

Article

Intelligent Cognitive Radio Ad-Hoc Network: Planning, Learning and Dynamic Configuration

Kwang-Eog Lee ¹, Joon Goo Park ²  and Sang-Jo Yoo ^{3,*} ¹ Agency for Defense Development, Daejeon 34060, Korea; kelee@add.re.kr² School of Electronics Engineering, Kyungpook National University, Daegu 41566, Korea; jgpark@knu.ac.kr³ Department of Information and Communication Engineering, Inha University, Incheon 22212, Korea

* Correspondence: sjyoo@inha.ac.kr; Tel.: +82-32-860-8304

Abstract: Cognitive radio (CR) is an adaptive radio technology that can automatically detect available channels in a wireless spectrum and change transmission parameters to improve the radio operating behavior. A CR ad-hoc network (CRAHN) should be able to coexist with primary user (PU) systems and other CR secondary systems without causing harmful interference to licensed PUs as well as dynamically configure autonomous and decentralized networks. Therefore, an intelligent system structure is required for efficient spectrum use. In this paper, we present a learning-based distributed autonomous CRAHN network system model for network planning, learning, and dynamic configuration. Based on the system model, we propose machine learning-based optimization algorithms for spectrum sensing, cluster-based ad-hoc network configuration, and context-aware signal classification. Using the sensing engine and the cognitive engine, the surrounding spectrum usage and the neighbor network operation status can be analyzed. The proposed policy engine can create network operation policies for the dynamically changing surrounding wireless environment, detect policy conflicts, and infer the optimal policy for the current situation. The decision engine finally determines and configures the optimal CRAHN configuration parameters through cooperation with a learning engine, in which we implement the proposed machine-learning algorithms. The simulation results show that the proposed machine-learning CRAHN algorithms can construct CR cluster networks that have a long network lifetime and high spectrum utility. Additionally, with high signal context recognition performance, we can ensure coexistence with neighboring systems.

Keywords: cognitive radio; ad-hoc network; machine learning; optimization; coexistence

**Citation:** Lee, K.-E.; Park, J.G.;

Yoo, S.-J. Intelligent Cognitive Radio Ad-Hoc Network: Planning, Learning and Dynamic Configuration.

Electronics **2021**, *10*, 254. <https://doi.org/10.3390/electronics10030254>

Received: 3 December 2020

Accepted: 20 January 2021

Published: 22 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, as the demand for wireless communication services has increased rapidly, the problem of a shortage of frequency resources has greatly increased. For efficient use of limited frequency resources, a cognitive radio (CR) technology, which is a frequency-sharing method achieved through dynamic spectrum access, has drawn attention. A CR network (CRN) is composed of unlicensed secondary users (SUs) and uses a spatially and temporally empty spectrum to avoid interference with licensed primary users (PUs) by sensing the surrounding wireless environment. The CRN should coexist with licensed users without causing harmful interference. It needs to dynamically set up a system configuration suitable for the wireless environment, and it should make an optimal decision for the current situation. In this paper, we consider a CR ad-hoc network (CRAHN), which is decentralized and self-configured [1]. A CRAHN can respond quickly to dynamic changes in surrounding wireless environments and is more scalable.

In recent years, CRAHNs have been applied in various fields, including disaster emergency networks and military tactical communications because they enable immediate network configuration without using the existing infrastructure and can efficiently use frequency resources while responding to changes in dynamic radio resource demand [2,3].

With respect to existing wireless ad-hoc networks, such as MANET (mobile ad-hoc network), FANET (flying ad-hoc network), VANET (vehicular ad-hoc network), dynamic routing, and medium access control (MAC) technology, studies have been primarily conducted to provide seamless services with changes in network topology according to the mobility of user devices. Conversely, in a CRAHN, the network topology changes in response to the spatial and temporal changes in wireless environments caused by the primary system activation and neighbor CR network operations as well as the mobility of SU devices, so that each secondary device needs to find the available frequency resources and determine their quality. SU devices must dynamically reconfigure system parameters for optimal ad-hoc network operation. Conventional wireless ad-hoc systems generally follow predefined policies for system parameter configurations, such as operating frequency and maximum transmission power. Therefore, the pre-defined policies are embedded in the device, and it is easy to enforce a transmission policy. However, because CR systems operate under conditions in which the surrounding wireless environments change from time to time, policies suitable for the current environmental conditions must be dynamically reconfigured for the device. Dynamic policy updates and reasoning are challenging operations in a CRAHN.

Recently, machine learning (ML), which is one of the most rapidly growing artificial intelligence (AI) technologies, has been extensively used to solve critical challenges in CR networks [4–7]. ML techniques can be applied to many functional elements in a CRAHN, including spectrum sensing, optimum resource allocation, precise environment context awareness, spectrum usage prediction, and ad-hoc routing. These techniques can make a CRAHN highly intelligent, provide fast adaptability to the dynamicity of the environment, and improve the quality of service of CR users. In [8], we proposed a Q-learning-based dynamic optimal band and channel selection method in the CR network by considering the surrounding wireless environments and system demands in order to maximize the available transmission time and capacity at the given time and geographic area. For CRAHN cluster formation, in [9] we presented a Q-learning-based clustering mechanism for cluster head selection and inter-cluster coexistence.

In this paper, we present a system model for an intelligent CRAHN and propose machine-learning algorithms for the proposed model. The proposed intelligent CRAHN system model consists of sensing, cognitive, decision, policy, and learning engines. The learning engine, which is a core part of the proposed model, and other engines are integrated with the model and use statistics from sensing results and neighboring secondary system information to make optimum decisions for network parameter configuration. The learning-based policy engine predicts the optimal policy according to the region/time/mission and performs policy reasoning to prevent conflicts between policies. By designing and implementing the organized interactions between engines, we can provide a more stable ad-hoc network and improve the efficiency of the system. The proposed learning algorithms capture the short-term and long-term changes in the surrounding wireless environments. We propose a reinforcement learning-based CRAHN network configuration method that (re)configures a cluster-based ad-hoc network by sharing the spectrum sensing results and other cluster network information. After establishing the CR cluster network for fine sensing band selection at the sensing engine, we present a bio-inspired particle swarm optimization (PSO)-based algorithm. For cognitive engine operation to distinguish the received signal source and type, we propose a convolutional neural network (CNN)-based automatic modulation classification method. We evaluated the performance by implementing the proposed system model, and it showed that the proposed system can increase network reliability and frequency use efficiency.

This paper is organized as follows. In Section 2, we present an intelligent CRAHN system model. Machine learning-based CRAHN configuration and optimum network parameter decision algorithms using the proposed system model are presented in Section 3. In Section 4, the policy engine design and implementation of the proposed system are presented. The simulation results are evaluated in Section 5, and Section 6 concludes this paper.

2. Intelligent Cognitive Radio Ad-Hoc Network System Model

For a CRAHN to recognize the surrounding networks and spectrum environment and to configure optimal system parameters, an intelligent system model is required. In this section, we propose an intelligent wireless CRAHN system model based on artificial intelligence. As a reference for how a CR could achieve the required functionality, Mittra [10] introduced the basic cognition cycle as a top-level control loop for CR. Figure 1 shows the learning-based intelligent CR functional cycle considered in this study.

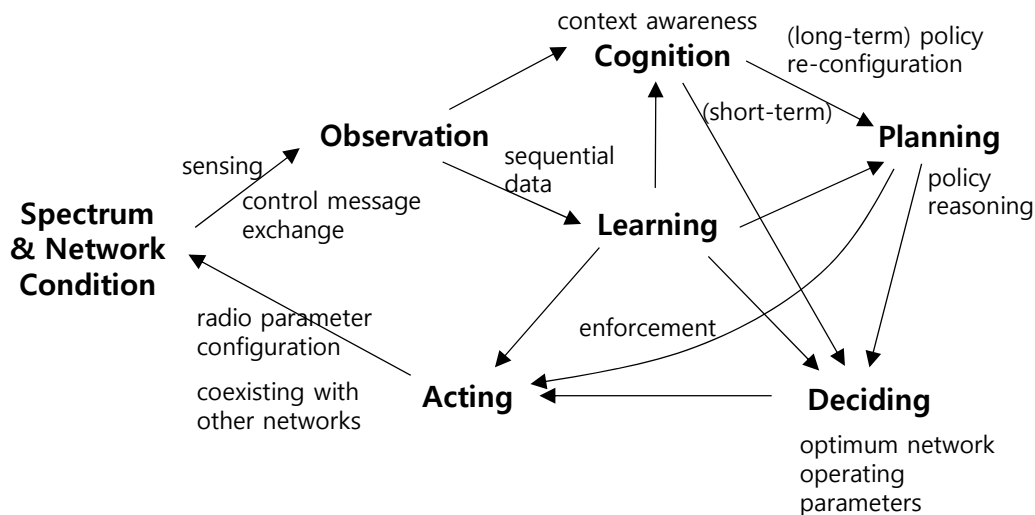


Figure 1. Intelligent cognitive radio ad-hoc network (CRAHN) functional cycle.

