Design and analysis of multi-carrier multiple access systems without feedback

Design and Analysis of Multi-carrier Multiple Access Systems without Feedback

G. T. Peeters and B. Van Houdt
University of Antwerp - IBBT
Middelheimlaan 1
B-2020 Antwerp, Belgium

***Abstract*—In this paper we study a multiple access system without feedback supporting multiple carriers for delay critical applications with a small loss tolerance. In such a system, users transmits R times within the next N timeslots, in order to improve their success probability. In an earlier work [1], we considered the same problem in a single carrier system and demonstrated that the distribution of properly designed user codes significantly improved the success probability over a random selection strategy. These user codes, that determine the R slots used for transmission, corresponded to so-called 2-$(N, R, 1)$ designs.**

Mainly motivated by DVB-RCS satellite systems, this paper considers a system with multiple carriers with the limitation that at any time, at most one slot, and thus carrier, can be used by a single user. We introduce two static and four dynamic slot assignment schemes and under some mild assumptions provide closed form formulas for the success probability in each of these systems. For the last dynamic scheme we will show how group divisible designs are the multi-carrier equivalent of the 2-$(N, R, 1)$ designs of the single carrier system and provide a simple procedure to construct these user codes.

Finally, we compare all of the proposed assignment schemes and show that the group divisible designs are superior, especially for large population sizes. Some engineering rules with respect to their usage are also provided. For two of the dynamic schemes we also compare two different strategies in case there are not enough user codes available for all users.

I. Introduction

The first multiple access systems without feedback considered [2], [3] demanded that all packets were transmitted successfully with probability one and consisted in designing user transmit patterns of length M such that irrespective of how many users are active, at least one of the transmissions is successful. Afterward, algorithms for random access systems without feedback that allowed for a predefined loss tolerance, e.g., of at most $\epsilon = 1\%$, were studied in [4], [5]. Their motivation stems from random access channels were the round-trip time, and thus the feedback, is (too) large for delay critical applications that can tolerate some losses. For example, Voice-over-IP can cope with some limited packet loss, whereas retransmitting lost packets would take too long, especially in two-way satellite networks. Since there is no feedback, packets are transmitted multiple times to improve their success probability. The most natural way to select the R transmission occasions out of the next N timeslots (for some N), is by selecting these R slots at random, as was done in [4], [5].

A better, more recent approach is to introduce user codes [1], to make this selection of R slots, meaning we exploit the limited population size by assigning each user its individual code. Furthermore, these codes were designed such that any two user codes share at most one slot. Hence, they correspond to binary constant weight codes with weight R and minimum distance $2(R-1)$. Moreover, for any 2 slots, there exists a user code using both slots. Hence, we considered sets of user codes of maximum cardinality such that every two slots are part of exactly one user code. In combinatorial design [6], such codes are known as 2-$(N, R, 1)$ designs (or $(2, N, R)$ Steiner systems). In [1], we focused on this type of user codes as it creates as little overlap between two user codes as possible, without having an extremely small number of codes (which would be the case if we allowed no overlap).

In principle, we consider synchronous systems with a maximum allowed delay of $2N$ timeslots, in the sense that we rely on frames of N timeslots; that is, the R slots have to be chosen from these N slots. Since a user has to wait at most N timeslots until a new frame starts, the maximum delay is limited by $2N$. However, the synchronous restriction can be circumvented as follows. Suppose a user code is represented by a bit vector of length N and weight R, where bit number i is set if the user must use slot i as one of his R slots. When a new packet becomes ready for transmission at the end of the k-th timeslot of a frame of N slots, it will change its original user code by moving the first k bits to the back of its user code. This *shifted* bit vector is subsequently used for the packet transmission and may commence in the very next slot (that is, slot $k+1$). In this way, we guarantee that any two packets still interfere in at most one slot, even though the transmissions are no longer synchronized to the start of a frame.

The work presented in [1] was limited to TDMA systems consisting of one carrier. However, several systems employ multiple carriers to increase their capacity, in terms of the number of users or the absolute amount of traffic. For example in DVB-RCS [7] some slots can be used in contention mode; a SAC message can be used to make bandwidth reservations, whereas TRF slots can be used to transmit data directly. As the employed frame structure is MF-TDMA, these slots can be found on a set of carrier frequencies, each of which is divided into timeslots.

One could define a simple extension from a one carrier system to an F carrier system, by using codes for NF

instead of N slots, and by mapping slot i to carrier $\lceil i/N \rceil$ and timeslot[1] $i \mod N$. This solution has several drawbacks. First, some users will receive a user code containing two or more simultaneous slots, which is impossible since users typically only possess one transmitter. Therefore, some users cannot transmit R times and are treated unfairly. A remedy would be to remove these problematic codes, thereby slightly reducing the maximum supported population size. This typically produces asymmetrical codes, where some slots belong to more user codes than others, which creates another type of unfairness (and may prohibit computing closed formulas for the loss probabilities). Second, the required 2-$(FN, R, 1)$ designs become harder to generate as FN becomes large, meaning these codes are harder to generate for increasing F values.

