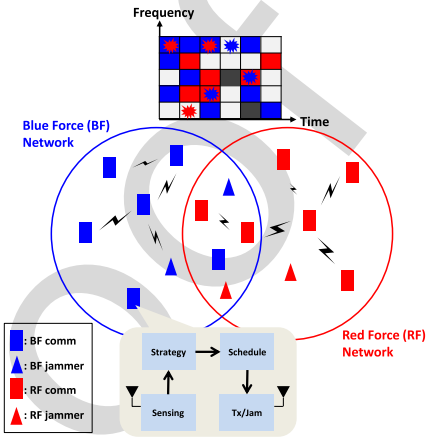



Fig. 3. Overview of Competing Cognitive Resilient Network (CCRN).

Under the centralized control, CCRN works in the following steps.

1) Sense channel activities (each node)
2) Collect sensing information (controller)
3) Compute node actions (controller)
4) Disseminate node actions (controller)
5) Act on channel (each node)

In the distributed control, CCRN works as follows.

1) Sense channel activities (each node)
2) Exchange sensing information (each node)
3) Compute its own action (each node)
4) Act on channel (each node)

When acting on a channel, a CCRN node under the distributed control assumes *cooperative* behavior by considering the holistic goal of maximizing the overall network performance. When new information is available for a node (e.g., channel sensing, outcome), a CCRN node exchanges the information *collaboratively* with one another. The distributed CCRN would be a realistic application for Mobile ad hoc network (MANET) where there is little or no fixed infrastructural support. In the later sections of this paper, we will evaluate CCRN under both centralized and distributed control architectures.

## III. MATHEMATICAL FORMULATION

This section presents mathematical formulation for CCRN. In particular, we develop two compatible, yet different

TABLE II
SUMMARY OF NOTATIONS

| Notation | Description |
|---|---|
| $a_B^t, a_R^t$ | node actions at $t$ for Blue (B) and Red (R) Force networks |
| $\Gamma^t$ | regret over $t$ time slots |
| $R^t$ | cumulative reward over $t$ time slots |
| $r^j$ | instantaneous reward at time slot $j$ |
| $\gamma$ | reward discount ratio |
| $\sigma^t$ | channel access strategy at time $t$ |
| $\pi$ | probability distribution computed over action set |
| $\Omega^t$ | outcome bit vector (each 1 is success, and 0 failure) at $t$ |
| $\theta, a, b, \beta$ | hyperparameters of probability model |
| $Q(s,a)$ | measure for quality of action at state $s$ |
| $V(s)$ | measure for value of state |
| $\alpha, \lambda$ | learning rate, decay parameters for Q-learning |
| $N$ | total number of accessible channels |
| $C, J$ | number of comm and jamming nodes |
| $p_{\mathrm{Tx}}$ | transmit probability of a comm node |

frameworks for our BF-RF competition scenario. First, we explain a *state-agnostic* model that operates a CCRN without considering any system states. Then, we describe a *stateful* model by explicitly defining CCRN states and discuss plausible ways in computing them. Markov Decision Process (MDP) underlies the stateful CCRN whereas the state-agnostic counterpart is solely driven by sequential reward sampling.

### A. Notation and Preliminaries

*Strategy* or *policy* for a CCRN means rules to select its node actions. There are $C$ comm nodes and $J$ jammers in each CCRN. Let $N$ designate the number of channels in the spectrum. We define the BF action set $A_B$ such that a BF action $a_B \in A_B$, and similarly for RF, $a_R \in A_R$. At time $t$, the BF and RF actions are $a_B^t = \{a_{B,comm}^t, a_{B,jam}^t\}$ and $a_R^t = \{a_{R,comm}^t, a_{R,jam}^t\}$ containing both comm and jamming actions, the size-$C$ vectors $a_{B,comm}^t$, $a_{R,comm}^t$ and the size-$J$ $a_{B,jam}^t$, $a_{R,jam}^t$. (Note we use a superscripted $^t$ for 'at time $t$.') An $i$th element in $a_{B,comm}^t$ designates the channel number that the $i$th BF comm node tries to transmit at $t$. Similarly, a $j$th element in $a_{B,jam}^t$ is the channel that the $j$th BF jammer tries to jam at $t$. The BF and RF node actions result in an outcome $\Omega : A_B \times A_R \longrightarrow \mathbb{R}^N$. Subsequently, we can map the outcome to a reward $R : \Omega \longrightarrow \mathbb{R}$ given a reward function. In Table II, we summarize a list of important notations used in this paper.

### B. State-Agnostic CCRN

The state-agnostic CCRN model is based on multi-armed bandit (MAB) originated from Thompson's medical experiments [9], although MAB is best explained with a gambler facing $N$ slot machines (arms). The gambler's objective is to find a strategy that maximizes $R^t = \sum_{j=1}^t r^j$ for some $t$, the *cumulative* reward over a finite horizon. Lai and Robbins [7] introduced the concept of *regret* for a strategy $\sigma$ measuring the distance from optimality

$$\Gamma^t = t\mu^* - \mathbb{E}\big[R_\sigma^t\big] \tag{1}$$

where $\mu^*$ is the hypothetical, maximum average reward if gambler's action led to the best possible outcome each round and achieved the actual reward $R_\sigma^t$ under $\sigma$. It turns out that minimizing $\Gamma^t$ is mathematically more convenient than maximizing the *expectation* of $R^t$.

An arm corresponds to a channel in the spectrum under competition. Comm nodes and jammers are the players that a CCRN allocates to play (i.e., transmit or jam) the channels. Since the CCRN has multiple nodes, our problem is classified as multi-player MAB [16], which is different from the classic single-player MAB formulated by Lai and Robbins [7]. In addition, we have two system variations depending on whether a centralized or distributed control mechanism is deployed.

Lai and Robbins [7] further derived the mathematical qualification for an optimal strategy:

$$\lim_{t\to\infty} \sup \mathbb{E}\big[T_i^t\big] \le \frac{\log t}{\mathrm{D}_{KL}(p_i \parallel p^*)} \tag{2}$$

where sup means supremum, $T_i^t$ is the total number for arm $i$ being played, and $\mathrm{D}_{KL}(\cdot \parallel \cdot)$ is the Kullback-Leibler divergence [17] measuring the dissimilarity between the probability distributions $p_i$ and $p^*$ for the $i$-th arm's reward and the maximum reward resulted by choosing only the best possible arm each time. Eq. (2) provides the least upper bound for the number of times should an optimal arm—which could be different each time—be played asymptotically. Lai and Robbins also provided an algorithm that satisfies the condition of Eq. (2), which will be discussed in Section IV.

The BF strategy $\sigma_B^t$ is a function over time. It takes necessary information such as sensing results and past action-outcome/reward statistics as input and determines the BF node actions. Under the centralized decision making, we express

$$\left\{x_B^j\right\}_{j=1}^t, \left\{a_B^j, \Omega^j\right\}_{j=1}^{t-1} \xrightarrow{\sigma_B^t} a_B^t \tag{3}$$

where $x_B^t$ is the BF sensing results at $t$.

Under the distributed decision making, each node in the network computes its own action. For BF node $i$ (whether it is a comm node or jammer), we write

$$x_{B,i}^t, \left\{x_B^j, a_B^j, \Omega^j\right\}_{j=1}^{t-1} \xrightarrow{\sigma_{B,i}^t} a_{B,i}^t \tag{4}$$

where $x_{B,i}^t$ is the sensing information only available to BF node $i$ at time $t$, and $\sigma_{B,i}^t$ the strategy of BF node $i$'s own. At time $t$, BF node $i$ does not yet have all sensing results except its own $x_{B,i}^t$. For the distributed case, node strategies can differ, and there is no guarantee that conflicting actions of the nodes in the same network such as collision and misjamming are resolved.

### C. Stateful CCRN

The stateful CCRN model uses the tuple $\langle S, A_B, A_R, R, T\rangle$ to describe the dynamics of the competition between BF and RF networks. $S$ denotes the state set, and $A_B = \{A_{B,comm}, A_{B,jam}\}, A_R = \{A_{R,comm}, A_{R,jam}\}$ are the action sets for BF and RF networks. The reward function $R : S \times \prod A_{\{B,R\},\{comm,jam\}} \to \mathbb{R}$ maps CCRN node actions to a reward value at a given state. The state transition $T : S \times \prod A_{\{B,R\},\{comm,jam\}} \to \mathrm{PD}(S)$ is the probability distribution over $S$. Under stochastic setup, a strategy $\pi : S \to \mathrm{PD}(A)$ is the probability distribution over the action set.

We formulate the stateful CCRN with a Markov game [10].

TABLE III
COLLISION PARAMETERS

| Parameter | Description |
|---|---|
| $I_{B,C}$ | # of control channel collisions caused by BF comms only |
| $I_{B,D}$ | # of data channel collisions caused by BF comms only |
| $I_{R,C}$ | # of control channel collisions caused by RF comms only |
| $I_{R,D}$ | # of data channel collisions caused by RF comms only |
| $I_{BR,C}$ | # of control channel collisions caused by BF and RF comms |
| $I_{BR,D}$ | # of data channel collisions caused by BF and RF comms |

TABLE IV
JAMMING PARAMETERS

| Parameter | Description |
|---|---|
| $J_{B,R,C}$ | # of BF control channel jammed by RF jammers |
| $J_{B,R,D}$ | # of BF data channel jammed by RF jammers |
| $J_{B,B,C}$ | # of BF control channel jammed by BF jammers |
| $J_{B,B,D}$ | # of BF data channel jammed by BF jammers |
| $J_{B,BR,C}$ | # of BF control channel jammed by BF and RF jammers |
| $J_{B,BR,D}$ | # of BF data channel jammed by BF and RF jammers |
| $J_{R,B,C}$ | # of RF control channel jammed by BF jammers |
| $J_{R,B,D}$ | # of RF data channel jammed by BF jammers |
| $J_{R,R,C}$ | # of RF control channel jammed by RF jammers |
| $J_{R,R,D}$ | # of RF data channel jammed by RF jammers |
| $J_{R,BR,C}$ | # of RF control channel jammed by BF and RF jammers |
| $J_{R,BR,D}$ | # of RF data channel jammed by BF and RF jammers |

*1) State Representation:* Each of $N$ channels in the spectrum under competition is described by a Markov chain. If there are $L$ discrete states for a channel, we require to track $L^N$ states to describe the CCRN interactions fully. Such tracking, however, leads to a computational complexity class in $O(L^N)$ with respect to channelization $N$. We instead choose a terser state representation $s = \langle I_C, I_D, J_C, J_D \rangle$ where $I_C$ denotes the number of control channels collided, $I_D$ the number of data channels collided, $J_C$ the number of control channels jammed, and $J_D$ the number of data channels jammed.