In a CRAHN, each device independently or cooperatively observes the environment, including spectrum usage and neighboring network status. The observation is performed by analyzing the received signal for a certain period of time or collecting information from neighboring SU devices by a control message exchange. In the cognition stage, accurate context awareness of the surrounding environment is performed using the observed data. For context awareness, using artificial intelligence machine-learning technologies, we can more efficiently and accurately perform cognition of the current and future status, including the classification of received signals and prediction of dynamic changes in user requirements and network behaviors.

The intelligent CRAHN considered in this paper performs policy-based system operation. Due to the nature of distributed ad-hoc systems that use unlicensed bands and non-centralized system control, the operation may cause several problems that interfere with mutual coexistence and may cause harmful interference to primary users. Therefore, for applications requiring strict control, as in disaster communication networks or military ad-hoc networks, a network operation capable of dynamically configuring policy restrictions is required [11]. The intelligent policy engine proposed and implemented in this study can dynamically perform reasoning for the optimal policy; accordingly, the decision engine sets the optimal wireless network operation parameters suitable for the current time and region where the CR system is located. For all processes in Figure 1, the learning engine, using the machine-learning algorithms proposed in this paper, helps to achieve improved performance.

Figure 2 shows the distributed network model of the CRAHN considered in this study. There are multiple PU systems in a given area. PUs are licensed systems that have been assigned an operating frequency in advance, and it is assumed that there is no other PU system using the same frequency within the system coverage through detailed interference control. As shown in Figure 2, SUs coexist with the PU systems and form distributed ad-hoc networks that do not rely on a pre-existing infrastructure. Since a CR network must not cause harmful interference to PUs during data transmission, it is very difficult or impossible to operate an ad-hoc network over a large area using a frequency channel [12].

Therefore, in this paper, we consider cluster-based CRAHNs, as in [13]. Cluster head (CH) nodes are selected in a dynamic and fully distributed manner based on connectivity with neighboring nodes, the stability of the use of available frequency channels, and residual energy. Afterward, a cluster network with one-hop neighbor nodes as member nodes (MNs) is formed around the selected CH.

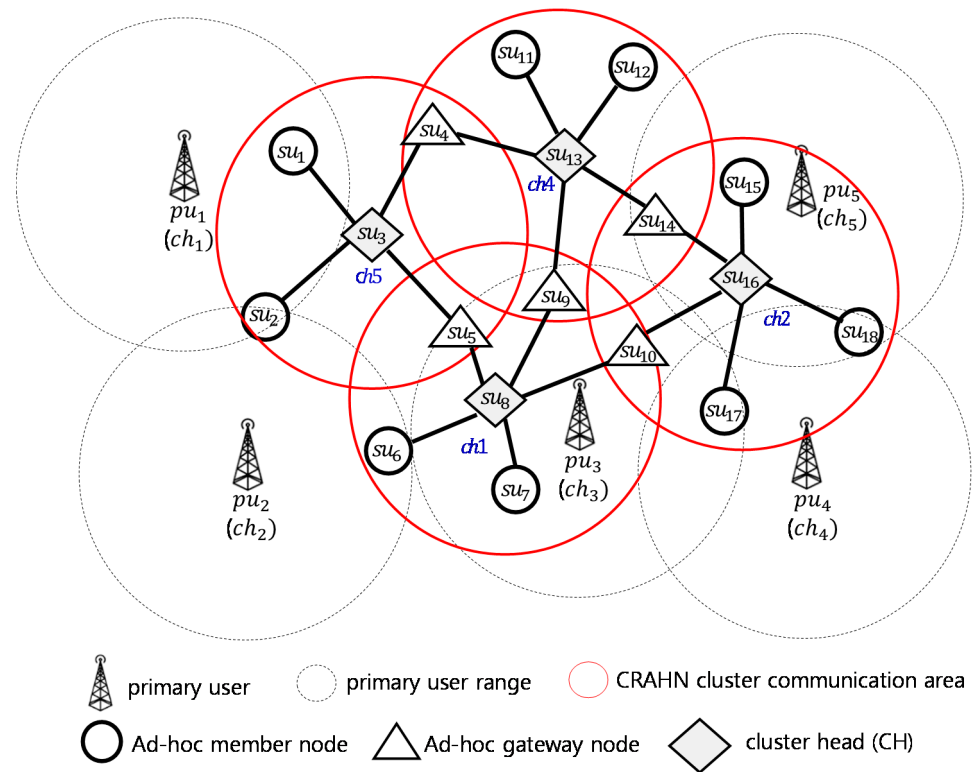


Figure 2. CRAHN coexistence model.

In the network model of Figure 2, for inter-cluster communication, a special MN called a gateway node (GN) that guarantees a connection with neighboring clusters is selected. When selecting a common active data channel of a cluster, the decision is made in consideration of the channels used by neighboring clusters to reduce interference between adjacent clusters in the CRAHN. Therefore, the GN must belong to two or more cluster networks to be connected, and all active data channels of each cluster must be available at the GN. When configuring the CRAHN, it must comply with the dynamic policy of the policy engine, including the conditions of specific frequencies that should not be used in certain regions or time zones, or restrictions on transmission power. In this study, it is assumed that a predefined common control channel (CCC) exists for the exchange of control messages between SUs. Therefore, when configuring the initial CRAHN or reconfiguring the network, information exchange with neighboring SU nodes uses the CCC allocated to the secondary system. In some applications such as military tactical networks, the predefined CCC may not be possible or it may be vulnerable to security or jamming attacks. In that case, we can apply distributed dynamic common control channel selection protocols [14], in which a network or cluster wise CCC is established dynamically based on the neighboring node's channel availability.

Figure 3 shows the proposed intelligent CRAHN system model. The proposed system model is composed of the following five engines: sensing, cognitive, decision, policy, and learning engines. The functions of each engine and the interactions between the engines are as follows:

- Sensing engine: To coexist with PUs, each SU periodically senses the spectrum. In the sensing engine, any sensing technique can be used, such as energy detec-

tion, cyclostationary-based feature detection, or coherent-based detection. In each MN, local spectrum sensing is performed, and in the CH, cooperative sensing is implemented by fusing the sensing results of MNs in the cluster. The main decision parameters in the sensing engine are the wide- and/or narrowband sensing schedules and the ability of bands to be sensed more precisely. These parameters are determined by the decision engine, combined with the learning engine, and then delivered to the sensing engine. In addition, when a context awareness of the signal type or configuration of the surrounding networks is required beyond simple signal detection, the raw data from the sensing engine is passed to the cognitive engine.

- Cognitive engine: The cognitive engine performs a more accurate recognition of surrounding wireless environments based on the results obtained from the sensing engine. The neighbor discovery module analyzes messages from MNs and GNs through the RF module and derives spectrum and network-aware information regarding the adjacent CR ad-hoc clusters, which include modulation types, active data channels, and reachable cluster identifications through the neighbor clusters. The cognitive engine proposed in this paper clearly distinguishes whether the signal received is a PU signal, an adjacent SU cluster network signal, or a noise signal, thereby enhancing the efficiency of system coexistence and frequency used between systems. The cognitive engine classifies the signal source and type using deep learning in the learning engine.
- Decision engine: The decision engine is responsible for the final optimization in the CRAHN. It determines the optimal system parameters for sensing, network configuration, and resource allocation using the received context information from the cognitive engine. When configuring the optimization parameters in the system, the decision engine should finally verify whether they conform to the network operation policy derived from the policy engine. Regarding sensing, when precise sensing of a specific band among the broadband spectrum is required, the best narrow sensing band is dynamically determined using the proposed PSO algorithm of the learning engine. In addition, the ad-hoc network is configured or reconfigured by dynamically selecting the network CH and the common data channel using the proposed reinforcement learning.
- Policy engine: The policy engine implemented in this study has a structure for dynamically establishing, distributing, and applying policies. The CH of the cluster-based CRAHN becomes an agent that infers and sets policies within the cluster. The configured policy is distributed to the MNs in the cluster. The policy engine dynamically creates policies using the authoring tool, detects conflicts between policies, and performs reasoning to infer network policies available at the current location and time. In addition, long-term policy updates are performed using the prediction function of the learning engine. The regression function is used for updating the policy based on the long-term behavior prediction.
- Learning engine: The learning engine is a core engine required for intelligent CRAHN configuration. It performs regression, classification, and optimization requested by each engine based on sensed signal data, context-aware information, and related policy information. The machine-learning techniques implemented in this study include polynomial regression techniques, CNNs, unsupervised clustering, and Q-learning. The learning engine provides a common platform related to machine learning for CRAHN operation. In addition, the learning results for a specific purpose can also be used as additional data or supplementary input for other optimization purposes. Therefore, we have defined the learning platform and database as separate engine functions.

