

Chinese Remainder Theorem-Based Sequence Design for Resource Block Assignment in Relay-Assisted Internet-of-Things Communications

Yaser M. M. Fouad¹, Ramy H. Gohary, *Senior Member, IEEE*, and Halim Yanikomeroglu², *Fellow, IEEE*

Abstract—Terminal relays are expected to play a key role in facilitating the communication between base stations and low-cost power-constrained cellular Internet of Things (IoT) devices. However, these mobile relays require a mechanism by which they can autonomously assign the available resource blocks (RBs) to their assisted IoT devices in the absence of channel state information (CSI) and with minimal assignment conflicts. To address this problem, in this paper, we develop an autonomous sequence-based RB assignment scheme that dispenses with CSI. The sequences underlying the proposed scheme are designed using the Chinese remainder theorem (CRT). In particular, the CRT is used to combine the cyclic sequences generated by simple cyclic group structures into longer ones. The combining process introduces additional degrees of freedom in sequence generation, thereby enriching the set of RB assignment sequences. Simulation results show that the sequences generated by the proposed CRT-based scheme outperform those generated by currently available autonomous ones.

Index Terms—Cooperative D2D Relaying, distributed resource allocation, machine-type communications, sequence-based resource allocation, terminal relaying, CRT-based sequence design.

I. INTRODUCTION

INTERNET of Things (IoT) systems are envisioned to create an enriched human experience by offering connectivity to billions of devices in a wide spectrum of applications; e.g., health monitoring, smart grids, and intelligent transportation systems [1]–[3]. However, to realize these applications, it is necessary to provide ubiquitous coverage to a large number of low-cost power-constrained devices [4]. To satisfy this coverage requirement, one possibility is to enable IoT devices to utilize the readily available cellular infrastructure; i.e., we refer to those systems as cellular IoT systems. In these systems, each IoT device must possess the capability to directly communicate with the base station (BS). Unfortunately, communicating directly with the BS may result in increasing the complexity and power consumption of IoT devices and subsequently their cost. One potential candidate for solving this problem is a system architecture that exploits wireless

terminals to opportunistically act as relaying terminals (RTs) to assist the communications between their neighboring IoT devices and the BSs. In other words, such wireless terminals are not dedicated for relaying. This approach is desirable because the number of terminals tends to naturally increase with the number of IoT devices and thus these terminals can play a key role in providing connectivity to neighboring IoT devices with no additional cost to the system. This relaying-based approach has been proposed by the 3rd Generation Partnership Project (3GPP) for the Long Term Evolution (LTE) system. In particular, this approach was first initiated in 3GPP for Release 12 of LTE under proximity-based services [5]. However, in this release it was limited to providing coverage extensions to terminals with unfavorable channel conditions [6]; e.g., cell edge users. This approach was further extended in Release 14 to provide support for vehicle to infrastructure applications [7]. In these applications, vehicles form clusters, whereby one vehicle acts as a cluster head and accordingly relay the signals of its neighboring vehicles to the BS. Another 3GPP work stream, i.e., the 3GPP 5G New Radio (NR), envisions that wireless terminals can act as relays and accordingly collect data from their neighboring devices over an operator-controlled sidelink in the licensed cellular spectrum [8]. Since each IoT device communicates only with its neighboring RT, terminal relaying offers three favorable features: 1) IoT devices can utilize low power communications and low complexity transmitters, thereby reducing their cost and power requirements; 2) the data collected by the wireless terminals can be locally processed thus reducing the required resources on the expensive long range link (the link between the wireless terminal and the BS); and 3) The massive number of IoT devices attempting simultaneous communication with the BS can be significantly reduced, thereby reducing the interference at the BS and the probability of collision on the random access channel.

Despite the advantages of RT-assisted communications, the prospected large number of RTs and their incidental operation suggest that the signalling overhead for coordinating their RB assignments may not be practically feasible. In particular, the network topology of an RT-assisted system is considered to be highly dynamic due to the mobility and the large numbers of RTs and IoT devices. Hence, the signalling required for coordination between neighboring RTs can significantly infringe on the available resources. This is because coordinating RB assignments between neighboring RTs requires two types of signalling: 1) frequent discovery and

Manuscript received August 4, 2017; revised November 22, 2017 and February 19, 2018; accepted February 24, 2018. Date of publication March 9, 2018; date of current version May 8, 2018. The associate editor coordinating the review of this paper and approving it for publication was D. Niyato. This work is supported in part by Huawei Canada Inc. (*Corresponding author: Yaser M. M. Fouad.*)

The authors are with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON K1S 5B6, Canada (e-mail: yfouad@sce.carleton.ca).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TWC.2018.2811798

handshaking to obtain an updated list of neighboring RTs; and 2) RB reservation messages to coordinate RB assignments [9]. To address this drawback, distributed resource block (RB) assignment schemes were proposed to enable the RTs to autonomously assign the RBs to their assisted IoT devices; e.g., [10]–[12]. However, these schemes assume the availability of the instantaneous channel state information (CSI) of all IoT devices at each RT, an assumption that can seriously hinder the RBs available for communications. Alternatively, the distributed schemes in [13] and [14] statistically model channel uncertainties and accordingly obtain potentially inaccurate estimates of CSI at any given instant for RB assignment. This work was extended in [15] to heterogeneous systems that support both human-type devices and IoT ones. In these systems, the large numbers and high mobility of RTs and IoT devices might result in substantial changes in the actual CSI thus leading to highly suboptimal assignments. In other words, the channels between the IoT devices and the RTs are expected to suffer from limited coherence time and bandwidth, thereby rendering CSI acquisition expensive due to signalling overhead and pilot transmissions. To mitigate these drawbacks, the work in [16] proposed a random RB assignment scheme that dispenses with both centralized coordination and CSI. Random-based schemes for medium access such as ALOHA and carrier sensing multiple access (CSMA) were also considered in many practical non-cellular IoT communication systems [17] such as Sigfox [18], LoRa [19], and dedicated short range communications (DSRC) [20]. A variation of the CSMA scheme, namely truncated CSMA, was also proposed in [21] to enable distributed RB reservations for IoT data transmissions in a network with stringent quality of service requirements. In [22], the authors proposed a dynamic RB assignment scheme for wireless sensor networks that relies on a CSMA-based approach and a tool from game theory, i.e., regret matching, for RB assignments. Similarly, the work in [23] introduced a dynamic resource allocation scheme for cluster-based mobile ad-hoc networks (MANETS). In this scheme, devices randomly contend for resources and rely on a sensing-based approach to avoid collisions with their neighbors. Similarly, the authors of [24] developed a distributed RB assignment strategy for wireless sensor networks (WSNs), whereby devices select their resources randomly from a predefined pool that is continuously updated based on the observed collisions with their neighbors.

Despite the advantages of the aforementioned random RB assignment schemes, the lack of coherence in their structure results in, otherwise avoidable, RB allocation conflicts especially in scenarios with large numbers of devices; e.g., IoT systems. To address this problem, the work in [25] proposed an autonomous RB assignment scheme for relay-assisted systems that dispenses with CSI while minimizing the number of assignment conflicts between neighboring devices. In this scheme, each RT locally generates an RB assignment sequence based on the cyclic group structure. These sequences were shown to be easily generated by a single group generator and to substantially reduce the assignment conflicts between neighboring RTs when compared with the random RB assignments; e.g., [16]. This scheme was extended

in [26] by providing number theoretic techniques for selecting the cyclically-generated sequences (CGS) for RB assignments. However, a limitation of the work in [25] and [26] is that it considers only CGS by a single group generator which significantly reduces the set of available sequences for RB assignment.

In this paper, we propose an autonomous RB assignment scheme for a distributed IoT system in which RTs assist the communication between IoT devices and the BSs. In this scheme, the RTs autonomously assign the available RBs to their incoming IoT devices according to locally-generated RB assignment sequences. The structure of these sequences is independent of the instantaneous channel quality and thus dispenses with the CSI requirement. This is inline with the evolving IoT paradigm, wherein swarms of communicating nodes result in a highly dynamic environment that renders CSI acquisition expensive and impractical in many scenarios. In addition, the RB assignment sequences are jointly designed prior to system startup such that their potential overlap at any given instant is minimal. This results in minimizing the interference caused by RB assignment conflicts. To see this, we note that the occurrence of assignment conflicts depends on the RB assignments, the system load, and the number of RTs, all of which vary with time. Hence, to reduce the number of these conflicts, the proposed sequences are jointly designed to achieve implicit coordination between neighboring RTs, thereby minimizing the assignment conflicts at particular RBs while other RBs are not assigned. This design also ensures that each sequence spans all the available RBs, providing each RT with unrestricted access to the system's RBs. Allowing each RT to have access to all the system's RBs is essential to guarantee high system efficiency and spectrum utilization when the network load is not uniformly distributed across all RTs, which is typically the case in practical networks; e.g., [7]. To generate the RB assignment sequences, in this paper, we consider an approach that combines multiple cyclic sequences using the Chinese remainder theorem (CRT). In particular, to obtain its sequence, each RT combines the ones generated by multiple group generators belonging to two or more cyclic group structures. This combining is done by solving multiple sets of linear congruences, where a congruence is an equivalence relation modulo a given base; i.e., for a given base n , two integers a and b are congruent if and only if their difference is an integer multiple of n , [27]. In particular, for each sequence element, a set of linear congruences is solved by the RT using the CRT. The number of these congruences solely depends on the number of group generators assigned to the RT. Unlike the single group generator approach, the proposed CRT-based approach offers an enriched set of sequences that can be used for RB assignments. Numerical results suggest that the performance of the sequences generated by the proposed CRT-based scheme outperform those generated by the cyclic group structure of [25] and the random RB assignments of [16]. To summarize, the proposed scheme

- uses CRT-structured sequences for RB assignments that can be easily-generated locally at the RTs with minimal signalling overhead;

- offers additional degrees of freedom by allowing the RTs to combine multiple cyclic sequences thus yielding a rich set of sequences for RB assignments;
- enables each RT to have unrestricted access to all the available RBs in the system;
- uses CRT-based sequences that are jointly designed such that their overlap is minimal to yield an implicit coordination between neighboring RTs when assigning their RBs;
- reduces the number of RB assignment conflicts while allowing RTs to operate autonomously;
- uses a CRT-based structure of sequences that is independent of the communication channel quality and thus dispenses with the CSI requirement; and
- yields RB assignment sequences that outperform those obtained by a single generator.

II. AUTONOMOUS SEQUENCE-BASED RB ASSIGNMENTS

We consider a relay-assisted OFDMA-based frequency division duplexing (FDD) system. In this system, wireless user terminals switch to the relaying mode and assist the communication between the BS and IoT devices within their vicinity; cf. Figure 1. We note that IoT devices are not required to use the assistance of the same RT for both uplink (UL) and downlink (DL). In particular, in the considered FDD system, UL and DL transmissions are performed on separate bands and thus the resources available for UL transmissions will be separate from those available for DL transmissions. Hence, an IoT device can have a decoupled DL/UL association with two RTs, each of which assisting the communication in one direction, thereby enabling IoT devices to realize the gains of decoupled DL/UL associations [28]. An RT is selected by an IoT device according to the observed signal-to-noise-plus-interference-ratio (SINR). To reduce the signalling overhead, the RTs assign the RBs to their assisted IoT devices autonomously, i.e., without centralized coordination, and blindly, i.e., without acquiring the CSI. IoT devices are divided into categories prior to system startup and RTs can allocate one or multiple RBs based on the category of the incoming IoT device; i.e., each category is associated with a predefined number of RBs per assignment. For instance, when an IoT device requests an RT's assistance, it sends an assignment request that includes its ID. Using this ID, the RT identifies the device category and accordingly assigns the corresponding number of RBs. However, if an IoT device still requires more RBs, it can send more requests to the RT for additional RB assignments.

