BALKAN JOURNAL OF ELECTRICAL & COMPUTER ENGINEERING, Vol. 12, No. 2, June 2024
Research Article

Multi-Channel CooperativeSpectrum Sensing and Scheduling for Cognitive IoT Networks

Abdulkadir Celik

***Abstract*—This paper presents a novel multi-channel cooperative spectrum sensing and scheduling ($\mathrm{MC_2S_3}$) framework for spectrum and energy harvesting cognitive Internet of Things (IoT) networks. We address the challenge of maximizing network throughput by formulating a combinatorial problem that jointly optimizes the sensing scheduling of primary channels (PCs), the assignment of IoT devices for sensing scheduled PCs, and the clustering and allocation of IoT nodes to efficiently use discovered idle PCs; subject to spectrum utilization and collision avoidance constraints. Recognizing the inherent complexity of the underlying NP-hard mixed-integer non-linear programming (MINLP) problem, we propose a decomposition strategy that decouples the master problem into PC exploration and exploitation subproblems. In the exploration phase, we derive closed-form solutions for optimal sensing durations and detection thresholds that satisfies spectrum utilization and collision avoidance constraints, which are then used to develop a priority metric to rank PCs. The proposed PC ranking informs a sequential PC scheduling and IoT sensing assignment approach that exploits a linear bottleneck assignment (LBA) strategy, proceeding until further scheduling does not enhance network utility. For the exploitation phase, we leverage a non-orthogonal multiple access (NOMA) strategy to multiplex multiple IoT nodes on a single PC, employing an iterative linear sum assignment (LSA) method for efficient allocation to maximize utilization of idle PCs. Numerical results validate the efficacy of our proposed methodologies, reaching an accuracy of approximately 99% in the order of milliseconds, significantly outperforming time complexity of brute-force benchmarks.**

***Index Terms*—Internet of Things, Cognitive Radio Networks, Cooperative Spectrum Sensing, Energy Detection, Energy Harvesting.**

I. Introduction

THE PROLIFERATIO of the Internet of Things (IoT) has dramatically intensified the demand for efficient spectrum management strategies in cognitive radio networks (CRNs) [1], [2]. In particular, stands at the forefront of this challenge, facilitating a collaborative mechanism for IoT devices to detect and utilize underutilized spectrum bands across multiple primary channels (PCs), thereby addressing the spectrum scarcity problem [3]. This endeavor unfolds through two critical phases: *spectrum exploration* and *spectrum exploitation*, each harboring distinct complexities and optimization prospects.

During the spectrum exploration stage, the essence of multi-channel cooperative spectrum sensing scheduling ($\mathrm{MC_2S_3}$) unfolds as multiple IoT nodes concurrently sense various PCs, reporting their findings to a central decision maker, typically a secondary base station (SBS) [4]. The overarching goal is to minimize the exploration time by swiftly identifying the greatest number of potentially idle PCs with a high probability of being unoccupied. The duration of PC exploration predominantly hinges on the assignment of IoT nodes to sense scheduled PCs, ultimately determined by the node with the longest sensing time. Therefore, the overall channel exploration time is set by the slowest IoT as the SBS cannot make a decision until receiving local decisions from all IoT nodes. Moreover, requiring more IoT nodes to cooperate on sensing improves the detection accuracy of $\mathrm{MC_2S_3}$. This necessitates a judicious balance between the speed of detection and the accuracy of cooperatively identifying idle spectrum opportunities. Another subtle yet critical issue is that a PC with a high probability of being idle may incur a high sensing duration if there is no IoT nodes with good sensing quality on that PC. Thus, scheduling such a PC to be sensed may deteriorate overall spectrum discovery and available throughput for the IoT network, underscoring the importance of a joint optimization strategy for both PC scheduling and IoT sensing assignment. It is also worth noting that jointly optimizing sensing and scheduling tasks to minimize the spectrum exploration time not only maximizes the time remaining for the spectrum exploitation but also minimizes the $\mathrm{MC_2S_3}$ energy consumption and maximizes the available time for wireless energy harvesting [5].

The *spectrum exploitation* phase focuses on the effective allocation of identified idle PCs and the strategic clustering of IoT devices, aiming to maximize the utility of limited spectrum resources. In densely populated IoT networks, where the demand for vacant PCs far exceeds the supply, achieving high-quality service (QoS) becomes a formidable challenge. Here, non-orthogonal multiple access (NOMA) techniques offer a promising solution by allowing multiple users to share the same spectrum resources through distinct identification methods such as power levels or codes, thereby enhancing spectrum efficiency [6]. However, the implementation of traditional power-domain NOMA schemes involves complexities related to grant acquisition and power weight calculations, which could introduce computational and signaling overheads, especially in uplink transmissions [7]. To circumvent these challenges, our study explores a grant-free operational model, where transmit powers are predetermined, and the reception disparity is tactically managed during the clustering phase by grouping IoT nodes with distinct channel characteristics into the same cluster.

Addressing the intertwined challenges of efficient spectrum sensing, optimal scheduling, and effective exploitation of discovered spectrum often falls within the real of combinatorial problems, whose computational complexity is prohibitive even

Abdulkadir Celik Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division, King Abdullah University of Science and Technology (KAUST), Thuwal, KSA 23955-6900.
(e-mail: abdulkadir.celik@kaust.edu.sa)
https://orcid.org/ 0000-0001-9007-9979

Manuscript received Feb 17, 2024; revised May 11, 2024.
DOI: 10.17694/bajece.1438843

for moderate size of networks. Accordingly, this paper contributes to the advancement of cognitive IoT networks through developing fast yet efficient solutions for enhancing network throughput, energy efficiency, and overall performance in the face of growing spectrum demands.

A. Related Works

Cooperative Spectrum Sensing (CSS) has emerged as a pivotal technology to address the challenges of spectrum scarcity in cognitive IoT networks, focusing on optimizing spectral and energy efficiency while ensuring minimal interference with primary network (PN). This subsection reviews recent advancements in CSS, highlighting various strategies aimed at optimizing sensing operations, energy consumption, and network throughput.

Heuristic Solutions and Optimization Techniques have been proposed to address the challenges in CSS, aiming at improving energy efficiency and network utility. Chauhan et al. introduced a CSS scheme using a greedy-based heuristic solution to enhance network utility and energy efficiency in heterogeneous multi-channel CRNs. This approach adaptively optimizes the sensing duration based on the received signal-to-noise ratio (SNR) [8]. On the other hand, Cao et al. proposed a PSO-based CSS strategy, focusing on system throughput and energy efficiency optimization. This strategy balances sensing performance and energy consumption effectively [9]. Likewise, Kaschel et al. explored energy-efficient solutions for dynamic CRNs, proposing a framework to predict energy consumption in CSS tasks, minimizing energy consumption with low-complexity algorithms [10]. Moreover, a green approach with an emphasis on high energy efficiency through spectrum sensing and power allocation optimization is proposed in [11], demonstrating significant energy efficient improvements. Finally, Al-Kofahi et al. proposed Spectrum Sensing as a Service (SSaS) approach to optimize sensor selection to minimize energy consumption with a sub-optimal greedy algorithm [12]. In [13] and [14], statistical and systematic approaches are offered to evaluate network lifetime and identifying correlations between spectrum sensing, clustering algorithms, and energy-harvesting technology, respectively. Finally, Wu et al. explored the impact of user characteristics on CSS performance in mobile CRNs, proposing a dynamic detection scheme to optimize performance [15].

Machine Learning (ML) Approaches have also been utilized to enhance sensing accuracy and efficiency in CSS. Ning et al. presented a reinforcement learning-enabled CSS scheme for CRNs, optimizing the scanning order of channels and the selection of cooperative partners to improve detection efficiency [16]. Shi et al. employed multiple ML algorithms, including unsupervised and supervised learning techniques, to tackle the challenges of complex sensing models in CSS [17]. Moreover, Ahmed et al. integrated CR technology with IoT, employing SVM for spectrum sensing and allocation, demonstrating high reliability in frequency band allocation [18].

Game-Theoretic Models have been applied to extend network lifetime and improve sensing quality and cooperation among sensors. Bagheri et al. proposed a novel game-theoretic sensor selection algorithm for MCSS in WSNs, focusing on extending the network lifetime while ensuring the quality of sensing [19]. Rajendran et al. introduced a distributed coalition formation game with a genetic algorithm for optimal coalition head selection in CSS among SUs, enhancing sensing accuracy and minimizing energy consumption [20].

Cooperative Prediction and Sensing Schemes have also been studied to minimize energy consumption and maintain spectral efficiency. Chauhan et al. proposed a cooperative spectrum prediction-driven sensing scheme employing long short-term memory networks for local spectrum prediction, aiming at reducing energy consumption [21]. Gharib et al. proposed a heterogeneous multi-band multi-user CSS scheme for IoT in CR networks, improving detection performance and CRN throughput through optimized leader selection and cooperative sensing [22].

B. Main Contributions

The main contributions of this paper can be summarized as follows:

- We consider MC_2S_3 for spectrum and energy harvesting cognitive IoT networks. An optimal problem formulation is provided to maximize overall IoT network throughput through joint optimization of sensing scheduling of PCs, assignment of IoTs to sense scheduled PCs, and clustering and allocating IoT nodes to exploit discovered idle PCs; subject to collision and spectrum utilization constraints of CSS. The proposed system model considers practical aspects including channel reporting errors as well as channel switching and reporting latency.
- Since the optimal problem falls into the class of mixed-integer non-linear programming (MINLP) problem, which is known to be NP-hard, we propose a decomposition approach to decouple it into PC exploration and exploitation sub-problems. For the PC exploration sub-problem, we derive optimal sensing durations and detection thresholds in closed-form to inform a PC ranking mechanism based on a newly proposed priority metric. PCs are scheduled sequentially according to their priority, and IoT nodes are assigned using a linear bottleneck assignment (LBA) strategy to minimize exploration time. This process continues until additional PC scheduling does not further benefit the network's utility.
- For the exploitation phase, upon identifying a set of idle PCs, we organize IoT nodes into clusters and allocate them to discovered idle PCs. Given the potential disparity between the number of IoT nodes and available idle PCs, we employ a non-orthogonal multiple access (NOMA) strategy. This allows for the multiplexing of multiple IoT nodes on a single PC. Allocation is done using an iterative linear sum assignment (LSA) method, adding IoT nodes to an idle PC until all nodes are allocated, thus maximizing the utilization of available spectrum resources.
- Our numerical analysis demonstrates that the proposed methodologies achieve approximately 99% accuracy in

milliseconds, a significant improvement over traditional brute-force approaches that exhaustively search all possible combinations to identify optimal sensing schedules and IoT node clustering.

C. Paper Organization and Notations

The remainder of paper is organized as follows: Section II delineates the system model. Section III provide details of MC$_2$S$_3$, multi-modal energy harvesting, and NOMA-based IoT data transmission. Section IV formulates the optimal problem and outline the proposed solution methodology. Section V presents proposed decomposed approach and explain its algorithmic implementation. Section VI provide numerical results. Finally, Section VII papers with a few remarks.

