PAPER 65 – AN OVERVIEW OF APPLICATIONS OF COGNITIVE WIRELESS COMMUNICATION SIMULATORS

F. O. Ehiagwina[1,2*], N. T. Surajudeen-Bakinde[1], A. S. Afolabi[2], L. O. Afolabi[1], A. M. Usman[1,3]

[1]Department of Electrical and Electronics Engineering, University of Ilorin, Ilorin, Kwara State, Nigeria
[2]Department of Electrical/Electronic Engineering, The Federal Polytechnic Offa, Offa, Kwara State, Nigeria
[3]Department of Electrical & Electronic Engineering, Stellenbosch University, Stellenbosch, South Africa.
[*]Email: Frederick.ehiagiwna@fedpoffaonline.edu.ng

ABSTRACT
Cognitive radio is proposed in the literature to address the ironical problem of spectrum scarcity owing to inefficient utilisation. This idea has recently gained popularity because it provides wireless users with the ability to detect and efficiently adapt operating parameters based on interactions with the surrounding radio environment. In the evaluation of cognitive radio networks simulators are used. The simulators differ in terms of platforms and technicalities. Thus making selection process difficult. In this review, attention is paid to cognitive radio network simulators such as Network Simulator 2 and 3 (NS-2) and (NS-3), Java Simulator (J-Sim), Operations Network (OPNET) and MATLAB. In the selection of literature covered in this review, attention was paid to publications not later than 2017 from reputable databases. This review of cognitive radio network simulators will provide useful insights to new researchers in this field. Furthermore, a case study is reported on the application of the Monte Carlo simulation method to the cognitive radio network.

KEYWORDS: Cognitive radio, network simulators, random numbers, Monte Carlo simulation

1. INTRODUCTION

Cognitive radio is advocated in the literature to address the ironical problem of spectrum scarcity owing to inefficient utilisation. This idea has recently gained popularity because it provides wireless users with the ability to detect and efficiently adapt operating parameters based on interactions with the surrounding radio environment. In the last few years since the introduction of the concept by Mitola, there have been numerous significant advances in cognitive radios (Hu et al., 2018; Arjoune & Kaabouch, 2019). The structure of a typical CRN and the cognitive cycle are shown in Figures 1 and 2. The cognitive cycle is at the core of CRN operation.

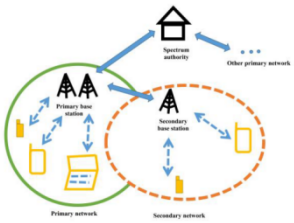


Figure 1: Architecture of a typical cognitive radio network

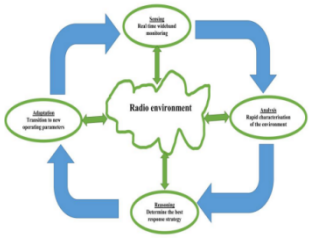


Figure 2: Cognitive cycle

It is crucial to have a complete grasp of CRNs via intense experimental analysis to gain deeper insight into CRNs and to create more reliable procedures. The two main categories of CRN experimental analysis techniques are platform-based emulation and computer-assisted simulation. Due to several benefits, computer-assisted simulation is used more frequently in CRN experimental analysis than platform-based experimental analysis. Some of the benefits include: 1) Computer-assisted simulators are more practical and less expensive than CRN platforms; whereas many simulators are open source and free to use or have academic versions that may be obtained from the Internet; 2) As the technology matures, computer-assisted simulators may be equipped with stronger technical assistance; yet, despite certain commercial CRN platforms having strong technical support, their lack of module libraries stems from the lengthy development process; and 3) The use of computer-assisted simulators, which allow for the adjustment of a wide range of CRN parameters without changing platform hardware architecture, makes them more appropriate for large-scale CRN experiments with fewer restrictions. Certain radio scenarios are difficult for radio platforms to simulate in the actual world, but computer-assisted simulation is completely relevant.

However, selecting a powerful and user-friendly simulator is still a challenge for many researchers due to the following reasons: 1) simulators are maintained by either commercial companies or open source communities;

historical technical development and current technique support (CRN patches, libraries, etc.) situations differ greatly among different simulators; and 2) CRN is a complex system, and some researches are typically undesirable to cover all functionalities; some simulators are not user-friendly while others are (Assegie & Nair, 2019; Dong et al., 2018; Kafetzis et al., 2022).