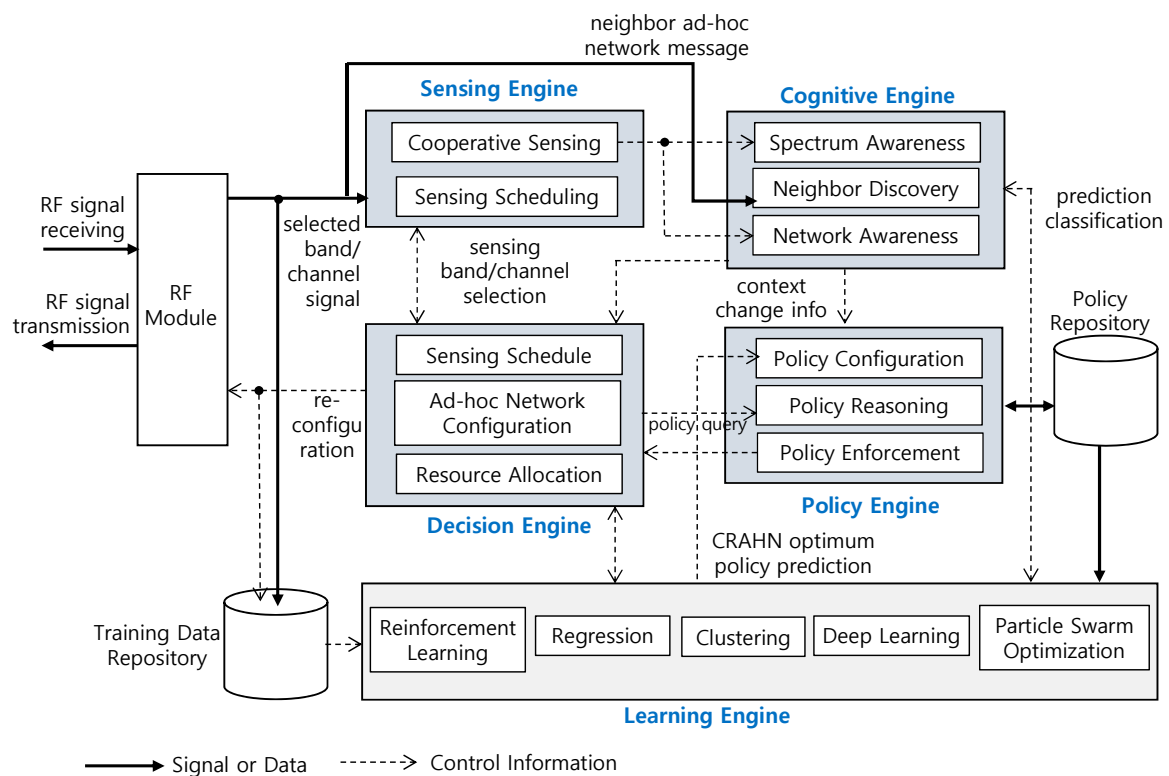


Figure 3. Proposed intelligent CRAHN system model.

Although security in CRNs has received less attention than other areas of CR technology, ensuring security becomes a major and crucial issue. An open channel for secondary users is used for communications that can easily be accessed by attackers and the particular attributes of CRNs raise new opportunities to malicious users, which can disrupt network operation. In this paper, even though we have not deeply considered the security issues in CRN, each engine needs to conduct security functionalities, which are application or network operation environment-dependent.

3. Learning-Based CR Ad-Hoc Network Configuration and Optimum Network Parameter Decision

3.1. Optimum Narrow Spectrum Band Decision Using Particle Swarm Optimization

Cognitive radio devices need to sense a wideband spectrum in the range of several hundred MHz to several GHz to find a channel that guarantees high throughput and long service time. However, a high sampling rate and implementation complexity are required for precise sensing of a wideband spectrum, which makes actual implementation difficult [15,16]. In a CRAHN, wideband spectrum sensing is used to find an operating channel in the initial stage of the network configuration, to find a new channel by the appearance of a primary user, or to periodically search for a better channel. In the proposed sensing method, during wideband spectrum sensing, rough and fast spectrum sensing with a small number of fast Fourier transform (FFT) bins in the unit frequency range is performed. Then, the optimal narrow and fine sensing band that has the greatest possibility of the existence of high-quality available channels is derived using a machine-learning technique.

Figure 4 shows the proposed narrow sensing band decision procedure for fine sensing. The CH requests wideband spectrum sensing to all nodes in the cluster (Figure 4a), and each member node performs wideband N -point FFT. At node i , if the value FFT_n^i of each n -th FFT bin is less than the threshold Th_{PU} for determining the presence of the PU signal, the bin availability f_n^i is set to 1; otherwise, it is expressed as 0. Each node makes an FFT

bin availability vector F_i for the entire wideband, as in Equation (1), and sends it to the CH (Figure 4b).

$$F_i = [f_1^i, f_2^i, \dots, f_n^i, \dots, f_N^i], f_n^i = \begin{cases} 1, & \text{if } FFT_n^i < Th_{PU} \\ 0, & \text{else} \end{cases} \quad (1)$$

where FFT_n^i is the n -th FFT bin value of node i , Th_{PU} is the threshold to determine the possible existence of the PU signal, f_n^i is the FFT bin availability index, and F_i is the FFT bin availability vector of node i .

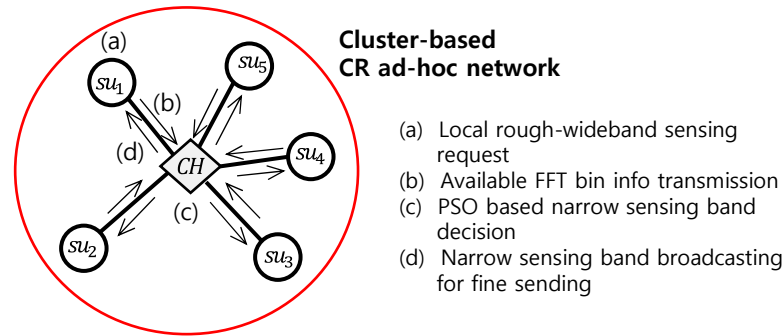


Figure 4. Wide and narrow-spectrum sensing band decision procedure.

The CH calculates the cluster-wise wideband FFT bin availability vector CV for the entire cluster by fusing the availability vectors received from all member nodes,

$$CV = F_1 \cap F_2 \cdots \cap F_M = \cap_{i=1}^M F_i, \quad (2)$$

where M is the number of member nodes in the cluster. CV is used to derive the optimum narrow spectrum band for fine sensing and eventually to obtain the common data channel for the cluster so that the wideband FFT bins of CV should be available for all member nodes as in Equation (2).

In this paper, the utility function of Equation (3) is defined to select the narrowband fine sensing range in which the FFT bin length is L . L is determined based on the RF measurement capability of CR devices for fine spectrum sensing.

$$U(n) = \omega_1 Z_{NAB}(n) + \omega_2 Z_{NCB}(n), \quad (3)$$

where $U(n)$ is the utility for the bin range $[n, n + L - 1]$; $Z_{NAB}(n)$ is the number of available bins (bin value = 1) in bin range $[n, n + L - 1]$ of cluster CV vector, $Z_{NCB}(n)$ is the maximum number of consecutive available bins of CV in bin range $[n, n + L - 1]$, and ω_1 and ω_2 are weight parameters.

CH calculates utility $U(n)$ at each wideband FFT bin point using a sliding window mechanism, in which the window size is L , and then derives the bin range $[n^*, n^* + L - 1]$ that has the largest utility value. Fine sensing is performed for this narrow range NSB .

$$n^* = \underset{n}{\operatorname{argmax}} U(n) \quad (4)$$

$$NSB = [n^*, n^* + L - 1] \quad (5)$$

However, the utility calculation in each FFT bin of the wideband using the sliding window requires a large number of calculations. This makes its real-time implementation difficult. Therefore, in this study, the PSO algorithm, which is a bio-inspired machine-learning technique, is used to quickly find the bin range with the optimal utility (Figure 4c). Finally, the CH broadcasts the narrow sensing band (NSB) range for fine sensing to all member nodes.