Notations: Throughout the paper, sets and their cardinality are denoted with calligraphic and regular uppercase letters (e.g., $|\mathcal{A}| = A$), respectively. Vectors and matrices are represented in lowercase and uppercase boldfaces (e.g., $\boldsymbol{a}$ and $\boldsymbol{A}$), respectively. The ith member of a vector and set is denoted by $\boldsymbol{a}[i]$ and $\mathcal{A}\{i\}$, respectively. Likewise, matrix $\boldsymbol{A}$'s entry on i^{th} row and j^{th} column is denoted by $\boldsymbol{A}[i, j]$. Subscripts m, n, and i are used for indexing PCs, IoT nodes, and idle PCs, respectively.

II. System Model

This section introduces system models for PN, secondary network (SN), and underlying models for sensing, reporting, and communication channels.

A. Primary Network

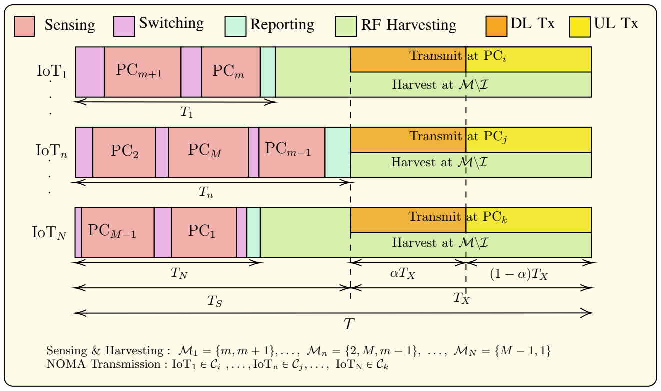
We consider a PN which is licensed to operate on M downlink PCs, whose index set is denoted by $\mathcal{M}$, each with a bandwidth of W Hz. The PN consists of a single primary base station that communicates with its associated primary users over dedicated PCs. Primary transmissions operate on a synchronous communication protocol with time-slot duration of T. The actual occupancy state of a PC is denoted by $O_m \in \{0, 1\}$, with $O_m = 1$ if PC$_m$ is busy, $O_m = 0$ otherwise. The binary hypotheses correspond to the idle and busy PC states are denoted as $\mathcal{H}_m^0$ and $\mathcal{H}_m^1$, respectively. The PC traffic is modeled as an independent and identically distributed (i.i.d.) random process such that π_0^m and π_1^m represent the apriori probability of being idle and busy for PCs, respectively. We assume that these apriori probabilities are obtained over long-term observations and known to the SN.

B. Secondary Network

The SN comprises of N time synchronous energy and spectrum harvesting IoT devices, whose index set is denoted by $\mathcal{N}$. The SN is orchestrated by a SBS that also follows a time-slotted operation synchronized to the PN. As shown in Fig. 1, each time-slot of duration T is split into two stages: multi-channel *exploration* and *exploitation*. In the multi-channel exploration stage, the IoT devices collaboratively searches for the available bands by means of a MC$_2$S$_3$ policy, wherein the SBS assigns a set of PCs to each IoT device for sequential sensing, which requires IoT devices to switch channel and adjust their operating frequency to the assigned channels. Accordingly, the binary sensing assignment matrix is denoted by $\mathbf{X} \in \{0, 1\}^{M \times N}$, whose elements are given by

$$x_m^n = \begin{cases} 1 & , \text{ if IoT}_n \text{ is assigned to sense PC}_\text{m} \\ 0 & , \text{ otherwise} \end{cases}, \quad (1)$$

$\forall m \in \mathcal{M}, \forall n \in \mathcal{N}$. Hence, the set of PCs assigned to IoT$_n$ for sequential sensing is represented by $\mathcal{M}_n = \{m | x_m^n = 1, \forall m\}$, whose cardinality is $M_n = \sum_n x_m^n$. Likewise, the cluster of IoT nodes that cooperatively sense PC$_m$ is denoted by $\mathcal{N}_m = \{n \mid x_m^n = 1, \forall n\}$, whose cardinality is $N_m = \sum_m x_m^n$. To reach a better global decision, MC$_2$S$_3$ typically requires PCs to be sensed by a certain number of IoTs, i.e., $N_m \geqslant \bar{N}$.



🟥 Sensing
🟪 Switching
🟦 Reporting
🟩 RF Harvesting
🟧 DL Tx
🟨 UL Tx
IoT$_1$
PC$_{m+1}$
PC$_m$
Transmit at PC$_i$
Harvest at $\mathcal{M}\backslash\mathcal{I}$
T_1
IoT$_n$
PC$_2$
PC$_M$
PC$_{m-1}$
Transmit at PC$_j$
Harvest at $\mathcal{M}\backslash\mathcal{I}$
T_n
IoT$_N$
PC$_{M-1}$
PC$_1$
Transmit at PC$_k$
Harvest at $\mathcal{M}\backslash\mathcal{I}$
T_N
αT_X
$(1-\alpha)T_X$
T_S
T_X
T
Sensing & Harvesting : $\mathcal{M}_1 = \{m, m+1\}, \ldots, \mathcal{M}_n = \{2, M, m-1\}, \ldots, \mathcal{M}_N = \{M-1, 1\}$
NOMA Transmission : IoT$_1 \in \mathcal{C}_i$, . . . , IoT$_\text{n} \in \mathcal{C}_j$, . . . , IoT$_\text{N} \in \mathcal{C}_k$

Figure 1: Time-slotted operation of the secondary network.

However, sensing all PCs may require IoTs to search over a large number of PCs and results in limited exploration time, especially when $\bar{N} \gg \frac{N}{M}$. At this point, PC sensing scheduling becomes paramount, which is represented by $\mathbf{y} \in \{0, 1\}^M$ with $y_m = 1$ if PC$_m$ is scheduled for sensing, $y_m = 0$ otherwise. Following the completion of sequential sensing, IoTs report their local decisions on channel occupancy states to the SBS that makes global decisions and broadcasts the list of I idle PCs, $\mathcal{I}$. Denoting overall sensing, switching, and reporting duration for IoT$_n$ by T_n^s, the overall multi-channel search time is determined by the IoT taking the longest channel search time (e.g., IoT$_n$ in Fig. 1) as the SBS needs to receive all reports to reach a conclusion.

In the multi-channel exploitation stage, IoT devices are scheduled to transmit data over the residual time after the multi-channel exploration. Denoting the number of sample matrix as $\boldsymbol{\mathcal{S}} \in \mathbb{N}^{M \times N}$, the overall transmission duration is given by $T_X(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\mathcal{S}}) = T - T_S(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{S})$, during which IoT devices exploit the discovered idle PCs, $\mathcal{I}$, and busy PCs, $\mathcal{M}\backslash\mathcal{I}$, for data transmission and RF energy harvesting, respectively. During the exploitation stage, the SN operates in a NOMA fashion, wherein IoT devices are clustered to exploit idle PCs. Accordingly, the binary PC allocation and IoT clustering matrix is denoted by $\mathbf{Z} \in \{0, 1\}^{M \times N}$, whose elements are given by

$$z_m^n = \begin{cases} 1 & , \text{ if IoT}_n \text{ transmits at PC}_\text{m} \\ 0 & , \text{ otherwise} \end{cases}, \forall m \in \mathcal{M}, \forall n \in \mathcal{N}, \quad (2)$$

based on which, the IoT cluster transmitting at PC$_m$ is denoted by $\mathcal{C}_m = \{n \mid z_m^n = 1, \forall n\}$, $\forall m$, $|\mathcal{C}_m| = C_m \leqslant \lceil \frac{N}{M} \rceil$.

C. Channel Model

The sensing channel gain matrix between the PBS and IoT devices is denoted by $\mathbf{H} \in \mathbb{R}^{\mathbf{M}\times\mathbf{N}}$, whose elements $h_m^n, \forall m \in [1, M], \forall n \in [1, N]$, are given by

$$h_m^n = G_m G_n K_m \left[\frac{d_0}{d_m^n}\right]^\theta 10^{\psi_m^n/10} \tag{3}$$

where G_t is the transmitter gain of the PBS, G_n is the receiver gain of IoT$_n$, $K_m = \frac{c}{4\pi d_0 f_m}$, c is the speed of light, f_m is the carrier frequency of the PC$_m$, d_0 is a reference distance, d_m^n is the distance between the PBS and IoT$_n$, θ is the path loss exponent, $10^{\psi_m^n/10}$ is the log-normal shadowing, and ψ_m^n is a normal random variable representing the variation in received power with a standard deviation of σ_m^n, i.e., $\psi_m^n \sim \mathcal{N}(0, \sigma_m^n)$. The reporting between SBS and IoT$_n$ occurs on a dedicated control channel, which is denoted by g_n^s and follows the composite channel gain similar to (3). Likewise, the transmission channel gain matrix between the SBS and IoT devices is denoted by $\tilde{\mathbf{H}} \in \mathbb{R}^{\mathbf{M}\times\mathbf{N}}$, whose elements, $\tilde{h}_m^n$, can also be obtained following the same model presented in (3). We assume all channels to be quasi-static which experience flat fading during the sensing, reporting, and transmission phases.

III. Multi-Channel Cooperative Spectrum Sensing and Energy Harvesting

In this section, we delineate the spectrum exploitation and exploration stages by providing the technical details of multi-channel cooperative spectrum sensing, multi-modal energy harvesting, and IoT data transmission.

A. Spectrum Exploration Stage

Since our main focus in this paper is the scheduling aspects of the CSS, we prefer *energy detection* as a generic method because it does not require any apriori information regarding the PN's physical layer features. Without loss of generality, let us focus on spectrum sensing of PC$_m$ with IoT devices belong to $\mathcal{N}_m$. First, IoT$_n \in \mathcal{N}_m$ measures the energy of received signal on PC$_m \in \mathcal{M}$ for a number of samples, S_m^n, and compares it with a detection threshold, ε_m^n, to reach a local decision on the idle and busy states of PC$_m$.