In contrast to conventional wireless networks, CRN is set up using a novel media access strategy. Via spectrum sensing, analysis, and cooperative decision-making, the transceiver uses dynamic spectrum access (DSA) to operate. The transceiver performs routing activities, QoS monitoring, and application management, as well as low-level coordination and configurations, in high-level functionality. When creating CRN simulators, each of these must be taken into account. Moreover, effective simulators should be able to replicate a flexible but controllable radio environment, be simple to install and setup, and be extendable with adaptive modular architectures (Agrawal et al., 2022; Aloqaily et al., 2020; Dong et al., 2018). There are also CRN simulators that are cloud-based. Examples of these are discussed in (Aron & Abraham, 2017; Campanile et al., 2020; Kafetzis et al., 2022).

The simulation platforms are described, coupled with studies deploying such a platform within the limitation of prescribed page numbers. Emphasis was placed on articles not later than 2017.

2. COGNITIVE RADIO NETWORK SIMULATOR

There are general-purpose simulators such as Network Simulator 2 and 3 (NS-2) (Patel et al., 2018; Sharma et al., 2019) and (NS-3), Java Simulator (J-Sim), Operations Network (OPNET), NetSim, Objective Modular Network Testbed (OMNEST) (Omnest, 2020), Objective Modular Network Testbed in C++ (OMNeT++), GNU and MATLAB. Others include Mininet (2020), Data Center Simulation tool (DCSim) (Byrne et al., 2017), and Cloud Simulator (CloudSim). In this section of the review, attention is paid to some cognitive radio network simulators based on the aforementioned platforms (Dong et al., 2018).

2.1. Cognitive Radio Network Simulation Based on NS-2 Simulator

NS-2 is an object-oriented simulator that has very detailed network modules. This simulator is usually used for designing protocols and new network architectures, comparing performances of various networks and studying network traffic. Further, NS-2 can be customized for specific simulation scenarios just like CRNs (Patel et al., 2018). On the contrary, this simulator suffers from its implementation complexity (Sharma et al., 2019). Figure 3 shows the architecture of the NS-2. It is a well-structured module that was mainly made to build Media Access Control (MAC) protocols.

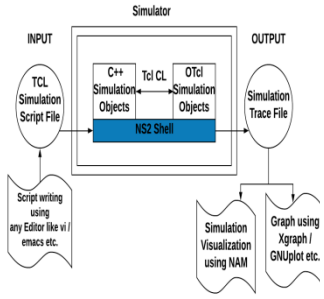


Figure 3: NS-2 Architecture (Patel et al., 2018)

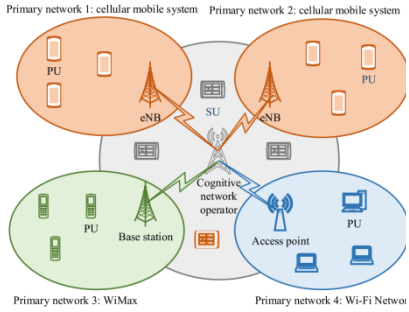


Figure 4: Cognitive Heterogeneous network

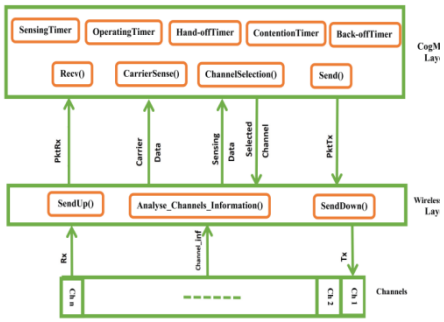


Figure 5: Structure of the CR node object in CogNS with accessories

Cognitive radio features can be incorporated into the NS-2. Examples of this include the CRCN, CRAHN and the CoGNS. All are at various stages of development (Darabkh et al., 2020). Research work on cognitive radio simulation based on NS-2 includes cognitive radio sensor network (Bukhari et al., 2018), channel selection/assignment, routing technique for multichannel cognitive radio networks, the effect of SU on the higher layers of the protocol stack and so on.