To maintain autonomy, each RT performs its RB assignments based on a locally-generated sequence. However, in the absence of coordination, multiple neighboring RTs can simultaneously assign the same RB to their assisted IoT devices. In this work, we refer to such occurrences as 'hits'. These hits significantly reduce the SINR observed by IoT devices. Hence, to maintain the quality of service for IoT devices, the RB assignment sequences must be designed to reduce the number of hits without requiring coordination between RTs. In addition, these sequences must be structured such that they can be easily-generated locally by each RT. To meet

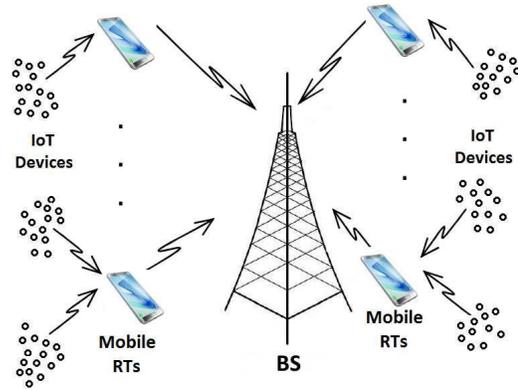


Fig. 1. A cellular IoT system assisted by RTs.

these requirements, one approach is to utilize the cyclically-structured RB assignment sequences of [25] that can be obtained by a single generator. For this structure, the number of available CGS for a given sequence order N , is upper bounded by N as will be shown in the following section. These cyclic sequences were shown to provide better performance when used for RB assignments in relay-assisted cellular networks than the randomly generated ones [25]. Similar to this approach, the authors in [29]–[32], proposed sequence-based RB assignment schemes for frequency hopping systems that relies on shuffled pseudonoise (PN) sequences. In these schemes, each device is assigned a sequence that specifies its assigned RB at each dwelling interval. To simplify this assignment, only the sequence seed is communicated to the device. Using this seed and a predefined shuffling function, each device locally generates its hopping sequence. However, a major drawback of this work is that it relies on a single seed for sequence generation, thereby limiting the number of available sequences. To further reduce the number of hits in the absence of coordination, it is desirable to expand the set of sequences for RB assignment while maintaining a structure that facilitates their generation. To meet this objective, in this work we propose an approach based on the CRT to combine the readily available CGS obtained by the cyclic group structure. This combining process yields an enlarged set of sequences that can reduce the number of hits when used for RB assignments. The details of the proposed approach are discussed next.

III. PRELIMINARIES

We now review preliminaries pertaining to the generation of CRT-based sequences. Cyclic sequences are generated from a specific subset of groups, i.e., the cyclic groups, whereby a group \mathbb{G} is cyclic if all its elements can be generated by repeated application of the group operation on at least one element in \mathbb{G} . Such an element is called a group generator. The order of \mathbb{G} is the number of its elements. For a cyclic group with N elements, the number of group generators is given by the Euler Totient function, $\phi(N)$ [27]. From the cyclic groups definition, it can be concluded that each of the generators of a group \mathbb{G} with order N yields a unique sequence with N entries. This sequence can be cyclically obtained by repeatedly

applying the group operation on the generator. This implies that out of $N!$ sequences with N entries, there exists only $\phi(N)$ distinct CGS, each corresponding to a group generator. The generation of these sequences render them attractive for RB assignment in relay-assisted IoT systems since they can be easily generated locally by each RT. To further enrich the set of easily generated sequences for RB assignments, one might consider a technique that yields multi-generator sequences. One example is the CRT, which can be used to combine the cyclic sequences obtained by multiple group generators. In particular, the CRT can be used to solve a set of congruences for the sequence elements belonging to different cyclic groups, thereby yielding sequences of longer lengths. We begin by introducing the CRT.

Theorem 1 (CRT [33]): Let $\{N_1, \dots, N_m\}$ be a set of pairwise coprime numbers. There exists an integer n that satisfies the system of congruences

$$n \equiv a_1 \pmod{N_1}, \dots, n \equiv a_m \pmod{N_m}, \quad (1)$$

where $a_i \in \mathbb{Z}$, $i = 1, \dots, m$ and n is unique modulo $\prod_{i=1}^m N_i$. \square

In the following lemma, we will show that the solution obtained by the CRT for the congruences system is also unique for each ordered set of (a_i, \dots, a_m) .

Lemma 1: Let n be a solution for a set of congruences $\{n \equiv a_1 \pmod{N_1}, \dots, n \equiv a_m \pmod{N_m}\}$. This solution is unique for all $0 \leq a_i < N_i, i = 1, \dots, m$. \square

To prove Lemma 1, we will use a contradiction as follows:

Proof: Let n be a solution for the two unique sets of congruences $\{n \equiv a_1 \pmod{N_1}, \dots, n \equiv a_m \pmod{N_m}\}$ and $\{n \equiv b_1 \pmod{N_1}, \dots, n \equiv b_m \pmod{N_m}\}$, where $0 \leq a_i \leq N_i$ and $0 \leq b_i \leq N_i$. This yields $n \equiv a_i \pmod{N_1} \equiv b_i \pmod{N_1}$ and thus $a_i = b_i$, $i = 1, \dots, m$. \square

Now, we will briefly describe a simple technique for solving a linear congruence and then show how this technique can be extended to solve the set of linear congruences described in (1).

Consider a linear congruence $ax \equiv 1 \pmod{N}$, where a and N are relatively prime constants and x is the congruence solution. Equivalently, this congruence can be solved by finding a constant c and the value of x such that

$$1 = xa + Nc. \quad (2)$$

The values of x and c can be obtained by using a systematic approach in Number theory referred to as the Euclidean algorithm. In particular, consider the two integers ℓ_1 and ℓ_2 with a given greatest common divisor $\gcd(\ell_1, \ell_2)$. The Euclidean algorithm can be used to find the integers c_1 and c_2 such that $\gcd(\ell_1, \ell_2) = \ell_1 c_1 + \ell_2 c_2$. This is done by performing the repeated divisions in the procedure described in [33]. Since $\gcd(a, N) = 1$, applying the Euclidean algorithm on (2) yields x , and c . This algorithm solves one linear congruence, but cannot be used to solve a set of such congruences; e.g., the one in (1). However, using the pairwise coprime property, the CRT furnishes a systematic approach for solving (1) by repeated applications of the Euclidean algorithm. To illustrate, consider the m congruences in (1)

and let $M_k = (\prod_{j=1}^m N_j)/N_k, k = 1, \dots, m$. From the definition of M_k , it follows that

$$M_k \equiv 0 \pmod{N_j} \text{ if } j \neq k. \quad (3)$$

Let x_k be the solution for the following linear congruence:

$$M_k x_k \equiv 1 \pmod{N_k}, \quad k = 1, \dots, m. \quad (4)$$

Since $\{N_1, \dots, N_m\}$ is a set of coprime numbers, it follows that x_k can be obtained using the Euclidean algorithm, cf. (2). Multiplying both sides of (4) by a_k yields that x_k is a solution for

$$M_k x_k a_k \equiv a_k \pmod{N_k}. \quad (5)$$

Subsequently, using (3) and (5), the solution n , for the set of congruences described in (1) can be obtained from the values of x_k , $k = 1, \dots, m$ as follows:

$$n = a_1 M_1 x_1 + a_2 M_2 x_2 + \dots + a_m M_m x_m. \quad (6)$$

In the following section, we will use Theorem 1 and Lemma 1 to show that solving the congruences described in (1) for each of the elements of the m cyclic sequences of lengths N_1, \dots, N_m , referred to as the seed sequences, yields a sequence of distinct elements that is unique modulo $\prod_{i=1}^m N_i$. The resulting sequence will depend on the number of combined seed sequences, m , and the order by which the CRT is applied on their elements. This implies that the CRT offers additional degrees of freedom for sequence generation, thereby enriching the available sequences set for RB assignment in relay-assisted IoT systems.

IV. A CRT-BASED RB ASSIGNMENT SCHEME

The philosophy of the proposed work is to distribute the available RBs among the RTs in the absence of coordination. In other words, the objective is to avoid the unintentional overutilization of particular RBs while others are being underutilized. To achieve this objective, in the proposed scheme, neighboring RTs assign the available RBs based on a set of prescribed CRT-based sequences. The sequences within this set are jointly designed such that their overlap is minimal. This joint design yields an implicit coordination between neighboring RTs when assigning their RBs. Hence, it reduces the number of RB assignment conflicts with minimal signalling overhead. This sequence structured RB assignment philosophy implies that each RT needs to: 1) associate each sequence element to a specific RB; 2) use a signalling protocol to convey the assigned RBs to the assisted IoT devices; and 3) locally generate a CRT-based RB assignment sequence that spans all the available RBs in the system. Next, we will discuss the details of these three steps.

A. Association Between Sequence Elements and RBs

In the proposed scheme, each RB is assigned a label between 0 and $N - 1$, where N is the number of RBs. In other words, a one-to-one mapping is selected between the set of RBs and $\{0, \dots, N - 1\}$. This mapping is arbitrary and is



Fig. 2. A mapping example between sequence elements and RBs.

announced to all BSs and RTs prior to system startup (any one-to-one mapping yields on average the same performance due to the absence of CSIs). This arbitrary mapping implies that consecutive sequence elements do not necessarily map to consecutive RBs; i.e., the RTs can assign non-consecutive RBs to a particular IoT device. For example, consider a 2-RT scenario, wherein the selected mapping function associates RBs A , B , and C to the sequence elements 0, 2, and 1, respectively and RTs 1 and 2 are assigned the sequences $\{0, 2, 1\}$ and $\{0, 1, 2\}$, respectively. Now, assuming that each RT assists two IoT devices, the resulting RB assignments for this scenario are depicted in Figure 2.

From Figure 2, it can be seen that any mapping function would result in a hit between the two RTs. However, in the following sections, we will show how the CRT-based scheme can reduce the number of these hits. Finally, we note that the non-consecutive RB assignments approach is in line with LTE Release 10 and is chosen because it offers a greater flexibility in RB assignments and an increase in the potential for a frequency selective scheduling gain [34].

B. Signalling Protocol for RB Assignments

When an RT has an incoming IoT device, it uses a dedicated control channel to assign the next available RBs in its sequence to the IoT device. The number of these assigned RBs will depend on the category of the device; cf. Section II. After receiving its RB assignment, the IoT device sends an acknowledgement to the RT. At this point, two cases can occur: 1) a response is received by the RT (either an acknowledgement or a negative acknowledgement); or 2) no response is received. If an acknowledgement is received, the RT identifies that this assignment is successful and starts an activity timer for the corresponding IoT device. This timer-based approach is similar to that of LTE in which timers are used to monitor a user's inactivity duration [35]. If the activity timer expires before an RT receives data on an assigned RB, it identifies the RB as abandoned and infers that the corresponding IoT device is no longer in its vicinity. In this case, a gap will exist in the RT's RB assignment sequence. In other words, one or more of the previously assigned entries of the RB assignment sequence are currently not utilized. To use these entries, the RT can either allocate them to its first incoming IoT device, or it can shift the RB assignments of all the following IoT devices such that no gaps are present in the sequence. On the other hand, if a negative acknowledgement is received, the RT infers that a hit has occurred on this RB and assigns the following RB in the sequence to the IoT device. If no response is received, the RT infers that the RB assignment has failed and that this device is no longer in its vicinity. The RT then reassigns this RB to the following incoming IoT device. Finally, we note that, although this work

focuses on terminals acting as RTs, it is readily extensible to networks with small cell deployments. In particular, each small cell can be considered as an RT with an RB assignment sequence and accordingly serve the IoT devices within its coverage area. However, unlike mobile RTs, small cells are fixed and thus can have a wired backhaul connection that can be exploited to further improve the performance by allowing nearby small cells to partially coordinate their RB assignments. For instance, small cells can obtain the load levels of their neighbors over the backhaul connection and accordingly avoid assigning the RBs that would result in hits.

