



**HAL**  
open science

# On the Performance of QPSK Modulation Over Downlink NOMA: From Error Probability Derivation to SDR-Based Validation

Jean-Romain Garnier, Alexis Fabre, Haïfa Farès, Rémi Bonnefoi

## ► To cite this version:

Jean-Romain Garnier, Alexis Fabre, Haïfa Farès, Rémi Bonnefoi. On the Performance of QPSK Modulation Over Downlink NOMA: From Error Probability Derivation to SDR-Based Validation. IEEE Access, 2020, 8, pp.66495-66507. 10.1109/ACCESS.2020.2983299 . hal-02796887

**HAL Id: hal-02796887**

**<https://univ-rennes.hal.science/hal-02796887>**

Submitted on 17 Jul 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Received January 30, 2020, accepted March 7, 2020, date of publication March 25, 2020, date of current version April 20, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2983299

# On the Performance of QPSK Modulation Over Downlink NOMA: From Error Probability Derivation to SDR-Based Validation

JEAN-ROMAIN GARNIER<sup>1</sup>, ALEXIS FABRE<sup>1</sup>, HAÏFA FARÈS<sup>1</sup>, AND RÈMI BONNEFOI<sup>2</sup>

<sup>1</sup>CentraleSupélec/IETR, CentraleSupélec Campus de Rennes, 35510 Cesson-Sévigné, France

<sup>2</sup>Edison Ways, 82000 Montauban, France

Corresponding author: Haïfa Farès (haifa.fares@centralesupelec.fr)

This work was supported by the Région Bretagne, France.

**ABSTRACT** In this paper, we study the performance of QPSK modulation in the context of multi-user downlink NOMA with a successive interference canceller (SIC) at the receiver side. The first objective is to evaluate the benefit of such a technique in terms of error probability, regardless of the number of involved users. Analytical derivations on its closed-form have been verified by both simulation and experimental validation. The article uses numerical simulations not only to corroborate the tightness of our theoretical expressions, but also to analyze the problem of power allocation in the two and three users cases. Finally, this paper provides an interplay between NOMA and software radio by building an experimental validation testbed.

**INDEX TERMS** Non-orthogonal multiple access (NOMA), power allocation, experimental validation.

## I. INTRODUCTION

Non-Orthogonal Multiple Access (NOMA) has been recently introduced as a promising multiple access technique for cellular networks [1], [2].

The benefits of NOMA and more precisely of Power Domain NOMA (PD-NOMA), which aims at employing different power levels to manage multiple access, has been widely studied considering the Shannon capacity as a performance indicator [3], [4]. Capacity analysis can be far from the reality and sometimes differ from practical implementation results [5].

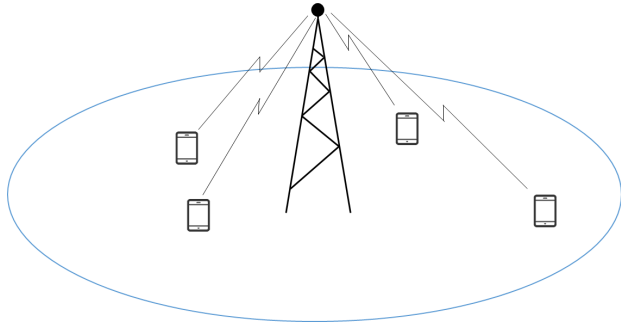
A more pragmatic approach lies in considering the bit error rate (BER) in order to evaluate communication techniques, as it has been done in [6], where the authors compute the closed-form expression for the error probability of uplink NOMA with QPSK (Quadrature Phase Shift Keying) constellation. Compared to [6] where the authors consider the uplink, in the present article, we consider the downlink communication. The BER of downlink NOMA for QPSK constellation has been expressed in a closed-form in [7] and [8]. However, the authors limited their study to the *two-users and three-users cases*, whereas we propose a closed form expression which is valid *regardless of the number of users*.

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wang<sup>1</sup>.

Besides, error probability-based analysis has the advantage of being experimentally verifiable, *e.g.*, using software defined platforms such as the Universal Software Radio Platform (USRP) and GNU Radio [9]. Some NOMA demonstrations have already been carried on with USRPs. In [10], [11], USRPs are used for assessing the over-the-air performance of NOMA for 5G network. In [12], the authors evaluate the performance of NOMA in Wi-Fi network. In these three articles, the analysis was limited to only two users contrary to our implementation which extended the analysis to a larger number of users. Another experimental demonstration of NOMA with more than two users is described in [13], where the authors study the performance of this multiple access technique for uplink communications which remains different from our testbed derived for downlink NOMA.

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

- 1) We first derive the theoretical expression of the error probability of a multi-user NOMA scenario where all the users are served using a QPSK modulation. The proposed formula is valid for any number of users.
- 2) Then, we use numerical simulations so as to obtain the BER in order to verify the proposed formula. It is then



**FIGURE 1.** The proposed system model where a base station serves  $N_u$  users.

used to analyze the problem of power allocation in the two and three user cases.

- 3) We finally use USRPs so as to build an experimental validation of the proposed theoretical analyses.