The overall channel search time for IoT$_n$ comprises of three phases: 1) Energy detection duration, T_n^{ed}; 2) channel switching time T_n^{sw}; and 3) reporting time, T_n^r. Denoting the sampling frequency and interval by W and $\tau_s = 1/W$, the number of samples determines the energy detection duration for IoT$_n, \forall n \in \mathcal{N}_m$ on PB$_m$ as $\tau_m^n = S_m^n \tau_s$. We assume that the channel switching time obeys triangularity and linearity properties [23], i.e., $\beta|f^c_{\mathcal{M}_n\{i\}} - f^c_{\mathcal{M}_n\{j\}}|$, $i \neq j, 1 \leqslant i, j \leqslant M_n$, where β is a switching factor that depends on hardware parameters and power consumption and $f^c_{\mathcal{M}_n\{i\}} - f^c_{\mathcal{M}_n\{j\}}$ is the central frequency of the PC$_i$/PC$_j$, $(i, j) \in \mathcal{M}_n$. Accordingly, the channel search time for IoT$_n$ can be expressed as

$$T_n^s(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{S}) = T_n^{ed} + T_n^{sw} + T_n^r \tag{4}$$

where

$$T_n^{ed} = \tau_s \sum_m S_m^n x_m^n y_m, \tag{5}$$

$$T_n^{sw} = \beta \left| f^c_{\mathcal{M}_n\{i\}} - f^c_0 \right| + \sum_{i=1}^{M_n} \beta \left| f^c_{\mathcal{M}_n\{i\}} - f^c_{\mathcal{M}_n\{i-1\}} \right|, \tag{6}$$

$$T_n^r = \tau_r \sum_m x_m^n y_m, \text{ and} \tag{7}$$

f_0^c is the central frequency of the last PC sensed in the previous time slot. Following from (4), the overall multi-channel exploration time is given by

$$T_S(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{S}) = \max_{n \in \mathcal{N}} \{T_n^s(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{S})\}. \tag{8}$$

For a normalized noise variance and large enough S_m^n, the probability of false alarm, $P^f_{m,n} \triangleq \mathcal{P}[\mathcal{H}^1_m | \mathcal{H}^0_m]$, and the probability of detection, $P^d_{m,n}(S_m^n, \varepsilon_m^n) \triangleq \mathcal{P}[\mathcal{H}^1_m | \mathcal{H}^1_m]$, are respectively given by [24]

$$P^f_{m,n}(S_m^n, \varepsilon_m^n) = \mathcal{Q}\left[(\varepsilon_m^n - 1)\sqrt{S_m^n}\right], \tag{9}$$

$$P^d_{m,n}(S_m^n, \varepsilon_m^n)(t) = \mathcal{Q}\left[(\varepsilon_m^n - \gamma_m^n - 1)\sqrt{\frac{S_m^n}{2\gamma_m^n + 1}}\right], \tag{10}$$

where γ_m^n is the received sensing signal-to-noise-ratio (SNR) at IoT$_n$ on the PC$_m$ and $\mathcal{Q}(x) = \frac{1}{\sqrt{2\pi}} \int_x^{+\infty} e^{-y^2/2} dy$ denote the right-tail probability of a normalized Gaussian distribution.

Notice in (9) and (10) that both probabilities are jointly determined by the detection threshold and number of samples, i.e., sensing duration. Since false alarm and detection probabilities are key metrics to measure sensing accuracy, a joint optimization of these variables is necessary to reach predetermined thresholds. After the sequential local sensing process, the IoT$_n$ sends its binary result $u_m^n \in \{0, 1\}$ to the SBS over a binary symmetric channel (BSC), which is received by the SBS as $\tilde{u}_m^n$. The error probability of the BSC is defined as $p_m^n = \mathcal{P}[\tilde{u}_m^n = 0 | u_m^n = 1] = \mathcal{P}[\tilde{u}_m^n = 1 | u_m^n = 0]$. Accordingly, the SBS receives the local false alarm and detection probabilities as

$$\tilde{P}^f_{m,n}(S_m^n, \varepsilon_m^n) = p_m^n \left[1 - P^f_{m,n}\right] + \left[1 - p_m^n\right] P^f_{m,n} \tag{11}$$

$$\tilde{P}^d_{m,n}(S_m^n, \varepsilon_m^n) = p_m^n \left[1 - P^d_{m,n}\right] + \left[1 - p_m^n\right] P^d_{m,n} \tag{12}$$

The SBS collects $\tilde{u}_m^n$, $\forall m, \forall n$, and makes the global decision using the k-out-of-N decision fusion test

$$\mathcal{K}_m = \sum_{n \in \mathcal{N}_m} \tilde{u}_m^n x_m^n \underset{\mathcal{H}_m^0}{\overset{\mathcal{H}_m^1}{\gtrless}} \kappa_m, \tag{13}$$

where $1 \leqslant \kappa_m \leqslant N_m$ is the voting rule. For instance, $\kappa_m = 1$, $\kappa_m = N_m$, and $\kappa_m = \lceil \frac{N_m}{2} \rceil$ are referred to as OR, AND, and majority voting rules, respectively. $\mathcal{K}_m$ follows Binomial distribution if $\tilde{u}_m^n$'s are i.i.d, which naturally requires all reported local detection and false alarm probabilities to be identical If the i.i.d condition is satisfied, the global false alarm $Q_m^f(S_m^n, \varepsilon_m^n) = \mathcal{P}\left[\mathcal{K}_m \geqslant \kappa_m | \mathcal{H}_m^0\right]$ and detection $Q_m^d(S_m^n, \varepsilon_m^n) = \mathcal{P}\left[\mathcal{K}_m \geqslant \kappa_m | \mathcal{H}_m^1\right]$ probabilities can be expressed by using the probability mass function (pmf) of the Binomial variable as follows

$$Q_m^f(S_m^n,\varepsilon_m^n) = \mathcal{P}\left[\mathcal{K}_m \geqslant \kappa_m|\mathcal{H}_m^0\right] = \tag{14}$$

$$\sum_{i=\kappa_m}^{N_m}\binom{N_m}{i}\left(\tilde{P}_m^f(S_m^n,\varepsilon_m^n)\right)^i\left(1-\tilde{P}_m^f(S_m^n,\varepsilon_m^n)\right)^{N_m-i},$$

$$Q_m^d(S_m^n,\varepsilon_m^n) = \mathcal{P}\left[\mathcal{K}_m \geqslant \kappa_m|\mathcal{H}_m^1\right] = \tag{15}$$

$$\sum_{i=\kappa_m}^{N_m}\binom{N_m}{i}\left(\tilde{P}_m^d(S_m^n,\varepsilon_m^n)\right)^i\left(1-\tilde{P}_m^d(S_m^n,\varepsilon_m^n)\right)^{N_m-i}.$$

Indeed, opportunistic spectrum access schemes typically require spectrum utilization constraint, i.e., $Q_m^f(S_m^n,\varepsilon_m^n) \leqslant \bar{Q}_f$, and collision avoidance constraint, i.e., $Q_m^d(S_m^n,\varepsilon_m^n) \geqslant \bar{Q}_d$, are satisfied at the same time. For instance, predetermined thresholds of $\bar{Q}_f = 0.01$ and $\bar{Q}_d = 0.99$ ensure that idle and busy channels are utilized and correctly detected 99% of the time, respectively. Since both $Q_m^f(S_m^n,\varepsilon_m^n)$ and $Q_m^d(S_m^n,\varepsilon_m^n)$ are determined by the cluster size, N_m, the global false alarm and detection probability constraints require reported local false alarm and detection probabilities to respectively satisfy

$$\tilde{P}_m^{f,\star}(N_m) \leqslant \begin{cases} 1-\sqrt[N_m]{1-\bar{Q}_f} & ,\kappa_m = 1 \\ \sqrt[N_m]{\bar{Q}_f} & ,\kappa_m = N_m \end{cases}, \tag{16}$$

$$\tilde{P}_m^{d,\star}(N_m) \geqslant \begin{cases} 1-\sqrt[N_m]{1-\bar{Q}_d} & ,\kappa_m = 1 \\ \sqrt[N_m]{\bar{Q}_d} & ,\kappa_m = N_m \end{cases}, \tag{17}$$

which can be solved numerically for other voting rules, $1 < \kappa_m < N_m$. In order to satisfy both i.i.d. conditions and global constraints, IoT$_n \in \mathcal{N}_m$ needs to adjust its local false alarm and detection probabilities according to the erroneous BSC channel conditions as

$$P_{m,n}^{f^\star}(N_m) \leqslant \frac{\tilde{P}_m^{f,\star(N_m)} - p_m^n}{1-2p_m^n}, \tag{18}$$

$$P_{m,n}^{d^\star}(N_m) \geqslant \frac{\tilde{P}_m^{d,\star}(N_m) - p_m^n}{1-2p_m^n}, \tag{19}$$

which follows from (11) and (12), respectively.

B. Multi-Modal Energy Harvesting

In addition to scavenging energy from other ambient sources, e.g., flexible solar patches, IoT devices are also assumed to harvest RF energy from busy PCs. The energy arrival rate for RF and non-RF energy modalities are denoted bu $\chi_{m,n}^{RF}$ and χ_{sol}^n [J/s] which depends on received power on PC$_m$ and light intensity along with hardware specifications (e.g., energy conversion and storing efficiency, etc.), respectively. Indeed, energy detectors can be designed to support both spectrum sensing and RF energy harvesting on the same platform through power splitting or time switching [25]. Following the spectrum searching, each IoT switch themselves to the PC with the highest reception power, yielding the total harvested energy of $E_h^n = \chi_{sol}^n T + \text{argmax}_m\{\chi_{m,n}^{RF}(T - T_S(\boldsymbol{X},\boldsymbol{y},\boldsymbol{S}))\}$, where the first and second terms are harvested energy from RF and solar ambient sources, respectively. Since the non-RF energy harvesting is independent of MC$_2$S$_3$, we will focus our attention on the RF energy harvesting. In case of time switching mode, the harvested RF energy can only be maximized by minimizing the channel exploration phase, which will be the main focus of the formulated problem and developed solution methodology in the sequel.

C. IoT Data Transmission

The efficient use of discovered idle spectrum is of utmost importance, especially for dense SNs trying to utilize limited vacant PCs of a PN exhibiting frequent traffic characteristics, i.e., $I \ll N$. In such scenarios, exploitation of OMA to avoid interference may not provide required QoS demands for numerous IoT nodes that merely depends on the free spectrum. At this point, NOMA can offer high spectrum efficiency by multiplexing several users in the same time/frequency resource by distinguishing each with discrete codes, power levels, etc. For instance, NOMA allows SN to serve at most $C = \lceil \frac{I}{N} \rceil$ IoT nodes at each vacant PC. The traditional power-domain NOMA schemes requires grant acquisition and optimal power weight calculations, which may incur computational complexity and signaling overhead, especially in the UL transmission. Therefore, we consider a grant-free operation where transmit powers are set a predetermined level while power reception disparity is implicitly implemented during the clustering phase by admitting IoT nodes with distinct channels into the same cluster.