The contributions of this paper are the following. In Section II we introduce two static and four dynamic assignment schemes to select R slots out of the FN slots occurring in a frame. Except for the last dynamic scheme, they all rely on random selection and/or the 2-$(N, R, 1)$ designs of the single carrier system. In Section III we identify the class of combinatorial designs, known as group divisible designs, that are required to define *fair* user codes in an F carrier system. The codes are fair in the sense that all users experience exactly the same success probability. Moreover, we discuss a simple construction method for these large sets of users codes via finite affine and projective spaces. After paying some attention to the case where there are fewer users than user codes in Section IV, we provide closed formulas for the success probability for each of the proposed assignment schemes in Section V under some mild assumptions on the population size. Finally, Section VI compares the performance of each of the assignment schemes and provides some engineering rules, e.g., for selecting R the number of transmission attempts. For two of the dynamic assignment schemes we also consider and compare two different strategies in case there are not enough user codes available for all users.

There is some minor resemblance with frequency hopping spread spectrum (FHSS), a code division multiple access (CDMA) transmission technique used in Bluetooth, GSM, and the IEEE802.11 standard. FHSS divides each message into multiple smaller blocks, where a frequency is chosen for each block, based on some predetermined codes. Hence, FHSS provides a robust solution for time-varying multi-path fading and attempts to prevent eavesdropping and signal jamming [8], [9]. However, these systems are designed such that each user has its own reserved (virtual) channel, with little cross interference. Our solution differs in that we primarily address the contention access problem, by transmitting a message several times, typically on different carriers.

II. Approaches

In this section we introduce a variety of approaches for the C users to determine their R transmission slots in a frame consisting of N timeslots and F carriers, i.e., to select R out of the FN slots. Apart from random selections, some approaches also rely on the 2-$(N, R, 1)$ designs discussed in [1] for the single carrier system, i.e., $F = 1$. This set of user codes $S_{N,R}$ is a set of maximum cardinality $|S_{N,R}|$, such that for every two slots there is exactly one user code holding both slots. As there are only N slots and every code consists of R slots, there are exactly $N(N-1)/R(R-1)$ user codes as there are $N(N-1)/2$ choices for the two slots and each code holds $R(R-1)/2$ sets of two points. A known necessary condition for such designs to exist is $N = 1$ or $R \mod R(R-1)$ and a simple construction for $R \leq 5$ and $N \leq 85$ was provided in [1]. Thus, given the maximum delay, we will make use of the largest N value that meets this criteria, i.e., that satisfies $N = 1$ or $R \mod R(R-1)$.

A. Static assignments

The first class of approaches considered assigns the F carriers between the C users in a static manner. Meaning each user is assigned to a carrier, such that there are C/F users per carrier (assuming F divides C). Hence, a user always uses the same pre-assigned carrier for this R transmission attempts. Therefore, the system behaves as F independent single carrier systems with C/F users each. We will consider two static schemes:

1) We do not utilize user codes and each user selects R slots out of the N slots on his carrier at random.
2) We make use of the $|S_{N,R}| = N(N-1)/R(R-1)$ user codes discussed in [1] by assigning each of the $|S_{N,R}|$ user codes to one user on each frequency, meaning we can support up to $FN(N-1)/R(R-1)$ users with our codes. If the system contains more users, the additional users select R slots at random (as this was shown to be superior to reusing codes in a single carrier system).

Due to the results for a single carrier system in [1], the second scheme is superior to the first, which is considered mainly for comparison with some of the dynamic schemes.

B. Dynamic assignments

In the dynamic setup, we do not partition the users over the F carriers, meaning each user will transmit on any of the F carriers at some point in time. Four different schemes are considered:

1) We do not utilize user codes and each user selects R slots out of the FN slots on all of the F carriers at random by first selecting R different timeslots among the N and then by selecting a carrier out of the F carriers for each of the N timeslots. Hence, there are $\binom{N}{R}F^R$ possible choices for each user.
2) We make use of the $N(N-1)/R(R-1)$ user codes discussed in [1] by assigning each of these codes to one user, meaning we can only support up to $N(N-1)/R(R-1)$ users with our codes. The user code determines the R timeslots, while the carrier is chosen at random and all R attempts are made on the same chosen carrier.

[1] We will refer to the combination of carrier and time as slot, whereas a timeslot refers only to a specific time, i.e., containing F slots.

3) We use exactly the same solution as in the previous case, but for each of the R timeslots we select a carrier at random, meaning the R transmissions can be located at different carriers.
4) We use the set of user codes discussed in the next section. These codes are designed for multiple carrier systems and will consist of $F^2N(N-1)/R(R-1)$ user codes.

If the system contains more than $N(N-1)/R(R-1)$ users in the second or third scheme, we will either (a) reuse the codes by assigning the same code to different users, or (b) make use of the random selection of the first scheme for the additional users. Notice, in the dynamic assignment case, code reuse cannot be excluded without consideration as users with the same code might select a different carrier. Finally, in the exceptional case of having more than $F^2N(N-1)/R(R-1)$ users, we will also rely on random selection for the additional users for the last scheme.

III. CONSTRUCTION

In this section we introduce the codes used by the fourth dynamic assignment scheme. The idea is to generate a set of maximal cardinality such that for every two slots, not occurring at the same time, there exists exactly one user code using both slots, while none of the user codes hold two slots occurring at the same time. As there are $F^2N(N-1)/2$ ways to select two slots that do not occur at the same time and we have $R(R-1)/2$ sets of two slots per user code, we are looking for a set of $F^2N(N-1)/R(R-1)$ codes. These will be constructed using group divisible designs (GDDs) in case $N = 1$ or $R \mod R(R-1)$ (as in the single carrier case) and F is a prime power.