Given the current state and the action sets of BF and RF nodes, the next state of CCRN is computable. The actions of the opponent is inferred from channel measurements and sensing. To estimate $I_C$, $I_D$, $J_C$, and $J_D$, we need to observe the parameters in Tables III and IV to calculate

$$I_C = \sum_{x\in\{B,R,BR\}} I_{x,C}$$

$$I_D = \sum_{x\in\{B,R,BR\}} I_{x,D}$$

$$J_C = \sum_{x\in\{B,R\},y\in\{B,R,BR\}} J_{x,y,C}$$

$$J_D = \sum_{x\in\{B,R\},y\in\{B,R,BR\}} J_{x,y,D}.$$

*2) State Transition Probabilities:* In this section, we derive the full, analytical formula for the CCRN state transition probability distribution that can be used for numerical approximation.

*a) Counting parameters for state transition:* The following conditional probability distribution determines the transition function $T$:

$$p\left(s^{t+1}\middle|s^t, a_B^t, a_R^t\right) = p\left(I_C^{t+1}, I_D^{t+1}, J_C^{t+1}, J_D^{t+1}\middle|I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\right)$$

To express $I_C^{t+1}, I_D^{t+1}, J_C^{t+1}$, and $J_D^{t+1}$, we need to define the counting parameters related to collision and jamming:

- $m_{C1} \stackrel{\text{def}}{=}$ # of collided control channels previously uncollided and unjammed;
- $m_{C2} \stackrel{\text{def}}{=}$ # of collided control channels previously collided;
- $m_{C3} \stackrel{\text{def}}{=}$ # of collided control channels previously jammed;
- $m_{D1} \stackrel{\text{def}}{=}$ # of collided data channels previously uncollided and unjammed;
- $m_{D2} \stackrel{\text{def}}{=}$ # of collided data channels previously collided;
- $m_{D3} \stackrel{\text{def}}{=}$ # of collided data channels previously jammed;
- $n_{C1} \stackrel{\text{def}}{=}$ # of jammed control channels previously uncollided and unjammed;
- $n_{C2} \stackrel{\text{def}}{=}$ # of jammed control channels previously collided;
- $n_{C3} \stackrel{\text{def}}{=}$ # of jammed control channels previously jammed;
- $n_{D1} \stackrel{\text{def}}{=}$ # of jammed data channels previously uncollided and unjammed;
- $n_{D2} \stackrel{\text{def}}{=}$ # of jammed data channels previously collided;
- $n_{D3} \stackrel{\text{def}}{=}$ # of jammed data channels previously jammed.

Now we can write the number of collided control channels $I_C^{t+1} = m_{C1} + m_{C2} + m_{C3}$, the total number of collided data channels $I_D^{t+1} = m_{D1} + m_{D2} + m_{D3}$, the jammed control channels $J_C^{t+1} = n_{C1} + n_{C2} + n_{C3}$, and the jammed data channels $J_D^{t+1} = n_{D1} + n_{D2} + n_{D3}$.

We define the counting parameters that describe how BF and RF networks choose control and data channels at time $t$:

- $\alpha_{C1}^t \stackrel{\text{def}}{=}$ # of control channels chosen from previously uncollided and unjammed channel space;
- $\alpha_{D1}^t \stackrel{\text{def}}{=}$ # of data channels chosen from previously uncollided and unjammed channel space;
- $\alpha_{C2}^t \stackrel{\text{def}}{=}$ # of control channels chosen from previously collided channel space;
- $\alpha_{D2}^t \stackrel{\text{def}}{=}$ # of data channels chosen from previously collided channel space;
- $\alpha_{C3}^t \stackrel{\text{def}}{=}$ # of control channels chosen from previously jammed channel space;
- $\alpha_{D3}^t \stackrel{\text{def}}{=}$ # of data channels chosen from previously jammed channel space.

We define the parameters to describe how BF and RF jamming actions are chosen at $t$:

- $\alpha_{J1}^t \stackrel{\text{def}}{=}$ # of channels chosen from previously uncollided channel space for jamming;
- $\alpha_{J2}^t \stackrel{\text{def}}{=}$ # of channels chosen from previously collided channel space for jamming;
- $\alpha_{J1}^t \stackrel{\text{def}}{=}$ # of channels chosen from previously unjammed channel space for jamming;
- $\alpha_{J2}^t \stackrel{\text{def}}{=}$ # of channels chosen from previously jammed channel space for jamming.

We have a constraint $\alpha_{C1}^t + \alpha_{D1}^t < N_1^t$ where $N_1^t = N - (I_C^t + I_D^t + J_C^t + J_D^t)$ gives the total number of uncollided and unjammed channels. We also have $\alpha_{C2}^t + \alpha_{D2}^t < N_2^t$ where $N_2^t = I_C^t + I_D^t$ is the total number of collided channels, and $\alpha_{C3}^t + \alpha_{D3}^t < N_3^t$ where $N_3^t = J_C^t + J_D^t$ is the total number of jammed channels.

*b) Combinatorial analysis:* We should consider combinations of $(m_{C\{1,2,3\}}, m_{D\{1,2,3\}})$ and $(n_{C\{1,2,3\}}, n_{D\{1,2,3\}})$ subject to the constraints represented by $I_C$, $I_D$, $J_C$, and $J_D$. Using the binomial coefficient $\binom{n}{k} = \frac{n!}{k!(n-k)!}$, the probability of $m_{C1}$ control and $m_{D1}$ data channels collided given that BF and RF networks choose from *previously uncollided and unjammed* channels is:

$$p\big(m_{C1}, m_{D1} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) = \frac{\binom{\alpha_{C1}^t}{m_{C1}}\binom{\alpha_{D1}^t}{m_{D1}}\binom{N_1^t-\alpha_{C1}^t-\alpha_{D1}^t}{\alpha_{J1}^t+\alpha_{J1}^t-m_{C1}-m_{D1}}}{\binom{N_1^t}{\alpha_{J1}^t+\alpha_{J1}^t}}$$

The probability of $m_{C2}$ control and $m_{D2}$ data channels collided given that BF and RF networks choose from *previously collided* channels is:

$$p\big(m_{C2}, m_{D2} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) = \frac{\binom{\alpha_{C2}^t}{m_{C2}}\binom{\alpha_{D2}^t}{m_{D2}}\binom{N_2^t-\alpha_{C2}^t-\alpha_{D2}^t}{\alpha_{J2}^t-m_{C2}-m_{D2}}}{\binom{N_2^t}{\alpha_{J2}^t}}$$

The probability of $m_{C3}$ control and $m_{D3}$ data channels collided given that BF and RF networks choose from *previously jammed* channels is:

$$p\big(m_{C3}, m_{D3} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) = \frac{\binom{\alpha_{C3}^t}{m_{C3}}\binom{\alpha_{D3}^t}{m_{D3}}\binom{N_3^t-\alpha_{C3}^t-\alpha_{D3}^t}{\alpha_{J2}^t-m_{C3}-m_{D3}}}{\binom{N_3^t}{\alpha_{J2}^t}}$$

The probability of $n_{C1}$ control and $n_{D1}$ data channels jammed given that BF and RF networks choose from *previously uncollided and unjammed* channels is:

$$p\big(n_{C1}, n_{D1} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) = \frac{\binom{\alpha_{C1}^t}{n_{C1}}\binom{\alpha_{D1}^t}{n_{D1}}\binom{N_1^t-\alpha_{C1}^t-\alpha_{D1}^t}{\alpha_{J1}^t+\alpha_{J1}^t-n_{C1}-n_{D1}}}{\binom{N_1^t}{\alpha_{J1}^t+\alpha_{J1}^t}}$$

The probability of $n_{C2}$ control and $n_{D2}$ data channels jammed given that BF and RF networks choose from *previously collided* channels is:

$$p\big(n_{C2}, n_{D2} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) = \frac{\binom{\alpha_{C2}^t}{n_{C2}}\binom{\alpha_{D2}^t}{n_{D2}}\binom{N_2^t-\alpha_{C2}^t-\alpha_{D2}^t}{\alpha_{J2}^t-n_{C2}-n_{D2}}}{\binom{N_2^t}{\alpha_{J2}^t}}$$

The probability of $n_{C3}$ control and $n_{D3}$ data channels jammed given that BF and RF networks choose from *previously jammed* channels is:

$$p\big(n_{C3}, n_{D3} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) = \frac{\binom{\alpha_{C3}^t}{n_{C3}}\binom{\alpha_{D3}^t}{n_{D3}}\binom{N_3^t-\alpha_{C3}^t-\alpha_{D3}^t}{\alpha_{J2}^t-n_{C3}-n_{D3}}}{\binom{N_3^t}{\alpha_{J2}^t}}$$

*c) Posterior conditional probabilities:* The combinatorial analysis leads to the posterior state transition probability distribution presented in Eq. (5), as shown at the top of next page. To solve for an optimal strategy, we need to evaluate this posterior distribution. For large networks and rapid variation in the system parameters (e.g., changing number of channels), this approach imposes high computational cost. We can alternatively sample the distribution, using a statistically rigorous technique such as Markov Chain Monte Carlo (MCMC); however, the MCMC performance relies on the choice of a proposal distribution that must work well for CCRN, which by itself is an active area of research. In the next section, we propose Q-learning [11] based methods that can avoid complex state transition computations by a technique called *value iteration* [18]. In Section V-B, we will numerically evaluate transition probabilities for the two-CCRN scenario using Eq. (5) and Monte Carlo sampling.