2.2. Cognitive Radio Ad Hoc Network (CRAHN)

Another NS-2-based emulator is the Cognitive Radio Ad Hoc Network (CRAHN), which detects primary user activity and pauses transmission when the principal user occupies the spectrum. Because nodes are always seeking vacant spectrum to communicate on, this simulator allows them to alter transmission and receiving channels. Examples of works on CRAHN include spectrum routing and aggregation (Ramkumar & Vadivel, 2018; Devapriya & Kannan, 2020; Singh & Dutta, 2020). Sridevi and Venkatesan (2017) and Devapriya and Kannan (2020) examined CRAHN as it relates to spectrum aggregation. According to Devapriya and Kannan (2020), the primary goal of a spectrum aggregation-based pleasant routing protocol for CRAHNS is to increase imperative

profitability, improve throughput and reduce framework delay for the cognitive heterogeneous network. A typical cognitive network operator-based 5G heterogeneous network is shown in Figure 4. The Stackelberg gaming hypothesis was used to realize high throughput and decreased delay, resulting in an improvement of the spectrum and energy efficiency.

2.3. Cognitive Network Simulator (CogNS)

CogNS is another NS-2-based simulator. It may be used to explore and assess the influence of lower-layer protocols, such as the MAC and physical layer, on transport and network layer protocols. Because packet loss probability, end-to-end delay and throughput are important QoS requirements in real-time dependable applications, the CogNS system evaluates these metrics over CRNs. However, CR-related characteristics such as primary user activity model, sensing time and frequency should be addressed while designing novel network and transport layer protocols over CRNs (Singhal & Thanikaiselvan, 2022; Hassani & Berangi, 2019).

Figure 5 shows the structure of each CogNS CR node object, as well as modules, timers and their relationships. Based on the topology of the mobile node object in NS2, the MAC layer, physical layer and channels are changed. The transmission of control information and data packets is shown by the arrows between different levels. The main change was made to the Mobile node's MAC layer. A cognitive CSMA/CA-based MAC protocol dubbed CogMAC is implemented in the MAC layer of CogNS.

Researches, where the CogNS is utilized, are in congestion control schemes for transport protocol and in the evaluation of the effect of secondary user block throughput of TCP in CRSN (Hassani & Berangi, 2018 and 2019).

2.4. Cognitive Radio Simulation Based on NS-3

Network Simulator 3 (NS-3) is not an evolution of NS2. Hence, it is not a backwards-compatible extension of NS2. Some models from NS2 have been ported to NS3. It is a discrete-event network simulator targeted primarily for research and educational use. The development of the open-source project began in 2006 (NS-3 Tutorial Release ns-3.18.1, 2013). The programme is written in C++ and Python. In general, NS3 has the advantage of supporting many radio models such as 802.11, 802.16, 802.15.3 and 802.15.4 and it has various options for network topologies and traffic patterns. This allows researchers to build different simulation scenarios to test their protocols. NS3 has three cognitive radio modules.

NS3 is created as a collection of libraries that may be integrated and used in conjunction with other software libraries. NS3 is more modular in this sense than certain simulation platforms, which offer customers a single, integrated graphical user interface environment where they can do all functions. NS-3 can be utilized with a variety of external animators, as well as data analysis and visualization tools. Users should anticipate working using C++ and/or Python software development tools and at the command line. NS3 is used on Linux operating systems. However, it can be utilized in FreeBSD and Cygwin (for Windows). The development of native Windows Visual Studio support is currently underway (Masek et al., 2019).

2.5. Cognitive Radio Emulator in NS-3 (CRENS3)

CRENS3 extension is written to add cognitive radio (CR) capabilities to the NS-3. CRENS3 follows the NS-3 standards in terms of documentation through the utilization of Doxygen. Khan and Islam (2019) incorporate the fee dBack-based approach to CRENS3 as a modifier. The authors noted that the existing cognitive module of CRENS3 made provision for three interfaces for each device. They are the transmitter interface, control interface and receiver interface. The block diagram of the CRENS3 extension in NS-3.17n is shown in Figure 6. In this case, the actual cognitive radio transceiver is being emulated by the transmitter and receiver interfaces of the module (Khan & Islam, 2019).