Without loss of generality, let us elucidate how NOMA can facilitate data transmission for the ith IoT cluster, $\mathcal{C}_i$, by dividing exploration duration into $\alpha T_X(\boldsymbol{X},\boldsymbol{y},\boldsymbol{\mathcal{S}})$ and $(1-\alpha)T_X(\boldsymbol{X},\boldsymbol{y},\boldsymbol{\mathcal{S}})$ for DL and UL traffic, respectively. $\alpha \in [0,1]$ is a design parameter and can be set based on temporal traffic characteristics of the SN. Due to the quasi-static channel assumption and time-division multiplexing during the same time slot, the DL/UL common channel gains of $\mathcal{C}_i$ sorted in descending order can be denoted by $\tilde{\boldsymbol{h}}_{\boldsymbol{i}} = [\tilde{h}_i^1,\ldots,\tilde{h}_i^j,\ldots,\tilde{h}_i^{C_i}]$. Hence, the received DL and UL signals for IoT$_j, j \in \mathcal{C}_i$, are respectively given by

$$\check{r}_i^j = \left(\sum_{j\in\mathcal{C}_i}\sqrt{P_s^i}\check{\omega}_i^j\check{s}_j\right)\tilde{h}_i^j + n, \tag{20}$$

$$\hat{r}_i^j = \sum_{j\in\mathcal{C}_i}\sqrt{P_j}\tilde{h}_i^j\hat{\omega}_i^j\hat{s}_j + n, \tag{21}$$

where $P_s^i \triangleq P_{\max}^s/I$ is the maximum DL transmission power at PC$_i$ such that the maximum transmit power of the SBS, $P_{\max}^s$, is equally shared by clusters and members within cluster, P_j is the maximum transmit power of IoT$_j$, $\check{\omega}_i^j/\hat{\omega}_i^j$ is the DL/UL power allocation of IoT$_j$, $\check{s}_j/\hat{s}_j$ is the DL/UL message for IoT$_j$, and $n \sim \mathcal{N}(0,\sigma^2)$ is the additive white Gaussian thermal noise, $\sigma^2 = N_0W$ is the thermal noise power, and N_0 is the thermal noise power spectral density. It is worth noting that cluster members' channel gains play the crucial role of power weights to ensure power reception disparity, which primarily determines the spectral efficiency of NOMA schemes. At the receiver side, a successive interference cancellation (SIC) receiver iteratively decodes the messages in the order of reception power and removes decoded messages

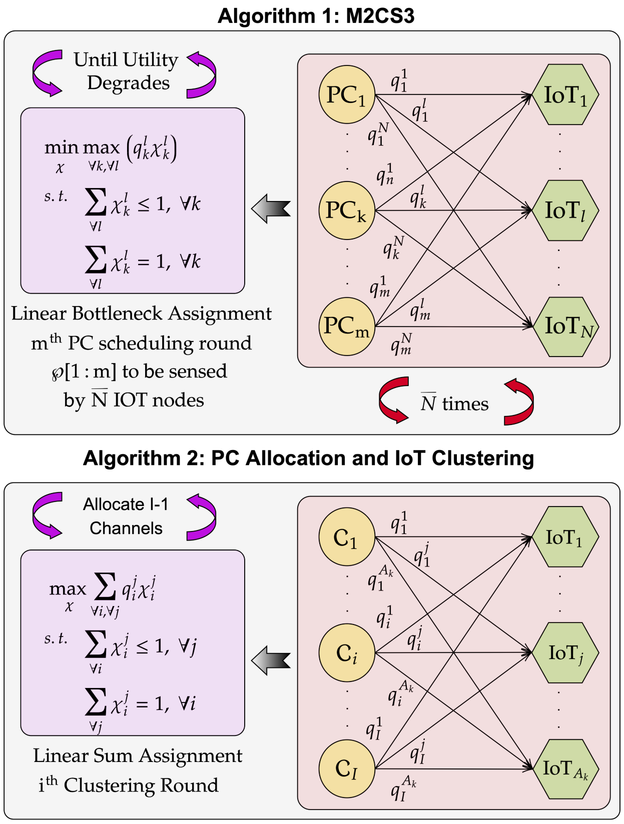

Algorithm 1: M2CS3
Until Utility Degrades
$\min_{\chi} \max_{\forall k,\forall l} (q_k^l \chi_k^l)$
s.t. $\sum_{\forall l} \chi_k^l \le 1, \forall k$
$\sum_{\forall l} \chi_k^l = 1, \forall k$
Linear Bottleneck Assignment
mth PC scheduling round
$\wp[1:m]$ to be sensed
by $\overline{N}$ IOT nodes
PC$_1$
PC$_k$
PC$_m$
q_1^1
q_1^l
q_1^N
q_n^1
q_k^l
q_k^N
q_m^1
q_m^l
q_m^N
IoT$_1$
IoT$_l$
IoT$_N$
$\overline{N}$ times
Algorithm 2: PC Allocation and IoT Clustering
Allocate I-1 Channels
$\max_{\chi} \sum_{\forall i,\forall j} q_i^j \chi_i^j$
s.t. $\sum_{\forall i} \chi_i^j \le 1, \forall j$
$\sum_{\forall j} \chi_i^j = 1, \forall i$
Linear Sum Assignment
ith Clustering Round
C$_1$
C$_i$
C$_I$
q_1^1
q_1^j
$q_1^{A_k}$
q_i^1
q_i^j
$q_i^{A_k}$
q_I^1
q_I^j
$q_I^{A_k}$
IoT$_1$
IoT$_j$
IoT$_{A_k}$

Figure 2: Depiction of Alg. 1 (up) and Alg. 2 (down).

to reduce interference at each time step, leading to following DL and UL SINRs

$$\breve{\gamma}_i^j = \frac{|\tilde{h}_i^j|^2 \breve{\omega}_i^j}{\epsilon \sum_{k=1}^{j-1} |\tilde{h}_i^j|^2 \breve{\omega}_i^k + \sum_{l=j+1}^{C_i} |\tilde{h}_i^j|^2 \breve{\omega}_i^l + \sigma^2/P_s^i}, \tag{22}$$

$$\widehat{\gamma}_i^j = \frac{|\tilde{h}_i^j|^2 \widehat{\omega}_i^j}{\sum_{k=1}^{j-1} |\tilde{h}_i^k|^2 \widehat{\omega}_i^k + \epsilon \sum_{l=j+1}^{C_i} |\tilde{h}_i^l|^2 \widehat{\omega}_i^l + \sigma^2/P_j}. \tag{23}$$

Accordingly, the combined bidirectional data rate of IoT$_j$ on PC$_i$ is given by

$$R_i^j = \left[\alpha \log_2(1+\widehat{\gamma}_i^j) + (1-\alpha)\log_2(1+\breve{\gamma}_i^j)\right], j \in \mathcal{C}_i \tag{24}$$

which can also be denoted as $R_i^j(\boldsymbol{Z})$ by setting $\boldsymbol{Z}[i,\mathcal{C}_i] = 1$ and multiplying ω_a^b terms with $\boldsymbol{Z}[a,b]$ elements in (22).

IV. Problem Formulation and Solution Methodology

The throughput maximization of cognitive IoT networks depends on two main criteria:

1) Discovery of the highest possible number of idle PCs to maximize the spectrum available for IoT nodes. At this point, joint optimization of sensing scheduling variables (i.e., $\boldsymbol{X}, \boldsymbol{y}$), are of paramount importance to schedule PCs with a high probability of being idle and low sensing duration.
2) The optimal utilization of discovered idle PCs through effective PC allocation and IoT clustering.

Accordingly, overall throughput achievable by the SN can be expressed as in (25), where the first and second parts are related to spectrum exploration and exploitation problems. Following from (25), optimal formulation of the spectrum exploration and exploitation problem can be obtained as follows

$$\begin{array}{lll}
\mathbf{P_o}: & \max\limits_{\substack{\boldsymbol{X},\boldsymbol{y},\boldsymbol{Z}\\ \boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{E}}}} & R(\boldsymbol{X},\boldsymbol{y},\boldsymbol{Z},\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{E}}) \\
C_o^1: & \text{s.t.} & \bar{Q}_d \leqslant Q_m^d(S_m^n, \varepsilon_m^n),\ \forall m \in \{m|\ y_m = 1\} \\
C_o^2: & & Q_m^f(S_m^n, \varepsilon_m^n) \leqslant \bar{Q}_f,\ \forall m \in \{m|\ y_m = 1\} \\
C_o^3: & & x_m^n \leqslant y_m,\ \forall m; \forall n \\
C_o^4: & & \bar{N} y_m \leqslant \sum\limits_{n \in \mathcal{N}} x_m^n,\ \forall m \\
C_o^5: & & \overline{S} \leqslant S_m^n \leqslant T/T_s,\ \forall m; \forall n \\
C_o^6: & & 0 \leqslant T - T_S(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\mathcal{S}}), \\
C_o^7: & & \sum\limits_{m \in \mathcal{M}} z_m^n \leqslant 1,\ \forall n \\
C_o^8: & & \sum\limits_{n \in \mathcal{N}} z_m^n \leqslant \left\lceil \frac{N}{M} \right\rceil,\ \forall m \\
C_o^9: & & \boldsymbol{X} \in \{0,1\}^{M \times N}, \boldsymbol{y} \in \{0,1\}^{M}, \\
& & \boldsymbol{Z} \in \{0,1\}^{M \times N}, \boldsymbol{\mathcal{S}} \in \mathbb{N^+}^{M \times N}, \boldsymbol{\mathcal{E}} \in \mathbb{R}^{B \times M}
\end{array},$$

where $\boldsymbol{\mathcal{S}}$ is a positive real matrix with entries S_m^n, and $\boldsymbol{\mathcal{E}} \in \mathbb{R}^{B \times M}$ is a real matrix with entries ε_m^n. In $\mathbf{P_o}$, C_o^1 and C_o^2 are the collision and spectrum utilization constraints, respectively. C_o^3 ensures that if the PC$_m$ is not scheduled for sensing then no IoT can be assigned to sense the PC$_m$. C_o^4 requires that a scheduled PC must be sensed by the cooperation of at least $\bar{N}$ IoT nodes. C_o^5 sets the upper bound for number of samples to T/T_s which is the maximum number of samples possible within a timeslot duration and sets the lower bound of $\overline{S}$ on the required number of samples to invoke the central limit theorem to ensure that assumptions are hold for (9) and (10). C_o^6 limits the spectrum exploration time by time slot duration T. C_o^7 assures that an IoT can be admitted to at most one cluster, i.e., can utilize at most one PC. On the other hand, C_o^8 limits cluster size to $\left\lceil \frac{N}{M} \right\rceil$ as explained in the previous section. Finally, C_o^9 defines the variable domains.

$\mathbf{P_o}$ is an MINLP problem which incurs impractical computational complexity even for moderate size of networks. Therefore, developing fast yet high-performance sub-optimal solutions is necessary to achieve satisfactory results for practical purposes. It is obvious from (25) that spectrum exploitation and exploration can be decoupled; hence, $\mathbf{P_o}$ can be decomposed into two problems: 1) MC$_2$S$_3$ and 2) clustering of IoT nodes and allocation of discovered idle PCs.