Before we can see how GDDs can provide these patterns; let us first recall a few definitions from [10] that are reformulated in out network setting. A GDD is based on an association scheme for υ slots, which is defined by:

Definition 1. *Given υ slots, a set of ordered pairs satisfying the following conditions is said to be an association scheme with g classes:*

1) *Any two slots are either each others 1st, 2nd, $\cdots$, or g-th associates.*
2) *Each slot α has n_i i-th associates, n_i being independent of α.*
3) *If any two slots α and β are i-th associates, then the number of slots that are both a j-th associate of α and a k-th associate of β, is $p^i_{j,k}$ and is independent of the pair of i-th associates α and β.*

The following example is the one we will use to design the user codes. Assume we have $\upsilon = FN$ slots and two slots occurring at the same time are 1st associates, while any two other slots are 2nd associates. Thus, $n_1 = F-1$ and $n_2 = F(N-1)$, while $p^1_{1,1} = F-2$, $p^1_{1,2} = p^1_{2,1} = 0$, $p^1_{2,2} = F(N-1)$, $p^2_{1,1} = 0$, $p^2_{1,2} = p^2_{2,1} = F-1$ and $p^2_{2,2} = F(N-2)$. Notice, an association scheme consisting of one class is nothing but the set of all pairs of slots.

Now, a partially balanced incomplete block design (PBIBD), which is a set of user codes on the υ slots, can be defined on an association scheme with g classes as follows.

Definition 2. *Given an association scheme with g classes, we have a PBIBD with g associate classes if the υ slots form b user codes of length R such that*

1) *Every slot occurs at most once in a user code.*
2) *Every slot occurs in exactly r user codes.*
3) *If two slots α and β are i-th associates, then they occur together in λ_i user codes, λ_i being independent of the particular pair of i-th associates α and β.*

Notice, if we take $g = 1$, $\lambda_1 = 1$, $\upsilon = N$ and $b = N(N-1)/R(R-1)$, we obtain balanced incomplete block design (BIBD) codes used for the single carrier case, as in [1]. For the multi-carrier case, we consider the association scheme of the earlier example with $\upsilon = FN$ and $g = 2$ and set $\lambda_1 = 0$ and $\lambda_2 = 1$, as this implies that no two simultaneous slots occur in a single user code, while all other pairs of slots are used exactly once by a user code. In this case, $b = F^2N(N-1)/R(R-1)$ and as a result $r = F(N-1)/(R-1)$. Finally, a specific PBIBD, a GDD, can be now be defined as:

Definition 3. *A PBIBD with two associate classes is said to be group divisible if there are $\upsilon = mn$ symbols and the symbols can be divided into m groups of n symbols each, such that any two symbols of the same group are first associates and two symbols from different groups are second associates.*

Hence, our running example is a GDD. There are several construction methods for GDDs in the literature. For completeness, we will now introduce a simple procedure to generate such a GDD from the existing codes for a single carrier (for which a simple construction method was given in [1]).

Instead of just assigning one single carrier and user code to a specific user, as in the second dynamic assignment scheme, we want to find a set of *translation* codes, with each codeword of the form $F_1F_2 \ldots F_R$ with $1 \leq F_i \leq F$. Such a code translates a single carrier user code, by assigning the F_i-th carrier to the i-transmission, for all R transmissions. Notice, by construction two slots occurring at the same time will never be part of the same user code. The second static assignment scheme of Section II would correspond to the F translation codes $\{11\ldots1, 22\ldots2, \ldots, FF\ldots F\}$. As we are aiming for $F^2N(N-1)/R(R-1)$ user codes, we need to find a set of F^2 translation codes to combine with the $N(N-1)/R(R-1)$ single carrier user codes. For this purpose, we rely on orthogonal arrays (OAs). OAs are defined as follows:

Definition 4. *A $k \times v$ matrix A with entries from a set of s (≥ 2) elements is called an orthogonal array of size v if any $t \times v$ submatrix of A contains all possible $t \times 1$ column vectors equally often, i.e., λ times. Such an array is denoted by (v, k, s, t).*

We are interested in OAs with parameters $(F^2, R, F, 2)$, which implies that $\lambda = 1$, as the F^2 columns represent the

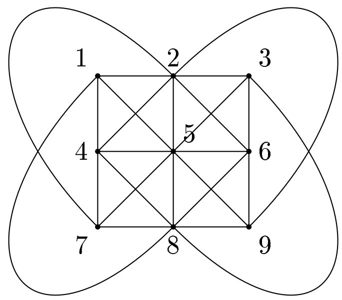


Fig. 1: The two dimensional affine space over GF(3)

F^2 translation codes $F_1 F_2 \ldots F_R$ with $1 \leq F_i \leq F$ and every two rows must hold all of the F^2 pairs exactly once. The latter property guarantees that any two slots that do not occur at the same time are part of exactly one user code as required.

It is also immediate to see that an $(F^2, R', F, 2)$ OA can be obtained from an $(F^2, R, F, 2)$ OA with $R' < R$, by removing $R - R'$ rows. Hence, we only need to construct a few OAs to cover most of the cases. We will now start by describing a simple construction method for $(F^2, F, F, 2)$ OAs with F a prime power, relying on two dimensional finite affine spaces. Next, we will extend them to $(F^2, F+1, F, 2)$ OAs by relating them with finite projective spaces. This two step approach will show its benefits in Section IV.