PSO is a computational method that optimizes a problem by iteratively trying to improve a candidate solution for a given utility function. It solves a problem by having a population of candidate solutions and moving these particles around in the search space according to simple mathematical formulae over the particle's position and velocity. Each particle's movement is influenced by its local best-known position but is also guided toward the global best-known position in the search space, which is updated as better positions are found by other particles. The particle position in the proposed PSO-based method represents the FFT bin sliding window starting point. The velocity and position of the i -th particle are updated as in Equations (6) and (7), respectively, until the utility of Equation (3) converges or the PSO iteration number reaches a predefined number.

$$v_i(k+1) = \omega v_i(k) + c_1 r_1 [p_i(k) - x_i(k)] + c_2 r_2 [p_g(k) - x_i(k)] \quad (6)$$

$$x_i(k+1) = x_i(k) + v_i(k+1) \quad (7)$$

where $x_i(k)$ and $v_i(k)$ are the FFT bin sliding window starting point and velocity of the particle i at the k -th iteration time, respectively; ω denotes the inertia weight factor; $\{c_1, c_2\}$ are the position acceleration constants; and $\{r_1, r_2\}$ are random numbers uniformly distributed over interval $[0, 1]$.

3.2. Reinforcement Learning-Based Distributed CR Ad-Hoc Network Configuration and Operational Channel Decision

In the distributed CRAHN, the set of available frequency channels of the network and the list of connectable neighbor nodes using each channel continuously change over time because of the dynamics of the PU system activity, the mobility of SU nodes, and the network channel configuration of the neighbor cluster networks. To adjust to these changes, the network topology and the common data channel of a cluster should be configured dynamically [17]. This section presents a dynamic cluster-based CRAHN (re)configuration method using reinforcement learning (RL).

RL essentially deals with the solution of optimal control problems using on-line measurements by interacting with an environment. It is suitable for application to CRAHN clustering because RL can capture the dynamics of the network topology and spectrum usage well. Q-learning is a model-free RL algorithm that includes an agent, a set of states S , and a set of actions A . By performing an action $a \in A$, the agent transitions from state to state. The agent in a state s interacts with the environment with an action a to learn the environment, while depending on the outcome, it acquires a reward value $r(s, a)$. Suppose that at each time t , the agent selects an action a_t , observes a reward r_t , and enters a new state s_{t+1} . Then, the Q-value of $Q(s_t, a_t)$ is updated as:

$$Q(s_t, a_t) = (1 - \alpha)Q(s_t, a_t) + \alpha \left\{ r_t + \gamma \cdot \underbrace{\max_a Q(s_{t+1}, a)}_a \right\} \quad (8)$$

where α is the learning rate and γ is the discount factor for the future reward.

Each node of the CRAHN periodically senses the spectrum and measures the quality of each channel with a predefined bandwidth. In this paper, the state s_t of Equation (8) represents each secondary user su_k in the network, and the action set $A = \{a_t\}$ that can be selected in each state is the available channels for the current state (i.e., each member node) at time t . The quality of each sensed channel is defined as a reward according to the periodic sensing result. The sensing reward r_t for the channel ch_c of the node is expressed by

$$r_t = \delta_1 \cdot T_{su_s}^{ch_c} + \delta_2 \cdot P_{su_s}^{ch_c} \quad (9)$$

$$T_{su_s}^{ch_c} = \frac{\text{average } ch_c \text{ channel idle time}}{\text{total sensing time}}, P_{su_s}^{ch_c} = \frac{\text{number of idle observations for } ch_c}{\text{number of } ch_c \text{ sensing trials}} \quad (10)$$

where δ_1 and δ_2 are the weight parameters, and $\delta_1 + \delta_2 = 1$.

For cluster (re)formation, each node broadcasts its own device status, local sensing learning result, and neighboring cluster and neighbor node information in a packet using the predefined CCC. The device status includes the node identification and the current residual energy, and the local sensing learning result information includes a list of available channels and Q-values for available channels, which are updated with Equation (8). The neighboring cluster information contains the neighboring cluster identifications and the cluster active data channels to which the node can connect. The neighbor node information includes the one-hop neighbor nodes and their available channel list. Each node that receives the broadcasting packets from neighbor nodes calculates the channel fitness value CF_j (goodness of available channels of node j) and the cluster head fitness value V_j (goodness node j to become a CH), in which node j is the node itself as well as one-hop neighbor nodes.

$$CF_j = \sum_{k \in CAC_j} (Q(j, k) \times N_j^k) \quad (11)$$

$$V_j = \beta_1 \frac{E_j^R}{E_{max}} + \beta_2 \frac{CF_j}{CF_{max}} + \beta_3 \frac{RNC_j}{RNC_{max}} + \beta_4 \frac{N_j}{NN_{max}} \quad (12)$$

where CAC_j is the set of commonly available channels between node j and its one-hop neighbors; N_j^k is the number of neighbor nodes that can be connected with node j using channel k ; $\beta_1 + \beta_2 + \beta_3 + \beta_4 = 1$; E_j^R is the residual energy of node j ; RNC_j is the number of reachable neighbor clusters through node j itself or node j 's neighbor nodes; and N_j is the number of neighbor nodes of node j within the transmission coverage. E_{max} , CF_{max} , RNC_{max} , and NN_{max} are the predetermined maximum values for normalization.

Each node i selects the node that has the highest CH fitness value and sends a CH_REQ (CH Request) message to the selected node using the CCC. If the CH fitness value of the node itself is highest among its neighbors, then it virtually sends a CH_REQ to itself. If a node has received more CH_REQ messages than the predetermined ratio η for the number of neighbor nodes, then it should act as a CH and start to determine the common data channel for its ad-hoc CR cluster. The common data channel CDC_j for node j 's cluster is derived as

$$CDC_j = \underset{k \in CAC_j}{\operatorname{argmax}} (Q(j, k) \times N_j^k) \quad (13)$$

Finally, the CH broadcasts the selected optimal channel CDC_j to its neighbors using CCC. The neighbor nodes, where CDC_j is one of their available channels will join the cluster network. The selected CDC_j is used for data communication between member nodes within the cluster. The other detailed protocol procedures for CR ad-hoc cluster formation have been previously published [9].

3.3. Modulation Type Classification Using Convolutional Neural Network

In a CRAHN, interference between primary and secondary users should be minimized, and coexistence between secondary systems should be considered important. To this end, it is necessary to accurately analyze the context of the sensed signal in a cognitive engine.

Energy detection is one of the most widely used techniques for spectrum sensing because it does not require any prior knowledge about the characteristics of the primary and secondary signals. However, this technique cannot distinguish between primary and secondary signals. Worse, when the noise power is relatively large or the signal power is weak, the energy detection technique may not be able to distinguish the signal from the noise. It shows low performance at a low signal-to-noise ratio (SNR), and the selection of the detection threshold becomes an issue because the noise is uncertain. Automatic modulation classification (AMC) is of great importance for achieving automatic receiver configuration, interference mitigation, and spectrum management [18]. AMC also performs a role in distinguishing the modulation types of received signals from primary or secondary users. In the proposed system model, AMC is performed at the cognitive engine through

cooperation with the learning engine. In [19], the SCF pattern vector is used as an input to the deep belief network (DBN) for AMC.

In this section, we propose a CNN-based signal classification method to identify different modulation types. Instead of using raw sampled data of the received signal, we use the spectral correlation function (SCF) to capture the signal characteristics and to represent the signal as image data. In addition, some important statistical features are added to the neural network as an input to enhance the classification accuracy.

Cyclic autocorrelation of a signal $x(t)$ is defined as:

$$\hat{R}_x^\alpha(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} x\left(t + \frac{1}{2}\tau\right)x\left(t - \frac{1}{2}\tau\right)e^{-j2\pi\alpha t} dt \tag{14}$$

Also, two frequency-shift signals of $x(t)$ are defined as:

$$u(t) = x(t)e^{-j\pi\alpha t} \tag{15}$$

$$v(t) = x(t)e^{+j\pi\alpha t} \tag{16}$$

Then, $\hat{R}_x^\alpha(\tau)$ can be represented as the cross-correlation of the two signals as follows:

$$\hat{R}_x^\alpha(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} u\left(t + \frac{1}{2}\tau\right)v^*\left(t - \frac{1}{2}\tau\right) dt \tag{17}$$

The spectral correlation function is the Fourier transformation of cyclic autocorrelation.

$$\hat{S}_x^\alpha(f) = \int_{-\infty}^{\infty} \hat{R}_x^\alpha(\tau)e(-j2\pi f\tau) d\tau \tag{18}$$

If $\alpha = 0$, $\hat{R}_x^0(\tau)$ is a conventional autocorrelation function and $\hat{S}_x^0(f)$ is the power spectral density.