In the following subsections, we will provide solution methodology for these two subproblems. In the former subsection, optimal sensing durations and detection thresholds are derived in closed-form for a given set of IoT nodes. Then, we propose a PC ranking approach based on a priority metric, which is the best achievable utility if PC$_m$ is assigned to the best $\bar{N}$ IoT nodes with the lowest sensing duration. As depicted in Fig. 2, we iteratively schedule the first m PCs to be sensed in the m^{th} iteration, where we assign $\bar{N}$ IoT nodes to sense m PCs iteratively. If there is an improvement in terms of total utility, we proceed with scheduling $m+1$ best PCs to be sensed, otherwise we terminate. In the latter

$$R(\boldsymbol{X},\boldsymbol{y},\boldsymbol{Z},\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{E}}) = \underbrace{\overbrace{\sum_{\forall m\in\mathcal{M}} y_m \pi_m^0 (1-Q_m^f(S_m^n,\varepsilon_m^n))T_X(\boldsymbol{X},\boldsymbol{y},\boldsymbol{\mathcal{S}})}^{\text{Utility Function (Average Total Achievable Throughput)}}}_{\text{Spectrum Exploration}} \underbrace{\overbrace{\left(\sum_{\forall n\in\mathcal{N}} R_m^n(\boldsymbol{Z})\right)}^{\text{Total Achievable IoT Rate}}}_{\text{Spectrum Exploitation}}, \tag{25}$$

subsection, we cluster N IoT nodes and allocate each cluster to previously discovered I idle PCs, which is also done through $\lceil N/I \rceil$ consecutive bi-partite matching as illustrated in Fig. 2.

V. Decomposed Spectrum Exploration and Exploitation

By ignoring the second part of (25) and excluding the irrelevant constraints related to spectrum exploitation, the MC$_2$S$_3$ can be directly obtained from $\mathbf{P_o}$ as

$$\begin{aligned}
\mathbf{P_1}:& \max_{\boldsymbol{X},\boldsymbol{y},\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{E}}} \sum_{\forall m\in\mathcal{M}} y_m \pi_m^0 (1-Q_m^f(S_m^n,\varepsilon_m^n))T_X(\boldsymbol{X},\boldsymbol{y},\boldsymbol{\mathcal{S}})\\
C_o^1:& \quad \text{s.t.} \quad \bar{Q}_d \leqslant Q_m^d(S_m^n,\varepsilon_m^n),\ \forall m \in \{m|\ y_m = 1\}\\
C_o^2:& \quad Q_m^f(S_m^n,\varepsilon_m^n) \leqslant \bar{Q}_f,\ \forall m \in \{m|\ y_m = 1\}\\
C_o^3:& \quad x_m^n \leqslant y_m,\ \forall m; \forall n\\
C_o^4:& \quad \bar{N} y_m \leqslant \sum_{n\in\mathcal{N}} x_m^n,\ \forall m \qquad ,\\
C_o^5:& \quad \overline{S} \leqslant S_m^n \leqslant T/T_s,\ \forall m; \forall n\\
C_o^6:& \quad 0 \leqslant T - T_S(\boldsymbol{X},\boldsymbol{y},\boldsymbol{\mathcal{S}}),\\
C_o^9:& \quad \boldsymbol{X} \in \{0,1\}^{M\times N}, \boldsymbol{y} \in \{0,1\}^{M},\\
& \quad \boldsymbol{\mathcal{S}} \in \mathbb{N}^{+^{M\times N}}, \boldsymbol{\mathcal{E}} \in \mathbb{R}^{B\times M}
\end{aligned}$$

which attains the optimal point when $Q_m^f(S_m^n,\varepsilon_m^n) = \bar{Q}_f$ and $Q_m^d(S_m^n,\varepsilon_m^n) = \bar{Q}_d$ as $Q_m^f(S_m^n,\varepsilon_m^n) < \bar{Q}_f$ and $\bar{Q}_d > Q_m^d(S_m^n,\varepsilon_m^n)$ require more strict thresholds and sensing duration, respectively; which reduces throughput and causes suboptimality. As explained in (16) and (17), the false alarm and detection probabilities satisfying $Q_m^f(S_m^n,\varepsilon_m^n) = \bar{Q}_f$ and $Q_m^d(S_m^n,\varepsilon_m^n) = \bar{Q}_d$ can be obtained as $P_{m,n}^{f^\star}(\bar{N})$ and $P_{m,n}^{d^\star}(\bar{N})$, respectively. By substituting $P_{m,n}^{f^\star}(\bar{N})$ and $P_{m,n}^{d^\star}(\bar{N})$ into (18) and (19), respectively, then solving for ε_m^n and S_m^n respectively yields the following optimal detection threshold and number of samples

$$\varepsilon_{m,n}^\star \leqslant 1 + \frac{\mathcal{Q}^{-1}\left(P_{m,n}^{f^\star}\right)}{\sqrt{S_{m,n}^\star}}, \tag{26}$$

$$S_{m,n}^\star \geqslant \left[\frac{\mathcal{Q}^{-1}\left(P_{m,n}^{d^\star}\right)\sqrt{2\gamma_m^n+1} - \mathcal{Q}^{-1}\left(P_{m,n}^{f^\star}\right)}{\gamma_m^n}\right]^2, \tag{27}$$

which are necessary and sufficient conditions to satisfy $Q_m^f(S_m^n,\varepsilon_m^n) = \bar{Q}_f$ and $Q_m^d(S_m^n,\varepsilon_m^n) = \bar{Q}_d$. By leveraging optimal number of samples, $\boldsymbol{\mathcal{S}}^\star$, $\mathbf{P_1}$ can be reduced to the following problem

$$\begin{aligned}
\mathbf{P_1'}:& \max_{\boldsymbol{X},\boldsymbol{y}} \sum_{\forall m\in\mathcal{M}} y_m \pi_m^0 (1-\bar{Q}_f)T_X(\boldsymbol{X},\boldsymbol{y},\boldsymbol{\mathcal{S}}^\star)\\
C_o^1:& \quad \text{s.t.} \quad x_m^n \leqslant y_m,\ \forall m; \forall n\\
C_o^2:& \quad \bar{N} y_m \leqslant \sum_{n\in\mathcal{N}} x_m^n,\ \forall m \qquad ,\\
C_o^3:& \quad 0 \leqslant T - T_S(\boldsymbol{X},\boldsymbol{y},\boldsymbol{\mathcal{S}}^\star),\\
C_o^4:& \quad \boldsymbol{X} \in \{0,1\}^{M\times N}, \boldsymbol{y} \in \{0,1\}^{M}
\end{aligned}$$

which is still NP-Hard and requires a fast yet efficient heuristic solution for practical purposes.

The proposed solution is detailed in Algorithm 1, which starts with a channel ranking procedure in Line 2. As explained in Section V-A, the channels are ranked based on their best possible average throughput considering their probability of being idle and channel exploitation time. Then, the optimal sensing duration and priority list returned from PC Rank is used by IoT Assign procedure in Line 3. As explained in Section V-B, IoT Assign procedure iteratively schedules PCs following the priority list such that m^{th} iteration schedules PCs with the highest priority and assign $\bar{N}$ IoTs for each scheduled PC, which is terminated if scheduling $(m+1)^{th}$ PC degrades the average throughput obtained in the m^{th} iteration. Based on returned sensing scheduling matrices, the SBS broadcasts the sensing lists to IoT nodes in Line 4, which are then report their local decisions in Line 5. The SBS eventually make a decision on the list of idle PCs using the voting rule in (13).

A. PC Priority Ranking

To ensure that $\bar{N}$ reports satisfy global false alarm and detection probabilities (i.e., C_o^1 and C_o^2), Line 9 of the PC Rank first computes required detection and false alarm probabilities reports as per (16) and (17), which are then used in Line 10 to obtain required local detection and false alarm probabilities by taking the BER of the reporting channel into account. Based on the local probabilities, we compute the matrix of optimal number of samples as derived in (27). Thereafter, the for-loop between Lines 12-15 acquires the PC scheduling priority metrics for each PC by repeating the following steps: Line 13 sorts the m^{th} column of $\boldsymbol{S}^\star$, $\boldsymbol{S}^\star[m,:]$, to reorder IoTs in ascending order of number of samples. Then, Line 14 takes the first $\bar{N}$ element of $\boldsymbol{S}^\star[m,:]$ to calculate the average maximum throughput of PC$_m$ if it is sensed by the best $\bar{N}$ possible IoT nodes. Finally, Line 16 returns the priority list following sorting PCs in descending order of their utility performance. With this priority list, having a higher *apriori* probability of being idle becomes as important as having a less sensing duration for better utilization of the idle PCs.

Algorithm 1 : Multi-Channel Cooperative Spectrum Sensing and Scheduling (MC$_2$S$_3$)

```
 1: Input:  $M, N, \bar{N}, \bar{Q}_f, \bar{Q}_d, \bar{S}, T, T_s, \pi_m^0, \boldsymbol{H}, p_m^n$
 2: $\wp^\star, \boldsymbol{S}^\star \leftarrow$ PC RANK$(\pi_m^o, T, T_s, \bar{N}, \bar{Q}_f)$
 3: $\boldsymbol{y}^\star, \boldsymbol{X}^\star \leftarrow$ IOT ASSIGN$(\wp^\star, \boldsymbol{S}^\star, \pi_m^0, T, \bar{Q}_f)$
 4: The SBS broadcasts sensing and scheduling decisions
 5: IoT nodes share decisions $(u_m^n, \forall m, n)$ with the SBS
 6: $\mathcal{I} \leftarrow$ as per (13)
 7: return $\mathcal{I}$
 8: procedure PC RANK$(\pi_m^o, p_m^n, T, T_s, \bar{N}, \bar{Q}_f, \bar{Q}_d)$
 9:     $\tilde{P}_m^{d,\star}(\bar{N})/\tilde{P}_m^{f,\star}(\bar{N}) \leftarrow$ as per (16) and (17), $\forall m$.
10:     $P_{m,n}^{i^\star}(\bar{N}) \leftarrow$ as per (18) and (19), $i \in \{f, d\}$
11:     $\boldsymbol{S}^\star \leftarrow$ Substitute $P_{m,n}^{i^\star}(\bar{N}), i \in \{f, d\}$, into (27)
12:     for $m = 1 : M$ do
13:         $\aleph \leftarrow$ SORTASCEND$(\boldsymbol{S}^\star[m, :])$
14:         $\wp[m] \leftarrow \pi_m^o(1 - \bar{Q}_f)(T - \sum_{i=1}^{\bar{N}} \aleph[i]T_s)$
15:     end for
16:     $\wp^\star \leftarrow$ SORTDESCEND$(\wp_m, \forall m)$
17: return $\wp^\star, \boldsymbol{S}^\star$
18: end procedure
19: procedure IOT ASSIGN$(\wp^\star, \boldsymbol{S}^\star, \pi_m^0, T, \bar{Q}_f)$
20:     $\tau \leftarrow 0$
21:     for $m = 1 : M$ do
22:         $\boldsymbol{X} \leftarrow 0^{M \times N}, \boldsymbol{y} \leftarrow 0^M$
23:         $\xi \leftarrow \wp[1 : m]$
24:         $\boldsymbol{y}[\xi] \leftarrow 1$
25:         $\boldsymbol{Q} \leftarrow \boldsymbol{S}^\star[\xi, :]$
26:         $\tilde{\boldsymbol{S}} \leftarrow \boldsymbol{S}^\star[\xi, :]$
27:         for $i = 1 : \bar{N}$ do
28:             $\boldsymbol{\delta} \leftarrow$ LINEAR BOTTLENECK ASSIGNMENT$(\boldsymbol{Q})$
29:             for $j = 1 : m$ do
30:                 $\boldsymbol{X}[j, \boldsymbol{\delta}[j]] \leftarrow 1$
31:                 $\boldsymbol{Q}[:, \boldsymbol{\delta}[j]] \leftarrow \boldsymbol{Q}[:, \boldsymbol{\delta}[j]] + \tilde{\boldsymbol{S}}[j, \boldsymbol{\delta}[j]]$
32:                 $\boldsymbol{Q}[j, \boldsymbol{\delta}[j]] \leftarrow \infty$
33:             end for
34:         end for
35:         $\Phi(m) \leftarrow \sum_{i \in \wp[1:m]} \pi_i^0(1 - \bar{Q}_f)(T - T_S(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{S}^\star))$
36:         if $\Phi(m) \geqslant \tau$ then
37:             $\tau \leftarrow \Phi(m)$
38:         else
39:             Break
40:         end if
41:     end for
42: return $\boldsymbol{y}, \boldsymbol{X}$
43: end procedure
44: procedure LINEAR BOTTLENECK ASSIGNMENT$(\boldsymbol{Q})$
45:     $\boldsymbol{\chi} \leftarrow \max_{\boldsymbol{\chi}} \left\{ \min_{\forall i, \forall j} \{\boldsymbol{Q}[i, j]\boldsymbol{\chi}[i, j]\} \right\}$
46:         s.t. $\sum_i \boldsymbol{\chi}[i, j] \leqslant 1, \forall j$
47:              $\sum_j \boldsymbol{\chi}[i, j] = 1, \forall i$
48: return $\boldsymbol{\chi}$
49: end procedure
```