C. CRT-Based Sequence Generation for RB Assignments

In this section, we will begin by noting that any cyclic group can be mapped to the additive cyclic group (ACG) structure [36]. In particular, there exists at least one one-to-one and onto function that maps all cyclic group structures to the additive one. Hence, without loss of generality, we will focus attention to the CGS by the ACG. Using the ACG structure, an RT can locally generate its RB assignment sequences using only a group generator as follows:

$$\{kg_i \pmod{N}\}_{k=1}^N, \quad (7)$$

where N is the sequence length and g_i is the i^{th} group generator of the ACG \mathbb{Z}_N . To further enrich the set of sequences for RB assignments, cyclic shifts were applied on each of the N sequences of the ACG structure [25]. In particular, it was shown in [25] that cyclically-shifted sequences do not coincide with one another, thereby creating an additional degree of freedom for sequence generation. This implies that for a group \mathbb{G}_N , the number of available sequences is $N\phi(N)$. Each of these sequences can be obtained by adding a cyclic shift, s_j , to (7) as follows:

$$\{(k + s_j)g_i \pmod{N}\}_{k=1}^N, \quad j = 1, \dots, N. \quad (8)$$

Despite the simplicity of cyclic sequences generation, their limited numbers impose a serious restriction on the degrees of freedom available for avoiding collisions when used for autonomous RB assignments. This restriction becomes more pronounced in prospected autonomous systems with large numbers of RTs and IoT devices. To address this concern, we use the CRT to generate a super set of sequences that contains the cyclic ones and their shifted versions as a proper subset. In particular, the proposed CRT-based scheme is used to combine two or more CGS into longer ones, thereby offering additional degrees of freedom for sequence generation. This increases the cardinality of the sequences set available for RB assignments, while retaining the ease of generation property. In the proposed CRT-based scheme, each RT uses two or more generators and cyclic shifts of seed sequences, which are either selected autonomously by the RT or provided by the BS. The elements of these sequences are used to form a linear set of independent congruences. Such congruences have a unique solution that can be obtained by using the CRT. This solution is taken to be the element of the combined sequence; cf. Figure 3.

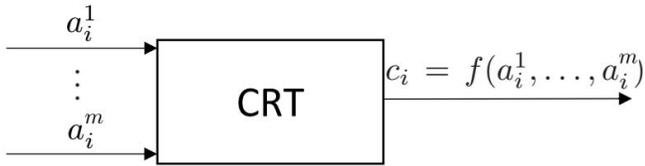


Fig. 3. Sequence Generation using the CRT.

In this figure, c_i is the i -th element of the sequence generated by the CRT, m is the number of seed sequences, and $f(\cdot)$ is the CRT function that maps the elements of the seed sequences, a_i^1, \dots, a_i^m , to c_i . This mapping function represents the solution obtained by the CRT, cf. (6), for the set of congruences formed by the elements a_i^1, \dots, a_i^m . By repeatedly solving the set of congruences for all elements of the seed sequences, we obtain a sequence that is unique modulo the multiplication of the lengths of its seed sequences. We will illustrate the proposed methodology with a simple example. Consider an RT, which is assigned the group generators (g_1, g_2) of the cyclic groups \mathbb{Z}_{N_1} and \mathbb{Z}_{N_2} . For simplicity we will assume zero cyclic shift in this example. To obtain its RB assignment sequence, the RT will use the CRT to combine the seed sequences of lengths N_1 and N_2 , respectively. This sequence will be parsed into N_1 sections each of length N_2 elements. In particular, the RT solves the following set of congruences

$$c_k^i \equiv ig_1 \pmod{N_1}, \quad (9)$$

$$c_k^i \equiv kg_2 \pmod{N_2}, \quad (10)$$

with $i = 1$ and $k = 1, \dots, N_2$ to obtain the first section of N_2 elements of its RB assignment sequence; i.e., $(c_1^1, \dots, c_{N_2}^1)$. The following sequence section of N_2 elements is then obtained by the RT by setting $i = 2$ in (9). This procedure is then repeated until all the N_1 sections of the combined sequence are obtained; i.e., until $i = N_1$. To elaborate on the details of the proposed CRT-based sequence generation scheme, we will consider an example that combines two seed sequences of lengths $N_1 = 5$ and $N_2 = 6$. Assume that an RT is assigned two cyclic shifts, s_1 and s_2 , and two group generators, g_1 and g_2 , of two distinct ACGs, e.g., \mathbb{Z}_5 and \mathbb{Z}_6 . For instance, for \mathbb{Z}_5 we choose $(g_1 = 4, s_1 = 0)$, whereas for \mathbb{Z}_6 we choose $(g_2 = 5, s_2 = 0)$. Using (8), the RT can generate its seed cyclic sequences from the assigned pairs. Using this expression, the RT generates two seed cyclic sequences, viz., $\mathbb{S}_1 = (4, 3, 2, 1, 0)$ and $\mathbb{S}_2 = (5, 4, 3, 2, 1, 0)$. Now, to obtain the first sequence element of the CRT-based sequence, the RT uses the lengths of the two seed cyclic sequences, i.e., 5 and 6, and their first elements, i.e., 4 and 5, to construct a set of joint linear congruences $c_1 \equiv 4 \pmod{5}$ and $c_1 \equiv 5 \pmod{6}$, where c_1 is the joint solution of the congruences. The RT then uses a systematic approach based on the CRT and the Euclidian algorithm to obtain the value of c_1 ; cf. Section III. This value then becomes the first element of the CRT-based RB assignment sequence of the RT under consideration. For the following elements, the RT fixes the first element of \mathbb{S}_1 and solves the constructed congruences for each of the remaining

five elements of \mathbb{S}_2 until exhaustion. Next, this process is repeated with the following element of \mathbb{S}_1 until exhaustion of all its elements. Once all congruences are solved, the RT obtains a sequence of length $N_1 N_2 = 30$, and the resulting sequence becomes $(29, 4, 9, 14, 19, 24, 23, 28, 3, 8, 13, 18, 17, 22, 27, 2, 7, 12, 11, 16, 21, 26, 1, 6, 5, 10, 15, 20, 25, 0)$.

Unlike the cyclic-based RB assignments in IoT systems, the proposed CRT-based scheme offers additional degrees of freedom for sequence generation. In particular, the proposed CRT-based scheme enables combining one or more seed cyclic sequences, thereby enriching the set of available sequences for RB assignments. The resulting combined sequences will depend on the number of combined seed sequences as well as the order in which they are combined. In other words, each ordered set of seed sequences will yield a unique set of sequences that can be used for RB assignments. Subsequently, the additional degrees of freedom increase the cardinality of the set of sequences available for RB assignments, which offers an opportunity for performance enhancement as will be shown by simulations in Section V. To elaborate on the benefits of the proposed CRT-based scheme, we will discuss the cardinality of the CRT-based sequences and compare it to that of the cyclic ones discussed in [25] in the following section.

D. The Cardinality of the CRT-Based Sequences Set

In this section, we will evaluate the number of RB assignment sequences of length N that can be generated by the proposed CRT-based sequence generation scheme. This number will depend on the following four features of the proposed scheme:

- 1) *Length of component seed sequences*: The lengths of component seed sequences determine the lengths of the combined sequences by the CRT. In other words, combining m seed sequences of lengths N_1, \dots, N_m yields a sequence of length $N = \prod_{i=1}^m N_i$.
- 2) *Number of CGS*: For a sequence of length N_i , the number of available CGS is given by $N_i \phi(N_i)$; cf. Section III.
- 3) *Coprimeness of N_i, \dots, N_m* : This condition is required for the congruences described in (1) to have a unique solution.
- 4) *Combining order of the seed sequences*: Lemma 1 implies that the CRT-based sequences directly depend on the order in which the seed cyclic sequences are combined. In other words, the sequence obtained by combining (S_1, S_2) is different from the one obtained by combining (S_2, S_1) , where S_1 and S_2 are two seed cyclic sequences.

Using these features, we will compute the cardinality of the set of length N sequences that can be generated by the proposed CRT-based scheme. In particular, these features imply that this cardinality will depend on the number of ordered sets (feature 4) of the pairwise coprime (feature 3) factors of N (feature 1) and their lengths (feature 2). Next, we will investigate the implications of each of these four features on this number.

We will begin by using the number of pairwise coprime factors of N to evaluate all the possible lengths of the seed cyclic sequences. For a given N , the number of pairwise coprime factors can be obtained by considering all possible combinations of its unique prime factors. In particular, let $N = \prod_{i=1}^{q(N)} p_i^{t_i}$, where $q(N)$ is the number of unique prime factors of N , p_i is the i -th prime factor of N and t_i is its repetitions. Note that, each prime factor p_i with t_i repetitions will be considered as a single element to ensure the pairwise coprimeness of the obtained factors of N . Now, to obtain the m coprime factors of N , one will need to consider all possible combinations of the prime factors into m distinct numbers. For example, consider a sequence of length $N = p_1 p_2^2 p_3$. All the ordered sets of the coprime factors of this length can be represented by all the possible combinations of its prime factors as follows $\{(p_1, p_2^2, p_3), (p_1, p_3, p_2^2), (p_2^2, p_1, p_3), (p_2^2, p_3, p_1), (p_3, p_2^2, p_1), (p_3, p_1, p_2^2), (p_1, p_2^2 p_3), (p_2^2 p_3, p_1), (p_3, p_2^2 p_1), (p_2^2 p_1, p_3), (p_2^2, p_3 p_1), (p_3 p_1, p_2^2), p_1 p_2^2 p_3\}$. The number of these combinations is equivalent to the number of ways in which the $q(N)$ prime factors can be partitioned into m non-empty subsets. The number of possible partitions for the $q(N)$ prime factors into non empty subsets can be evaluated from the Stirling number of the second kind, which is defined as follows:

Definition 1 (Stirling Number of the Second Kind [37]):

The number of ways to partition a set of n objects into k non-empty subsets is given by the Stirling number of the second kind and is denoted by $\left\{ \begin{smallmatrix} n \\ k \end{smallmatrix} \right\}$, where

$$\left\{ \begin{smallmatrix} n \\ k \end{smallmatrix} \right\} = \frac{1}{k!} \sum_{j=0}^k (-1)^{k-j} \binom{k}{j} j^n. \quad (11)$$

□

The expression in (11) yields the number of coprime factors of N by substituting $n = q(N)$ and $k = 1, \dots, q(N)$. However, it does not consider all possible permutations of these factors. Since, the output CRT-based sequences depend on the order in which the seed sequences are combined (feature 4), all ordered sets of the coprime factors of N must be considered when calculating the number of sequences. Using (11), the number of all possible ordered lengths of sequences that can be used as an input to the CRT to obtain a sequence of length N can be evaluated by:

$$\sum_{k=1}^{q(N)} \left\{ \begin{smallmatrix} q(N) \\ k \end{smallmatrix} \right\} k! = \sum_{k=1}^{q(N)} \sum_{j=0}^k (-1)^{k-j} \binom{k}{j} j^{q(N)}. \quad (12)$$

So far, we have used features 1, 3, and 4 to compute the number of possible lengths of the seed sequences for the proposed CRT-based scheme. It remains to consider the effect of feature 2 on the number of available length N CRT-based sequences; i.e., the number of available CGS for each of the lengths of the combined seed sequences. For a given N_i , there exists $N_i \phi(N_i)$ possible cyclic sequences and thus the total number of combinations of the possible seed cyclic sequences is given by $\prod_{i=1}^m N_i \phi(N_i)$. Since all values of N_i , $i = 1, \dots, m$, are relatively prime and $\prod_{i=1}^m N_i = N$, it follows that $\prod_{i=1}^m \phi(N_i) = \phi(N)$; see [38]. Subsequently,

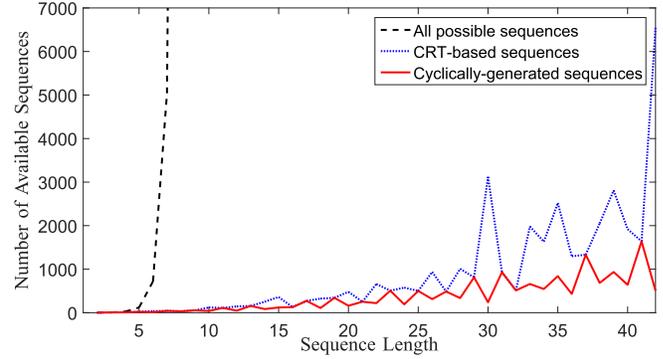


Fig. 4. Comparison between the number of CGS, \bar{N} , and $N!$.