### D. Optimal Strategies

The goal of the state-agnostic CCRN is to minimize the growth of regret with an optimal strategy $\sigma^*$:

$$\sigma^* = \arg\min_{\sigma} \Gamma^t = \min_{\sigma} \left\{ \mathbb{E}\left[ \sum_{i=1}^{M} \sum_{j=1}^{t} r_{(i)}^j \right] - \mathbb{E}\big[R_\sigma^t\big] \right\} \quad (6)$$

where $\Gamma^t$ represents the CCRN regret at time $t$, and we use $r_{(i)}^t$ an *ordered* sequence of the $N$ instantaneous channel rewards at time $t$ such that $r_{(1)}^t \geq r_{(2)}^t \geq \ldots \geq r_{(N)}^t$. Knowing that there are $M = C + J$ total number of nodes in the network, sum of the $M$ ($< N$) highest rewarding channels reflects the optimal allocation of the nodes. We note that one can adopt the concept of *discounted* reward by expressing $\mathbb{E}\big[R_\sigma^t\big] = \mathbb{E}\sum_{j=0}^{t} \gamma^j r_\sigma^j$. By adjusting the value of discount ratio $\gamma$ ($0 \leq \gamma < 1$) over time, one can control the coverage to which a strategy tries to optimize short term versus long term (i.e., exploit or explore) rewards.

The goal for the stateful CCRN is stated differently. We wish to find an optimal distribution $\pi^*$ over all possible node actions to maximize the *expected* cumulative sum of discounted rewards:

$$\pi^* = \arg\max_{\pi} \mathbb{E}\left[ \sum_{j=0}^{t} \gamma^j R\Big(s^j, a_B^j, a_R^j\Big) \right] \quad (7)$$

where $\gamma$ again is a reward discount ratio, strategy $\pi$ decides BF node actions, RF node actions are measurable to determine the state, and the reward can be observed over time. It would be an interesting problem to see whether the stateful optimization of Eq. (7) is compatible to that of state-agnostic in Eq. (6). We will later provide an in-depth, comparative analysis of the two approaches.

## IV. Algorithmic Approaches to Find Optimal CCRN Strategies

An optimal CCRN strategy yields node actions that maximize networking efficiency measurable in data throughput and jammed enemy communications. This section describes our algorithmic approaches in state-agnostic and stateful settings. We first consider the MAB formulation for its simplicity in computing strategies from only reward observation over time. In this section, we show that constructing a MAB algorithm (Thompson sampling in particular) with an extreme-valued reward likelihood as guide is ideal because the maximum possible reward should consist of only successful transmission and

$$
\begin{aligned}
&p\Big(I_C^{t+1}, I_D^{t+1}, J_C^{t+1}, J_D^{t+1} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\Big) \\
&\quad= \sum_{\substack{I_C^{t+1}=m_{C1}+m_{C2}+m_{C3}\\ I_D^{t+1}=m_{D1}+m_{D2}+m_{D3}\\ J_C^{t+1}=n_{C1}+n_{C2}+n_{C3}\\ J_D^{t+1}=n_{D1}+n_{D2}+n_{D3}}} p\big(m_{C1}, m_{D1} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) \times p\big(m_{C2}, m_{D2} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) \\
&\qquad\times p\big(m_{C3}, m_{D3} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) \times p\big(n_{C1}, n_{D1} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) \\
&\qquad\times p\big(n_{C2}, n_{D2} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) \times p\big(n_{C3}, n_{D3} | I_C^t, I_D^t, J_C^t, J_D^t, a_B^t, a_R^t\big) \\
&\quad= \sum_{\substack{I_C^{t+1}=m_{C1}+m_{C2}+m_{C3}\\ I_D^{t+1}=m_{D1}+m_{D2}+m_{D3}\\ J_C^{t+1}=n_{C1}+n_{C2}+n_{C3}\\ J_D^{t+1}=n_{D1}+n_{D2}+n_{D3}}} \frac{\binom{\alpha_{C1}^t}{m_{C1}}\binom{\alpha_{D1}^t}{m_{D1}}\binom{N_1^t-\alpha_{C1}^t-\alpha_{D1}^t}{\alpha_{J1}^t+\alpha_{J1}^t-m_{C1}-m_{D1}}}{\binom{N_1^t}{\alpha_{I1}^t+\alpha_{J1}^t}} \times \frac{\binom{\alpha_{C2}^t}{m_{C2}}\binom{\alpha_{D2}^t}{m_{D2}}\binom{N_2^t-\alpha_{C2}^t-\alpha_{D2}^t}{\alpha_{I2}^t-m_{C2}-m_{D2}}}{\binom{N_2^t}{\alpha_{I2}^t}} \\
&\qquad\times \frac{\binom{\alpha_{C3}^t}{m_{C3}}\binom{\alpha_{D3}^t}{m_{D3}}\binom{N_3^t-\alpha_{C3}^t-\alpha_{D3}^t}{\alpha_{J2}^t-m_{C3}-m_{D3}}}{\binom{N_3^t}{\alpha_{J2}^t}} \times \frac{\binom{\alpha_{C1}^t}{n_{C1}}\binom{\alpha_{D1}^t}{n_{D1}}\binom{N_1^t-\alpha_{C1}^t-\alpha_{D1}^t}{\alpha_{I1}^t+\alpha_{J1}^t-n_{C1}-n_{D1}}}{\binom{N_1^t}{\alpha_{I1}^t+\alpha_{J1}^t}} \\
&\qquad\times \frac{\binom{\alpha_{C2}^t}{n_{C2}}\binom{\alpha_{D2}^t}{n_{D2}}\binom{N_2^t-\alpha_{C2}^t-\alpha_{D2}^t}{\alpha_{I2}^t-n_{C2}-n_{D2}}}{\binom{N_2^t}{\alpha_{I2}^t}} \times \frac{\binom{\alpha_{C3}^t}{n_{C3}}\binom{\alpha_{D3}^t}{n_{D3}}\binom{N_3^t-\alpha_{C3}^t-\alpha_{D3}^t}{\alpha_{J2}^t-n_{C3}-n_{D3}}}{\binom{N_3^t}{\alpha_{J2}^t}}
\end{aligned}
\tag{5}
$$

jamming, not a mix of successes and failures. For comparison, we present three classical algorithms known for the stochastic MAB problem.

For the stateful counterpart, we apply Q-learning, a multi-agent reinforcement learning technique, in the Markov game framework. As shown previously, evaluating transition functions of a stateful CCRN is non-rivial due to combinatorial blowup. Thus, we use a value-iteration technique to evaluate the Bellman equations for our stateful algorithms. This section presents pseudo-code for all proposed algorithms.

### A. State-Agnostic CCRN

*1) Deterministic Algorithm:* Algorithm 1 describes an adaptation of Lai and Robbins's asymptotically optimal rules [7] for state-agnostic case. The algorithm keeps track of cumulative reward $R_i^t = \sum_{j=1}^t r_i^j$ and total number of accesses $T_i^t$ for channel $i$. It draws two candidate channels $c_{MPE}$ and $c_{RR}$, based on the maximum point estimate (MPE) criterion (e.g., channel with highest sample mean) and round robin (RR) selection, respectively. The Kullback-Leibler divergence between the two distributions serves a test statistic to finalize the choice.

*2) Indexing Algorithm:* The success of Lai and Robbins depends on the accuracy of $\mathrm{D}_{KL}$ estimated from empirical sampling, which is a challenging task of its own. Another class of MAB algorithms uses index as a computable substitute for $\mathrm{D}_{KL}$. Strictly speaking, indexing algorithm is a subclass of deterministic algorithms. Auer *et al.* [19] formulated an indexing scheme called Upper Confidence Bound (UCB). In Algorithm 2, we present UCB for state-agnostic CCRN.

**Algorithm 1** (Lai & Robbins for State-Agnostic CCRN)

**while** $t < 1$ ▷ initialized offline
  Access each channel at least once
  Record $R_i^t = \sum_{j=1}^t r_i^j$ and $T_i^t$ for every channel $i$
**end**
**while** $t \geq 1$ ▷ online
  Compute $\mu_i = R_i^t / T_i^t \ \forall i$
  Find MPE candidate $c_{MPE} = i^*$ s.t. $\mu_{i^*} = \max \mu_i$
  Find RR candidate $c_{RR} = (t \mod N) + 1$
  **if** $\mathrm{D}_{KL}(p_{RR} \parallel p_{MPE}) > \log(t-1)/T_{c_{RR}}^t$
    Access $c_{MPE}$ and observe $r_{c_{MPE}}^t$
    Update $R_{c_{MPE}}^t$ and $T_{c_{MPE}}^t$
  **else**
    Access $c_{RR}$ and observe $r_{c_{RR}}^t$
    Update $R_{c_{RR}}^t$ and $T_{c_{RR}}^t$
  **end**
**end**

**Algorithm 2** (UCB for State-Agnostic CCRN)

**while** $t < 1$ ▷ initialized offline
  Same as **Algorithm 1**
**end**
**while** $t \geq 1$ ▷ online
  Compute point estimate $\mu_i = R_i^t / T_i^t \ \forall i$
  Compute index $g_i = \mu_i + \sqrt{\alpha \log \frac{t}{T_i^t}} \ \forall i$
  Access channel $i^* = \arg\max_i g_i$
  Update $R_{i^*}^t$ and $T_{i^*}^t$
**end**

Despite its simpler form, UCB results in the margin of error that decays logarithmically in time.