B. IoT Sensing Assignment

The IoT sensing assignment procedure starts with initialization of utility (i.e., average throughput) threshold in Line 20, followed by a for-loop between Lines 21-39. Line 22 initializes sensing scheduling variables; then Line 23 schedules first m PCs from $\wp$ to be sensed, which is reflected on the PC scheduling variable in Line 24. The columns of scheduled PCs are extracted from $\boldsymbol{S}^\star$ to obtain a cost matrix $\boldsymbol{Q} \in \mathbb{R}^{m \times N}$ in Line 25, which is also stored as $\tilde{\boldsymbol{S}}$ for future purposes.

After that, the inner for-loop between Lines 27-34 assign an IoT to each PC at each iteration, until all PCs are assigned with a total of $\bar{N}$ IoTs for sensing. In Line 28, the LBA is executed to find the best IoT assignments that minimizes the maximum sensing duration among all IoT nodes. The LBA returns a vector $\delta \in \mathbb{N}^m$ whose j^{th} element's value correspond to the index of IoT assigned for sensing PC$_j$. The first step of for-loop between Lines 29-33 updates the assignment matrix by setting $\boldsymbol{X}[j, \boldsymbol{\delta}[j]] = 1$. Line 31 adds sensing duration of assigned IoT to account for its sensing duties in the next round of assignments. Then, Line 32 updates cost matrix with $\boldsymbol{Q}[j, \boldsymbol{\delta}[j]] = \infty$ for preventing selected IoTs to be assigned to sense the same PC again. Exploiting the sensing scheduling matrices, Line 35 computes the utility function, which is compared with the threshold between Lines 36-40; where threshold is updated if adding m^{th} PC into scheduling matrix improves the throughput, otherwise Line 39 breaks the for loop and corresponding sensing scheduling matrices are returned in Line 42.

C. PC Allocation and IoT Clustering

Given the set of idle PCs are obtained through Algorithm 1, now it is time to focus on the second part of (25) and excluding the irrelevant constraints related to spectrum sensing. Accordingly, PC allocation and IoT clustering can be directly obtained from $\mathbf{P_o}$ as follows

$$\begin{array}{llll} \mathbf{P_2}: & \max\limits_{\boldsymbol{Z} \in \{0,1\}^{I \times N}} & \sum\limits_{\substack{i \in \mathcal{I} \\ n \in \mathcal{N}}} R_i^n(\boldsymbol{Z}) & \\ C_o^1: & \text{s.t.} & \sum\limits_{m \in \mathcal{M}} z_m^n \leqslant 1, \ \forall n & , \\ C_o^2: & & \sum\limits_{n \in \mathcal{N}} z_m^n \leqslant \left\lceil \dfrac{N}{I} \right\rceil, \ \forall m & \end{array}$$

which is also a combinatorial problem and NP-hard. The proposed solution is illustrated in Fig. 2 and detailed in Algorithm 2, where initialization starts with sorting IoT nodes based on their channel gain and assigning the IoT node with i^{th} strongest channel to the i^{th} PC among I idle PCs, $i \in [1, I]$. The for-loop between Lines 7-14 iteratively admits IoT nodes into I clusters over $K-1$ iteration. At each iteration, Line 8 first updates the set of admission awaiting nodes, then Line 9 calls COMPUTE COST procedure to generate cost matrix whose index $\boldsymbol{Q}[i, j]$ represent the data rate if $\mathcal{A}_k\{j\}$ is admitted to $\mathcal{C}_i$. In Line 10, the linear sum assignment obtains the matching with the highest sumrate, which is followed by cluster and clustering matrix updates in Lines 12 and 13, respectively.

D. Computational Complexity Analysis

The computational complexity of solving simplified problem $\boldsymbol{P_1'}$ in a brute-force fashion can be obtained by considering following combinations: 1) $\sum_{m=1}^{M} \binom{M}{m}$ is total number of scheduling combinations where $\binom{M}{m}$ selects m PCs out of M PCs, 2) For each of selected m PCs, we need to choose $\bar{N}$ nodes out of N IoT nodes. Accordingly, the overall computational complexity is given by $\mathcal{O}\left(K \sum_{m=1}^{M} \binom{M}{m} \binom{N}{\bar{N}}^m\right)$ where

Algorithm 2 : PC Allocation and IoT Clustering

```
1: Input: $N, \mathcal{I}, \alpha, \tilde{\boldsymbol{H}}$
2: $\mathcal{I} \leftarrow$ as per (13)
3: $K \leftarrow \lceil \frac{N}{I} \rceil$
4: $\tilde{\mathcal{N}} \leftarrow$ Sort IoTs in descending order of channel gains
5: $\mathcal{C}_i \leftarrow \tilde{\mathcal{N}}\{i\}, i \in [1, \ldots, I]$
6: $\boldsymbol{Z}_i^j \leftarrow 1, j \in \mathcal{C}_i$
7: for $k = 1 : K-1$ do
8:     $\mathcal{A}_k \leftarrow \tilde{\mathcal{N}} - \bigcup_{i=1}^{I} \mathcal{C}_i$
9:     $\boldsymbol{Q} \leftarrow$ COMPUTE COST($\mathcal{A}_k, \mathcal{C}_i, \forall i$)
10:    $\boldsymbol{\chi} \leftarrow\leftarrow$ LINEAR SUM ASSIGNMENT($\boldsymbol{Q}$)
11:    $\mathcal{C}_i \leftarrow \mathcal{C}_i \bigcup \mathcal{A}_k\{j\}, \chi_i^j = 1, \forall(i,j)$
12:    $\boldsymbol{Z}[i,j] \leftarrow 1, \forall j \in \mathcal{C}_i$
13: end for
14: The SBS broadcasts PC Allocation and IoT clustering
15: return $\boldsymbol{Z}$,
16: procedure COMPUTE COST($\mathcal{A}_k, \mathcal{C}_i, \forall i$)
17:    $\boldsymbol{Q} \leftarrow \boldsymbol{0}^{I \times A_k}$
18:    for $i = 1 : I$ do
19:        for $j = 1 : A_k$ do
20:            $\mathcal{T}_i \leftarrow \mathcal{C}_i \bigcup \mathcal{A}_k\{j\}$
21:            $\boldsymbol{Q}[i,j] \leftarrow R_i^j, j \in \mathcal{T}_i$
22:        end for
23:    end for
24: return $\boldsymbol{Q}$
25: end procedure
26: procedure LINEAR SUM ASSIGNMENT($\boldsymbol{Q}$)
27:    $\boldsymbol{\chi} \leftarrow \max_{\boldsymbol{\chi}} \sum_{\forall i, \forall j} \boldsymbol{Q}[i,j]\boldsymbol{\chi}[i,j]$
28:        s.t. $\sum_i \boldsymbol{\chi}[i,j] \leqslant 1, \forall j$
29:             $\sum_j \boldsymbol{\chi}[i,j] = 1, \forall i$
30: return $\boldsymbol{\chi}$
31: end procedure
```

K is the number of operations required for each combination. On the other hand, the proposed PC ranking algorithm has $\mathcal{O}\left(MN\log N + M\log M\right)$ complexity where the first and second terms are induced from the SORTASCEND and SORTDESCEND operations in Lines 13 and 16, respectively. The complexity of proposed PC allocation and IoT assignment procedure is dominated by the cubic complexity of the LBA [26], yielding $\mathcal{O}\left(\sum_m \bar{N}(\max(m,N))^3\right)$.

On the other hand, the computational complexity of solving simplified problem $\boldsymbol{P_2}$ in a brute-force fashion can be obtained by assigning $K = \lceil \frac{N}{I} \rceil$ IoT nodes to a cluster at each iteration, decrementing the pool of available workers by K after each task assignment, and multiplying the combinations possible for each task. Assuming $\lceil \frac{N}{I} \rceil$ is an integer, the computational complexity becomes $\mathcal{O}\left(\prod_{i=0}^{M-1}\binom{N-iK}{K}\right)$. In case of frequency selective channels, the order of iteration is also need to be taken care of, yielding an overall $\mathcal{O}\left(M!\prod_{i=0}^{M-1}\binom{N-iK}{K}\right)$ complexity for the exhaustive PC allocation and IoT clustering. On the other hand, the computational complexity of Algorithm 2 is mainly driven by two steps: 1) IoT sorting and 2) iterative IoT matching. The IoT sorting in Line 5 has a complexity of $\mathcal{O}\left(N\log N\right)$. For PC allocation and IoT clustering, the complexity of for-loop between lines 7-13 is $\mathcal{O}\left(\sum_k^{K-1}(\max(I,A_k))^3\right)$ where .

The overall complexity of the proposed decomposition approach can be furhter simplified in two steps: 1) Algorithm 1 has $\mathcal{O}\left(M\bar{N}N^3\right)$ assuming that all PCs are scheduled and $N \geqslant M$ and 2) Algorithm 2 has $\mathcal{O}\left(KN^3\right)$ assuming $N \geqslant A_k \geqslant I$. Finally, the overall algorithm has a complexity of $\mathcal{O}\left(\left(K + M\bar{N}\right)N^3\right)$

Table I: Table of default parameters

Par.	Value	Par.	Value	Par.	Value
Q_f	0.01	W	180 KHz	G_m	3 dBi
$\bar{Q}_d$	0.99	p_m^n	10^{-3}	G_n	3 dBi
$\bar{N}$	4	τ_r	1 ms	d_0	10 m
$\bar{S}$	30	P_s	30 dBm	σ_m^n	8 dB
T	1 s	P_n	10 dBm	f_c	0.9 GHz

VI. NUMERICAL RESULTS

In order to gain a clear insight into the impact of various system parameters on the SN performance, IoT nodes are assumed to have identical system parameters. Since our focus is to demonstrate the effect of spectrum sensing and scheduling on the available RF energy harvesting time, we consider a unitary energy arrival rate for all IoT nodes. Moreover, the channel switching factor is set to be $\beta = 0.1$ ms/MHz [23]. Unless it is explicitly stated otherwise, we employ the parameter values summarized in Table I.