Therefore, SCF can be calculated from the following expression:

$$\hat{S}_x^\alpha(f) = \lim_{\Delta f \rightarrow 0} \lim_{\Delta t \rightarrow \infty} \frac{1}{\Delta t} \int_{-\frac{\Delta t}{2}}^{\frac{\Delta t}{2}} \Delta f X_{\frac{1}{\Delta f}}\left(t, f + \frac{\alpha}{2}\right) X_{\frac{1}{\Delta f}}^*\left(t, f - \frac{\alpha}{2}\right) dt \tag{19}$$

where

$$X_{\frac{1}{\Delta f}}(t, v) = \int_{t-\frac{1}{2\Delta f}}^{t-\frac{1}{2\Delta f}} x(u)e^{-j2\pi v u} du \tag{20}$$

Figure 5 shows the proposed CNN-based learning architecture for modulation-type classification. For the sampled signal, the SCF image is computed and forwarded to the convolutional layer. From the sampled signal, eleven statistically important features shown in Table 1 are concatenated with the convolutional layer output and are input to the fully connected layer. Some of the statistical features of Table 1 were presented in [20]. Using SCF and CNN learning methods, the received signal can be easily classified in a relatively good SNR region. Otherwise, the statistical features in Table 1 are resistant to noise, so that the combination of SCF and statistical features makes a more accurate classifier. Using these two types of input data, we obtain a powerful performance for all SNR regions. In accordance with the classification results, we can determine whether the detected signal is from a primary signal, secondary signal, or noise. Depending on the source of the signal, we can apply different coexistence policies to the policy engine.

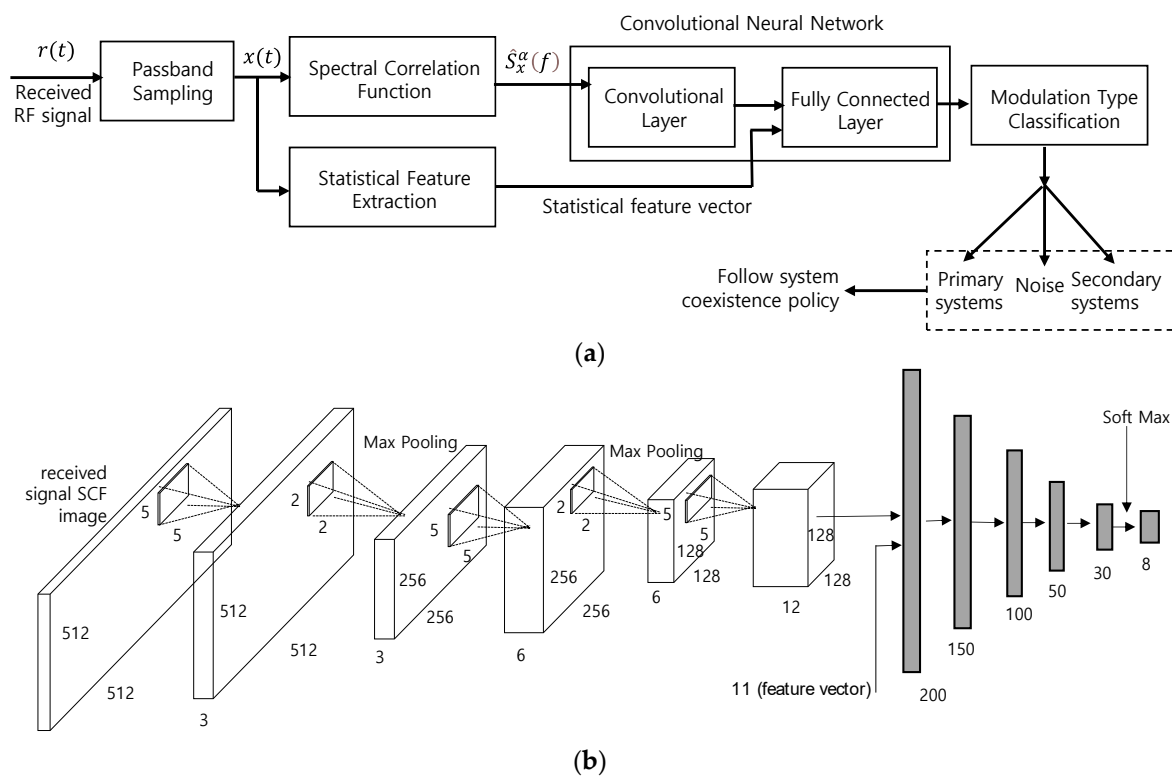


Figure 5. Convolutional neural network for automatic modulation classification. (a) Functional structure for received signal classification; (b) convolutional neural network (CNN) layer structure.

Table 1. Statistical features for automatic modulation classification.

Number	Statistical Feature
1	Ratio of in-phase component and quadrature component signal power
2	Standard deviation of the direct instantaneous phase
3	Standard deviation of the absolute value of the non-linear component of the instantaneous phase
4	Standard deviation of the absolute value of the normalized instantaneous amplitude of the simulated signal
5	Standard deviation of the absolute normalized centered instantaneous frequency for the signal segment
6	Standard deviation of the normalized signal amplitude
7	Mean of the signal magnitude
8	Normalized square root value of sum of amplitude of signal samples
9	Maximum value of power spectral density of the normalized signal samples
10	Peak-to-RMS ratio
11	Peak-to-average ratio

4. CR Ad-Hoc Network Policy Engine Design and Implementation

A device operating in a CRAHN needs to be able to perform opportunistic transmissions based on policies that regulate the behavior of the device, even in a dynamic wireless environment. To accomplish this, dynamic policy management and control technology capable of actively responding to changing wireless environmental conditions are required. This section presents the proposed policy engine structure and system implementation considering the scalability of policy expression for a policy-based CRAHN. The policy engine guarantees that CR devices operate within the domain defined by policies and prevents the configuration of wireless devices from changing to an unacceptable state in the current space and time. It is also used to ensure the establishment, distribution, and selection of appropriate policies in a dynamically changing wireless environment.

The most important function performed by the policy engine is the reasoning function, which derives an appropriate policy for communication requested by the wireless devices and finds conflicts between policies. The policy engine works by organically linking with other engines in the system, as presented in the system model shown in Section 2.

A policy defines an action appropriate to the current condition. An action generally does not determine the exact radio parameters but rather specifies the availability or range of allowable parameters (e.g., maximum or minimum). Policies can be created and updated by network operators using a policy authoring tool. In some cases, the existing policy can be dynamically updated automatically based on the context recognition of the learning engine and the cognitive engine. Learning-based dynamic policy updating in the proposed system modifies the related policies for the current condition. The policy is updated and applied based on long-term behaviors for wireless environments and CR user spectrum use trends. These long-term behaviors are predicted by a simple machine-learning technique in the proposed system. We implemented a polynomial regression algorithm for long-term behavior prediction. In statistics, polynomial regression is a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modeled as an n th degree polynomial in x . As a simple example scenario, depending on the traffic demand of a CRAHN cluster, the policy engine needs to update the policy for the bandwidth of a channel. In this case, the independent variable x is at time instance t_i , and the dependent variable y has the observed traffic amount y_i at time t_i . The general polynomial model is represented as

$$y_i = \beta_0 + \beta_1 t_i + \beta_2 t_i^2 + \cdots + \beta_n t_i^n + \varepsilon_i (i = 1, 2, \dots, m) \quad (21)$$

where ε_i is an unobserved random error and m is the number of observations. Equation (21) can be expressed in matrix form in terms of a time matrix T , an observation vector \vec{y} , a parameter vector $\vec{\beta}$, and a vector of random errors $\vec{\varepsilon}$ as follows:

$$\vec{y} = T\vec{\beta} + \vec{\varepsilon} \quad (22)$$

where $T = \begin{bmatrix} 1 & t_1 & \cdots & t_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_m & \cdots & t_m^n \end{bmatrix}$. The vector of estimated polynomial regression coefficients using ordinary least squares estimation is computed as

$$\vec{\beta} = (T^T T)^{-1} T^T \vec{y} \quad (23)$$

The polynomial regression coefficients $\vec{\beta}$ for the long-term behavior prediction can also be obtained using the iterative gradient descent algorithm as

$$\vec{\beta}(k+1) := \vec{\beta}(k) - \alpha \frac{1}{m} \left[T\vec{\beta}(k) - \vec{y} \right]^T T \quad (24)$$

where $\vec{\beta}(k)$ is the regression coefficient at the k -th iteration and α is the learning rate.

Newly created or updated policies should be automatically verified to determine if they conflict with existing policies or whether merging or splitting is necessary. The policy engine designed for the distributed CRAHN in this study has three reasoning processes: transmission parameter reasoning, conflict reasoning, and optimal policy reasoning. Figure 6 shows the structure of the implemented policy engine.