*3) Randomized Algorithm:* Randomization serves an effective means to simultaneously explore and exploit. In particular, we focus on a probability matching heuristic known

**Algorithm 3** (Thompson Sampling for State-Agnostic CCRN)

**Require:** $d = \{x, a, r\}$ for context $x$, action $a$, reward $r$, estimator $p(\theta|d) \propto p(r|x, a, \theta)p(\theta)$ parameterized by $\theta$

**while** $t \geq 1$ ▷ online
  Acquire $x^t$
  Draw $\theta^t \sim p(\theta)$
  Choose $a^t$ to access $i^* = \arg\max_i \mathbb{E}[r_i^t|x^t, \theta^t]$
  Observe actual $r^t$
  Update $d = d \cup \{x^t, a^t, r^t\}$
  Update $p(\theta) = p(\theta|d)$
**end**

as Thompson sampling [9], originally proposed in 1933 for stochastic MAB problems. Thompson sampling selects actions according to some optimal probability that is believed to yield the maximum reward. The strategy maker based on Thompson sampling is required to observe the actual outcome of a selected action in order to adjust the belief. For this reason, the algorithm is also called *posterior* sampling.

Despite its often superior performance over other algorithms, Thompson sampling lacks a rigorous theoretical analysis. The full proof on the convergence properties and bounds still remains to be an open problem. Thompson sampling reemerged in recent machine learning literature such as Agrawal and Goyal [20], which provides the most thorough mathematical treatment available to date. Algorithm 3 describes Thompson sampling in its naturally Bayesian form.

We note that running Algorithms 1, 2, and 3 once determines an action for one node only. For determining multiple node actions, we use the following technique. Select the best channel from an algorithm and assign it to a node of interest. Remove the selected channel and rerun the algorithm for the *remaining* channels. Select the best channel among the remaining and assign it to the next node of interest. We repeat the process until we allocate all nodes (comm or jammer).

### B. Stateful CCRN

In the model-based reinforcement learning, an agent explicitly learns the transition probabilities of the underlying Markov chain that characterizes the system. The model-free learning, on the other hand, needs to evaluate the actual action executed at a given state. We choose Q-learning [11], a model-free, temporal-difference learning as our baseline method to compute an optimal strategy for the stateful case.

Q-learning evaluates the *quality* of an action possible at a particular state and the *value* of that state. The quality function $Q(\cdot)$ is a function of state $s$ and action $a$, and of only $s$ for the value function $V(\cdot)$. The Bellman equations [21] characterize such optimization:

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} p\big(s'|s, a\big)V\big(s'\big) \tag{8}$$

$$V(s) = \max_{a'} Q\big(s, a'\big) \tag{9}$$

The key strength of Q-learning is the value iteration technique that an agent performs an update $Q(s, a) = R(s, a) + \gamma V(s')$ (note that $s'$ is the next state transited from $s$) in place of Eq. (8) without explicit knowledge of transition probability $p(s'|s, a)$, which is often too complex (e.g., Eq. (5)) to compute as discussed in Section III-C. Noting that a strategy $\pi$ is the probability distribution of $a$ at state $s$, linear programming can solve for $\pi^* = \arg\max_\pi \sum_a Q(s, a)\,\pi$, reflecting the value maximization in Eq. (9).

**Algorithm 4** (Q-learning for Stateful CCRN)

**Require:** $Q(s, a_B, a_R) = 1, V(s) = 1, \pi(s, a_B) = \frac{1}{|\mathcal{A}|}$ $\forall$ state $s \in \mathcal{S}$, BF action $a_B \in \mathcal{A}$, RF action $a_R \in \mathcal{A}$; learning rate $\alpha < 1$ with decay $\lambda \leq 1$ ($\alpha$, $\lambda$ nonnegative)

**while** $t \geq 1$
  Draw $a_B^t \sim \pi(s^t)$ and execute
  Observe $r_B^t$
  Estimate $a_R^t$ given observed reward
  Compute $s^{t+1}$
  $Q(s^t, a_B^t, a_R^t) = (1 - \alpha)Q(s^t, a_B^t, a_R^t) + \alpha(r_B^t + \gamma V(s^{t+1}))$
  linprog: $\pi(s^t, .) = \arg\max_\pi \sum_{a_B} \pi(s^t, a_B)Q(s^t, a_B, a_R)$
  Update $V(s^t) = \min_{a_R} \sum_{a_B} \pi(s^t, a_B)Q(s^t, a_B, a_R)$
  Update $\alpha = \lambda \times \alpha$
**end**

In Algorithm 4, we present a baseline Q-learning algorithm that searches for an optimal strategy of the Blue Force (BF) network. Note that the BF and RF networks are symmetrical (e.g., same number of comm nodes and jammers), thus have the same node action space. In Section IV-D, we propose three variations of Q-learning algorithms that improve the performance of Algorithm 4.

### C. New State-Agnostic Algorithm

For the state-agnostic approach, we propose a new MAB algorithm based on extreme value theory [22], conjugate priors, and Thompson sampling.

*1) Distribution of Maximum Reward Sequence:* Let $Y^t = \max\{r_1^t, \ldots, r_N^t\}$ where $r_i^t$ represents the reward from channel $i$ at $t$. Since the sequence $Y^1, Y^2, \ldots, Y^t$ consists only of the maximum channel reward each time, it must have achieved the distribution $p^*$ in Eq. (2). Furthermore, the sequence should result in an *upper bound* of the optimal mean reward $\mu^*$. Therefore, we need a strategy $\sigma$ to empirically follow the distribution of $Y^t$. But how is it distributed?

Fisher and Tippet [23] and Gnedenko [24] proved the existence of limiting distributions for block maxima (or minima) of random variables. Their findings became the foundation of extreme value theory used widely in financial economics.

*Theorem 1 (Fisher & Tippett, Gnedenko):* Let $X_1, \ldots, X_n$ be a sequence of i.i.d. random variables and $M_n = \max\{X_1, \ldots, X_n\}$. If real number pairs $(a_n, b_n)$ exist such that $a_n, b_n > 0$ and $\lim_{n\to\infty} P(\frac{M_n - b_n}{a_n} \leq x) = F(x)$, where $F(\cdot)$ is a non-degenerate distribution function, then the limiting distribution $F(\cdot)$ belongs to only Fréchet, Gumbel, or Weibull family of probability distribution functions.

*Proof:* See Fisher and Tippet [23] and Gnedenko [24]. ■

*2) Conjugate Priors:* In Bayesian inference, the posterior is updated by the observed likelihood given the prior distribution:

$$\underbrace{p(\theta|r)}_{\text{posterior}} \propto \underbrace{p(r|\theta)}_{\text{likelihood}} \times \underbrace{p(\theta)}_{\text{prior}}$$

TABLE V
BAYESIAN CONJUGACY UNDER EXTREME-VALUED LIKELIHOOD

| Distribution family | Likelihood model | Conjugate priors |
|---|---|---|
| Fréchet | Pareto | Gamma |
| | Lognormal | Gamma or normal |
| Gumbel | Exponential | Gamma |
| | Normal | Normal |
| | Gamma | Gamma |
| Weibull | Weibull | Inverse gamma |
| | Beta | Unknown |

**Algorithm 5** (CCRN State-Agnostic Algorithm)

**Require:** $a_i, b_i = 0\ \forall i$
**while** $t < 1$ ▷ initialized offline
  Access each channel until $a_i, b_i \neq 0\ \forall i$, where $a_i$ and $b_i$ are sample reward mean and variance
**end**
**while** $t \geq 1$ ▷ online
  Draw $\theta_i \sim$ inv-gamma($a_i$,$b_i$)
  Estimate $\hat{r}_i =$ weibull($\theta_i$,$\beta_i$) $\forall i$ for given $0.5 \leq \beta_i \leq 1$
  Access channel $i^* = \arg\max_i \hat{r}_i$
  Observe actual $r^t_{i^*}$ to update $\{R^t_{i^*}, T^t_{i^*}\}$
  Update $a_{i^*} = a_{i^*} + T^t_{i^*}$, $b_{i^*} = b_{i^*} + \sum_t (r^t_{i^*})^{\beta_{i^*}}$
**end**

When the probabilistic model for the likelihood is known, we can set the prior and posterior distributions conveniently of the *same* family of functions. This is known as conjugate prior. Since the reward distribution under our search is extreme-valued, our likelihood choices are left to Fréchet, Gumbel, or Weibull distributions. Table V summarizes the conjugate priors having an extreme-valued likelihood distribution [25].

*3) The Algorithm:* Algorithm 5 proposes a new state-agnostic approach. The algorithm performs Thompson sampling following an extreme-valued likelihood and updates the posterior distribution based on its conjugate prior. However, we need to decide on which extreme value distribution is suitable for CCRN.

Since both Fréchet and Gumbel distributions model *unbounded* random variables, we adopt a *Weibull* likelihood with the inverse gamma conjugate prior (see Table V), reasoning that the maximum reward value for CCRN should be *finite*. The lack of theoretical analysis on Thompson sampling makes it difficult to justify our design choice. In Section V, we show an empirical evidence that backs up our choice.

A Weibull distribution has finite endpoints. Its conjugate prior, the inverse gamma distribution, has two hyperparameters $a, b > 0$. Our algorithm draws the scale parameter $\theta$ from the inverse gamma prior $p(\theta|a,b) = \frac{b^{a-1}e^{-b/\theta}}{\Gamma(a-1)\theta^a}$ for $\theta > 0$ where $a$ and $b$ are the sample mean and variance of the reward of a channel, and $\Gamma(\cdot)$ the gamma function (not to be confused with the Lai & Robbins's regret $\Gamma$ in Eq. (1)). The Weibull random variable generated by $\theta$ drawn from the prior estimates the expected reward for the channel. After observing the actual reward, the posterior update follows.