2.6. Cognitive Radio Simulation Based OMNeT++

Objective Modular Network Testbed in C++ (OMNeT++) is a general-purpose tool that has nested architecture for modelling networks. It also has a mobility extension for wireless networks named Mixed Simulator MiXiM. The Objective Modular Network Testbed in C++ (OMNeT++) is a well-designed open-source, component-based, discrete event-driven simulator written in C++. As an open-source network simulator, OMNET + + has plenty of public documents like NS-2 and NS-3 and it is also equipped with a powerful graphical user interface as OPNET does. Thus, OMNeT++ becomes successful in both industry as well as academia. OMNeT++ provides a hierarchically nested architecture for modelling. An OMNeT++ model is composed of many modules that pass messages back and forth for communication (Dong et al., 2018). Simple modules are the basic active elements used to model a system. They are programmed in C++ using the OMNeT++ simulation class library. Users can design new simple modules in OMNeT++ IDE. Meanwhile, a large number of simple modules are provided in the OMNeT++ library. Single modules can be assembled into more functional compound modules and so forth for compound modules with unlimited hierarchy levels. NED is the adhesive language in OMNeT++ for declaring simple modules, assembling compound modules and constructing network topologies. In the process of simulation and after, Then, an integrated GUI of OMNeT++, provides users with powerful features for ongoing animation

tracing, output tracing and synthesizing. It is convenient for users to look for the output results of the system or even each module.

On top of OMNeT++, MiXiM (Mixed Simulator) has been developed as a simulation framework for wireless and mobile networks. It provides wireless MAC protocols, as well as detailed models of radio wave propagation, interface estimation and radio transceiver power consumption. In the MAC layer, IEEE 802.11 and IEEE 802.15.4 protocols have been developed. Due to the component-based structure, it is relatively less time-consuming to implement a new CRN protocol via OMNeT++. To study cognitive radio, some reported efforts have built up proper framework and simulation models using OMNeT++, which shed light on developing CRNs efficiently (Dong et al., 2018). In smart home cognitive wireless sensor networks, a distributed cooperative spectrum resource allocation was simulated in OMNeT++ by Aroua et al. (2017).

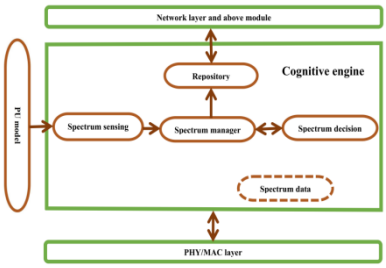


Figure 6: The block diagram of CRENS3 extension in NS-3.17 (redrawn by Dong et al., 2018)

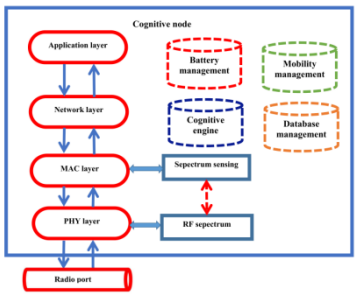


Figure 7: The node structure of CR Simulator in OMNeT++ (redrawn from Dong et al., 2018)

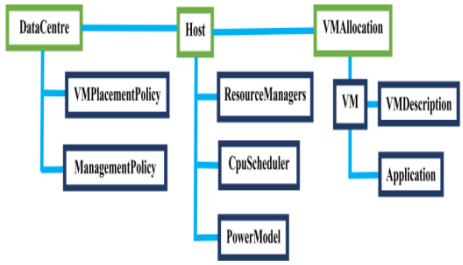


Figure 8: Data Center Simulation Architecture (redrawn from Byrne et al., 2017)

2.7. Cognitive Radio Simulation Based on Java (J-Sim)

J-Sim is another general-purpose simulator that was developed in Java. This simulator has a script interface that can run several script-based languages like Python or Perl. Further, J-Sim is described as neutral, extensible and reusable by authors. This is because it is closely coupled with autonomous component architecture, i.e., it relies on self-regulating, replaceable and linked components that help in managing complexity and inspiring reuse. More specifically, it uses a multi-agent-based computational model (MABM), which simulates the common interactions of a group of independent entities (agents) composing a system. This provides a perspective to study the agents' effects on the entire system. Interestingly, the MABM showed great efficiency in representing complex and dynamic systems. As for the weaknesses, MABMs, such as J-Sim, are inclined to be more resource and computational time exhaustive, less suitable for event-driven simulations and known to be fairly difficult to verify and validate.