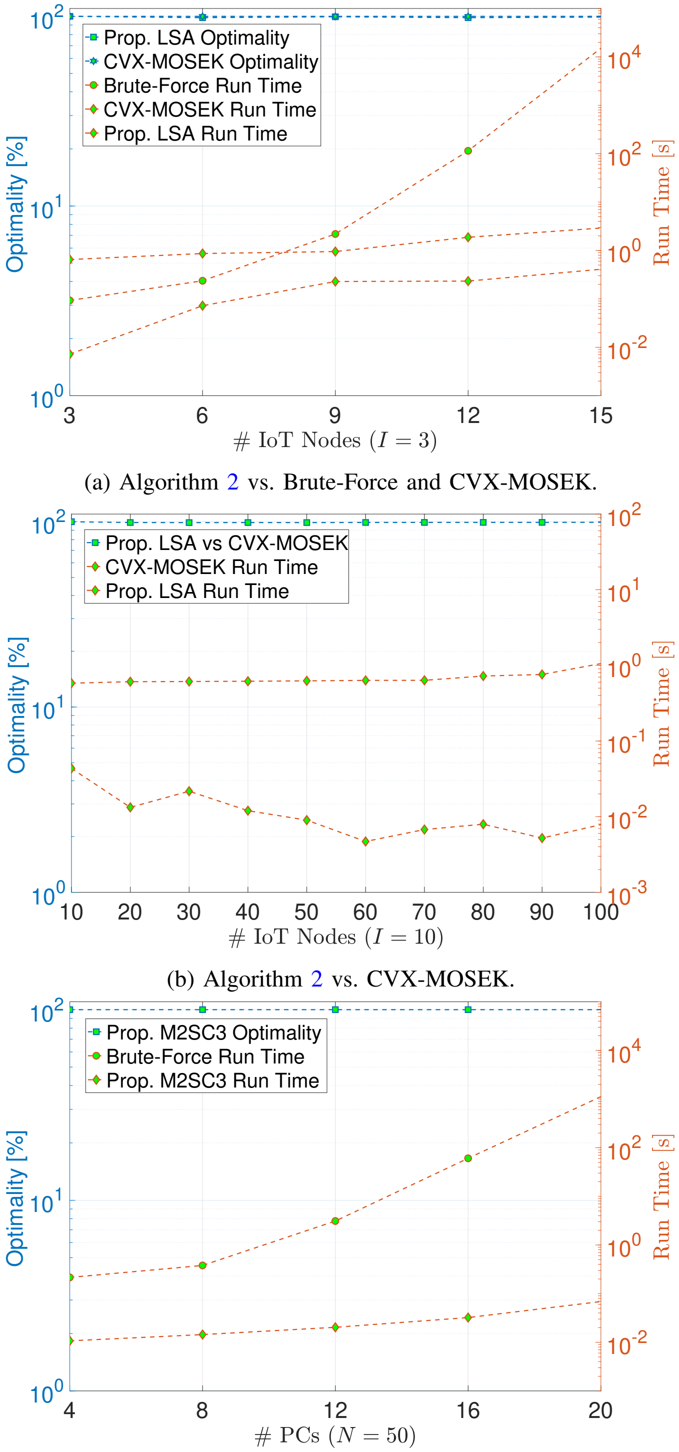
A. Optimality and Run Time Comparison

Before delving into performance evaluation of proposed solutions in a large scale network, we believe it is better to compare optimality and run time complexity with the brute-force approach, where we exhaustively explore all combinations to find the optimal solutions. We start with the comparison of Algorithm 2 as shown in Fig. 3a, where we considering clustering $N = \{3, 6, 9, 12, 15\}$ IoT nodes into $I = 3$ PCs/clusters, yileding following cluster sizes $C = \{1, 2, 3, 4, 5\}$. The proposed iterative LSA approach reaches %98.5 - %99.5 optimality while MOSEK[1] solver of CVX[2] reaches %100 optimality. Fig. 3a clear shows that computational complexity of the brute-force approach increases exponentially as the number of IoT increases even for a very low number of PCs, $I = 3$. To further investigate the optimality and complexity of the proposed iterative LSA approach, we drop brute-force approach and focus on the comparison with the CVX-MOSEK solution for a more realistic network size in Fig. 3b, where we show that our solution reaches %99- %100 optimality while keeping the complexity 2 orders of magnitude less than the CVX-MOSEK.

Given that the efficacy of linear assignment solutions have been shown in Fig. 3a and Fig. 3b, we will focus on comparing proposed M2CS3 approach with the brute-force solution in Fig. 3c, where we consider assignment of $N = 50$ IoT nodes to sense $M = \{4, 8, 12, 16, 20\}$ scheduled PCs. The proposed M2CS3 approach reaches %99.99 performance of brute-force solution in less than 0.1 seconds, proving the efficacy of the PC ranking and LBA-based sensing assignment.

[1] MOSEK is a large scale optimization software widely used to solve Linear, Quadratic, Semidefinite and Mixed Integer problems [27].

[2] CVX is a modeling system for constructing and solving disciplined convex programs [28].

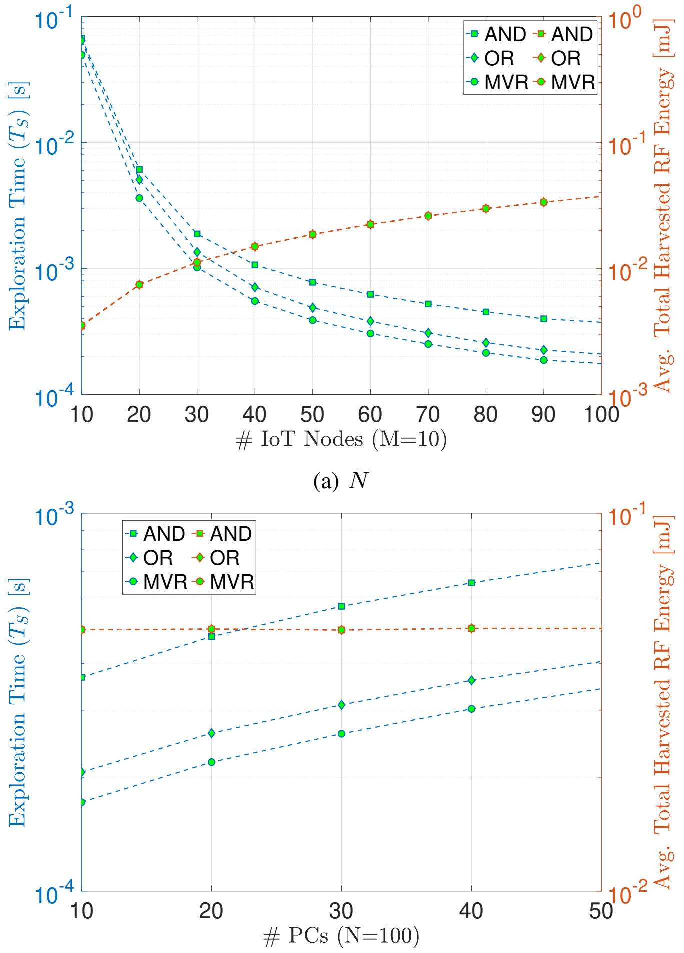


(a) Algorithm 2 vs. Brute-Force and CVX-MOSEK.

(b) Algorithm 2 vs. CVX-MOSEK.

(c) Algorithm 1 vs. Brute-Force.

Figure 3: The optimality and run time complexity comparison for: a) Algorithm 2 and b) Algorithm 1.



(a) N

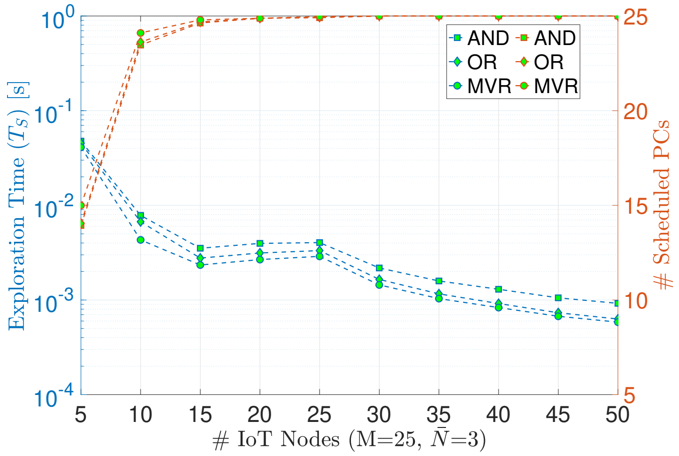
(b) M

Figure 4: The impact of number of IoT nodes and PCs on exploration time and average total harvested RF energy.

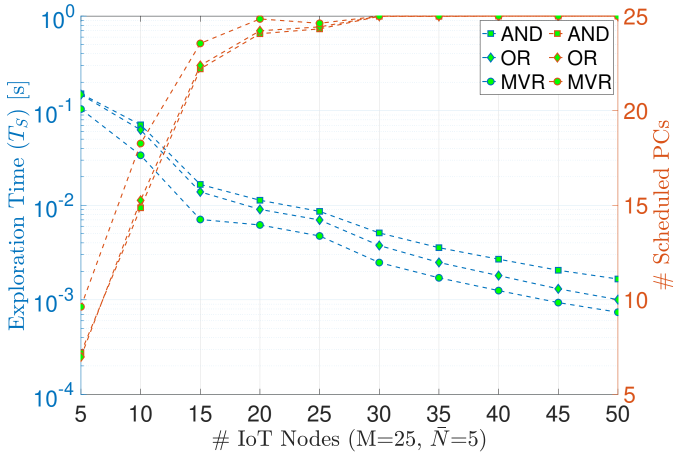
B. Exploration Time and Harvested Energy

As we already shown the optimality and speed of proposed solutions in the previous section, the rest of this section will consider larger network sizes and exclude comparisons with exhaustive solutions. The channel exploration time, T_S with respect to N for $M = 10$ is shown in Fig. 4a where maximum voting rule and AND rules requires lowest and highest sensing duration, respectively. T_S also reduces as N increases as having more IoT nodes increase the likelihood of having IoT nodes with better sensing attributes. On the other hand, Fig. 4b shows that having more PCs increases the T_S as the likelihood of having PCs with higher probability of being idle increases, which yields more PCs to be scheduled and more time spent for sensing. The impact of T_S on the average total harvested RF energy ($\bar{E}_{RF} = \frac{\chi_{RF}}{M} \sum_m \pi_m^1 \bar{Q}_d(T - T_S)$) is also shown in Fig. 4a and Fig. 4b. Fig. 4a shows that as N increases we have more nodes to harvest RF energy with more exploitation time to harvest energy. On the other hand, the impact of having more PCs is limited since our algorithms focused more on scheduling PCs with higher probability of being idle.