$\prod_{i=1}^m N_i \phi(N_i) = N \phi(N)$. Finally, using (12), the number of available length N sequences that can be generated by the CRT-based scheme, \bar{N} , can be calculated as follows:

$$\bar{N} = N \phi(N) \sum_{k=1}^{q(N)} \sum_{j=0}^k (-1)^{k-j} \binom{k}{j} j^{q(N)}. \quad (13)$$

To draw insights from the expression in (13), in Figure 4, we compare \bar{N} with both the number of CGS and the total number of available sequences.

From Figure 4, it can be seen that for a sequence with length N that is either prime, e.g., $N = 29$, or a power of a single prime, e.g., $N = 27$, the number of CGS, i.e., $N \phi(N)$, is equal to that of the sequences generated by the proposed CRT-based scheme. This is because prime numbers and powers of a single prime yield only one coprime factor; i.e., for those numbers $q(N) = 1$ and $\left\{ \begin{smallmatrix} 1 \\ 1 \end{smallmatrix} \right\} = 1$. Hence, such values of N account for the trivial cases in which the input to the CRT sequence generator in Figure 3 is an element of the possible $N \phi(N)$ seed cyclic sequences. Figure 4 also shows that, apart from primes and powers of a single prime, the CRT-based scheme offers a larger number of sequences for RB assignment compared to their cyclic counterparts. Notice that the ratio between the number of CRT-based sequences in (13) and the CGS is given by (12), which accounts for the additional degrees of freedom introduced by combining the CGS using the CRT. From Figure 4, it can be seen that neither the number of CGS nor the number of CRT-based sequences is monotonically increasing with N . This is because neither $\phi(N)$, nor $q(N)$ is monotonic; e.g., $\phi(31) = 30 > \phi(32) = 16$, and $q(30) = 3 > q(31) = 1$. Finally, the figure shows that the number of sequences generated by the proposed CRT-based scheme is significantly smaller than the total number of available sequences; i.e., $N!$. For example, for a given sequence length $N = 30$, the number of available CGS and the number of available sequences by the CRT-based scheme are 240 and 3120, respectively. This number is significantly smaller than the $30! = 2.65 \times 10^{32}$ available sequences.

Remark 1: Although conventional CRT-based sequences will not be cyclic since they are obtained by combining multiple seeds, the proposed CRT-based scheme still yields cyclic sequences in the trivial case in which only a single seed sequence is used as an input to the CRT. In other words,

the set of sequences obtained by the CRT can be considered as a superset to the CGS set. Hence, when used for autonomous RB assignments, the performance of the sequences obtained by the CRT constitutes a lower bound on the number of RB assignment conflicts when compared to the performance of their cyclically-generated counterparts. \square

As illustrated in Figure 4, the proposed CRT-based scheme expands the set of RB assignment sequences, thereby offering an opportunity for performance enhancement. To better quantify this expansion, in the following section, we will derive upper and lower bounds on the number of CRT-based sequences and compare these bounds with the total number of sequences, $N!$.

E. Bounds on the Cardinality of CRT-Based Sequences

1) *Lower Bound:* We begin this section by evaluating the lower bound on the cardinality of the set of CRT-based sequences, which as per Remark 1, constitutes a superset of CGS. Hence, the cardinality of CRT-based sequences is lower bounded by the number of CGS; i.e., $N\phi(N)$. This bound is tight when N is either a prime or a power of a single prime; cf. Section IV-D.

2) *Upper Bound:* To obtain an upper bound on the cardinality of the set of CRT-based sequences, we note that the gain in this cardinality with respect to length- N CGS can be derived from (12). In particular, this equation shows that the gain is equivalent to the number of permutations of the coprime factors of N . In other words, this gain depends on all possible ordered sets of the seed cyclic sequences with coprime lengths that can be used as input to the CRT block in Figure 3; cf. Section IV-D. We are now ready to derive an upper bound on the gain in (12). This bound will then be used to compare the cardinality of length- N CRT-based sequences with the total number of available sequences of length N ; i.e., $N!$.

Before deriving the bound, we note that the gain in (12) can be divided into three components: 1) the Stirling number of the second kind which provides the cardinality of the coprime factors of N ; 2) the number of unique prime factors of N , $q(N)$; and 3) the number of possible ordered sets of the coprime factors of N . To obtain an upper bound on the gain in (12) due to the first and second components, we invoke an upper bound on the Stirling number of the second kind as a function of $q(N)$. This bound will then be modified to include the gain due to the third component. A bound on the Stirling number of the second kind is given by [39], [40]:

$$\left\{ \begin{matrix} q(N) \\ k \end{matrix} \right\} \leq \frac{1}{2} \binom{q(N)}{k} k^{(q(N)-k)}, \quad k \leq q(N). \quad (14)$$

The right hand side of (14) gives an upper bound on the number of unordered sets of the coprime factors of N . Now, to compute an upper bound on the corresponding number of ordered sets (cf. feature 4 in Section IV-D) both sides of (14) must be multiplied by $k!$ yielding

$$k! \left\{ \begin{matrix} q(N) \\ k \end{matrix} \right\} \leq \frac{1}{2} \frac{q(N)!}{(q(N)-k)!} k^{(q(N)-k)}. \quad (15)$$

To obtain better insight into the gain in the number of sequences in (12), we will further bound the right hand side of (15). Since $\frac{q(N)!}{(q(N)-k)!} \leq q^k(N)$, with equality when $k \in \{0, 1\}$, we have

$$k! \left\{ \begin{matrix} q(N) \\ k \end{matrix} \right\} \leq \frac{q^k(N)}{2} k^{(q(N)-k)}. \quad (16)$$

So far, we upper bounded the number of ordered sets of the coprime factors of N as a function of the summation index k in (12). Note that this index represents the number of seed sequences that are combined by the CRT. Using (16) to upper bound the left hand side of (12) yields:

$$\begin{aligned} \sum_{k=1}^{q(N)} \left\{ \begin{matrix} q(N) \\ k \end{matrix} \right\} k! &\leq \max_{k \leq q(N)} \frac{1}{2} q^{k+1}(N) k^{(q(N)-k)} \quad (17) \\ &\leq \frac{1}{2} (q(N))^{q(N)+1}, \quad (18) \end{aligned}$$

where (18) is obtained by substituting $k = q(N)$. Using (18), the upper bound on the cardinality of the CRT-based sequences of length N , can be expressed as:

$$\bar{N} \leq \frac{N}{2} \phi(N) (q(N))^{q(N)+1}. \quad (19)$$

Now, to complete the derivation of an upper bound on (12), we need to bound $q(N)$. One possible bound can be given by the following Lemma:

Lemma 2: For any value of N , $q(N)$ can be upper bounded by $\lceil \log_{10}(N) \rceil + 1$.

Proof: See the Appendix A. \blacksquare

Substituting this upper bound in (19) yields:

$$\bar{N} \leq \frac{N}{2} \phi(N) (\lceil \log_{10}(N) \rceil + 1)^{\lceil \log_{10}(N) \rceil + 2}. \quad (20)$$

Note that the upper bound in (20) is still significantly lower than the total number of N -length sequences; i.e., $N! \approx \sqrt{2\pi N} \left(\frac{N}{e}\right)^N$. This is because, the dominant multiplicative factor of $N!$ is N^N , which is a monotonically increasing function of N . On the other hand, the dominant multiplicative factor of (20) is a function of $\log_{10}(N)$, which is much less than N . Hence, the right hand side of (20) is much less than $N!$. Despite this significant reduction in the number of available sequences, (20) indicates that the proposed CRT-based scheme offers an enrichment to the set of easily-generated sequences that can be used by the RTs for RB assignments. In particular, this scheme offers a multiplicative gain over its cyclic counterpart; i.e., a gain that is upper bounded by $\frac{1}{2} (\lceil \log_{10}(N) \rceil + 1)^{\lceil \log_{10}(N) \rceil + 2}$. For example, when $N = 30$, the numbers of the CGS, the CRT-based sequences, and the sequences of length $N = 30$ are 240, 9720, 2.65×10^{32} , respectively. Hence, the proposed scheme offers an opportunity for performance enhancement with much less sequences than $N!$ while maintaining the ease of sequence generation property.

Finally, notice that the bound on \bar{N} in (20) was obtained by using the upper bound on $q(N)$ derived in Appendix A. However, the gap between this bound and $q(N)$ is relatively small only for small and medium values of N ; cf. the discussion in Appendix B. For asymptotically large values of N ,

TABLE I
VALUES OF THE BOUND IN LEMMA 2 AND THAT GIVEN
IN (21) FOR DIFFERENT VALUES OF N

N	$10^{\lceil \log_{10}(N) \rceil}$	Bound in Lemma 2	Bound in (21)
$1 < N \leq 10$	10	2	∞
$10^2 < N \leq 10^3$	10^3	4	19.47
$10^5 < N \leq 10^6$	10^6	7	17.62
$10^8 < N \leq 10^9$	10^9	10	19.32
$10^{19} < N \leq 10^{20}$	10^{20}	21	27.19
$10^{49} < N \leq 10^{50}$	10^{50}	51	46.75

a tighter bound on $q(N)$ is given in [40]. In particular, for asymptotically large values of N :

$$q(N) \leq \frac{\log_{10}(N)}{\log_{10}(\log_{10}(N))} (1 + \mathcal{O}(\log_{10}^{-1}(\log_{10}(N))))). \quad (21)$$

However, this bound is loose for small and medium values of N . To illustrate, we tabulated both, the bound in Lemma 2 and the bound in (21), for $N \leq 10^{50}$ in Table I. Note that for simplicity, the bounds in Lemma 2 and in (21) are tabulated only for selected values of $N = 10^k$ where $k \in \{1, 3, 6, 9, 20, 50\}$ and the constant of the $\mathcal{O}(\cdot)$ is set to 1.

From Table I, it can be seen that the bound in Lemma 2 offers a smaller gap for small and medium values of N . However, for asymptotically large values of N , the bound in (21) is tighter than the one in Lemma 2. Since the sequences' lengths in practical relay-assisted scenarios are prospected to be less than 10^9 , we used the bound in Lemma 2 when evaluating (20). Next, we will evaluate the complexity of sequence generation by the proposed CRT-based scheme.

F. Computational Complexity of the CRT-Based Scheme

In this section, we begin by noting that three components contribute to the complexity of the proposed CRT-based RB assignment scheme, namely, association, search, and generation. The association and search are performed once prior to system startup and thus add negligible computational complexity. Generation, however, is performed at each RT and thus analyzing its computational complexity is essential. Next, we will discuss the three components in detail.

1) *Association*: Prior to system startup, a one-to-one mapping function is selected between the set of RBs and the sequence elements. We note that in the absence of CSIs, any one-to-one mapping yields the same average performance. Hence, the mapping function can be chosen arbitrarily, thereby adding negligible complexity.

2) *Search*: A metric-based search is performed to obtain a look-up table of CRT-based sequences with favorable performance. The computational complexity of this search depends on the underlying metric; e.g., exhaustive search or greedy. However, this search is performed only once prior to system startup and the resulting look-up table is stored at the BS. Hence, it does not add any computational burden on the RTs.

3) *Generation*: Before assigning its RBs, each RT uses the CRT to generate a sequence based on the parameters obtained from the BS; i.e., the group generators and the cyclic shifts of the seed cyclic sequences. Assessing the computational complexity of generating this CRT-based sequence requires calculating the computational complexity of two operations: 1) the generation of the elements of the seed cyclic sequences that will be combined by the CRT; and 2) the generation of the sequence elements using the CRT-based combining scheme. We will begin by evaluating the computational complexity of the first operation. Without loss of generality, we will focus on the sequences generated by the ACG structure; cf. Section IV-C. Using this structure, each element of the i -th sequence of length N_i can be obtained by performing a single addition of the group generator g_i modulo N_i as shown in (7). Hence, obtaining a cyclic sequence of length N_i involves N_i additions and N_i divisions. Now, feature 1 of the proposed CRT-based scheme asserts that combining m seed cyclic sequences yields a CRT-based sequence of length $N = \prod_{i=1}^m N_i$; cf. Section IV-D. The total number of elements of these seed sequences is $L_t = \sum_{i=1}^m N_i$. Since $\sum_{i=1}^m N_i \leq \prod_{i=1}^m N_i$, it follows that $L_t \leq N$; $L_t = N \iff m = 1$. Hence, obtaining the elements of the seed cyclic sequences of a CRT-based sequence of length N requires at most N additions and N divisions. For the division operation, we note that the computational complexity of dividing two numbers of j bits each requires $\mathcal{O}((\log_2(j))^2)$ additions [41]. Since there are N divisions, the computational complexity of generating the seed sequences is bounded as follows:

$$\text{Number of additions} \leq \mathcal{O}(N(1 + (\log_2 N)^2)). \quad (22)$$

Next, we will evaluate the computational complexity of combining the seed cyclic sequences.