2.8. Optimized Network Engineering Tool (OPNET)

Optimized Network Engineering Tool (OPNET) is a widely used object-oriented, discrete event-driven, general-purpose network simulation tool with many user-preferred features, tool sets and a huge library of available network protocols. It helps set up different network topologies and is convenient for users who need to design customized network protocols or applications. Generally speaking, using OPNET for the experimental study consists of three main procedures: modelling, simulating and analysis. In the simulation, all required models of protocols should be created and defined first. Then, the network is simulated based on the developed models with hierarchical structures. OPNET defines a network as a collection of submodels, in which the hierarchical structure is divided into the network, node and process domains. It runs in C/C++. It was initially free and made for multiple computer environments.

As a commercially supported software, OPNET has powerful graphical support of network topology, traffic flow control and various entities of network units. OPNET has a large user group and many discussion forums, so it is easy to be extended to implement CRNs. OPNET provides many integrated propagation models such as Free Space, Hata, CCIR, Longley-Rice, TIREM, Walfish-Ikegami and more added models such as Rayleigh, Ricean, two-ray and 802.15.4/ZigBee models from OPNET community (Dong et al., 2018).

3. COGNITIVE RADIO SIMULATION BASED ON MATLAB

GNU and MATLAB are signal processors that are easy to use, great in signal processing and support customized modules to mimic various functionalities. MATLAB is a software that is widely used with radio generation and network simulations; because of its rich mathematical library. In CRNs, MATLAB is used for signal analysis that exists in spectrum sensing and for modelling CRNs partially. As one of the most widely used scientific

experimental tools with powerful mathematical toolboxes and packages, MATLAB (Matrix Laboratory) is the pioneer of cognitive radio simulation because of its popularity in numerical computing and ease of use. Energy detection is the most fundamental technique in spectrum sensing in CRNs. To justify the feasibility and viability of energy detection, the PU signal is first generated with added noise in the radio environment. With a proper radio propagation model, the received signal can be simulated on the SU side and the energy detection technique is easily adopted by measuring the received signal level (Dong et al., 2018).

MATLAB is more than a signal processor; it is a comprehensive experiment tool that can be widely used in CRN studies. With a plethora of toolboxes, especially supported by Simulink, MATLAB can specify design details of network components as well as topology, which makes it ideal for operation on signal processing and spectrum management. Usually, Simulink is cooperatively used to drive CRN hardware platforms for experimental analysis. In the MATLAB/Simulink environment, software was created by Shuaibu-Sadiq and Anyasi (2020) to reuse a range of vacant frequencies using multiple model elements working together to produce spectrum sensing. A register transfer level - software-defined radio was used to interface the built Simulink model, which evaluates the estimated noise power of the received signal over a particular time and bandwidth. The threshold estimation produces a binary digit that can be used for decision and prediction. However, due to the lack of pre-defined network architecture, Matlab/Simulink is not convenient for higher-level network simulation or comprehensive construction studies (Dong et al., 2018).

4. EVALUATION OF SELECTED SIMULATION SCHEMES

A summary of some of the simulators discussed is presented in Table 1.

Table 1: Evaluation of selected network simulators

	Remark	Programme	CRN	Multiple Channel	Primary Receiver Activity
CogNS	Based on NS-2	C++/OTCL	Yes	Yes	No
CRAHN	Based on NS-2	C++/OTCL	Yes	No	Yes
CrSimulator	Based on OMNet++	C++/NED	Yes	Yes	No
J-SIM		Java	No	No	No
CRCN	Based on NS-2	C++/OTCL	Yes	No	No
MiXiM	Based on OMNet++	C++/NED	No	No	No

5. CASE STUDY: MONTE CARLO SIMULATION OF COGNITIVE RADIO NETWORK

A cognitive radio network involves a lot of dynamism. Given that, Monte Carlo simulation may be used to evaluate their performance. The probability of various outcomes in a process that cannot be easily anticipated due to the interference of random factors is modelled using a Monte Carlo simulation owing to ease of implementation and it is useful in simulating both deterministic and stochastic problems. It is a method for comprehending the effects of risk and uncertainty. In various disciplines, including finance, business, physics, and engineering, Monte Carlo simulation is used to solve a variety of issues (Makarfi et al., 2021; Helmy, Hassan & Ismail, 2022).

In this case study, a cognitive network was created to cover a 5000 by 5000 m^2 area and PU and SUs were randomly generated. It was observed that SU-to-SU calculations accounted for more than 96% of the collisions (Figure 12), hence the need for a spectrum decision manager to coordinate spectrum assignment to SUs.