It is also important to investigate N and required number of sensing IoT nodes per PC, $\bar{N}$, for given M. Fig. 5a and



(a)



(b)

Figure 5: Impact of N and $\bar{N}$ on exploration time and # scheduled PCs: a) $\bar{N} = 3$ and b) $\bar{N} = 5$

Fig. 5b show that $N \leqslant M$ yields under utilization of idle PCs as low number of sensing IoT nodes cause a limited discovery of spectrum. This is also closely related to how many IoT nodes required to sense each PC. Setting a lower $\bar{N}$ can allow us to discover more PCs as $\bar{N} = 3$ and $\bar{N} = 5$ compared in Fig. 5a and Fig. 5b, respectively. Here, the impact of voting rule becomes more obvious as having less T_S allow MVR to schedule more PCs, improving overall idle spectrum discovery. For the same set up, we also show normalized and average network throughput per IoT node in Fig. 6, where $\bar{N} = 3$ and MVR provides a better performance than other schemes. Moreover, it is obvious that even if all PCs are scheduled by $N = 30$, the NOMA scheme was capable of providing similar performance for additional IoT nodes thanks to effective multiplexing of NOMA.

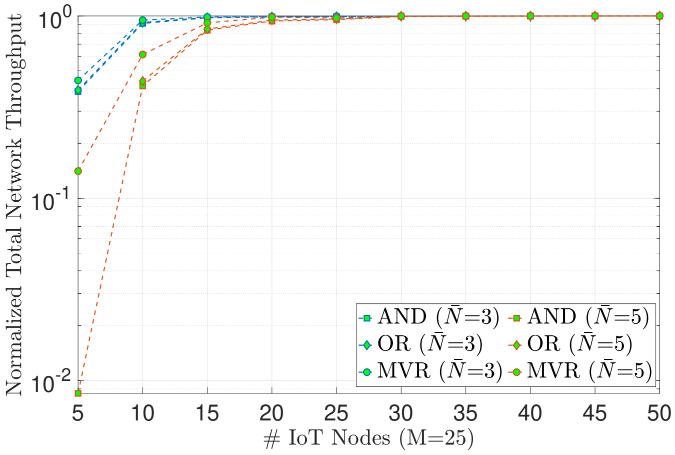


Figure 6: Algorithm 1 vs. Brute-Force.

Figure 7: Normalized network throughput per IoT node.

VII. Conclusions

In this paper, we introduced a comprehensive framework, MC_2S_3, designed to enhance the throughput of cognitive IoT networks through advanced multi-channel cooperative spectrum sensing and scheduling. By tackling the challenge of optimizing network throughput with a novel decomposition strategy, we effectively addressed the NP-hard nature of the joint optimization of sensing scheduling, device assignment, and node clustering within these networks. Our approach, which intelligently divides the problem into manageable subproblems of PC exploration and exploitation, not only ensures efficient spectrum utilization but also minimizes potential collision with primary networks. The adoption of NOMA for the exploitation phase further underscores our framework's capability to maximize the use of available spectrum resources by accommodating multiple IoT nodes on a single PC. Numerical analysis underscores the superiority of our proposed methodologies, showcasing near-perfect accuracy and significant performance gains over conventional brute-force methods. The implications of our findings are twofold: they pave the way for more efficient cognitive IoT network operations and offer a scalable solution for future spectrum and energy harvesting endeavors. Future work will focus on extending this framework to accommodate dynamic network environments and exploring the integration of machine learning techniques for enhanced decision-making processes.

References

[1] A. Celik, I. Romdhane, G. Kaddoum, and A. M. Eltawil, "A top-down survey on optical wireless communications for the internet of things," *IEEE Communications Surveys Tutorials*, vol. 25, no. 1, pp. 1–45, 2023.

[2] M. Khasawneh, A. Azab, S. Alrabaee, H. Sakkal, and H. H. Bakhit, "Convergence of iot and cognitive radio networks: A survey of applications, techniques, and challenges," *IEEE Access*, vol. 11, pp. 71097–71112, 2023.

[3] A. Gharib, W. Ejaz, and M. Ibnkahla, "Distributed spectrum sensing for iot networks: Architecture, challenges, and learning," *IEEE Internet of Things Magazine*, vol. 4, no. 2, pp. 66–73, 2021.

[4] W. Ejaz and M. Ibnkahla, "Multiband spectrum sensing and resource allocation for iot in cognitive 5g networks," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 150–163, 2018.

[5] N.-N. Dao, W. Na, A.-T. Tran, D. N. Nguyen, and S. Cho, "Energy-efficient spectrum sensing for iot devices," *IEEE Systems Journal*, vol. 15, no. 1, pp. 1077–1085, 2021.

[6] F. Zhou, Y. Wu, Y.-C. Liang, Z. Li, Y. Wang, and K.-K. Wong, "State of the art, taxonomy, and open issues on cognitive radio networks with noma," *IEEE Wireless Communications*, vol. 25, no. 2, pp. 100–108, 2018.

[7] S. Arzykulov, A. Celik, G. Nauryzbayev, and A. M. Eltawil, "Uav-assisted cooperative cognitive noma: Deployment, clustering, and resource allocation," *IEEE Transactions on Cognitive Communications and Networking*, vol. 8, no. 1, pp. 263–281, 2022.

[8] P. Chauhan, S. K. Deka, B. C. Chatterjee, and N. Sarma, "Utility driven cooperative spectrum sensing scheduling for heterogeneous multi-channel cognitive radio networks," *Telecommunication Systems*, vol. 78, no. 1, pp. 25–37, 2021.

[9] Y. Cao and H. Pan, "Energy-efficient cooperative spectrum sensing strategy for cognitive wireless sensor networks based on particle swarm optimization," *IEEE Access*, vol. 8, pp. 214707–214715, 2020.

[10] H. Kaschel, K. Toledo, J. T. Gómez, and M. J. F.-G. García, "Energy-efficient cooperative spectrum sensing based on stochastic programming in dynamic cognitive radio sensor networks," *IEEE Access*, vol. 9, pp. 720–732, 2021.

[11] A. Ostovar, Y. B. Zikria, H. S. Kim, and R. Ali, "Optimization of resource allocation model with energy-efficient cooperative sensing in green cognitive radio networks," *IEEE Access*, vol. 8, pp. 141594–141610, 2020.

[12] O. M. Al-Kofahi, H. M. Almasaeid, and H. Al-Mefleh, "Efficient on-demand spectrum sensing in sensor-aided cognitive radio networks," *Computer Communications*, vol. 156, pp. 11–24, 2020.

[13] A. Bagheri and A. Ebrahimzadeh, "Statistical analysis of lifetime in wireless cognitive sensor network for multi-channel cooperative spectrum sensing," *IEEE Sensors Journal*, vol. 21, pp. 2412–2421, Jan 2021.

[14] X. Fernando and G. Lăzăroiu, "Spectrum sensing, clustering algorithms, and energy-harvesting technology for cognitive-radio-based internet-of-things networks," *Sensors*, vol. 23, no. 18, 2023.

[15] J. Wu, C. Wang, Y. Yu, T. Song, and J. Hu, "Performance optimisation of cooperative spectrum sensing in mobile cognitive radio networks," *IET Communications*, vol. 14, no. 6, pp. 1028–1036, 2020.

[16] W. Ning, X. Huang, K. Yang, F. Wu, and S. Leng, "Reinforcement learning enabled cooperative spectrum sensing in cognitive radio networks," *Journal of Communications and Networks*, vol. 22, pp. 12–22, Feb 2020.

[17] Z. Shi, W. Gao, S. Zhang, J. Liu, and N. Kato, "Machine learning-enabled cooperative spectrum sensing for non-orthogonal multiple access," *IEEE Transactions on Wireless Communications*, vol. 19, pp. 5692–5702, Sep. 2020.

[18] R. Ahmed, Y. Chen, B. Hassan, and L. Du, "Cr-iotnet: Machine learning based joint spectrum sensing and allocation for cognitive radio enabled iot cellular networks," *Ad Hoc Networks*, vol. 112, p. 102390, 2021.

[19] A. Bagheri, A. Ebrahimzadeh, and M. Najimi, "Game-theory-based lifetime maximization of multi-channel cooperative spectrum sensing in wireless sensor networks," *Wireless networks*, vol. 26, no. 6, pp. 4705–4721, 2020.

[20] M. Rajendran and M. Duraisamy, "Distributed coalition formation game for enhancing cooperative spectrum sensing in cognitive radio ad hoc networks," *IET Networks*, vol. 9, no. 1, pp. 12–22, 2020.

[21] P. Chauhan, S. K. Deka, B. C. Chatterjee, and N. Sarma, "Cooperative spectrum prediction-driven sensing for energy constrained cognitive radio networks," *IEEE Access*, vol. 9, pp. 26107–26118, 2021.

[22] A. Gharib, W. Ejaz, and M. Ibnkahla, "Scalable learning-based heterogeneous multi-band multi-user cooperative spectrum sensing for distributed iot systems," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 1066–1083, 2020.

[23] D. Gözüpek, S. Buhari, and F. Alagöz, "A spectrum switching delay-aware scheduling algorithm for centralized cognitive radio networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 7, pp. 1270–1280, 2013.

[24] E. C. Y. Peh, Y.-C. Liang, and Y. L. Guan, "Optimization of cooperative sensing in cognitive radio networks: A sensing-throughput tradeoff view," in *2009 IEEE International Conference on Communications*, pp. 1–5, 2009.

[25] S. W. Kim, "Simultaneous spectrum sensing and energy harvesting," *IEEE Transactions on Wireless Communications*, vol. 18, no. 2, pp. 769–779, 2019.

[26] D. F. Crouse, "On implementing 2d rectangular assignment algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 4, pp. 1679–1696, 2016.

[27] M. ApS, "Mosek optimization toolbox for matlab," *User's Guide and Reference Manual, Version*, vol. 4, p. 1, 2019.

[28] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1." http://cvxr.com/cvx, Mar. 2014.

Abdulkadir Celik (Senior Member, IEEE) received M.S. degrees in electrical engineering in 2013 and in computer engineering in 2015, then the Ph.D. degree in co-majors of electrical engineering and computer engineering in 2016, all from Iowa State University, Ames, IA, USA. He was a Postdoctoral Fellow with the King Abdullah University of Science and Technology, Thuwal, KSA, from 2016 to 2020, where he is currently a Senior Research Scientist with the Communications and Computing Systems Laboratory. Dr. Celik is the recipient of IEEE Communications Society's 2023 Outstanding Young Researcher Award for Europe, Middle East, and Africa (EMEA) region. He currently serves as an editor for IEEE Communications Letters, IEEE Wireless Communications Letters, and Frontiers in Communications and Networks. His research interests are in the broad areas of next-generation wireless communication systems and networks.