### D. New Stateful Algorithm

Before presenting the new stateful algorithm, we decompose the CCRN Markov game into two subgames, namely

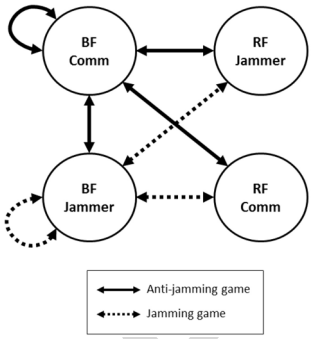



Fig. 4. Inter-node relationships in antijamming and jamming subgames.

*antijamming* and *jamming*, and examine inter-node relationship. Fig. 4 illustrates the decomposition. Each circle represents a unique node type from the Blue Force (BF) and Red Force (RF) networks. For clarity, we explain the relationships in the BF network's perspective—i.e., the same relationships can be derived for the RF network.

Antijamming game is primarily played between a BF comm node and an RF jammer. That is, the key objective of anti-jamming game for BF is to maximize data throughput by avoiding hostile RF jamming. Antijamming game is also played between a BF comm node and an RF comm node since the BF comm throughput is subject to degradation when the both comm nodes collide on the same channel. Fig. 4 also suggests that there are antijamming games among BF comm nodes and between a BF comm node and a BF jammer. When multiple BF comm nodes collide on the same channel, there will be a similar loss of throughput. Moreover, the BF network wastes corresponding comm resources because of the collision, thus it should be avoided. We call *misjamming* when a BF jammer jams BF's own comm nodes. Misjamming is devastating because it incurs a loss of throughput and wastes both comm and jamming resources of the network.

To avoid the collision among the BF nodes and misjamming, we rely on network control to coordinate the node actions. A rational strategy must first check any conflicting node actions within the same network. However, imperfect sensing and signaling can still lead to a collision and misjamming.

In jamming game, a BF jammer is trying to jam an RF comm node in order to suppress the RF data throughput. A BF jammer can target a data channel frequently accessed by the RF comm nodes. Alternatively, it can aim for an RF control channel, which would result a small immediate reward but a potentially larger value in the future by blocking subsequent RF data traffic. Additionally, jamming game is played between a BF jammer and an RF jammer, and among multiple BF jammers in order to minimize energy resources. There will be no need for a BF jammer to act on a channel already under jamming by either another BF jammer or an RF jammer.

Now we describe three comparable, new algorithms for stateful CCRN using Minimax [26], Nash [27], and Friend-or-foe [28] Q-learning methods.

*1) Minimax-Q Learning:* Minimax-Q assumes a zero-sum game that implies $Q_B(s^t, a^t_B, a^t_R) = -Q_R(s^t, a^t_B, a^t_R) = Q(s^t, a^t_B, a^t_R)$. This holds tightly for the jamming subgame

$$V(s^t) = \max_{\pi_{B1}(A_{B,comm})} \min_{a^t_{R,jam}} \max_{\pi_{B2}(A_{B,jam})} \min_{a^t_{R,comm}} \sum_{a^t_B} Q(s^t, a^t_B, a^t_R)\, \pi_B(a^t_B) \tag{10}$$

$$\begin{aligned} Q(s^t, a^t_B, a^t_R) &= r(s^t, a^t_B, a^t_R) + \gamma \sum_{s^{t+1}} T\left(s^t, a^t_B, a^t_R, s^{t+1}\right) V\left(s^{t+1}\right) \\ &= r(s^t, a^t_B, a^t_R) + \gamma \sum_{s^{t+1}} p\left(s^{t+1}|s^t, a^t_B, a^t_R\right) V\left(s^{t+1}\right) \end{aligned} \tag{11}$$

$$Q(s^t, a^t_B, a^t_R) = (1-\alpha^t) Q(s^t, a^t_B, a^t_R) + \alpha^t \left[ r(s^t, a^t_B, a^t_R) + \gamma V\left(s^{t+1}\right) \right] \tag{12}$$

$$\begin{aligned} Q(s^t, a^t_B, a^t_R) = &(1-\alpha^t)\, Q(s^t, a^t_B, a^t_R) \\ &+ \alpha^t \left[ r(s^t, a^t_B, a^t_R) + \gamma \max_{\pi_{B1}(A_{B,comm})} \min_{a^t_{R,jam}} \max_{\pi_{B2}(A_{B,jam})} \min_{a^t_{R,comm}} Q(s^t, a^t_B, a^t_R)\, \pi_B(a^t_B) \right] \end{aligned} \tag{13}$$

$$V(s^t) = \max_{\pi_{B1}(A_{B,comm})} \min_{\hat{\pi}_{R2}(A_{R,jam})} \max_{\pi_{B2}(A_{B,jam})} \min_{\hat{\pi}_{R1}(A_{R,comm})} \sum_{a^t_B} Q(s^t, a^t_B, a^t_R)\, \pi_B(a^t_B)\, \hat{\pi}_R(a^t_R), \tag{14}$$

$$\begin{aligned} Q(s^t, a^t_B, a^t_R) = &(1-\alpha^t)\, Q(s^t, a^t_B, a^t_R) \\ &+ \alpha^t \left[ r(s^t, a^t_B, a^t_R) + \gamma \max_{\pi_{B1}(A_{B,comm})} \min_{\hat{\pi}_{R2}(A_{R,jam})} \max_{\pi_{B2}(A_{B,jam})} \min_{\hat{\pi}_{R1}(A_{R,comm})} Q(s^t, a^t_B, a^t_R)\, \pi_B(a^t_B)\, \hat{\pi}_R(a^t_R) \right] \end{aligned} \tag{15}$$

$$\begin{aligned} Q_B(s^t, a^t_B, a^t_R) = &(1-\alpha^t)\, Q_B(s^t, a^t_B, a^t_R) \\ &+ \alpha^t \left[ r(s^t, a^t_B, a^t_R) + \gamma \max_{\pi_{B1}(A_{B,comm})} \min_{\hat{\pi}_{R2}(A_{R,jam})} \max_{\pi_{B2}(A_{B,jam})} \min_{\hat{\pi}_{R1}(A_{R,comm})} Q_B(s^t, a^t_B, a^t_R)\, \pi_B(a^t_B)\, \hat{\pi}_R(a^t_R) \right] \end{aligned} \tag{16}$$

$$\begin{aligned} Q_R(s^t, a^t_B, a^t_R) = &(1-\alpha^t)\, Q_R(s^t, a^t_B, a^t_R) \\ &+ \alpha^t \left[ r(s^t, a^t_B, a^t_R) + \gamma \max_{\pi_{B1}(A_{B,comm})} \min_{\hat{\pi}_{R2}(A_{R,jam})} \max_{\pi_{B2}(A_{B,jam})} \min_{\hat{\pi}_{R1}(A_{R,comm})} Q_R(s^t, a^t_B, a^t_R)\, \pi_B(a^t_B)\, \hat{\pi}_R(a^t_R) \right] \end{aligned} \tag{17}$$

where the jammer's gain is offset precisely by the opponent comm throughput loss. In order to solve antijamming and jamming subgames jointly, we propose a slight modification to the original Minimax-Q algorithm in Littman [26]. First, we divide the strategy of BF network $\pi_B$ into its antijamming and jamming substrategies, $\pi_{B1}$ and $\pi_{B2}$. Then, we add an extra minimax operator to our value function in Eq. (10), as shown at the top of this page. The modified Q-function in Eq. (11), as shown at the top of this page, can be computed iteratively, using Eqs. (12) and (13), as shown at the top of this page. Learning rate $\alpha$ decays over time such that $\alpha^{t+1} = \alpha^t \cdot \delta$ according to decay factor $0 < \delta < 1$.

*2) Nash-Q Learning:* Nash-Q [27] can solve a general-sum game in addition to zero-sum games. This makes an important distinction to Minimax-Q although the Nash-Q value function for a zero-sum game in Eq. (14), as shown at the top of this page, is different from Eq. (10) by only one extra term $\hat{\pi}_R(a^t_R)$. This means that Nash-Q requires to estimate the policy of the opponent. BF network needs to learn $\hat{\pi}_{R1}$ and $\hat{\pi}_{R2}$, the antijamming and jamming substrategies of RF network. The Q-function for the zero-sum Nash-Q is given by Eq. (15), as shown at the top of this page. For a general-sum game, the BF agent should compute $Q_B$ and $Q_R$ separately at the same time while observing its reward and estimating the RF's by Eqs. (16) and (17), as shown at the top of this page. The objective of Nash-Q is to find a joint *equilibrium* under the mixed strategies $(\pi_B, \hat{\pi}_R)$.

*3) Friend-or-Foe Q-Learning:* Although Nash-Q is applicable to both zero-sum and general-sum games, its convergence guarantee is considered too restrictive [28]. Littman [28] instead proposed Friend-or-foe Q-learning (FFQ). FFQ is a computational enhancement and provides better convergence properties by relaxing the restrictive conditions of Nash-Q. For this relaxation, FFQ requires extra information that other agents in the game should be classified *friendly* or *hostile*.