Feature 3 requires pairwise coprimeness of the lengths of the seed sequences. This coprimeness sets an upper limit on the number of seed sequences, m , to be equal to the number of unique prime factors of the length of the CRT-based sequence; i.e., $1 \leq m \leq q(N)$. In other words, generating an element of a CRT-based sequence involves solving at most $m = q(N)$ congruences; cf. (1). The computational complexity of this operation can be obtained by analyzing the complexities of evaluating the components of (6). In particular, obtaining an element of a CRT-based sequence using (6) involves calculating M_k and x_k , $k = 1, \dots, m$. First, $M_k = (\prod_{j=1}^m N_j) / N_k$, and thus obtaining the values of M_k , $k = 1, \dots, m$ involves m multiplications to calculate $\prod_{j=1}^m N_j$ and m divisions; one for each value of M . Second, to obtain x_k , $k = 1, \dots, m$ one needs to solve (2) using the Euclidean algorithm. In this algorithm, the value of x_k is obtained by sequentially dividing the value of N_k until the remainder is zero. Hence, the complexity of this algorithm can be upper bounded by $\frac{N_k}{2}$ divisions. From (6), the solution n on the left hand side that is provided by the CRT for the linear set of congruences described in (1) involves the addition of m terms; i.e., $\sum_{k=1}^m a_k M_k x_k$. Hence, it can be concluded that the computational complexity of combining m seed cyclic sequences by the CRT to obtain a sequence of N elements can be upper bounded by $N(m + \frac{1}{2} \sum_{k=1}^m N_k)$ divisions, $3mN$ multiplica-

tions and mN additions, where $m \leq (N)$. Combining the two operations necessary to obtain a CRT-based sequence of length N , the total complexity of the proposed CRT-based scheme can be bounded by $N(m + 1 + \frac{1}{2} \sum_{k=1}^m N_k)$ divisions, $3mN$ multiplications and $N(m+1)$ additions. Since $\sum_{k=1}^m N_k \leq N$ and the complexity of a multiplication is identical to that of a division [41], it follows that the computational complexity of the CRT-based scheme can be upper bounded by

$$\text{Number of additions} \leq \mathcal{O}((0.5N^2 + N(4m + 1))(\log_2 N)^2 + N(m + 1)). \quad (23)$$

Note that the complexity of generating a CRT-based sequence is generally affordable by the user terminals. This is because of two reasons. First, the CRT-based scheme has a low order polynomial complexity. Second, generating a CRT-based RB assignment sequence is performed only when the user terminal first switches to the relaying mode or when it is necessary for the RT to update its sequence. Next, we will provide numerical examples to show the performance gain obtained by the CRT-based scheme in the context of autonomous RB assignments.

V. SIMULATION RESULTS

In this section, we begin by comparing the performance of the RB assignment sequences obtained by the proposed CRT-based scheme, the multiplicative cyclic group (MCG) structure proposed in [25], the shuffled PN sequences discussed in [29]–[32], a soft fractional frequency reuse (SFFR) scheme similar to that in [42], and the uniformly-distributed random assignment scheme developed in [16]. We will begin by discussing the simulation environment.

A. Simulation Environment

In this work, two simulation environments are considered. In the first environment, we numerically evaluate the performance of the proposed scheme in terms of hit occurrences; cf. Examples 1, 2, 3, 4, and 5. In the second environment, the implications due to the channel effects and the RTs' deployment are considered when evaluating the performance; cf. Examples 6 and 7. In all examples, for the MCG-based scheme in [25], the CGS are generated for $N = P - 1$, where P is a prime number; see [25]. In addition, the lengths of the seed cyclic sequences are selected such that they are relatively prime and their multiplication is equal to $P - 1$ to comply with the length requirement of the MCG structure. For ease of exposition, we will restrict attention to the case of $M = 3$ RTs and apply an exhaustive search based technique for the selection of the cyclic and the CRT-based sequences (except for Example 3 in which N is large). In Example 1, the interference incurred by the RTs is measured by the average number of hits over all possible load combinations for different relative loads $\frac{L}{N}$, where L is the number of assigned RBs; i.e., over all ordered triples $(\ell_1, \ell_2, \ell_3) \in \mathcal{L} \triangleq \{(\ell_1, \ell_2, \ell_3) | \sum_{i=1}^3 \ell_i = L, \ell_i \geq 0\}$, where ℓ_1, ℓ_2 , and ℓ_3 are the load levels of RTs 1, 2 and 3, respectively. Note that when each IoT device is assigned an RB, the load ℓ_i represents the number of devices

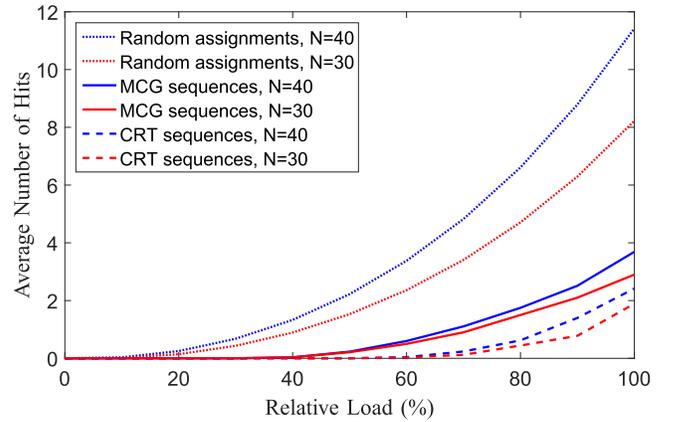


Fig. 5. Comparison between CRT-based sequences, CGS [25], and random assignments [16].

served by RT i and L becomes the number of IoT devices. In Examples 2 and 3, we measure the performance by the probability of hit occurrences, which can be calculated by dividing the average number of hits by L ; assuming all loading combinations are equiprobable. In Example 4, we consider an overloaded scenario, i.e., $100\% < \frac{L}{N} \leq 200\%$, and use the number of overutilized RBs as a metric; an RB is overutilized if assigned by 3 RTs simultaneously thus resulting in high interference. In Example 5, we depict the performance of the CRT-based scheme under a non-uniform loading condition. In particular, we evaluate the hits incurred by the RTs in a non-uniform loading scenario in which ℓ_1 and ℓ_2 are fixed while ℓ_3 increases from 0 until the system is fully loaded. In Examples 6 and 7, we evaluate the CRT-based scheme performance in a practical IoT system. In particular, we assume a fully loaded system, i.e., $\frac{L}{N} = 100\%$, and evaluate the aggregate data rate and the CDF of the successful RB assignments in an IoT system with 3 RTs.

B. Numerical Examples

Example 1: In this example, we compare the performance of the sequences obtained by the proposed CRT-based sequence generation scheme in Section II and the MCG scheme proposed in [25] when the number of RBs $N = 30$ and $N = 40$. The RTs apply the hit identification and avoidance (HIA) technique developed in [25]. In this technique, each RT is assumed to know the assignment sequences of the neighboring RTs and to be able to identify the RT with which it collided once a hit occurs. This is possible, for instance, if neighboring RTs use different modulation schemes. Using this information, the RTs update their assignment sequences to avoid the RBs that are already assigned by neighboring RTs. Note that the HIA technique cannot be applied to the randomly-generated sequences because they do not possess a specific structure.

From Figure 5, it can be seen that both, the CRT-based sequences and the cyclically-generated ones, outperform the randomly-generated sequences. For example, for a system with $N = 40$ RBs at a relative load, $\frac{L}{N} = 90\%$, the average number of hits resulting from using the the CRT-based, the cyclic,

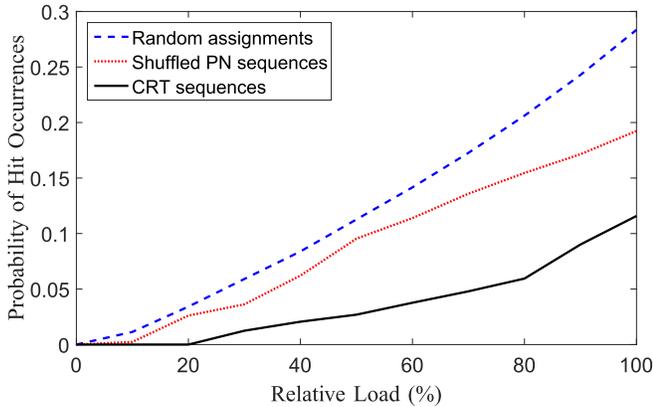


Fig. 6. Comparison between the random assignments of [16], the shuffled PN-based sequences, and the proposed CRT-based sequences for $N = 63$ RBs.

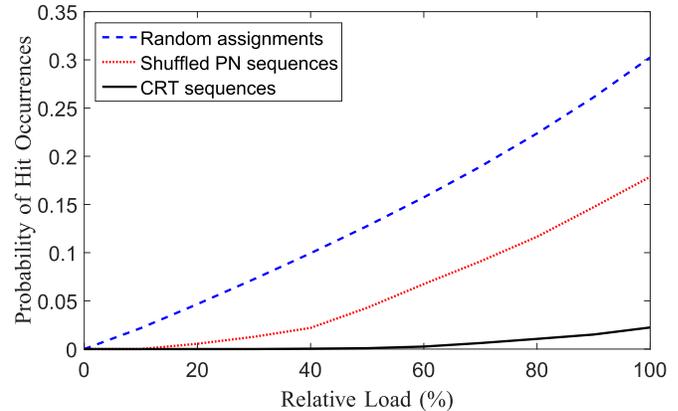


Fig. 7. Comparison between the random assignments of [16], the shuffled PN-based sequences, and the proposed CRT-based sequences for $N = 511$ RBs.

and the random sequences are 1.4, 2.5, and 8.8, respectively. We also observe from the figure that the performance gain of the proposed CRT-based scheme over that of the CGS of [25] increases with the number of RBs N at high relative loads; e.g., $\frac{L}{N} = 100\%$. This is mainly due to two reasons. First as N increases, the cardinality of the set of possible group generators increases, which increases the cardinality of the set of available seed cyclic sequences for the CRT-based scheme. Second, as N increases, the value of $q(N)$ tends to increase with N , cf. Figure 4, thereby increasing the set of coprime sequences' lengths that can be used as an input to the proposed CRT-based scheme. Finally, we note that the sequences obtained by the MCG structure of [25] constitutes a subset of the set of CRT-based sequences and thus their performance is upper bounded by that of the CRT-based ones, i.e., cf. Remark 1. \square

In the following two examples, we compare the performance of the proposed CRT-based sequences with that of the shuffled PN-based RB assignment scheme discussed in [29]–[32]. This sequence-based RB assignment scheme was specifically selected because its premises is similar to that of the CRT-based RB assignment scheme. In particular, the PN-based RB assignment scheme assumes the absence of the CSI and requires no coordination between neighboring devices. Hence, it is readily applicable in relay-assisted IoT systems with self-organizing RTs.

Example 2: In this example, we compare the performance of the CRT-based sequences, the shuffled PN-based sequences of [29], and the random RB assignments of [16]. Similar to Example 1, the number of RTs, $M = 3$, and they are all assumed to use the HIA technique developed in [25]. All cyclic shifts are set to zero and the number of RBs, $N = 63$. The length of the corresponding shuffled PN-based sequences is $2^k - 1$, $k = 6$.