A finite affine space $AG(2, F)$ of dimension 2 over a finite field $K = GF(F)$, with $F = p^k$ for some p prime and $k \geq 1$, consists of a set V of F^2 points having coordinates of the form (x_1, x_2) with $x_i \in K$, for $i = 1, 2$. Through every two points in such a space, there exists exactly one line and every line holds exactly F points. This implies that there are exactly $C = \binom{F^2}{2} / \binom{F}{2} = F^2(F^2 - 1)/F(F - 1) = F^2 + F$ different lines in $AG(2, F)$. Further, every two lines intersect in at most one point.

Each line is characterized by the F points that it holds. We can produce a list of lines by iterating over all 2-element subsets of V and determining the coordinates of the remaining $F - 2$ points. Given two points $a = (a_1, a_2)$ and $b = (b_1, b_2)$, we first compute the difference vector $b - a$ as $(b_1 - a_1, b_2 - a_1)$, where $a_i, b_i \in K$ and $b_i - a_i \in K$ is determined via the subtraction operation of the finite field K. The F points of the line through a and b are then given by $\{a + c * (b - a) | c \in K\}$, where $*$ denotes the product in K. Clearly, setting $c = 0$ and 1 simply reproduces the points a and b.

For example, for $F = 3$ there are nine points $(0, 0), (0, 1)$, $(0, 2)$, $(1, 0), \ldots, (2, 2)$. If we relabel these as 1 to 9, the 12 lines can be written as $123, 456, 789$, $147, 258, 369$, $159, 267, 348$, $357, 249$ and 168 (see also Figure 1).

Let us now choose F parallel lines (which we will refer to as a spread, as each point is included in exactly one of the F lines), and arrange the points in a matrix such that each parallel line corresponds with one column. Since these F parallel lines hold all the F^2 points of the space and every other line holds F points that never share 2 points with one of the chosen parallel lines, all other lines must cross each column exactly once. Mapping a line on a translation code is then accomplished by letting the i-th symbol of the translation code correspond to the row number of the point where the line crosses the i-th column. These remaining F^2 lines of the affine space $AG(2, F)$ then form F^2 translation codes for a $(F^2, F, F, 2)$ OA.

1	2	3	1	3	2	3	2	1
1	2	3	2	1	3	2	1	3
1	2	3	3	2	1	1	3	2
1	1	1	2	2	2	3	3	3

TABLE I: Example of the F^2 translation codes for $F = 3$ and $R = 3$ and 4 stemming from the two dimensional affine and projective space over GF(3)

In our earlier example of $F = 3$, we can pick the three lines $123, 456$ and 789 as the spread defining the three columns. For each of the remaining nine lines we can now use the above mapping. For instance, for the line 168, we find that it uses the first point of 123, the third point of 456 and the second point of 789, thus its corresponding translation code is 132. If we repeat this for all the nine points we end up with the nine length 3 translation codes found in the first three rows of Table I. Notice, any two rows hold each of the nine combinations exactly once.

An $(F^2, F + 1, F, 2)$ OA can be constructed from a finite projective space $PG(2, F)$ of dimension 2 over a finite field $K = GF(F)$, with $F = p^k$ for some p prime and $k \geq 1$. The projective space $PG(2, F)$ can be constructed from the affine space $AG(2, F)$ by adding *points at infinity*. For our purpose, for each difference vector $b - a$, and thus each spread, except for the one corresponding to the chosen set of parallel lines, we add a single point to $AG(2, F)$ and we also add this point to each of the F lines that correspond to this difference vector, such that each line now carries $F + 1$ points. Hence, we have F^2 lines each holding $F + 1$ points and we have $F(F + 1)$ points. We now add one column to the F columns formed by the chosen spread and this extra column holds the F added points. Finally, we apply the same procedure by letting the i-th symbol of the translation code correspond to the row number of the point where the line (holding $F + 1$ points) crosses the i-th column, meaning the translation codes are now of length $F + 1$.

Continuing our earlier example, we add three points labeled $10, 11$ and 12. The point 10 is also added to the lines $147, 258$ and 369, as they all have the same direction (being $(0, 1)$), likewise point 11 is added to the lines $159, 267$ and 348 (direction $(1, 1)$) and point 12 to $357, 249$ and 168 (direction $(1, 2)$). The fourth column now reads $10, 11, 12$ and therefore the OA is obtained by adding 111222333 as a fourth row to the OA of the affine space as shown in Table I.

A summary of all OAs obtained from combining this construction with the property that an $(F^2, R, F, 2)$ can be constructed from a $(F^2, F + 1, F, 2)$, with $R < F + 1$, can be found in Table II. It contains also a few more OAs which can be found in the literature in case F is not a prime power.

F/R	3	4	5	6	7	8	9	10
2	OA							
3	OA^r	OA						
4	OA^r	OA^r	OA					
5	OA^r	OA^r	OA^r	OA				
6	$SR30$							
7	OA^r	OA^r	OA^r	OA^r	OA^r	OA		
8	OA^r	OA^r	OA^r	OA^r	OA^r	OA^r	OA	
9	OA^r	OA^r	OA^r	OA^r	OA^r	OA^r	OA^r	OA
10	$SR34$	$SR51$						

TABLE II: Overview of orthogonal arrays for $R, F \leq 10$. OA refers to orthogonal arrays which can be constructed as described in this paper. OA^r are also resolvable. $SR30$, $SR34$ and $SR51$ refer to the orthogonal arrays described in [11].

Furthermore, the *resolvable* OAs are also identified in this table. The definition and usefulness of resolvability will be discussed in the next Section.