In FFQ, the BF agent maintains only one Q-function:

$$\begin{aligned} Q_B(s^t, a^t_B, a^t_R) = &(1-\alpha^t)\, Q_B(s^t, a^t_B, a^t_R) \\ &+ \alpha^t \left[ r(s^t, a^t_B, a^t_R) + \gamma \Psi_B \right] \end{aligned} \tag{18}$$

If the BF agent encounters an agent that is identified as a friend, the Q-function for the BF network is updated by

$$\Psi_B = \max_{a^t_B, a^t_R} Q_B(s^t, a^t_B, a^t_R) \tag{19}$$

On the other hand, if the BF agent encounters an agent that is identified as a foe, the Q-function is updated under the minimax criterion

$$\Psi_B = \max_{\pi_B(A_B)} \min_{\hat{\pi}_R(A_R)} \sum_{a^t_B} Q_B(s^t, a^t_B, a^t_R)\, \pi_B(a^t_R). \tag{20}$$

## V. Simulation Results

We evaluate the state-agnostic and stateful approaches against non-cognitive static and random strategies. Then, we

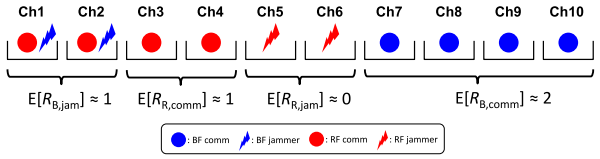



Fig. 5. Illustration of ideal channel access strategy against static RF network strategy when $C = 4$ and $J = 2$ in centralized scenario.

evaluate the case where one of the two CCRNs employs the state-agnostic algorithm and stateful algorithm for the other.

### A. Evaluation of State-Agnostic Approach

*1) Scenarios and Metric:* We evaluate the centralized and distributed control scenarios for BF and RF networks in a custom-built MATLAB simulator. The BF network runs Algorithms 1 (Lai and Robbins), 2 (UCB), 3 (Thompson Sampling), and 5 (proposed) described in Sections IV-A and IV-C while the RF network is configured with *static* and uniformly *random* strategies. In static strategy, RF nodes initially choose to access some channels and continue to access the same channels throughout. Random strategy chooses a uniformly random channel for each RF node at each time slot.

Under the centralized scenario, we assume that the central decision maker has perfect knowledge (i.e., sensing results from all nodes) in the decision making as expressed by Eq. (3). Under the distributed scenario, each node in the network makes its own decision by using its sensing results only (no information sharing) as described by Eq. (4).

We adopt the average reward per channel as the performance evaluation metric for a CCRN:

$$\bar{R}^t = \frac{1}{N \cdot t} \sum_{j=1}^{t} \sum_{i=1}^{N} r_i^j$$

where $r_i$ is the $i$th channel reward, and there are $N$ channels in the spectrum. We use the channel reward model described by Table I in Section II-D.

*2) Results:* The spectrum has $N = 10$ channels. For each CCRN, we vary the number of comm nodes $C = 2, 4, 6, 8$, but fix the number of jammers to $J = 2$. Comm nodes have a transmit probability $p_{\text{Tx}} = 0.5$ whereas jammers jam with probability 1. We run $t = 1,000$ time slots and measure steady-state, cumulative average reward per channel for comparison.

To better understand simulated results, we present an illustrative example for an optimal BF strategy against the *static* RF network with $C = 4$ and $J = 2$ in Fig. 5. In this example, RF comm nodes are fixed at channels 1, 2, 3, 4, and its 2 jammers at channels 5 and 6, leaving the rest of channels 7, 8, 9, 10 free of RF actions. Through learning by sensing all channels over time, an optimal BF strategy should place its two jammers somewhere between channel 1 and 4. Because the comm transmit probability at any given slot is 0.5 for all RF and BF comm nodes, the maximum average reward earned by the BF jammers should be $\mathbb{E}[R_{B,jam}] \approx 0.5 \times 2 = 1$. The BF comm nodes at channels 7, 8, 9, and 10 should earn

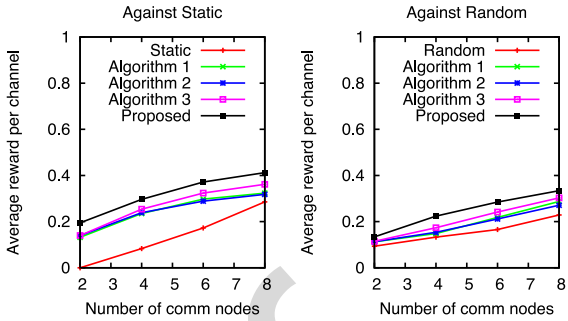



Fig. 6. Average reward performance comparison in centralized scenario for state-agnostic approaches.

$\mathbb{E}[R_{B,comm}] \approx 0.5 \times 4 = 2$ (because of the comm transmit probability of 0.5, the BF comm reward at each time slot come from two channels on the average). In summary, the total reward for BF network in this example is approximately 3, which is normalized to $\frac{3}{N} = \frac{3}{10} = 0.3$ (per channel), and similarly for RF network, average total reward is $\frac{1}{10} = 0.1$.

Fig. 6 compares the performance of tested algorithms in the centralized scenario. The curves for static and random strategies were obtained while they were tested against the proposed algorithm. We can clearly observe performance advantage of our algorithm over Algorithms 1, 2, and 3. The proposed algorithm (i.e., Algorithm 5) can learn static transmission and jamming patterns effectively. Static strategy yields near-zero reward at $C = 2$. As we have fixed $J = 2$, static strategy can realize nonzero rewards when $C > 2$.

The results for the static and proposed algorithms at $C = 4$ in Fig. 6 confirm the reward performance of the example illustrated in Fig. 5. The proposed algorithm consistently outperforms the others including the ones adapted from existing MAB solutions. As the number of comm nodes per network increases, the state-agnostic CCRN strategies face fewer options in choosing node actions. As a result, we observe that difference in the reward performance of the algorithms becomes smaller. To increase the potential reward, we need to explore more channels in optimizing the strategy.

Learning is harder against random strategy because randomization gives an effective exploration mechanism. As a result, the reward performance of the algorithms decreases. Random strategy, however, can only explore. The lack of exploitation explains its poorer performance compared to Algorithms 1, 2, 3, and 5.

In Fig. 7, we compare the performance in the distributed scenario that lacks explicit intra-network coordination. The reward performance becomes worse for all of the algorithms due to collisions and misjamming.

### B. Evaluation of Stateful Approach

*1) Scenarios and Metric:* As discussed earlier, there are model-based and model-free stateful approaches. We use the Markov game model derived in Section III-C to numerically simulate the CCRN state transition probabilities. We draw a reduced state diagram in Fig. 8, using only the 10 most probable state transitions computed from Eq. (5). Recall that a CCRN state is represented by the tuple $\langle I_C, I_D, J_C, J_D \rangle$, where

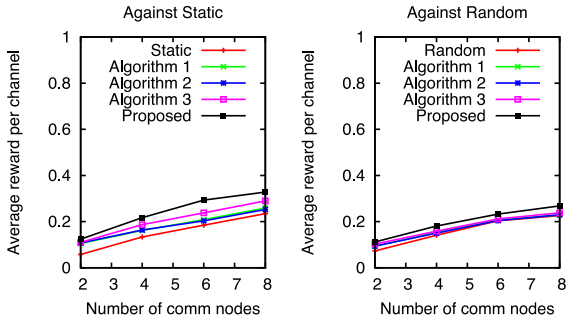



Fig. 7. Average reward performance comparison in distributed scenario for state-agnostic approaches.

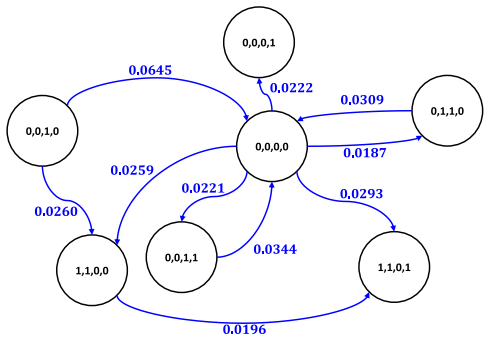



Fig. 8. Top 10 state transitions computed analytically using Eq. (5).

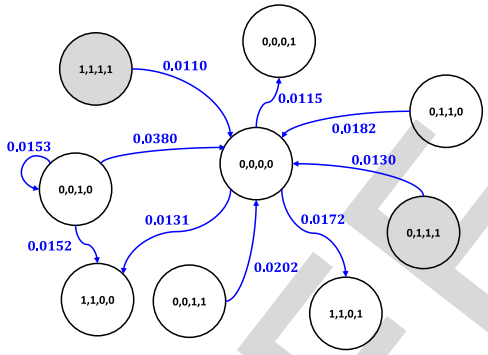



Fig. 9. Top 10 state transitions determined from Monte Carlo simulation (shaded states different from the analytical result of Fig. 8).

$I_C$ is the number of collided control channels, $I_D$ the number of collided data channels, $J_C$ the number of jammed control channels, and $J_D$ the number of jammed data channels.

In Fig. 9, we draw a similar state diagram, which is resulted from Monte Carlo sampling. We notice some differences between the two. First, the state diagram from the analytical evaluation is more compact and concentrated as it has two fewer states (see the two highlighted states in Fig. 9) and larger transition probability values overall. One reason for larger probability values in Fig. 8 is that our Monte Carlo simulation implements *uniform* sampling from the action space at each visited state, which does not necessarily reflect the behavior of a rational strategy maker. Subsequently, the resulting transitions are more likely to be distributed with *less* probability measures. As mentioned earlier, we can improve our Monte Carlo approach with MCMC. However, the optimal MCMC design for stateful CCRN is out of scope of this paper.

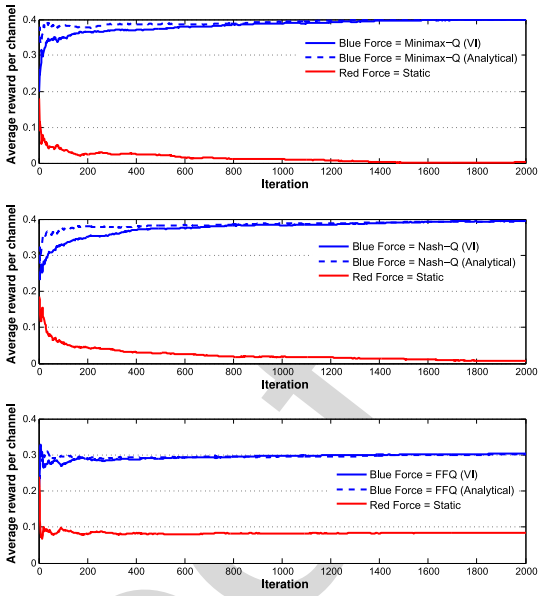



Fig. 10. Average reward performances of Minimax-Q, Nash-Q, and FFQ at BF network against *static* strategy at RF network.