The proposed algorithm for the CRN within the GSM 900/1800 Bands is shown in Table 2 (Ehiagwina et al., 2022). The simulation parameter for the case study is shown in Table 3.

Table 2: An adaptive spectrum decision algorithm for CRN simulation

1. Initialise the algorithm by specifying the spectrum band links, create a neural network (diagram), set constants and parameters

2. Perform spectrum sensing and acquire spectrum power densities $P(t_i, f_j)$ over time

3. Sort data into matrix form: $X = \begin{pmatrix} P(t_1, f_1) & \cdots & P(t_1, f_j) \\ \vdots & \ddots & \vdots \\ P(t_i, f_1) & \cdots & P(t_i, f_j) \end{pmatrix}$

4. Set decision threshold λ_k and $I_\phi(t_i, f_j)$: $P(t_i, f_j) \rightarrow \{0,1\}$ and apply the following rule:

$I_\phi(t_i, f_j) = \begin{cases} 1, & if P(t_i, f_j) \geq \lambda_k \\ 0, & if P(t_i, f_j) < \lambda_k \end{cases}$. Do for each entry on **3**. above

5. Determine the average spectrum duty cycle for each link using $\mu_j = N_f^{-1} \sum_{i=j=1}^{N_f} I_\phi(t_i, f_j)$

6. Obtain the spatio-temporal variations of duty cycles using $\mu_{(i,j)}(t) = N_f^{-1} \sum_{i=j=1}^{N_f} I_\phi(t_i, f_j)(t)$

7. From 6. above obtain CVDs according to $CVD_i = t_s(1-\mu_j)$

8. Use the outputs from **7**. as targets for the neural network with the data of **3**. as inputs

9. Develop an ANN-LSTM algorithm to obtain predicted CVDs, p_{CVD}

10. Is $p_{CVD} \geq k_{CVD}$, a defined threshold? If yes, proceed to the next step. If otherwise, proceed to step **13**. for a test of the existence of spatial opportunity using carrier-interference-ratio (CIR)

11. Compute the communicable area for the primary and secondary systems

12. Randomly generate the Locations of PR, ST and SR within the service area.

13. For an SU transmission, compute CIR_{PR} and CIR_{SR} then check if $CIR_{PR} > \lambda_2$ and $CIR_{SR} > \lambda_1$.

14. Check if the allowable number of trials has been reached, if no repeat steps 11 - 13 and if yes proceed to step 15.

15. Compute the probabilities where $CIR_{PR} > \lambda_2$ and $CIR_{SR} > \lambda_1$ are satisfied.

5.1. Performance Metrics for the Simulation

Making decisions is a crucial part of the CR's cognitive cycle. If the decision-making process is delayed, the quality of service (QoS) is negatively impacted; nevertheless, if the decision-making process is quick and the SU makes an incorrect conclusion, the SU may interfere with PU. Thus, the primary user interference probability (PUIP) will be evaluated to gauge the effectiveness of the created spectrum choice framework.

The ratio of the number of open slots found by the cognitive user to the total number of open slots present in the system over a limited period is known as spectrum utilization. This is expressed mathematically as in eqn. (1).

$$\eta_{SU} = \frac{number\ of\ vacant\ spectrum\ slot\ sensed}{total\ number\ of\ vacant\ slots\ in\ the\ spectrum\ fragment} \times 100\% \quad (1)$$

The PUIP is the ratio of the slots in which SU also uses the channel whereas a PU is transmitting. This is due to a false alarm. The PUIP is given as eqn. (2).

$$PUIP = \frac{total\ number\ of\ collision\ slots}{total\ number\ of\ PU\ activity\ slots} \quad (2)$$

The Simulation parameters for the hybrid spectrum opportunities extraction scheme may be seen in (Ehiagwina et al., 2022).

5.2. Result of the Case Study

The simulation revealed that over 96% of collisions were calculated from SU to SU interactions (Figure 11), demonstrating the necessity of a spectrum choice manager to coordinate SU spectrum assignment. As shown in the Figure, the percentage of PU to SU collision is quite low, with a maximum of 3. 456%.

The plot of users' collisions is shown in Figure 12. The results showed that SU-SU conflicts contribute to about 97% of the collisions, hence the development of the spectrum extraction decision algorithm to minimise these collisions.