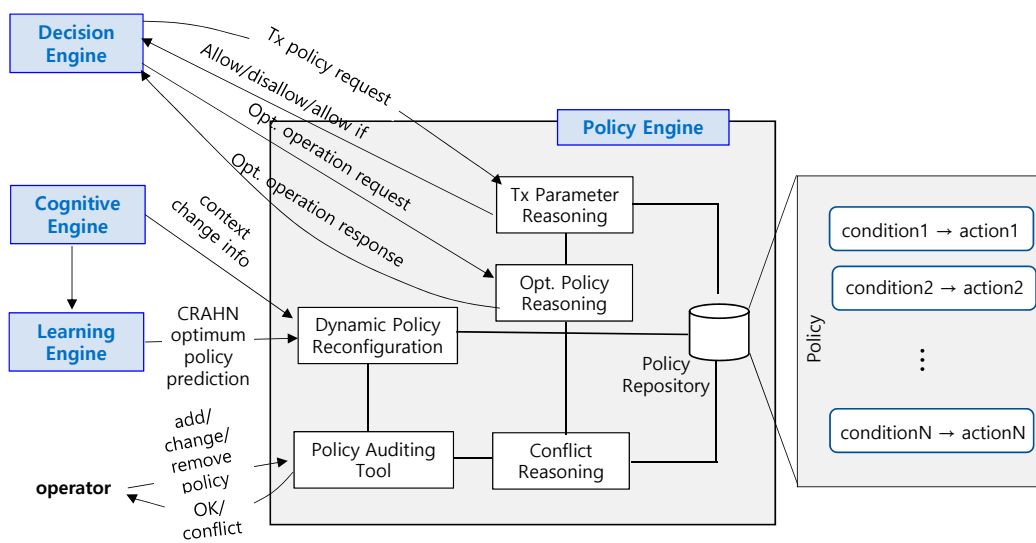


Figure 6. Proposed policy engine architecture.

Optimal transmission parameter reasoning is a process in which the decision engine examines whether the transmission parameters to be used by the device conform to the transmission policy stored in the policy repository. As a result of reasoning for the transmission parameters, the policy engine returns a response in the form of allow, disallow, or conditional approval (allow if certain conditions are satisfied). When the policy engine allows the transmission parameters, the device transmits using the determined transmission parameters. In the case of disallow, the decision engine reconfigures the transmission parameters and then sends a query to the policy engine again. Conditional approval means that transmission is granted when a specific constraint is additionally satisfied; then the device performs transmission within a limit that satisfies the constraint. Conflict reasoning refers to the process of detecting whether a conflict occurs with other existing policies when a new policy is created or an existing policy is updated. When policy conflict is recognized, the policy conflict must be resolved according to a predetermined priority or by the policy operator. The parameters to be queried by the decision engine may not be mapped to a single policy, and in some cases, more than one policy can be applied. When multiple policies can be applied, the optimal policy reasoning selects the optimal policy as a simple intersection concept, or it derives an optimal response through reduction and expansion of conditions. Figure 7 shows some policy engine modules implemented in this research. We used MATLAB and C++ language to describe policies and perform reasoning. As a further study, we have a plan to implement the policy engine on the ontology-based platform.

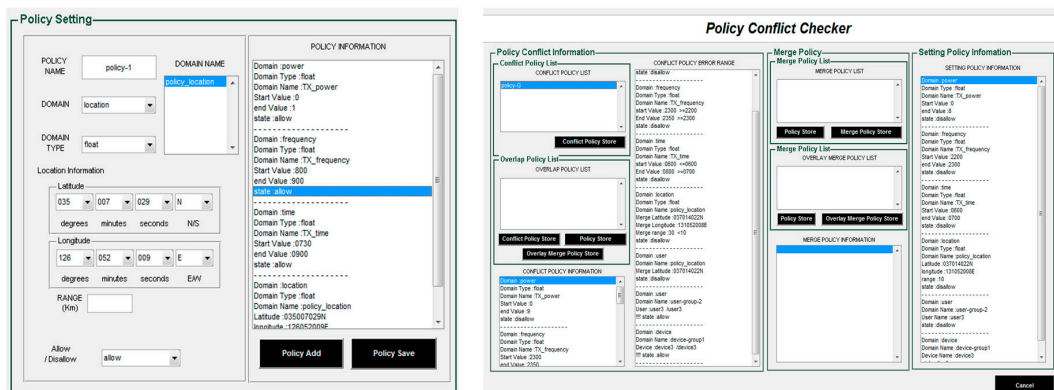


Figure 7. Policy engine implementation.

5. Simulation Results

This section presents the experimental results of the proposed intelligent CRAHN system model and machine learning-based optimization algorithms. We implemented the system in the form of combined sensing, cognitive, decision, policy, and learning engines. Each engine was implemented with C++ and MATLAB programs, and the learning algorithm was programmed using TensorFlow. The performance evaluations were conducted for a narrow sensing band decision, Q-learning-based ad-hoc clustering, and automatic modulation classification methods. Table 2 lists the simulation parameters used in this study. For the path loss model, we used the Friis transmission model with a shadowing effect.

Table 2. Simulation parameters.

Objective	Parameter	Value	Parameter	Value
Narrow sensing band decision	Number of FFT bins	1000	FFT window length L	100~300
	Number of particles	5	Number of iterations	20
	Inertia weight	0.5	Acceleration constants	$c_1, c_2 = 1.4$
	Utility weights	$\omega_1 = \omega_2 = 0.5$	Average length of available bins	$E[\text{ON}] = 70$ $E[\text{OFF}] = 30$
Q-learning based ad-hoc clustering	Q-learning rate	$\alpha = 0.5$	Discount factor	$\gamma = 0.5$
	Percentile threshold	$\eta = 0.5$	Reward weights	$\delta_1 = \delta_2 = 0.5$
	CH fitness weights	$\beta_1 = \beta_2 = \beta_3 = \beta_4 = 0.25$	Simulation area	100 m \times 100 m
	Number of SUs	10–40	Number of PUs	4–12
	Number of clusters	4–6	Primary $E[\text{on}], E[\text{off}]$	10–30 units
CNN-based automatic modulation classification	Size of SCF data	512×512	Number of data samples	10,000
	Learning rate	0.001	Activation function	ReLU, Softmax
		3 layers		
	Convolutional layer	$(5 \times 5 \times 3, 5 \times 5 \times 6, 5 \times 5 \times 12$ filters)	Fully connected layer	5 layers (200, 150, 100, 50, 30 nodes)
	Modulation types	BPSK, BASK, BFSK, QPSK, 16QAM, AM, FM, noise		
Data sample ratio	training:validation:test = 7:2:1			

We implemented a decision engine and a learning engine to determine the optimal sensing band for precise narrowband sensing in the CH. To compare the performance with the proposed method, a method that selects the narrowband range that has the maximum utility among the disjoint narrowband ranges having a predetermined length is implemented without using a sliding window. The compared method also used the proposed utility function and cooperative sensing method. As a result of wideband FFT sensing, the availability bin length was generated using the ON/OFF model, and we assumed that the length ON (available bin length) and OFF (unavailable bin length) follow an exponential distribution.

Figure 8 compares the average utility value according to the change in the window length L for narrowband sensing. As the window size increases, the number of available FFT bins and the maximum length of consecutive available bins in Equation (3) also increase. Therefore, the average utility values of the proposed method and the compared method increase as the observed FFT bin range window increases. Since the proposed method enables more precise band selection using PSO, the average utility value is higher than that of the disjoint window method by more than 20% on average. In addition, compared with the full search method, the average utility value of the proposed method was reduced by 4%, but only 10% of the computation amount was required.

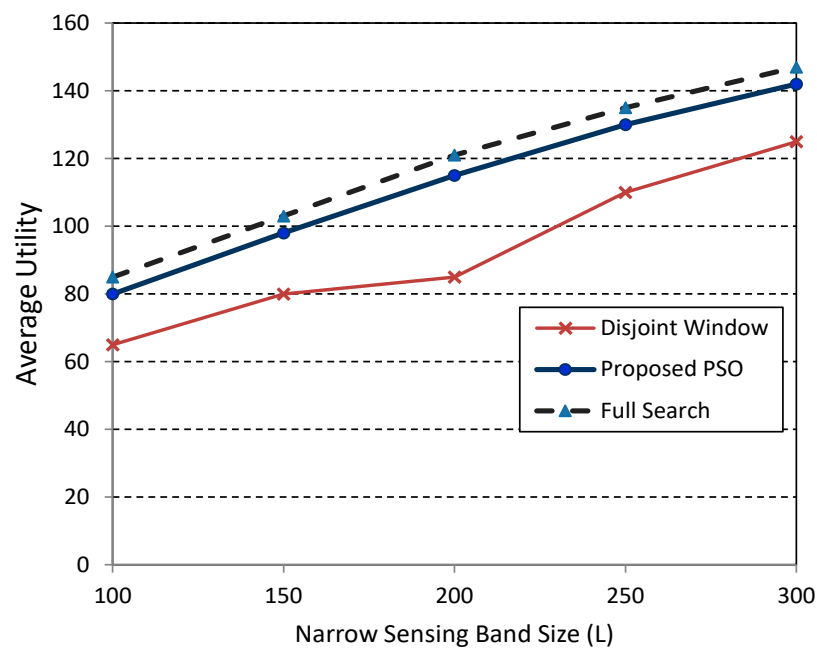


Figure 8. Average utility for narrow sensing band decision.