From Figure 6, it can be seen that both the shuffled PN-based sequences and the CRT-based ones outperform the random RB assignments. For example, at a relative load, $\frac{L}{N} = 90\%$, the probability of hit occurrences of the CRT-based, the shuffled PN-based, and the random sequences are 0.09, 0.17, and 0.24, respectively. We also observe that

the proposed CRT-based sequences outperform their shuffled PN-based counterparts at all relative loads. This is mainly due to the fact that the cardinality of the set of CRT-based sequences, cf. Section IV-D, is much larger than that of the shuffled PN-based sequences, which is limited to $\frac{\phi(2^k - 1)}{6} = \frac{\phi(63)}{6} = 6$ [43]. \square

Example 3: In this example, we consider a setup similar to that of the previous example. We compare the performance of the CRT-based sequences with that of the shuffled PN-based ones when the number of RBs is large; i.e., $N = 511 = 2^k - 1$, $k = 9$. However, unlike the other examples, the RB assignment sequences are selected using a computationally-efficient greedy-based approach similar to that discussed in [26]. This is because obtaining the optimal RB assignment sequences through exhaustive search is computationally expensive for sequences with large numbers of RBs. The philosophy underlying the greedy-based approach is that the RBs located at the beginning of each sequence, referred to as the dominating RBs, are the most significant contributors to the number of hit occurrences. This is because these RBs will always be assigned by the RTs even when the system is lightly loaded. Hence, to reduce the number of hit occurrences, the sequences should be selected such that the overlap between their dominating RBs is minimal. In this example, the number of dominating RBs is set to 160.

From Figure 7, it can be seen that both, the shuffled PN-based sequences and the CRT-based ones, outperform the random RB assignments. For example, at a relative load, $\frac{L}{N} = 100\%$, the probability of hit occurrences of the CRT-based, the shuffled PN-based, and the random sequences are 0.022, 0.179, and 0.303, respectively. Figure 7 also shows that the computationally-efficient greedy-based approach, despite being potentially suboptimal, yields CRT-based sequences with performance significantly superior to that of the random RB assignments and the shuffled PN-based sequences. Hence, this greedy-based approach renders the CRT-based sequences appealing for IoT systems with large numbers of devices. \square

Example 4: In this example, we compare the performance of CRT-based sequences with that of CGS of [25] and random

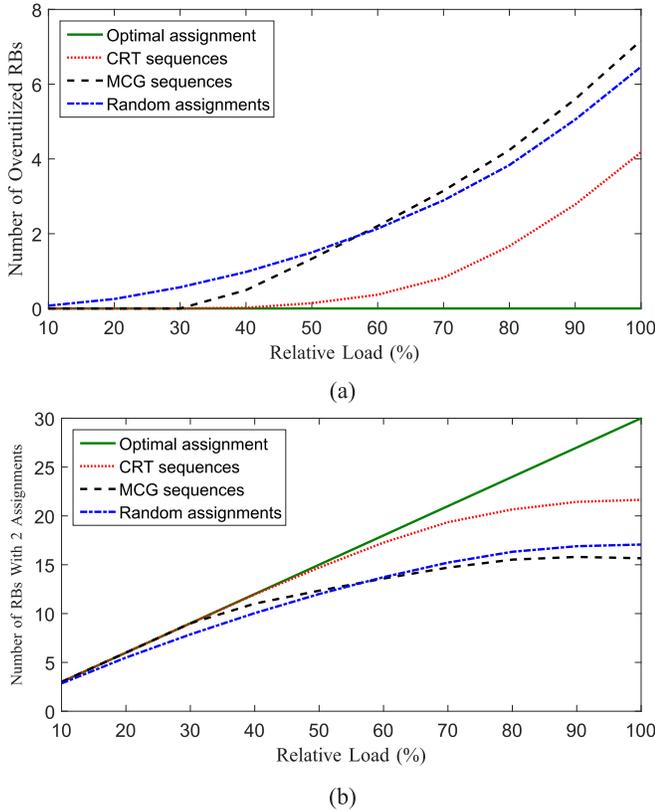


Fig. 8. Comparison between random RB assignment sequences [16] and the sequences obtained by the MCG structure in [25] and the proposed CRT-based scheme when $100\% < \frac{L}{N} \leq 200\%$.

sequences of [16] in an overloaded scenario, wherein the number of IoT devices is larger than the number of RBs; i.e., the relative load $100\% < \frac{L}{N} \leq 200\%$. This comparison is depicted in Figures 8a and 8b when $N = 30$, $M = 3$, and without HIA. In this example, the performance metric is based on reducing the overutilization of particular RBs by limiting the load level of each RB to 2; an RB is overutilized if assigned by 3 RTs simultaneously. Note that an optimal RB assignment case that assumes perfect centralized coordination between neighboring RTs is also depicted for reference.

In Figure 8a, we show the reduction in the number of overutilized RBs resulting from using the CRT-based sequences. For instance, for $\frac{L}{N} = 180\%$, the number of overutilized RBs when using the CRT-based, the cyclic, and the random sequences are 1.67, 4.24, and 3.84, respectively; i.e., the CRT-based sequences yield a 56% gain over their cyclic counterparts. Note that an optimal RB assignment will yield 0 overutilized RBs at any given load (this optimal assignment cannot be achieved with any autonomous scheme since no sequences will yield such performance).

In Figure 8b, we show the increase in the number of RBs assigned by 2 RTs, i.e., RBs that are not overutilized, when using the CRT-based sequences. For instance, for $\frac{L}{N} = 180\%$, the number of RBs assigned by 2 RTs for the three schemes are 20.66, 15.52, and 16.33, respectively; i.e., the CRT-based sequences yield a 26% gain over their cyclic counterparts. For this metric, an optimal RB assignment would result in a linear increase in the number of RBs assigned by two RTs with L . Finally, we note that the performance gain shown

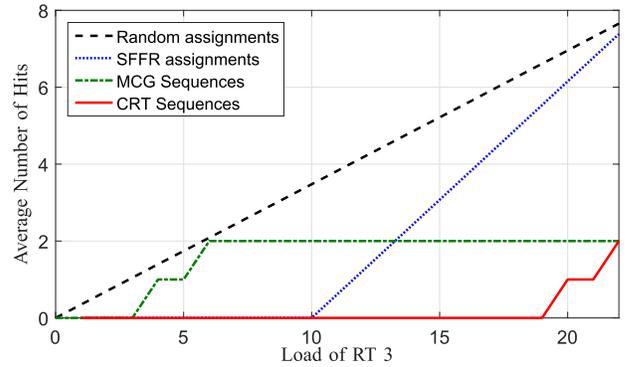


Fig. 9. A performance comparison between the CRT, the SFFR, the cyclic, and the random based RB assignment schemes for $N = 30$ RBs.

in Figures 8a and 8b is mainly due to the structure of the jointly designed CRT-based sequences that enables implicit coordination between RTs, thereby reducing the number of overutilized RBs even in overloaded scenarios. \square

Example 5: In this example, we consider a 3-RT scenario in which $\ell_1 = \ell_2 = 4$, while ℓ_3 increases from 0 to 22; i.e., RT 3 status changes from not loaded to lightly loaded and finally to heavily loaded. Hence, this scenario captures both uniform and non-uniform load distributions. We compare the performance of the proposed CRT-based scheme with that of an SFFR scheme similar to that in [42], which provides optimal performance in uniformly loaded scenarios. In particular, in this scheme, each RT is given a subset of the available RBs and only attempts to assign the remaining system RBs if all the ones within its given subset are already assigned. For completeness, we also depict the performance of the cyclic and the random based schemes.

From Figure 9, it can be seen that the CRT-based scheme outperforms the SFFR, the cyclic, and the random based ones. For example, when $\ell_3 = 21$, the average number of hits for the CRT-based scheme is 1, whereas those of the cyclic, the SFFR, and the random schemes are 2, 6.15, and 6.96, respectively; i.e., the CRT-based scheme offers a gain of at least 50% over the other schemes. It can also be seen from Figure 9 that no hits are observed by the SFFR scheme when $\ell_3 \leq 10$. This is because, each RT first attempts to assign from its given set of $\frac{N}{3}$ RBs before switching to random RB assignments. Hence, this scheme can provide optimal performance when either the system load is uniform or when the loads of each of the 3 RTs is less than or equal $\frac{N}{3}$. Figure 9 also shows that at $\ell_3 = 22$, i.e., a fully loaded system, both the CRT and the cyclic-based scheme result in only two hits. This is because both schemes use the HIA algorithm to avoid potential future hits; a feature that is not applicable to SFFR and random RB assignments. \square

Example 6: In this example, we evaluate the performance of the CRT-based sequences in a practical relay-assisted IoT system. In particular, we evaluate the system's aggregate data rate when the CRT-based sequences are used in an IoT system with $M = 3$ RTs and $N = 30$ RBs and compare it with the aggregate data rate when CGS of the same length are used. In the considered system, each RT assigns one RB to each incoming IoT device during a scheduling interval of 1 ms. In other words, in each scheduling interval, 3 RBs

corresponding to $M = 3$ RTs are assigned. Once RT i assigns an RB to an incoming IoT device, it increases the number of its assigned RBs, N_i , by one. At this point, either the assignment will be successful or a collision will occur. In case of a successful assignment, the assigned RB is retained and the RT proceeds during the next scheduling interval to assign the following RB in its sequence. In case of collision, the RT which was first to assign the RB retains it, whereas the other conflicting RT releases the RB and reduces its N_i by one at the end of its scheduling interval. The impact of a collision will be reflected on the interference observed by IoT devices. In particular, a collision on a given RB will result in an interference between the transmissions of their respective IoT devices for the duration of one scheduling interval. In this example, the system has an initial load of zero and the number of IoT devices is increased in a round Robin fashion until each of the RTs is loaded with 10 IoT devices. In that case, the system is fully loaded; i.e., $\frac{L}{N} = 100\%$.

The channel gains between the RTs and the IoT devices are generated using the line-of-sight Hotspot pathloss model [44] with frequency-flat Rayleigh fading and standard Gaussian noise spectral density of -174 dBm/Hz. The RB bandwidth, and the receiver noise figure are set to 180 KHz and 5 dB, respectively [44]. The transmit power, P_t , of all IoT devices is set to 0 dBm and the locations of these devices and the RTs are uniformly distributed within the cell boundaries. For a practical dense deployment scenario, IoT devices are assumed to be less than 300 m away from their assisting RTs and the maximum inter-RT distance is set to 100 m. The RTs implement the HIA algorithm of [25] and the performance is averaged over 10000 channel realizations. Using the aforementioned parameters, the aggregate data rate, R , at a given scheduling interval can be calculated using:

$$R = \sum_{i=1}^M \sum_{j=1}^{N_i} \log_2 \left(1 + \frac{|H_{l,i,B_j}|^2 P_t}{P_n + \sum_{m=1, m \neq i}^Q |H_{m,i,B_j}|^2 P_t} \right), \quad (24)$$

where P_n is the noise power, and B_j is the j -th RB assigned by RT i and is selected by the RT based on its sequence; i.e., the j -th entry in the RT's assignment sequence. In (24), the parameter Q is the total number of IoT devices, and H_{l,i,B_j} is the channel coefficient between RT i and IoT device l on RB B_j ; H_{l,i,B_j} is set to zero if B_j is not assigned by RT i to IoT device l .