For model-free approach, we configure BF network to run strategies based on Q-learning. We apply Minimax-Q and Nash-Q learning algorithms under the centralized network control while we use FFQ for the distributed case. RF network (for both model-based and model-free) is configured to run the same static and random strategies that we use to evaluate the state-agnostic case. Lastly, we use the same metric as the state-agnostic case to evaluate the reward performance of the stateful algorithms.

The simulation parameters are as follows. There are $N = 10$ channels in the spectrum. Both BF and RF networks have 2 comm nodes and 2 jammers. We set each comm node's Tx and each jammer's jamming with probability 1. We simulate each run for 2,000 time slots and observe reward performances.

*2) Results:* We plot the average reward performances for BF network employing Minimax-Q, Nash-Q, and FFQ against RF network's *static* strategy over time in Fig. 10. The solid curve shows the result under the model-free approach based on value-iterated (VI) Q-learning. The dashed curve shows the result under the model-based approach using the state transitions of Fig. 8 (i.e., numerical evaluation of Eq. (5)). In the steady state, we observe that the reward performances of the two approaches converge.

The results for Minimax-Q and Nash-Q, which are obtained under the centralized network control, indicate the optimal performance against the static strategy. On the other hand, the FFQ strategy displays suboptimal reward performance due to possible collisions and misjamming in the distributed network control.

Fig. 11 presents the average reward performances for BF network employing Minimax-Q, Nash-Q, and FFQ against RF network's *random* strategy. The performance of Q-learning algorithms decreases against the random strategy as observed in the state-agnostic case.

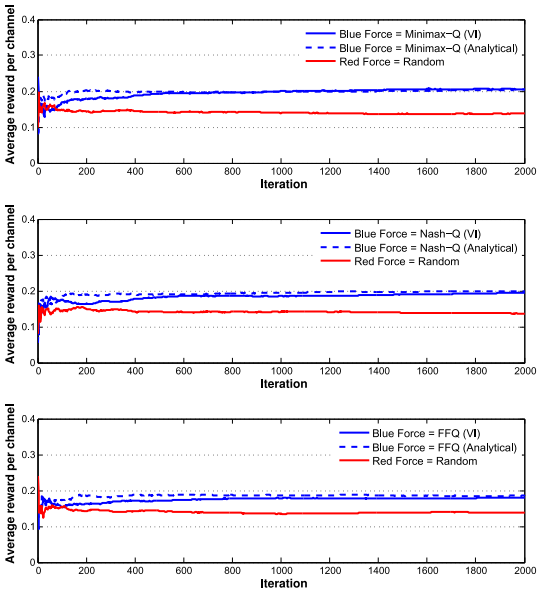



Fig. 11. Average reward performances of Minimax-Q, Nash-Q, and FFQ at BF network against *random* strategy at RF network.

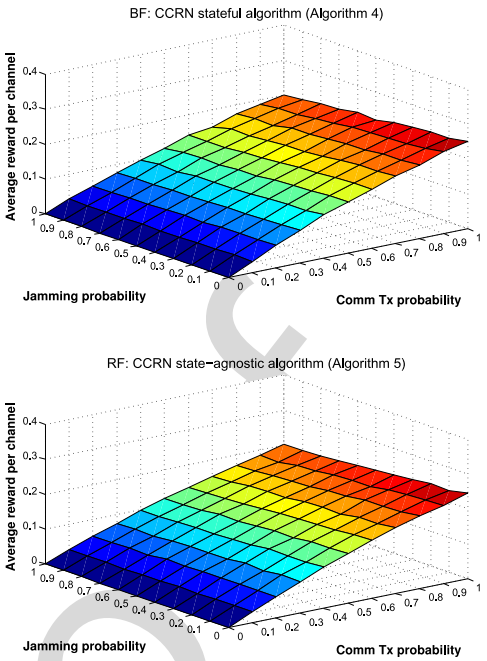



Fig. 12. Average reward performances of Minimax-Q (stateful approach) at BF network against Algorithm 4 (state-agnostic approach) at RF network under the centralized network control.

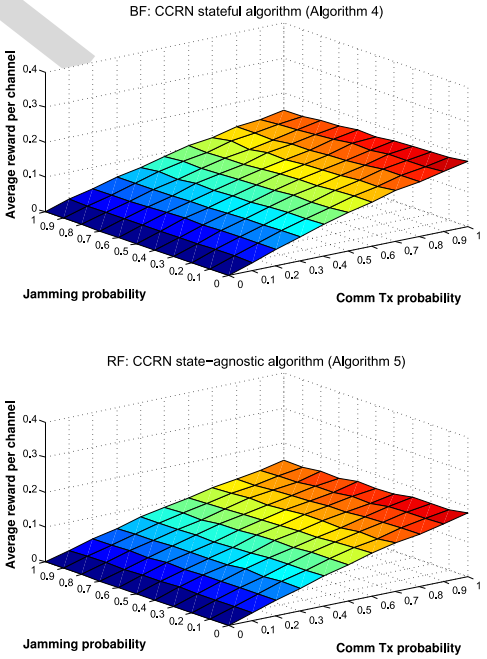



Fig. 13. Average reward performances of Minimax-Q (stateful approach) at BF network against Algorithm 4 (state-agnostic approach) at RF network under the distributed network control.

### C. State-Agnostic vs. Stateful Approaches

We now evaluate the reward performance for a scenario when one network employs the state-agnostic strategy and the other the stateful. We configure the BF network with Minimax-Q and the RF network with Algorithm 5 with simulation parameters $N = 10$, $C = 4$, and $J = 2$. We vary the comm Tx and jamming probabilities from 0 to 1 for the two CCRNs and run under both the centralized and distributed network control mechanisms.

Fig. 12 depicts the reward performances of the two networks as a function of comm Tx and jamming probabilities under the centralized control. The performances of both networks are comparable as the two approaches in the steady-state seem to achieve similar learning. In Fig. 13, we show the reward performances of the two networks under the distributed control. Again, the performances are on par because both networks can learn about each other's strategy under the distributed case.

## VI. Related Work

The state-agnostic approach of this paper is developed under the stochastic MAB framework. In 1933, Thompson [9] introduced a stochastic MAB problem and proposed an optimal heuristic known as Thompson sampling, which remains to be an effective action selection strategy. Robbins [29] 1952 presented the first sequential analysis of the single-player MAB problem. In Bellman [30] 1954, MAB problems were formulated as a class of Markov decision process (MDP). Gittins [31] 1979 proved the existence of a Bayes optimal indexing scheme for MAB problems if they can be modeled as a stationary MDP. Lai and Robbins [7] 1985 introduced the notion of regret, derived its lower bound using the Kullback-Leibler divergence, and constructed asymptotically optimal allocation rules. Anantharam *et al.* [16] 1987 extended Lai & Robbins for multi-player. Whittle [32] 1988 introduced PSPACE-hard restless MAB problems and showed that suboptimal indexing schemes are possible. Rivest and Yin [33] 1994 proposed Z-heuristic that achieved a better empirical performance than Lai and Robbins. Auer *et al.* [19] 2002 proposed Upper Confidence Bound (UCB), an optimistic indexing scheme.

The foundation of our stateful approach is reinforcement learning [34], which extends beyond self-confined views of the classical Markov Decision Process in which an agent's environment is stationary and contains no other agents. Q-learning

was originally proposed by Watkins and Dayan [11]. Littman [26] proposed Minimax-Q learning for a zero-sum two-player game. Littman and Szepesvári [18] showed that Minimax-Q converges to the optimal value suggested by game theory. Hu and Wellman [27] described Nash-Q that was distinguished from Minimax-Q by solving a general-sum game with a Nash equilibrium computation in its learning algorithm. Nash-Q has more general applicability, but its assumptions on the sufficient conditions for convergence guarantee are known to be restrictive. Friend-or-foe Q-learning (FFQ) [28] converges precisely to the steady-state value that Nash-Q guarantees. The key improvement of FFQ is relaxation of the restrictive conditions that Nash-Q has, but FFQ requires *a priori* knowledge on other agents identified as either a friend or foe.

This paper considers some similar problems discussed by Wang *et al.* [2] such as finding a strategy against hostile jamming. They formulated a stochastic antijamming game played between the secondary user and a malicious jammer, provided sound analytical models, and applied unmodified Minimax-Q learning to solve for the optimal antijamming strategy. Our work is novel and differentiated from existing work by the following. We embrace the notion of friendly jammers and provide an integrated antijamming-jamming strategy for cognitive radio networks under competition in a hostile environment. We use jamming as a means to cope with critical situations assumed in tactical mobile networking. At the same time, we try to avoid the hostile jammers that pose a serious threat to the network's comm activities in another optimization. We promote the notion of strategic jamming enabled by reinforcement learning. We modify existing Q-learning algorithms to solve for optimal antijamming and jamming strategies jointly. Lastly, we remark that this paper unites our previous approaches [12], [13] for competing cognitive radio networks.

## VII. Conclusion

We have described Competing Cognitive Resilient Networks (CCRNs) that operate in a hostile environment to maximize data throughput while simultaneously suppressing opponent's comm activities. We have provided two compatible, yet distinguished approaches for an optimal strategy that can coordinate the joint comm and jamming actions for a CCRN.