Figure 9 shows the cumulative distribution function of the utility value by fixing the window size L to 100. As can be seen, when the disjoint window method is used, the probability that the utility value of the selected narrowband is less than 65 is approximately 60%, but the proposed method has a probability that the utility value is less than 65 of only 1%. Therefore, the proposed method can determine a high-utility band for narrowband sensing.

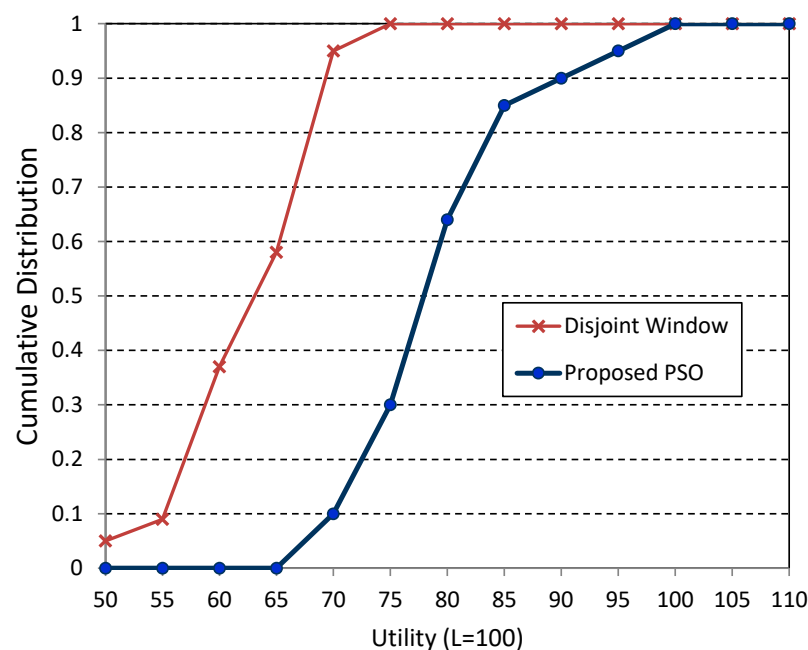


Figure 9. Utility cumulative distribution function.

The proposed Q-learning-based clustering algorithm was evaluated. We compared the clustering performance with K-means clustering for CR condition [21] and multichannel-based clustering (MCBC) [22], where the CH is determined based on node degree, which can communicate using the commonly available channels. Figure 10 shows the average lifetime of a cluster. After a cluster has been configured, when the current cluster data channel (CDC) is no longer available, the residual energy of the CH is not sufficient,

or some member nodes have moved, the cluster network can be broken and may need to be reconfigured. As we can see in Figure 10, the average lifetime of a cluster of the proposed method is approximately 30% longer than that of the compared methods.

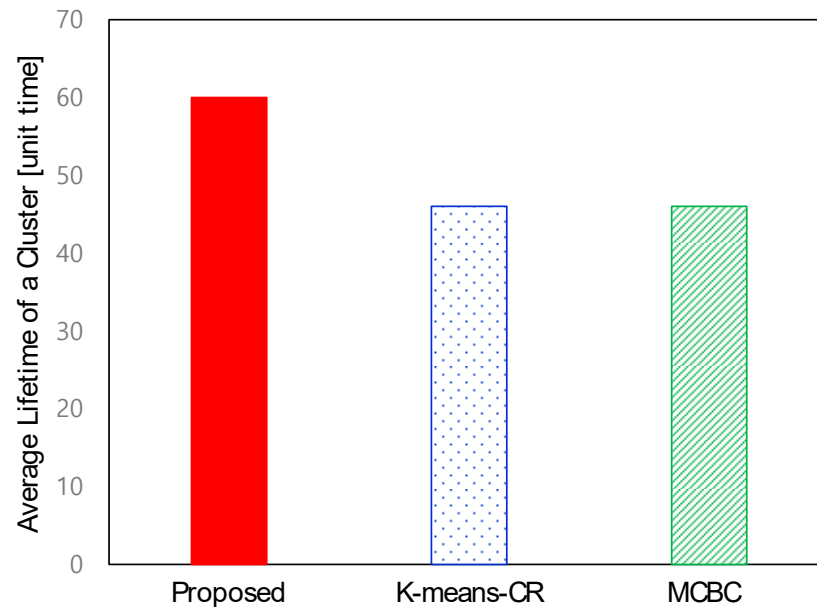


Figure 10. Average lifetime of a cluster.

Figure 11 shows the average Q-value of the selected CDC. The proposed Q-learning-based channel evaluation model and CH fitness function help select the optimum data channel of the cluster so that the Q-value of the CDC that represents channel goodness is higher than that of the MCBC.

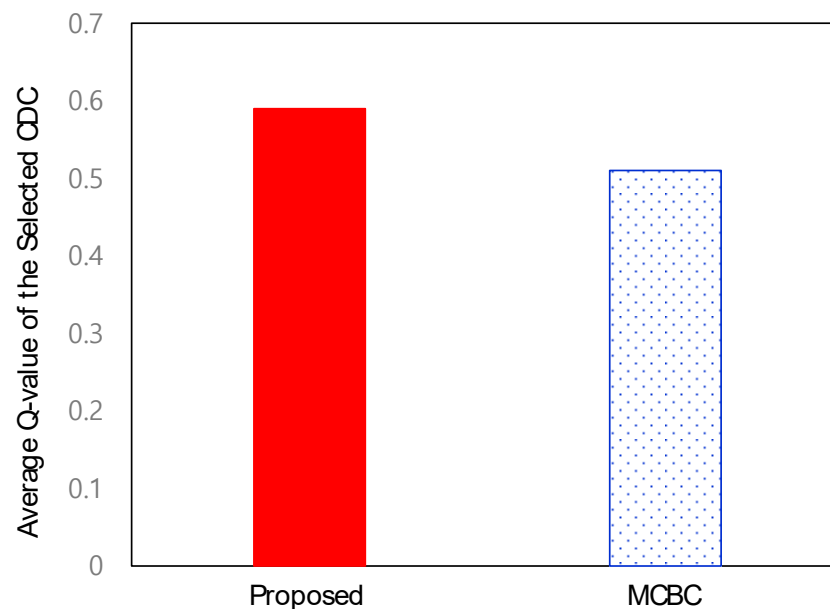


Figure 11. Average Q-value of the selected cluster data channel (CDC).

The proposed CNN-based automatic modulation classification method for signal context awareness is compared with three other classifiers. These include a fully connected network (FCN) classifier using 21 features [23], a 1D-CNN classifier using the SCF image, and a Gaussian mixture model (GMM) classifier using the sampled signal.

Figure 12 presents the classification accuracy of each classifier with changing SNR. As we can see, in the low-SNR region, only the proposed CNN classifier results in accuracy greater than 90%. For the low-SNR case (SNR = -6 dB), the classification accuracy for each modulation type is presented in Table 3. The accuracy of the proposed method is 83–100% for eight different modulation types including noise only. The GMM shows the worst performance, and the classification accuracy is less than 30% for all types. Moreover, it was observed that in the low-SNR region the convergence speed is lower than that of in the high-SNR region during the training process.

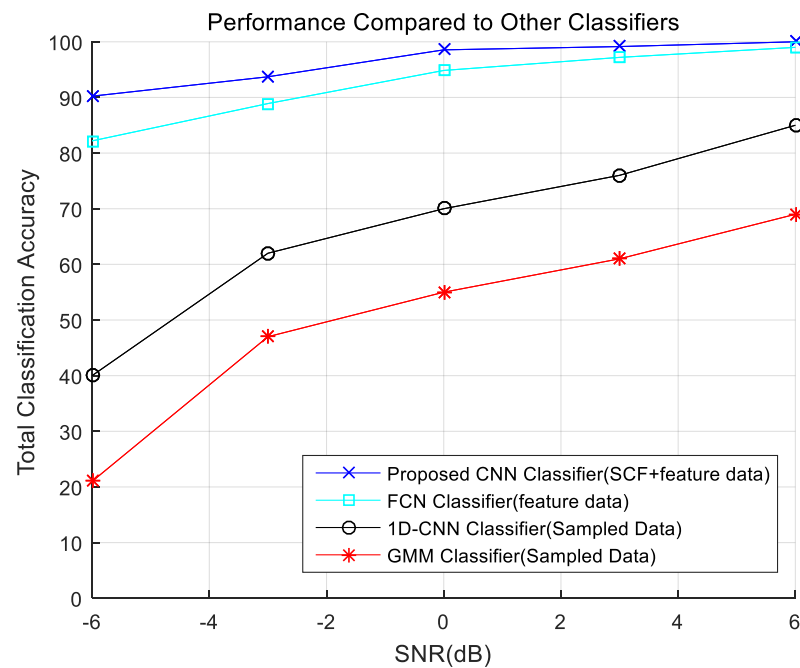


Figure 12. Modulation type classification for various signal-to-noise ratio (SNR) conditions.