IV. Arbitrary population sizes

For the assignment schemes that relied on user codes, we already mentioned that either random selection or code reuse will be used when there are more users in the system than user codes. In this section we address the issue of having less users C than user codes. In this case, we will select C user codes out of the total set, but a bad selection can result in degraded performance, as was already pointed out in [1]. Intuitively, we want each slot to be used by an equal number of users, as one can expect this to yield a good bandwidth usage. This also leads to closed form formulas for the success probability, keeping in mind that this does not always results in an optimal success probability.

There are several population sizes where a fair selection can be made, in the sense that each slot is used by equally many user codes. For this, we will rely on the resolvability of both the single carrier user codes of [1] and the OAs stemming from the affine spaces discussed earlier. A set of user codes is called fully resolvable when all the user codes can be partitioned into a number of subsets, where each subset contains every slot exactly once. A partial form of resolvability would be to demand that the codes can be partitioned into a number of sets, such that in each set every slot is used equally often. We will select the C codes by first assigning all the user codes in the first partition, followed by the codes in the second partition and so on until all users have received a code. The order in which the partitions are considered will have no impact on the performance.

The resolvability of single carrier user codes was already addressed in [1]. These $N(N-1)/R(R-1)$ user codes are typically only partially resolvable. More specifically, for $N = 1 \mod R(R-1)$ the user codes can be partitioned into $(N-1)/R(R-1)$ sets of size N each, such that each set uses every slot R times. For the case where $N = R \mod R(R-1)$ we had a single set of size N/R and $(N-R)/R(R-1)+1$ sets of size N.

The resolvability of OAs with $F = R$ can be seen directly from the affine geometry based construction method in Section III, that is, these designs are fully resolvable by associating a partition with each difference vector, meaning every partition holds F translation codes. Thus, as pointed out by Table II, OAs with $F = R$ and F a prime power are resolvable. One can also easily check that reduced OAs (i.e., with $R < F$) are also resolvable.

Due to these two results the user codes developed in the previous section are also partially resolvable and the partitions have the following sizes. For $N = 1 \mod R(R-1)$, we find that all the partitions have the same size being FN, where every slot is used R times. If $N = R \mod R(R-1)$, the first partition has size FN/R, while all the other partitions have size FN. In case the single carrier code is fully resolvable, so is the F carrier code and all the partitions have size FN/R and utilize each slot exactly once.

In general, the existence and resolvability of an $(F^2, R, F, 2)$ OA is not fully known, except when F is a prime power, in which case existence implies $R \leq F + 1$, and resolvability implies $R \leq F$[12].

V. Performance analysis

In this section we derive closed formulas for the success probability of all the dynamic assignment schemes introduced in Section II, as the static schemes were already covered in [1]. If the scheme relies on user codes and the number of users is less than the number of user codes, we assume that the number of user codes C required is such that $C = P_1 + \ldots + P_z$, for some integer z, where P_1 to P_z are the sizes of z different partitions introduced in the previous section. In other words, if the user codes are resolvable C is a multiple of FN/R, if not it suffices that it is a multiple of FN (or F plus a multiple of FN if $N = R \mod R(R-1)$). Hence, for these C values every slot is used equally often, allowing us to obtain closed formulas for the success probability.

These closed formulas even apply for any population size C, by introducing a simple rotation scheme. Suppose that C_k is the smallest value that meets the above criteria for which $C < C_k$ holds (and thus each slot is used equally often). We can apply this list of codes to a smaller size C population as follows. At the start of each frame, the active population selects a random subset of C out of C_k codes. If this selection is randomly chosen such that each code is still used at most once and each subset appears equally likely, we operate as if there were C_k users operating on the channel. As such the closed formulas still apply. A practical implementation for this permutation system could be based on a globally distributed random seed. After each frame, a new random number is generated simultaneously and identically by all users. Based on this number, a specific permutation is chosen.

A. Dynamic random selection

Let us first calculate the success probability for the random selection scheme, where first R timeslots are selected at random, followed by a random selection of the slots per timeslot. We first apply an inclusion-exclusion argument to determine the probability $p_{suc}^{(B_1)}(W)$ that a tagged users is successful provided that $W-1$ other users transmit, i.e.,

$$p_{suc}^{(B_1)}(W) = \sum_{i=1}^{R} (-1)^{i+1} \binom{R}{i} f^{(B_1)}(W,i), \qquad (1)$$

where $f^{(B_1)}(W,i)$ is the probability that none of the $W-1$ other users transmit in a particular set of i slots belonging to the user code of our tagged user. As $\binom{i}{j}\binom{N-i}{R-j}/\binom{N}{R}$ is the probability that a user selects j identical timeslots as the tagged user and for each of these j timeslots this user selects a different carrier with probability $(F-1)/F$, we have

$$f^{(B_1)}(W,i) = \left(\sum_{j=0}^{i} \frac{\binom{i}{j}\binom{N-i}{R-j}(F-1)^j}{\binom{N}{R}F^j} \right)^{W-1}. \qquad (2)$$

We further assume that each user generates packets according to a Poisson process with rate λ. As was done in [1], if multiple packets are generated by a single user in a length N interval, they are combined into one message that is transmitted R times in the next interval. Thus, with probability $p = 1 - e^{-\lambda N}$, a user will participate in a length N interval. The total load on the contention channel therefore matches $\rho = pC/(NF)$ (per slot). Hence, the overall success probability under Poisson arrivals matches

$$\begin{aligned} p_{suc}^{(B_1)} &= \sum_{W=1}^{C} \binom{C-1}{W-1} p^{W-1}(1-p)^{C-W} p_{suc}^{(B_1)}(W) \\ &= \sum_{W=1}^{C} \frac{W}{\rho N F} \binom{C}{W} p^{W}(1-p)^{C-W} p_{suc}^{(B_1)}(W). \end{aligned} \qquad (3)$$

When generating numerical results, we will typically fix p and consider various population sizes C resulting in different loads ρ.