In our state-agnostic approach, we have adopted the MAB framework and extended classical solutions for CCRN. In addition, we have proposed a new algorithm that outperforms the classical solutions. The new algorithm builds on Thompson sampling in its Bayesian form with an enhancement from the extreme value theory. Our performance results indicate that the proposed state-agnostic algorithm proves to be most effective in addressing the exploration-exploitation tradeoff when compared to other algorithms.

For the stateful approach, we have modeled the dynamics of CCRNs using the Markov game framework and decomposed it to antijamming and jamming subgames. We have derived an analytical expression to evaluate model-based reinforcement learning that can solve for an optimal strategy. We also have applied model-free reinforcement learning, namely Minimax-Q, Nash-Q, Friend-or-foe Q (FFQ) algorithms, to achieve optimal strategies. With Monte Carlo sampling, we have demonstrated that the model-based and model-free methods approach to the same steady-state value.

Both state-agnostic and stateful approaches can be applied to centralized or distributed network control mechanisms. The numerical results suggest the superior performance achieved for the case of centralized where CCRN nodes are cooperative and their actions are coordinated through a single entity. On the other hand, the performance under the distributed control mechanism suffers from collisions and misjamming.

Based on the reward performance, both state-agnostic and stateful approaches are significantly better than rudimentary strategies such as static and random. However, when competing against each other, the state-agnostic and stateful approaches achieve the same performance. Our future work includes various improvements for cognitive learning such as faster learning, better convergence properties, and lower computational complexity.

## Acknowledgment

Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Office of the Secretary of Defense or the United States Government.

## References

[1] Q. Zhao and B. M. Sadler, "A survey of dynamic spectrum access," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 79–89, May 2007.

[2] B. Wang, Y. Wu, K. J. R. Liu, and T. C. Clancy, "An anti-jamming stochastic game for cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 4, pp. 877–889, Apr. 2011.

[3] R. Chen, J.-M. Park, and J. H. Reed, "Defense against primary user emulation attacks in cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 25–37, Jan. 2008.

[4] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal jamming attack strategies and network defense policies in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 8, pp. 1119–1133, Aug. 2010.

[5] S. Khattab, D. Mosse, and R. Melhem, "Modeling of the channel-hopping anti-jamming defense in multi-radio wireless networks," in *Proc. Int. Conf. Mobile Ubiquitous Syst. Comput. Netw. Services (Mobiquitous)*, Dublin, Ireland, 2008, Art. no. 25.

[6] V. Navda, A. Bohra, S. Ganguly, and D. Rubenstein, "Using channel hopping to increase 802.11 resilience to jamming attacks," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Anchorage, AK, USA, 2007, pp. 2526–2530.

[7] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Adv. Appl. Math.*, vol. 6, no. 1, pp. 4–22, 1985.

[8] L. Lai, H. El-Gamal, H. Jiang, and H. V. Poor, "Cognitive medium access: Exploration, exploitation, and competition," *IEEE Trans. Mobile Comput.*, vol. 10, no. 2, pp. 239–253, Feb. 2011.

[9] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, nos. 3–4, pp. 285–294, 1933.

[10] L. S. Shapley, "Stochastic games," *Proc. Nat. Acad. Sci. USA*, vol. 39, no. 10, pp. 1095–1100, 1953.

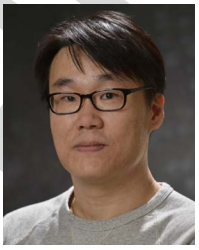
[11] C. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3, pp. 272–292, 1992.

[12] Y. Gwon, S. Dastangoo, C. Fossa, and H. T. Kung, "Competing mobile network game: Embracing antijamming and jamming strategies with reinforcement learning," in *Proc. IEEE Conf. Commun. Netw. Security (CNS)*, 2013, pp. 28–36.

[13] Y. Gwon, S. Dastangoo, and H. T. Kung, "Optimizing media access strategy for competing cognitive radio networks," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, Atlanta, GA, USA, 2013, pp. 1215–1220.
[14] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proc. ACM Int. Symp. Mobile Ad Hoc Netw. Comput. (MobiHoc)*, Chicago, IL, USA, 2005, pp. 46–57.
[15] M. Pajic and R. Mangharam, "Anti-jamming for embedded wireless networks," in *Proc. Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, San Francisco, CA, USA, 2009, pp. 301–312.
[16] V. Anantharam, P. Varaiya, and J. Walrand, "Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays–Part I: I.I.D. rewards," *IEEE Trans. Autom. Control*, vol. 32, no. 11, pp. 968–976, Nov. 1987.
[17] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 1991.
[18] M. L. Littman and C. Szepesvári, "A generalized reinforcement-learning model: Convergence and applications," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Bari, Italy, 1996, pp. 310–318.
[19] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, nos. 2–3, pp. 235–256, May/Jun. 2002.
[20] S. Agrawal and N. Goyal, "Analysis of Thompson sampling for the multi-armed bandit problem," in *Proc. Conf. Learn. Theory (COLT)*, 2012, Art. no. 39.
[21] R. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton Univ. Press, 1957.
[22] L. de Haan and A. Ferreira, *Extreme Value Theory: An Introduction* (Springer Series in Operations Research and Financial Engineering). New York, NY, USA: Springer-Verlag, 2006.
[23] R. A. Fisher and L. H. C. Tippett, "Limiting forms of the frequency distribution of the largest or smallest member of a sample," *Math. Proc. Camb. Philosoph. Soc.*, vol. 24, no. 2, pp. 180–190, 1928.
[24] B. V. Gnedenko, "Sur la distribution limite Du terme maximum D'une serie aleatoire," *Ann. Math.*, vol. 44, no. 3, pp. 423–453, 1943.
[25] E. I. George, U. E. Makov, and A. F. M. Smith, "Conjugate likelihood distributions," *Scandinavian J. Stat.*, vol. 20, no. 2, pp. 147–156, 1993.
[26] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proc. Int. Conf. Mach. Learn. (ICML)*, New Brunswick, NJ, USA, 1994, pp. 157–163.
[27] J. Hu and M. P. Wellman, "Multiagent reinforcement learning: Theoretical framework and an algorithm," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Madison, WI, USA, 1998, pp. 242–250.
[28] M. L. Littman, "Friend-or-foe Q-learning in general-sum games," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Williamstown, MA, USA, 2001, pp. 322–328.
[29] H. Robbins, "Some aspects of the sequential design of experiments," *Bull. Amer. Math. Soc.*, vol. 58, no. 5, pp. 527–535, 1952.
[30] R. E. Bellman, *A Problem in the Sequential Design of Experiments*. Fort Belvoir, VA, USA: Defense Tech. Inf. Center, 1954.
[31] J. C. Gittins, "Bandit processes and dynamic allocation indices," *J. Roy. Stat. Soc.*, vol. 41, no. 2, pp. 148–177, 1979.
[32] P. Whittle, "Restless bandits: Activity allocation in a changing world," *J. Appl. Probab.*, vol. 25, pp. 287–298, Jan. 1988.
[33] R. L. Rivest and Y. Yin, "Simulation results for a new two-armed bandit heuristic," in *Proc. Workshop Comput. Learn. Theory Nat. Learn. Syst.*, Cambridge, U.K., 1994, pp. 477–486.
[34] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

**Siamak Dastangoo** (S'88–M'00) received the bachelor's degree and the master's degree in electrical engineering from the University of Massachusetts, and the doctoral degree in electrical engineering from George Washington University. He spent several years in the commercial and defense industries working on a range of problems in communications and networking systems. In 2006, he joined Lincoln Laboratory, Massachusetts Institute of Technology, where he is a Technical Staff Member with the Tactical Networks Group, conducting research and development in the area of wireless ad hoc networks. His current research interests are in the areas of cognitive networks and performance predictions of networks.

**Carl E. Fossa** (S'98–M'02–SM'05) received the bachelor's degree from United States Military Academy, the master's degree from the Air Force Institute of Technology, the doctoral degree from Virginia Polytechnic Institute and State University, all in electrical engineering. He served as an Army Signal Officer for 21 years, retiring at the rank of Lieutenant Colonel. In 2008, he joined Lincoln Laboratory, Massachusetts Institute of Technology, where he is an Associate Leader of the Tactical Networks Group, focussing on the performance of mobile ad hoc networks in tactical military environments. Applications of this research include the development and deployment of a real-time emulation of the Army Warrior Information Network-Tactical to Aberdeen Proving Grounds, MD, USA. He served in a range of tactical military positions, which included deployment to Operation Desert Shield/Storm. He also served in a number of technical engineering positions at major command headquarters and as an Assistant Professor of electrical engineering with the United States Military Academy.



**Youngjune L. Gwon** (S'12–M'14) received the B.S. degree from Northwestern University and the M.S. degree from Stanford University, both in electrical engineering, and the Ph.D. degree in computer science from Harvard University, in 2015. He is a Technical Staff Member with the Human Language Technology Group, Lincoln Laboratory, Massachusetts Institute of Technology, where he does machine learning research and development for speech and audio-video processing systems. He currently works on multimodal representation learning for large, unstructured audio, video, and text corpora. His broader research interests include networks, wireless communications, embedded systems, and security. He was a Software Engineer with Silicon Valley for ten years.

**Hsiang-Tsung Kung** received the Ph.D. degree from Carnegie Mellon University. He taught at Carnegie Mellon University for 19 years. In 1992, he joined Harvard University, where he is the William H. Gates Professor of Computer Science and Electrical Engineering. His broad research areas include complexity theory, database systems, very large scale integration, parallel computing, computer networks, security, and wireless communications. He currently focuses on machine learning, high-performance computing, and the Internet of Things. He is well known for his pioneering work on systolic arrays in parallel processing and optimistic concurrency control in database systems. His academic honors include a membership in the National Academy of Engineering and the ACM SIGOPS 2015 Hall of Fame Award (with J. Robinson) that recognizes the most influential operating systems papers that were published at least ten years in the past.