Table 3. Classification accuracy for different modulation types at low SNR (SNR = -6 dB).

Modulation Type	Proposed CNN-Based	Fully Connected	1D-CNN	GMM
BASK	93%	82%	43%	23%
BFSK	91%	82%	41%	22%
BPSK	86%	77%	40%	21%
QPSK	85%	75%	32%	21%
16GAM	83%	76%	33%	20%
AM	95%	84%	33%	25%
FM	95%	82%	42%	26%
Noise only	100%	95%	52%	31%

6. Conclusions

In this paper, we presented an intelligent system model for distributed cognitive radio ad-hoc networks and proposed machine learning-based algorithms for network configuration, sensing band decision, and signal classification. The required functions in the sensing, cognitive, decision, policy, and learning engines were defined, and the cooperation structure between the engines to achieve the goal of intelligence and autonomy through a learning engine was presented. To determine the optimal narrow sensing band after periodic rough wideband sensing in the sensing engine, we proposed a bio-inspired PSO algorithm that can determine the optimum narrowband for fine sensing with a high probability of the existence of available channels. For CRAHN configuration and reconfiguration operations, we have presented a Q-learning algorithm that can improve the spectrum efficiency of ad-hoc clusters while minimizing interference with neighboring

networks by learning channel quality, number of connectable neighboring nodes and clusters, and energy consumption. In addition, a CNN-based automatic modulation-type classifier that can be used to coexist with neighboring systems by being aware of the context of the received signal in the cognitive engine is proposed. We designed and implemented a policy engine that can create a network operation policy, detect collisions between policies, and reason whether the decisions in the decision engine conform to the network operation policy. In addition, the proposed policy engine can dynamically update the contents of the policy using regression-based prediction of the changes in the usage pattern of the surrounding radio environments.

The proposed PSO-based narrowband sensing band determination algorithm showed a utility value improved by more than 20% compared with a simple disjoint narrowband search. In the network configuration, it was confirmed that the proposed Q-learning-based method shows a longer network lifetime and higher common data channel quality compared with other CR clustering methods. The proposed CNN-based algorithm using the statistical features for automatic modulation classification guaranteed accuracy of greater than 90% in all SNR ranges, including low-SNR cases. The intelligent system model and the learning algorithms proposed in this paper can be applied to various wireless ad-hoc network applications, including emergency disaster communications and military tactical networks because they can provide stable network services while adaptively responding to dynamic network environment changes.

Author Contributions: K.-E.L.: Conceptualization, development of the system model, system evaluation; J.G.P., review and validation; S.-J.Y., development of machine learning-based algorithms, supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the grant of Agency for Defense Development (ADD) for Cognitive radio under OTM (On the Move) environment and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2020R1F1A1053006).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mansoor, N.; Islam, A.K.M.M.; Zareei, M.; Baharun, S.; Wakabayashi, T.; Komaki, S. Cognitive Radio Ad-Hoc Network Architectures: A Survey. *Wirel. Pers. Commun.* **2015**, *81*, 1117–1142. [[CrossRef](#)]
2. Sudhakaran, C.; Suganthi, M. Distributed Algorithm to Reduce Contention in Emergency Situations by Deploying Cognitive Radio Ad-hoc Controllers. *IET Commun.* **2019**, *13*, 2814–2819. [[CrossRef](#)]
3. Bräysy, T.; Tuukkanen, T.; Couturier, S.; Verheul, E.; Smit, N.; Buchin, B.; Le Nir, V.; Krygier, J. Network Management Issues in Military Cognitive Radio Networks. In Proceedings of the International Conference on Military Communications and Information Systems (ICMCIS), Oulu, Finland, 15–16 May 2017.
4. Zhu, P.; Li, J.; Wang, D.; You, X. Machine-learning-based Opportunistic Spectrum Access in Cognitive Radio Networks. *IEEE Wirel. Commun.* **2020**, *27*, 38–44. [[CrossRef](#)]
5. Arjoune, Y.; Kaabouch, N. A Comprehensive Survey on Spectrum Sensing in Cognitive Radio Networks: Recent Advances, New Challenges, and Future Research Directions. *Sensors* **2019**, *19*, 126. [[CrossRef](#)] [[PubMed](#)]
6. Kaur, A.; Kumar, K. A Comprehensive Survey on Machine Learning Approaches for Dynamic Spectrum Access in Cognitive Radio Networks. *J. Exp. Theor. Artif. Intell.* **2020**. [[CrossRef](#)]
7. Hossain, M.A.; Noor, R.M.; Yau, K.-L.A.; Azzuhri, S.R.; Z'Abu, M.R.; Ahmedy, I. Comprehensive Survey of Machine Learning Approaches in Cognitive Radio-Based Vehicular Ad Hoc Networks. *IEEE Access* **2020**, *8*, 78054–78108. [[CrossRef](#)]
8. Jang, S.-J.; Han, C.-H.; Lee, K.-E.; Yoo, S.-J. Reinforcement Learning based Dynamic Band and Channel Selection in Cognitive Radio Ad-hoc Networks. *EURASIP J. Wirel. Commun. Netw.* **2019**, *2019*, 1–25. [[CrossRef](#)]
9. Hossen, M.A.; Yoo, S.J. Q-Learning Based Multi-Objective Clustering Algorithm for Cognitive Radio Ad Hoc Networks. *IEEE Access* **2019**, *7*, 181959–181971. [[CrossRef](#)]
10. Mitola, M., III. Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio. Ph.D. Thesis, Royal Institute of Technology, Stockholm, Sweden, 2000.
11. Wilkins, D.; Denker, G.; Stehr, M.; Elenius, D.; Senanayake, R.; Park, M.; Talcott, C. Policy-based Cognitive Radios. *IEEE Wirel. Commun.* **2007**, *14*, 41–46. [[CrossRef](#)]
12. Osman, M.M.A.; Syed-Yusof, S.K.; Malik, N.N.N.A.; Zubair, S. A Survey of Clustering Algorithms for Cognitive Radio Ad Hoc Networks. *Wirel. Netw.* **2018**, *24*, 1451–1475. [[CrossRef](#)]

13. Chen, T.; Zhang, H.G.; Maggio, G.M.; Chlamtac, I. CogMesh: A Cluster-based Cognitive Radio Network. In Proceedings of the IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN2007), Dublin, Ireland, 17–20 April 2007; pp. 168–178.
14. Kim, M.-R.; Yoo, S.-J. Distributed Coordination Protocol for Common Control Channel Selection in Multichannel Ad-Hoc Cognitive Radio Networks. In Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (IEEE WiMob), Marrakech, Morocco, 12–14 October 2009; pp. 227–232.
15. Sun, H.; Nallanathan, A.; Wang, C.-X.; Chen, Y. Wideband spectrum sensing for cognitive radio networks: A survey. *IEEE Wirel. Commun.* **2013**, *20*, 74–81.
16. De Vito, L. Methods and technologies for wideband spectrum sensing. *Measurement* **2013**, *46*, 3153–3165. [[CrossRef](#)]
17. Zhang, H.; Xu, N.; Xu, F.; Wang, Z. Graph cut based clustering for cognitive radio ad hoc networks without common control channels. *Wirel. Netw.* **2018**, *24*, 209–221. [[CrossRef](#)]
18. Meng, F.; Chen, P.; Wu, L.; Wang, X. Automatic Modulation Classification: A Deep Learning Enabled Approach. *IEEE Trans. Veh. Technol.* **2018**, *67*, 10760–10772. [[CrossRef](#)]
19. Mendis, G.J.; Wei, J.; Madanayake, A. Deep Learning-based Automated Modulation Classification for Cognitive Radio. In Proceedings of the IEEE International Conference on Communication Systems (ICCS), Rajasthan, India, 14–16 October 2017; pp. 1–6.
20. Azzouz, E.E.; Nandi, A.K. Automatic Modulation Recognition. *J. Frankl. Inst.* **1997**, *334*, 241–273. [[CrossRef](#)]
21. Benmammam, B.; Taleb, M.; Krief, F. Diffusing-CRN k-means: An Improved k-means Clustering Algorithm Applied in Cognitive Radio Ad Hoc Networks. *Wirel. Netw.* **2016**, *23*, 1849–1861. [[CrossRef](#)]
22. Berwer, R.K.; Kumar, S. Multi Channel-based Clustering in Cognitive Radio Networks. In Proceedings of the International Conference on Smart Technologies for Smart Nation (SmartTechCon), Bengaluru, India, 17–19 August 2017; pp. 665–670.
23. Kim, B.; Kim, J.; Chae, H.; Yoon, D.; Choi, J. Deep Neural Network-based Automatic Modulation Classification Technique. In Proceedings of the IEEE International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 19–21 October 2016; pp. 579–582.