B. Dynamic F carrier codes

Next, we consider the fourth dynamic scheme that uses the user codes of Section III. As discussed in Section IV we assume that the population size is such that each slot is used equally often. Notice, this means that C is always a multiple of FN/R and denote $C = kFN/R$. The analysis is identical to the random selection in the sense that Eqns. (1) and (3) still apply, except that $f^{(B_1)}(W,i)$ must be replaced by an appropriate function $f^{(B_4)}(W,i)$.

We can apply the same argument as in [1]. Due to the design of the selection algorithm, each *slot* is shared by exactly k user codes. Thus, if we tag a user, each slot belonging to its user code c will be shared by exactly $k-1$ other users. Also, the set of codes that contain one slot of c will be disjoint with a code that shares any other slot with c. Hence,

$$f^{(B_4)}(W,i) = \frac{\binom{(C-1)-i(k-1)}{W-1}}{\binom{C-1}{W-1}},$$

as the active users may not belong to the set of the $i(k-1)$ users that share a slot with the i particular slots. For $k=1$, this expression reduces to $p_{suc}^{B_4}(W) = 1$. If the population C exceeds $F^2N(N-1)/R(R-1)$ we will use random selection for the additional users and the success probabilities for this case can be calculated using the same approach as in [1].

C. Dynamic single carrier codes

In this section we simultaneously treat the second and third dynamic selection scheme. We start by assuming we have enough user codes, i.e., $C \le N(N-1)/R(R-1)$. Once more, it suffices to find an expression for $f^{(B_{2,3})}(W,i)$ and plugging this into Eqns. (1) and (3).

Analogue to the previous section we now denote $C = kN/R$, such that each *timeslot* is part of exactly k user codes, meaning $i(k-1)$ other users share one timeslot with the i particular timeslots of our tagged user. As opposed to the previous section, some of these users may also attempt transmission as they might select a different carrier when using the same timeslot. With probability $\binom{(C-1)-i(k-1)}{W-1-s}\binom{i(k-1)}{s}/\binom{C-1}{W-1}$, there are s users among the other $W-1$ active users that share a timeslot with our tagged user. As these users only share one timeslot, each one does not use one of the i particular slots with probability $(F-1)/F$, irrespective of whether the second or third dynamic scheme is used, hence:

$$f^{(B_{2,3})}(W,i) = \sum_{s=0}^{W-1} \frac{\binom{(C-1)-i(k-1)}{W-1-s}\binom{i(k-1)}{s}}{\binom{C-1}{W-1}} \left(\frac{F-1}{F}\right)^s.$$

Therefore, both schemes coincide for small population sizes. Moreover, if we have more than $|S_{N,R}| = N(N-1)/R(R-1)$ users and the additional users use random selection, both schemes still coincide and the success probability can be computed in a manner similar to [1] when we combine random users with user code based users.

Both scheme however no longer coincide for $C > |S_{N,R}|$ when we distribute the user codes multiple times, i.e., in case of code reuse. In this case, each code is used at least $\alpha = \lfloor C/|S_{N,R}| \rfloor$ times, while some codes may be used $\alpha+1$ times. Therefore, with probability $p_\alpha = \alpha((\alpha+1)|S_{N,R}| - C)/C$ the tagged user has a code that is used α times and its success probability can now be written as

$$\begin{aligned} p_{suc}^{(B_2/B_3)}(W) = \sum_{i=1}^{R} (-1)^{i+1} \binom{R}{i} \times \\ \left(p_\alpha f_\alpha^{(B_2/B_3)}(W,i) + (1-p_\alpha) f_{\alpha+1}^{(B_2/B_3)}(W,i) \right), \end{aligned}$$

where $f_j^{(B_2/B_3)}(W,i)$ is the probability that the i particular slots are not used by the other $W-1$ users provided that the tagged user's code is distributed j times among the user population, for $j=\alpha$ or $\alpha+1$. Thus, for the third dynamic scheme we find that $f_j^{(B_3)}(W,i)$ can be written as

$$\sum_{v=0}^{j-1} \sum_{s=0}^{W-1-v} \frac{\binom{(C-j)-i(k-j)}{W-1-s-v}\binom{i(k-j)}{s}\binom{j-1}{v}}{\binom{C-1}{W-1}} \left(\frac{F-1}{F}\right)^{s+vi},$$

for $j=\alpha$ and $\alpha+1$, as there are $j-1$ other users that use the same user code and thus share each of the i timeslots and $i(k-j)$ other users share a single timeslot. Thus, if v of

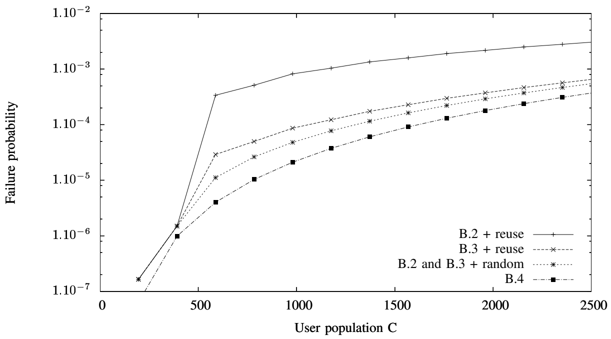


Fig. 2: $N = 49$, $R = 3$, $F = 4$, $p = 0.002$. The failure probability for the proposed dynamic schemes based on user codes.