As shown in Figure 10a, the aggregate data rate achieved by the CRT-based sequences constitutes an upper bound on that achieved by the CGS. This confirms the observation that the CGS constitute a special case of the CRT-based sequences. For example, at the eighth scheduling interval, the aggregate data rate achieved by the CRT-based sequences is 17% higher than the one achieved by the CGS. This performance gain is mainly due to the lower number of hit occurrences, i.e., the lower interference level, observed by the RTs due to the utilization of the CRT-based sequences. Figure 10a also illustrates that for the first few scheduling intervals, e.g., the third scheduling interval, both the CGS and the CRT-based sequences achieve the same bit rate. This is because for these scheduling intervals, all the RTs are lightly loaded, thereby no hits are

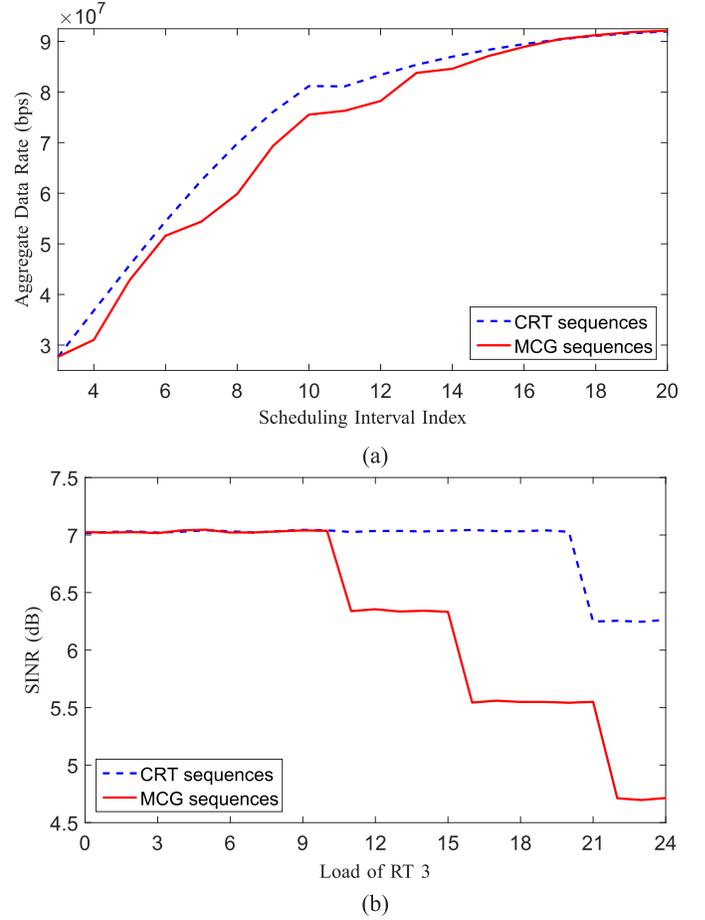


Fig. 10. Comparison between the aggregate data rates and the average SINR achieved by the cyclically-generated and the CRT-based RB assignment sequences for $N = 30$ RBs.

observed between the RTs. Finally, at high scheduling interval indices the two sequence structures ultimately achieve the same aggregate data rate. This is because, at these scheduling intervals, all the N RBs are assigned by the RTs and the data rate becomes flat.

In Figure 10b, we consider a 3-RT setup similar to that of Figure 10a. However, in this setup, $P_t = -50$ dBm and the loads of the first two RTs are set to 3, i.e., $\ell_1 = \ell_2 = 3$, while ℓ_3 increases from 0 to 24; i.e., until $L = 100\%$. The average SINR of the first two RTs is depicted in Figure 10b to show the effect of hit occurrences on the system performance. In particular, as ℓ_3 increases, hits occur between the RBs assigned by RT 3 and the ones already assigned by the first two RTs, resulting in a sudden drop in the average SINR. From this figure, it can also be seen that the performance of the CRT-based sequences outperform their cyclic counterparts. For instance, when $\ell_3 = 20$, the average SINR achieved by the CRT-based sequences and their cyclic counterparts is 7 dB and 5.5 dB, respectively. This is mainly due to the rich structure of the CRT-based sequences, which yields a lower number of hit occurrences. \square

Example 7: In this example, we evaluate the number of IoT devices with successful RB assignments versus the number of RB assignment attempts. The setup of this simulation is similar to that of the previous one. However, the total system load is

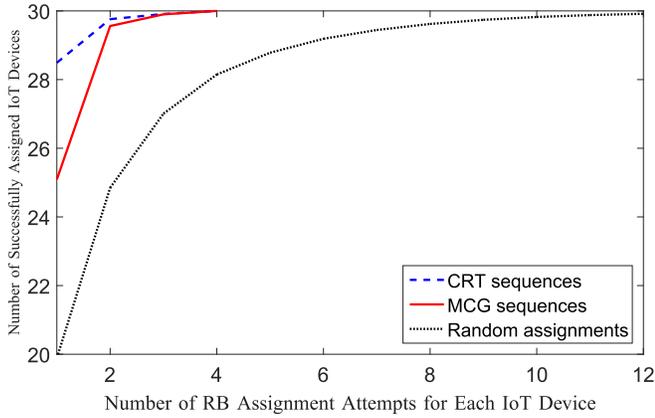


Fig. 11. CDF of the number of attempts needed for successful RB assignment for $N = 30$ RBs.

not uniformly distributed across the 3 RTs and the performance results are averaged over 10000 load distributions.

From Figure 11, it can be seen that the CRT-based sequences outperform both, the random and the CGS. For example, for the case of one RB assignment attempt per IoT device, i.e., no collisions, the numbers of successfully assigned IoT devices are 28.49, 25.09, and 19.9 when the RB assignments are done using the CRT-based, the cyclically-generated, and the random sequences, respectively. In other words, the CRT-based sequences offer a 13.5% gain over the CGS in the number of IoT devices not experiencing any collisions. This gain increases to 43.2% when compared to the performance of the random RB assignments. Note that the performance gain due to the CRT-based sequences is projected to increase for higher values of N . Figure 11 also shows that using the CRT, 99% of all IoT devices are successfully assigned with at most two RB assignment attempts, whereas in the case of cyclically-generated RB assignments the 99-th percentile is reached with three assignment attempts. Using the results reported in Figure 11, one can evaluate the power wastage incurred by collisions and the data transmission delay experienced by IoT devices; i.e., the duration between the first RB assignment and the successful one. This can be done by invoking the scheduling interval and the transmit power, P_t .

Denoting the transmission delay by d and the number of RB assignment attempts by a , it can be seen that $d = s(a - 1)$, where $s = 1$ ms is the scheduling interval duration. For example, a device with 3 assignment attempts will incur a delay, $d = 2$ ms. To evaluate the power wastage due to hits, denoted by P_w , one can use $P_w = P_t(a - 1)$. Using Figure 11, it can be concluded that 95% and 83.6% of the IoT devices incurred less than $P_w = 1$ mW power wastage and $d = 1$ ms delay when using the CRT-based sequences and the CGS, respectively. \square

VI. CONCLUSIONS

IoT devices are prospected to rely on nearby user devices to act as RTs and assist their communications with BSs. However, the underlying high mobility of the RTs and their coordination overhead suggests that they should autonomously perform their RB assignments to the IoT devices in the absence of CSIs. In this paper, we used the CRT to develop an autonomous RB assignment scheme that dispenses with the

CSI requirement. Each RT uses the CRT to combine multiple CGS. This combining process offers an additional degree of freedom for sequences generation, thereby yielding a rich set of RB assignment sequences. In addition, it was shown in this paper that the CRT-based sequences constitutes a superset of the set of CGS, and thus the performance of the CRT-based sequences is lower bounded by that of the CGS. Numerical results show that the sequences obtained by the proposed CRT-based scheme outperform the cyclically-generated ones, when used for autonomous RB assignment in relay-assisted IoT systems.

APPENDIX

In this section, we begin by proving the upper bound on $q(N)$ in Lemma 2. This is then followed by a discussion on the tightness of this bound for asymptotically large values of N .

A. Proof of Lemma 2: Upper Bound on $q(N)$

Let $\{p_1, p_2, \dots, p_n\}$ be a set of distinct prime numbers such that $p_1^{c_1} p_2^{c_2} \dots p_n^{c_n} \leq N$, where c_i is the repetitions of the i -th prime factor p_i . It follows that the cardinality of this set, i.e., n , is maximized when the prime numbers p_1, p_2, \dots, p_n are selected to be the smallest distinct primes, i.e., $p_1 = 2, p_2 = 3, p_3 = 5, \dots$, and the constants c_1, \dots, c_n are set to, $c_1 = \dots = c_n = 1$. We will refer to the maximum value of n as \bar{n} . Now, $q(N)$ is defined as the number of unique prime factors of N , cf. Section IV-D, and thereby is equal to the cardinality of the set of distinct primes $\{p_1, p_2, \dots, p_{q(N)}\}$ such that $p_1^{t_1} p_2^{t_2} \dots p_{q(N)}^{t_{q(N)}} = N$, where t_i is the repetitions of the i -th prime factor, p_i . Note that, for any N , $\{p_1, p_2, \dots, p_{q(N)}\} \subseteq \{p_1, p_2, \dots, p_n\}$; i.e., $q(N) \leq \bar{n}$. Hence, to obtain an upper bound on $q(N)$, it suffices to derive an upper bound on \bar{n} .

We begin by noting that for any $N > 1$ to have \bar{n} distinct primes, it is necessary that $N \geq p_1 p_2 \dots p_{\bar{n}}$. Now, apart from the first four primes (i.e., 2, 3, 5, and 7), all prime numbers are larger than 10. Hence, any increase in \bar{n} beyond four, will require increasing the value of N by a factor that is larger than 10. Hence, one possible bound on $\bar{n} > 4$, is given by $\lceil \log_{10}(N) \rceil + 1$ since it increases by at least 1 as N increases by a factor of 10 or more. However, we need to show that this bound holds for $\bar{n} \leq 4$. For N to have 1, 2, 3, and 4 distinct primes, it follows that N must be larger than or equal $p_1 = 2$, $p_1 p_2 = 6$, $p_1 p_2 p_3 = 30$, and $p_1 p_2 p_3 p_4 = 210$, respectively. For each of these four values of N , the corresponding values of $\lceil \log_{10}(N) \rceil + 1$ are 2, 2, 3, and 4, respectively; i.e., $\lceil \log_{10}(N) \rceil + 1 \geq \bar{n}$ for $\bar{n} \leq 4$. Hence, for any value of N , the value of \bar{n} , and subsequently the value of $q(N)$, can be both upper bounded by $\lceil \log_{10}(N) \rceil + 1$.

B. Discussion on the Tightness on the Bound on $q(N)$

As shown in the previous section, the bound in Lemma 2 is valid for any value of N . However, it becomes less tight as the value of N increases. This can be shown by considering the rate of increase of the upper bound with N and comparing it with that of the actual number of distinct prime numbers. Let N be an asymptotically large number, where $N = p_1 \dots p_{\bar{n}}$. For this value of N , the number of distinct primes $q(N) = \bar{n}$ and is upper bounded by $\lceil \log_{10}(N) \rceil + 1$. Let $p_{\bar{n}+1}$ be the

smallest prime following $p_{\bar{n}}$ and let $\hat{N} = p_1 \dots p_{\bar{n}+1}$. Note that $p_{\bar{n}+1} \gg 10$ for large values of N . Now, the contribution of this prime, i.e., $p_{\bar{n}+1}$, to $q(\hat{N})$ is one, i.e., $q(\hat{N}) = q(N) + 1$, whereas its contribution to the upper bound is $\log_{10}(p_{\bar{n}+1}) \gg 1$; i.e., $\lceil \log_{10}(\hat{N}) \rceil = \lceil \log_{10}(N) + \log_{10}(p_{\bar{n}+1}) \rceil$. Hence, the upper bound in Lemma 2 increases at a rate higher than that of $q(N)$ for asymptotically large values of N , thereby yielding a loose bound.