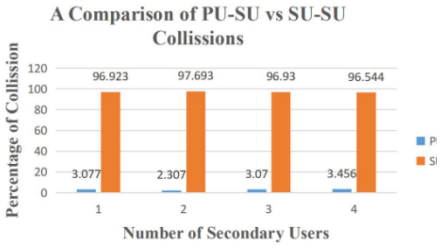


Figure 11: PU-SU vs SU-SU collisions

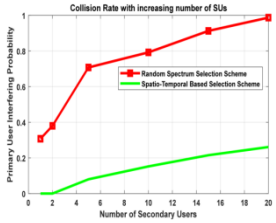


Figure 12: Collision rate with an increasing number of SUs

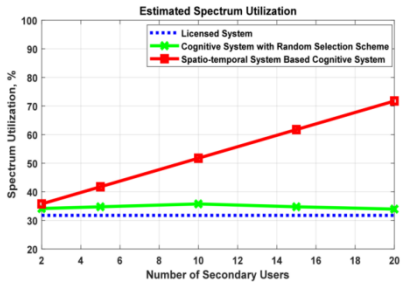


Figure 13: Spectrum utilization for the increasing number of SUs

About 4.5-fold reduction in collision rate was noticed when spatial and temporal-based spectrum selection scheme was used as against the random selection scheme. The estimated increase in spectrum utilization can be seen in Figure 13. Spectrum usage increased by more than a factor of two. This shows a preference for a spectrum extraction decision method that incorporates both spatial and temporal spectrum extractions and is adaptive.

6. CONCLUSION

It may be deduced from the foregoing overview of several of the existing CRN simulator that despite the much interest in CRN by researchers, who are keen on examining the various aspects, there seems to be no simulator robust enough to incorporate the various aspects and demands of cognitive radio networks. Several researchers are executing their algorithms via modifications of the existing simulators. In this study, we also reported on the case where Monte Carlo simulation was used for assessing the performance of CRN. The formulation and simulation of an adaptive algorithm for controlling the decision about the extraction of spectrum for CRN led to a reduction in PUIP and a rise in spectrum utilization. By creating an adaptive spectrum choice framework for the cognitive radio network, this research increased the efficiency of spectrum utilization.

ACKNOWLEDGEMENT

The Tertiary Education Trust Fund (TETFund) of Nigeria is acknowledged for supporting the study through the Academic Staff Training and Development (AST&D) program. We also thank the University of Ilorin's Department of Electrical and Electronics Engineering for supplying the measuring instrument utilized in the data collection phase of this investigation.

REFERENCES

Agrawal, S. K., Samant, A., & Yadav, S. K. (2022). Spectrum sensing in cognitive radio networks and metacognition for dynamic spectrum sharing between radar and communication system: A review. *Physical Communication*, 101673.

Aloqaily, M., Salameh, H. B., Al Ridhawi, I., Batieha, K., & Othman, J. B. (2020). A multi-stage resource-constrained spectrum access mechanism for cognitive radio IoT networks: Time-spectrum block utilization. *Future Generation Computer Systems*, *110*, 254-266.

Arjoune, Y., & Kaabouch, N. (2019). A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges and future research directions. *Sensors*, *19*(1), 126.

Aron, R., & Abraham, A. (2022). Resource scheduling methods for cloud computing environment: The role of meta-heuristics and artificial intelligence. *Engineering Applications of Artificial Intelligence*, *116*, 105345.

Aroua, S., El Korbi, I., Ghamri-Doudane, Y., & Saidane, L. A. (2017, July). A distributed cooperative spectrum resource allocation in smart home cognitive wireless sensor networks. In *2017 IEEE Symposium on Computers and Communications (ISCC)* (pp. 754-759). IEEE.

Assegie, S., & Nair, P. (2019). Performance analysis of emulated software-defined wireless network. *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, *16*(1), 311-318.

Bukhari, S. H. R., Siraj, S., & Rehmani, M. H. (2018). NS-2-based simulation framework for cognitive radio sensor networks. *Wireless Networks, 24*(5), 1543-1559.

Byrne, J., Svorobej, S., Giannoutakis, K. M., Tzovaras, D., Byrne, P. J., Östberg, P. O., Gourinovitch, A., & Lynn, T. (2017). A review of cloud computing simulation platforms and related environments. *International Conference on Cloud Computing and Services Science, 2*, 679-691.