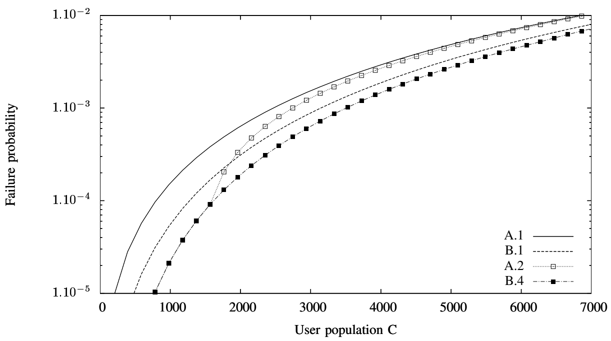


Fig. 3: $N = 49$, $R = 3$, $F = 4$, $p = 0.002$. Comparison between the fully random selection strategies and the best user code based schemes for both assigment policies.

the $j-1$ users are among the $W-1$ users, each does not coincide with probability $((F-1)/F)^i$, while each of the s users belonging to the group of $i(k-j)$ users does not use any of the i slots with probability $(F-1)/F$.

For the second dynamic scheme, each of the v users that make use of the same user code as our tagged user, does not use the set of i slots with probability $(F-1)/F$ as all the R transmission attempts occur on the same randomly chosen carrier, meaning the factor $((F-1)/F)^{s+vi}$ is replaced by $((F-1)/F)^{s+v}$ and $f_j^{(B_2)}(W,i)$ can be simplified to

$$\sum_{s'=0}^{W-1} \frac{\binom{(C-j)-i(k-j)}{W-1-s'}\binom{i(k-j)+j-1}{s'}}{\binom{C-1}{W-1}} \left(\frac{F-1}{F}\right)^{s'}.$$

VI. Numerical results

For brevity, we will refer to the single carrier user codes as BIBD codes, while the user codes developed in Section III are termed the GDD codes. Figure 2 compares the failure probability for the proposed dynamic schemes. Here, $N = 49$, $R = 3$, $F = 4$ and $p = 0.002$. If we focus on the second and third dynamic scheme, which we denote as $B.2$ and $B.3$, both making use of the $N(N-1)/R(R-1) = 392$ BIBD codes, we find that random selection is the best policy to support $C > 392$ users, as was the case in the single carrier case [1]. In case of code reuse, the policy where each of the R transmissions select a random carrier ($B.3$) is clearly superior to selecting the same random carrier for all R transmissions ($B.2$). This can be intuitively explained by looking at two users sharing the same code at the same time. The carrier selection of the first transmission is identical for both systems; if they both select the same carrier, policy $B.2$ implies R collisions, which does not necessarily occur with $B.3$. This performance penalty is severe, even if the number of users is only marginally larger than 392. The fourth dynamic policy $B.4$, based on the GDDs, is superior over the entire range of population sizes. In the above example, the $392F^2 = 6272$ codes suffice for all population sizes considered.

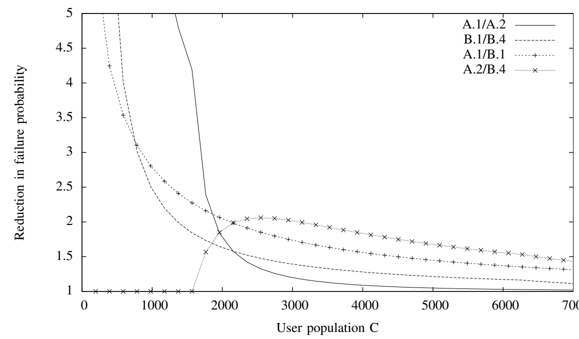
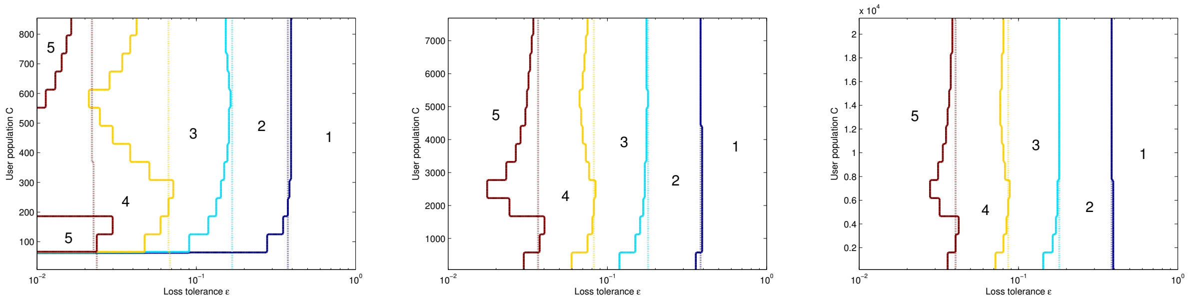


Fig. 4: $N = 49$, $R = 3$, $F = 4$, $p = 0.002$. The relative failure probability for a cross comparison between the random and user codes, and the dynamic and the random scheme.