REFERENCES

- [1] J. Kim, J. Lee, J. Kim, and J. Yun, "M2M service platforms: Survey, issues, and enabling technologies," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 61–76, 1st Quart., 2014.
- [2] S.-Y. Lien and K.-C. Chen, "Massive access management for QoS guarantees in 3GPP machine-to-machine communications," *IEEE Commun. Lett.*, vol. 15, no. 3, pp. 311–313, Mar. 2011.
- [3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.
- [4] M. Hasan, E. Hossain, and D. Niyato, "Random access for machine-to-machine communication in LTE-advanced networks: Issues and approaches," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 86–93, Jun. 2013.
- [5] *Study on LTE Device to Device Proximity Services, Radio Aspects (Release 12)*, document 3GPP TR 36.843 version 12.0.1, Mar. 2014. [Online]. Available: http://www.3gpp.org/ftp/specs/archive/36_series/36.843
- [6] X. Lin, J. Andrews, A. Ghosh, and R. Ratasuk, "An overview of 3GPP device-to-device proximity services," *IEEE Commun. Mag.*, vol. 52, no. 4, pp. 40–48, Apr. 2014.
- [7] *Study on Scenarios and Requirements for Next Generation Access Technologies (Release 14)*, document 3GPP TR 38.913 version 14.0.0, Oct. 2016. [Online]. Available: http://www.3gpp.org/ftp/specs/archive/38_series/38.913
- [8] J. Schliez and A. Roessler, "Device to device communication in LTE," Rohde & Schwarz, Munich, Germany, White Paper IMA264, Sep. 2015.
- [9] P. N. Alevizos, E. A. Vlachos, and A. Bletsas. (Mar. 2017). "Inference-based distributed channel allocation in wireless sensor networks." [Online]. Available: <https://arxiv.org/abs/1703.06652>
- [10] T. Wang, G. B. Giannakis, and R. Wang, "Smart regenerative relays for link-adaptive cooperative communications," *IEEE Trans. Commun.*, vol. 56, no. 11, pp. 1950–1960, Nov. 2008.
- [11] W. Rhee and J. M. Cioffi, "Increase in capacity of multiuser OFDM system using dynamic subchannel allocation," in *Proc. IEEE VTC-Spring*, Tokyo, Japan, May 2000, pp. 1085–1089.
- [12] A. Behnad, X. Gao, and X. Wang, "Distributed resource allocation for multihop decode-and-forward relay systems," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4821–4826, Oct. 2015.
- [13] M. Hasan and E. Hossain, "Distributed resource allocation for relay-aided device-to-device communication under channel uncertainties: A stable matching approach," *IEEE Trans. Commun.*, vol. 63, no. 10, pp. 3882–3897, Oct. 2015.
- [14] Z. Chang, T. Ristaniemi, and Z. Niu, "Radio resource allocation for collaborative OFDMA relay networks with imperfect channel state information," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2824–2835, May 2014.
- [15] N. Abuzainab, W. Saad, C. S. Hong, and H. V. Poor, "Cognitive hierarchy theory for distributed resource allocation in the Internet of Things," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 7687–7702, Dec. 2017.
- [16] C. Koutsimanis and G. Fodor, "A dynamic resource allocation scheme for guaranteed bit rate services in OFDMA networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Beijing, China, May 2008, pp. 2524–2530.
- [17] A. Laya, C. Kalalal, F. Vazquez-Gallego, L. Alonso, and J. Alonso-Zarate, "Goodbye, ALOHA!" *IEEE Access*, vol. 4, pp. 2029–2044, 2016.
- [18] G. Margelis, R. Piechocki, D. Kaleshi, and P. Thomas, "Low throughput networks for the IoT: Lessons learned from industrial implementations," in *Proc. IEEE 2nd World Forum Internet Things (WF-IoT)*, Dec. 2015, pp. 181–186.
- [19] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, "Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios," *IEEE Wireless Commun.*, vol. 23, no. 5, pp. 60–67, Oct. 2016.
- [20] T. V. Nguyen, F. Baccelli, K. Zhu, S. Subramanian, and X. Wu, "A performance analysis of CSMA based broadcast protocol in VANETs," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2013, pp. 2805–2813.
- [21] Q. Ye and W. Zhuang, "Distributed and adaptive medium access control for Internet-of-Things-enabled mobile networks," *IEEE Internet Things J.*, vol. 4, no. 2, pp. 446–460, Apr. 2017.
- [22] J. Chen, Q. Yu, B. Chai, Y. Sun, Y. Fan, and X. Shen, "Dynamic channel assignment for wireless sensor networks: A regret matching based approach," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 1, pp. 95–106, Jan. 2015.
- [23] B. Karaoglu and W. Heinzelman, "Cooperative load balancing and dynamic channel allocation for cluster-based mobile ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 5, pp. 951–963, May 2015.
- [24] Y. Sun and J.-S. Qian, "Distributed channel fast allocation in cognitive cellular sensor networks," *Wireless Pers. Commun.*, vol. 96, no. 4, pp. 5931–5945, Oct. 2017.
- [25] Y. M. M. Fouad, R. H. Gohary, and H. Yanikomeroglu, "An autonomous resource block assignment scheme for OFDMA-based relay-assisted cellular networks," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 637–647, Feb. 2012.
- [26] Y. M. M. Fouad, R. H. Gohary, and H. Yanikomeroglu, "Number-theoretic sequence design for uncoordinated autonomous multiple access in relay-assisted machine-type communications," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 9018–9034, Oct. 2017.
- [27] J. A. Gallian, *Contemporary Abstract Algebra*. Boston, MA, USA: Houghton Mifflin, 1998.
- [28] A. Celik, R. M. Radaideh, F. S. Al-Qahtani, and M.-S. Alouini, "Resource allocation and interference management for D2D-enabled DL/UL decoupled Het-nets," *IEEE Access*, vol. 5, pp. 22735–22749, Oct. 2017.
- [29] C. F. Shih, A. E. Xhafa, and J. Zhou, "Practical frequency hopping sequence design for interference avoidance in 802.15. 4e TSCH networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 6494–6499.
- [30] S. Zoppi, H. M. Gürsu, M. Vilgelm, and W. Kellerer, "Reliable hopping sequence for highly interfered wireless sensor networks," in *Proc. IEEE Int. Symp. Local Metropolitan Area Netw. (LANMAN)*, Jun. 2017, pp. 1–7.
- [31] V. Kotsiou, G. Z. Papadopoulos, P. Chatzimisios, and F. Tholeyre, "Is local backlisting relevant in slow channel hopping low-power wireless networks?" in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.
- [32] S. Linder, K. Wiklundh, and L. Junholm, "Performance of frequency hopping systems with adjacent channel interference," in *Proc. IEEE Int. Symp. Electromagn. Compat. (EMC EUROPE)*, Sep. 2016, pp. 376–381.
- [33] L. Gilbert and J. Gilbert, *Elements of Modern Algebra*. Pacific Grove, CA, USA: Brooks/Cole, 2005.
- [34] C. Johnson, *Long Term Evolution in Bullets*. Northampton, U.K.: Johnsson, 2012.
- [35] C.-C. Tseng, H.-C. Wang, F.-C. Kuo, K.-C. Ting, H.-H. Chen, and G.-Y. Chen, "Delay and power consumption in LTE/LTE-A DRX mechanism with mixed short and long cycles," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1721–1734, Mar. 2016.
- [36] W. W. Adams and L. J. Goldstein, *Introduction to Number Theory*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1976.
- [37] C. A. Charalambides, *Enumerative Combinatorics*. Cambridge, U.K.: Cambridge Univ. Press, 1997.
- [38] G. Bi and Y. Zeng, *Transforms and Fast Algorithms for Signal Analysis and Representations*. New York, NY, USA: Springer, 2004.
- [39] B. C. Rennie and A. J. Dobson, "On stirling numbers of the second kind," *J. Combinat. Theory*, vol. 7, no. 2, pp. 116–121, Sep. 1969.
- [40] P. Ribenboim, *The Little Book of Bigger Primes*. New York, NY, USA: Springer, 2004.
- [41] J.-M. Müller, *Elementary Functions: Algorithms and Implementation*. New York, NY, USA: Springer, 1997.
- [42] N. Saquib, E. Hossain, and D. I. Kim, "Fractional frequency reuse for interference management in LTE-advanced hetnets," *IEEE Wireless Commun.*, vol. 20, no. 2, pp. 113–122, Apr. 2013.
- [43] C. D. Lee, Y. P. Huang, Y. Chang, and H. H. Chang, "Perfect Gaussian integer sequences of odd period $2^m - 1$," *IEEE Signal Process. Lett.*, vol. 22, no. 7, pp. 881–885, Jul. 2015.
- [44] *Further Advancements for E-UTRA Physical Layer Aspects (Release 9)*, document 3GPP TR 36.814 version 9.0.0, Mar. 2010. [Online]. Available: http://www.3gpp.org/ftp/specs/archive/36_series/36.814



Yaser M. M. Fouad received the B.Sc. degree in electronics and communications engineering from the Arab Academy for Science and Technology and Maritime Transport, Cairo, Egypt, in 2009, and the M.A.Sc. degree in electrical and computer engineering from Carleton University, Ottawa, Canada, in 2011, where he is currently pursuing the Ph.D. degree in electrical and computer engineering. He is currently a Wireless Research Scientist with Intel Labs.

His research interests include efficient resource allocation in relay-assisted cellular networks, applications of graph theory in the design of medium access control schemes for multihop networks, and power allocation techniques in multiple antenna and multiuser systems.



Ramy H. Gohary (S'02–M'06–SM'13) received the B.Eng. degree (Hons.) from Assiut University, Egypt, in 1996, the M.Sc. degree from Cairo University, Egypt, in 2000, and the Ph.D. degree from McMaster University, ON, Canada in 2006, all in electronics and communications engineering. He is currently an Assistant Professor with the Department of Systems and Computer Engineering, and a member of the College of Reviewers. He is on the Panel Review Board of the Ontario Centre of Excellence.

He has co-authored over 90 IEEE journal and conference papers, and co-supervised over 10 master's and Ph.D. students. He has co-invented five U.S. patents. His research interests include applications of embedded systems in machine-to-machine communications, Internet-of-Things, cross-layer design of wireless networks, analysis and design of MIMO wireless communication systems, applications of optimization and geometry in signal processing and communications, information theoretic aspects of multiuser communication systems, and the applications of iterative detection and decoding techniques in multiple antenna and multiuser systems.

Dr. Gohary has been named the "Best Professor" by the Carleton Student Engineering Society for 2015. He is a registered limited engineering licensee in the province of Ontario, Canada.



Halim Yanikomeroglu (F'17) was born in Giresun, Turkey, in 1968. He received the B.Sc. degree in electrical and electronics engineering from the Middle East Technical University, Ankara, Turkey, in 1990, and the M.A.Sc. degree in electrical engineering (now ECE) and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 1992 and 1998, respectively. From 1993 to 1994, he was with the R&D Group of Marconi Kominikasyon A.S., Ankara. Since 1998, he has been with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada, where he is currently a Full Professor. He spent the 2011–2012 academic year at the TOBB University of Economics and Technology, Ankara, as a Visiting Professor. His research interests include wireless technologies with a special emphasis on wireless networks including the non-terrestrial networks. In recent years, his research has been funded by Huawei, Telus, Allen Vanguard, Mapsted, Blackberry, Samsung, the Communications Research Centre of Canada, and DragonWave. This collaborative research resulted in over 30 patents (granted and applied). He has been involved in the organization of the IEEE Wireless Communications and Networking Conference (WCNC) from its inception, including serving as a Steering Committee Member and the Technical Program Chair or Co-Chair of WCNC 2004, Atlanta, GA, USA, WCNC 2008, Las Vegas, NV, USA, and WCNC 2014, Istanbul, Turkey. He was the General Co-Chair of the IEEE Vehicular Technology Conference (VTC) 2010, Ottawa, and the General Chair of the IEEE VTC 2017, Toronto. He has served on the Editorial Boards of the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS. He was the Chair of the IEEE Technical Committee on Personal Communications (now called Wireless Technical Committee). He is a Distinguished Lecturer of the IEEE Communications Society and a Distinguished Speaker for the IEEE Vehicular Technology Society. He was a recipient of the IEEE Ottawa Section Outstanding Educator Award in 2014, the Carleton University Faculty Graduate Mentoring Award in 2010, the Carleton University Graduate Students Association Excellence Award in Graduate Teaching in 2010, and the Carleton University Research Achievement Award in 2009 and 2018, respectively. He is a registered professional engineer in the province of Ontario, Canada.

He has co-authored over 90 IEEE journal and conference papers, and co-supervised over 10 master's and Ph.D. students. He has co-invented five U.S. patents. His research interests include applications of embedded systems in machine-to-machine communications, Internet-of-Things, cross-layer design of wireless networks, analysis and design of MIMO wireless communication systems, applications of optimization and geometry in signal processing and communications, information theoretic aspects of multiuser communication systems, and the applications of iterative detection and decoding techniques in multiple antenna and multiuser systems.

Dr. Gohary has been named the "Best Professor" by the Carleton Student Engineering Society for 2015. He is a registered limited engineering licensee in the province of Ontario, Canada.