Campanile, L., Gribaudo, M., Iacono, M., Marulli, F., & Mastroianni, M. (2020). Computer network simulation with ns-3: A systematic literature review. *Electronics, 9*(2), 272.

Darabkh, K. A., Amro, O. M., Al-Zubi, R. T., Salameh, H. B., & Saifan, R. (2020). JavaSim-IBFD-CRNs: novel Java simulator for in-band full-duplex cognitive radio networks over the Internet of Things environment. *Journal of Network and Computer Applications, 172*, 102833.

Devapriya, S., Kannan, N. (2021). High energy and spectral efficiency analysis for CRAHN-based spectrum aggregation. *Journal of Ambient Intelligence and Humanized Computing, 12,* 6621–6628.

Dong, Q., Chen, Y., Li, X., & Zeng, K. (2018). A Survey on Simulation Tools and Testbeds for Cognitive Radio Networks Study. *arXiv preprint arXiv:1808.09858*.

Ehiagwina, F. O., Surajudeen-Bakinde, N. T., Afolabi, A. S., & Usman, A. M. (2022). A hybrid spectrum opportunities extraction scheme for cognitive wireless communication. *Telematics and Informatics Reports*, *7*, 100014.

Hassani, M. M., & Berangi, R. (2018). A new congestion control mechanism for transport protocol of cognitive radio sensor networks. *AEU-International Journal of Electronics and Communications, 85*, 134-143.

Hassani, M. M., & Berangi, R. (2019). Impact of Secondary User Block on the TCP Throughput in Cognitive Radio Sensor Networks. *Wireless Personal Communications, 109*(4), 2221-2238.

Hu, F., Chen, B., & Zhu, K. (2018). Full spectrum sharing in cognitive radio networks toward 5G: A survey. *IEEE Access*, *6*, 15754-15776.

Kafetzis, D., Vassilaras, S., Vardoulias, G., & Koutsopoulos, I. (2022). Software-defined networking meets software-defined radio in mobile ad hoc networks: State of the art and future directions. *IEEE Access*, *10*, 9989-10014.

Makarfi, A. U., Kharel, R., Rabie, K. M., Kaiwartya, O., Li, X., & Do, D. T. (2021, March). Reconfigurable intelligent surfaces-based cognitive radio networks. In *2021 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)* (pp. 1-6). IEEE.

Masek, P., Stusek, M., Zeman, K., Drapela, R., Ometov, A., & Hosek, J. (2019, October). Implementation of 3GPP LTE Cat-M1 technology in NS-3: System simulation and performance. In *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)* (pp. 1-7). IEEE.

Mininet. (2020).Mininet: An Instant Virtual Network on your Laptop (or other PC) - Mininet. http://mininet.org/. Accessed 22/11/2020.

Omnest. 2020. High-performance simulation for all kinds of networks. https://omnest.com/. Accessed 22/11/2020.

Patel, R. L., Pathak, M. J., & Nayak, A. J. (2018). Survey on network simulators. *International Journal of Computer Applications, 182*(21), 23-30.

Ramkumar, J., & Vadivel, R. (2018). Improved frog leap-inspired protocol (IFLIP)–for routing in cognitive radio ad hoc networks (CRAHN). *World Journal of Engineering*, *15*(2), 306-311.

Sharma, S., Mahajan, A. N., & Poonia, R. C. (2019). An Inclusive survey of Network Simulators. In *Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM)*, pp. (2366-2372), Amity University Rajasthan, Jaipur-India.

Shuaibu-Sadiq, M., & Anyasi, F. I. (2020). Analysis of radio frequency spectrum usage using cognitive radio. *Journal of Electrical, Control and Technological Research*, *1*, 1-8.

Singhal, C., & Thanikaiselvan, V. (2022, August). Cross-Layer Design for Primary User Modeling in Cognitive Radio Ad-hoc Sensor Network. In *2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC)* (pp. 577-582). IEEE.

Singh, L., & Dutta, N. (2020). Routing protocols for CRAHN: a comparative evaluation. In *Trends in Communication, Cloud and Big Data* (pp. 3-11). Springer, Singapore.

Sridevi, A., & Venkatesan, G. K. D. (2017). Certain investigations on high energy and spectral `efficient CRAHN-based spectrum aggregation. *Journal of Computational and Theoretical Nanoscience, 14*(8), 3861-3866.