To highlight the difference between the static and random assignments, we included Figure 3, which contains only the best user code based approaches and random selection strategies, for the same parameters as in Figure 2. For the code based approach, we see that the static and dynamic assignment schemes coincide, as long as the former does not run out of user codes, which happens $F = 4$ times faster for the static scheme (i.e., at $392F$ versus $392F^2$). This similarity can be explained by regarding the static scheme as a dynamic scheme, with translation codes which place each user on a single carrier (see also Section III). This contrasts with the fully random selection schemes, where static assignment induces a significant performance penalty. In either case, the user codes provide a significantly lower failure probability. Notice, the additional increase in the loss probability for the GDD scheme when C exceeds 6272 is hardly noticeable.

The relative failure probability for this example is shown in Figure 4, by cross comparing the random and user codes, as well as the dynamic and the random scheme. It further amplifies the large gains that can be achieved by a code-based approach in comparison with a random selection strategy. It also shows that the GDD codes are able to half the loss probability of the BIBD codes for certain population ranges.

Very similar conclusions can be drawn for other N, R, p and F values, where for $F = 1$ carrier the static and dynamic cases coincide. A higher number of carriers F typically increases the difference between the static and dynamic random selection strategy, as well as between the two dynamic BIBD policies



(a) 1 carrier (b) 3 carriers (c) 5 carriers

Fig. 5: Evolution of the optimal number of transmissions for $N = 61$, depending on the number of carriers. The solid lines define the regions for the user codes $(B.4)$, whereas the dashed lines represent the dynamic random selection $(B.1)$.

with code reuse. The dynamic BIBD schemes remains constant in terms of the number supported users, whereas this number grows linearly for the static BIBD scheme and quadratically for the GDD scheme. The latter thus outperforms all other schemes for any parameter setting, with a slightly lower performance gain compared to the dynamic random scheme if the number of users becomes large (i.e., several thousands). Although not shown in a figure, additional numerical examples indicate that for a given loss tolerance ϵ, about 20% more users can be supported by the GDD codes when compared to a dynamic random selection strategy for large ranges of p.

Finally, in Figure 5, the optimal R (i.e., which achieves the highest p, $1 \leq R \leq 5$) is shown for both the GDD scheme and the dynamic random scheme, for a large range of population sizes and loss tolerances. For 1 carrier, as indicated in [1], the optimal R depends heavily on both the loss tolerance and the population size C, where the latter was mainly influenced by the number of codes available. As the number of frequencies F becomes larger, the optimal R becomes less dependent on C, as is the case for the random selection strategy. In effect, the optimal number of transmissions R for both scenarios converge when further increasing the number of carriers, even though their corresponding loss probabilities do not. Notice that setting $F = 3$ and $R = 5$ is only of theoretical interest, as no OAs exist for these parameters.

VII. Conclusion

We showed how multi-carrier multiple access systems without feedback can benefit from specific user patterns, relying on single and multi-carrier user codes (BIBDs and GDDs). More specifically, we introduced two static and four dynamic assignment strategies, identified the multi-carrier equivalent of the BIBDs used in [1] for the single carrier case and provided a simple construction method based on finite affine and projective spaces. By means of closed formulas, we generated numerical results that indicated that the dynamic scheme is superior in case of random selection. When exploiting user codes however, both systems have similar success probabilities for fairly small population sizes, while for larger populations the GDDs clearly outperformed all other schemes and given a loss tolerance typically allows 20% more users in the system. We also studied the optimal number of transmission attempts as a function of the loss tolerance and user population size C and demonstrated that this value becomes far less dependent on C as the number of carriers increases.

References

[1] G. T. Peeters, R. Bocklandt, and B. Van Houdt, "Multiple access algorithms without feedback using combinatorial designs," *IEEE Transactions on Communications*, vol. 57, no. 9, Sep. 2009.

[2] J. Massey, "The capacity of the collision channel without feedback," in *Abstracts of Papers, IEEE Int. Symp. Info. Th.*, Les Arcs, France, June, 1982, p. 101.

[3] L. Bassalygo and M. Pinsker, "Restricted asynchronous multiple access," *Probl. Peredachi Informatsii*, vol. 19, pp. 92–96, Oct. 1983.

[4] G. Choudhury and S. Rappaport, "Diversity ALOHA–a random access scheme for satellite communications," *IEEE Trans. Commun.*, vol. 31, pp. 450–457, Mar 1983.

[5] S.-L. Su and V. Li, "Comments on diversity ALOHA," *IEEE Trans. Commun.*, vol. 32, pp. 1143–1145, Oct 1984.

[6] I. Anderson, *Combinatorial Designs: construction methods*. New York: John Wiley & sons, 1990.

[7] "Digital Video Broadcasting (DVB): Interaction channel for satellite distribution systems, European Telecommunication Standardization Institute (ETSI) EN 301 790 V1.3.1."

[8] J. Proakis, *Digital Communications*, 4th ed. McGraw-Hill Science/Engineering/Math, August 2000.

[9] M. Beuhrer, *Code division multiple access (CDMA)*. Morgan & Claypool Publishers, 2006.

[10] D. Raghavaro, *Constructions and Combinatorial problems in design of experiments*. Dover Publications, 1988.

[11] W. Clatworthy, *Tables of two-associate-class partially balanced designs*, ser. Appl. Math. Ser. National Bureau of Standards, 1973, vol. 63.

[12] A. Hedayat, N. J. A. Sloane, and J. Stufken, *Orthogonal Arrays: Theory and Application (Springer Series in Statistics)*. Springer-Verlag New York, Inc., 1999.