The rest of this article is organized as follows. The system model is introduced in Section II. In Section III, we derive a theoretical expression for the error probability in the network. Besides, the problem of power allocation is reported and solved in Section IV and the proposed derivations and interpretations are experimentally verified in Section V. Finally, conclusions are drawn in Section VI.

## II. SYSTEM MODEL

As illustrated in Figure 1, we assume the downlink of a base station that serves  $N_u$  users. We consider that PD-NOMA is used to separate the users, *i.e.*, the same band is employed for all of them and signals are segregated in the power domain. Besides, the further a user is, the more power it is assigned. The farthest user is noted  $N_u$  while the closest user is noted 1. We also note  $P_i$ , with  $i \in \llbracket 1; N_u \rrbracket$ , the power associated to the user  $i$ . Thus, it follows that  $\forall (i, j) \in \llbracket 1; N_u \rrbracket^2$ ,  $i < j \Rightarrow P_i < P_j$ .

For this study, we suppose that all the users in the network are served with the same modulation, a QPSK.<sup>1</sup> For each user  $i$ , we note  $x_i$  the signal transmitted to this user and  $\underline{b}_i \in \{1+j, 1-j, -1-j, -1+j\}$ , the information symbol sent to user  $i$ . With those notations,  $x_i = \sqrt{P_i} \underline{b}_i$  and the signal sent by the base station is  $\underline{x} = \sum_{i=1}^{N_u} x_i$ .

$g \in [0, 1]$  denotes the channel gain. The noise  $\eta_i$ , experienced by user  $i$ , is a zero-mean Gaussian random variable with a standard deviation of  $\Sigma$ . Thus, the signal received by user  $i$  is

$$y_i = g_i \underline{x} + \eta_i \tag{1}$$

At the receiver side, the user decodes the signal sequentially by decoding successively all symbols with greater

<sup>1</sup>QPSK modulation allows us to consider the BER as a performance metric. An analysis similar to the one conducted in this article allows to obtain the Symbol Error Rate (SER) for other Quadrature Amplitude Modulation (QAM). However, the link between SER and BER for higher order modulations remains difficult.

transmission power than their own, until they are able to extract their own symbol (as usually done for NOMA):

- 1) Decode the received symbol using a standard hard decision process associating the symbol to the nearest constellation point,
- 2) Subtract the decoded value multiplied by the user's power from the received symbol (taking the attenuation into account),
- 3) Restart until the user's signal is reached.

For example, the first symbol to be decoded by user  $i$  is the complex  $\underline{b}_{N_u}$  of user  $N_u$ :

$$\underline{b}_{N_u} = \begin{cases} 1+j & \text{if } \Re(\underline{y}) \geq 0 \text{ and } \Im(\underline{y}) \geq 0 \\ 1-j & \text{if } \Re(\underline{y}) \geq 0 \text{ and } \Im(\underline{y}) < 0 \\ -1-j & \text{if } \Re(\underline{y}) < 0 \text{ and } \Im(\underline{y}) < 0 \\ -1+j & \text{if } \Re(\underline{y}) < 0 \text{ and } \Im(\underline{y}) \geq 0 \end{cases} \tag{2}$$

This value is then ponderated by the power attributed to user  $N_u$  and by the gain  $g_i$ :

$$\dot{y}_{N_u} = \underline{b}_{N_u} g_i \sqrt{P_{N_u}} \tag{3}$$

We then subtract this value from  $y$ :

$$\dot{y}_{N_u-1} = \underline{y} - \dot{y}_{N_u} \tag{4}$$

and continue the decoding with  $\dot{y}_{N_u-1}$  in place of  $\underline{y}$ .

These steps are repeated  $N_u - i$  times, until  $\underline{b}_i$  is decoded. This algorithm is easy to implement, and is provided in appendix VI.

Please note that the decoding scheme presented here requires user  $i$  to know the transmit power of all the users  $j = i + 1, \dots, N_u$ . This information can be acquired by means of the pilot symbols transmitted by the base station.

In this paper, only the real part of the symbols will be studied since, for such a modulation, the analysis for the imaginary part is exactly similar to that of the real part.

To refer to the real part of complex values, we only need to remove the underline as follows:  $\Re(\underline{\eta}) = \eta$ ,  $\Re(\underline{b}_i) = b_i \dots$  We also note  $\Re(x_i) = r_i$  and  $\sigma$  the standard deviation of the real part of the noise, such that  $\sigma = \frac{\Sigma}{\sqrt{2}}$ . Moreover, power allocated to the real part of the symbol is  $P_i = \frac{P_i}{2}$ .

Furthermore,  $\llbracket M; N \rrbracket$  denotes the interval of integers between  $M$  and  $N$ . For all sets  $A$  and  $B$ ,  $\bar{A}$  denotes the complement of  $A$ ,  $A \cap B$  denotes the intersection of  $A$  and  $B$ ,  $A \cup B$  the union and  $A \oplus B = (A \cap \bar{B}) \cup (\bar{A} \cap B)$  is the symmetric difference of these two sets.

For the rest of this paper, we consider that we study the error probability from the point of view of a user, who observes  $\eta$  to be the real part of the complex zero-mean gaussian noise, and  $g$  the attenuation.

We note  $P_{e_i}$  the probability that the decoded value for the user  $i$  is wrong:  $P_{e_i} = P(\bar{r}_i = -r_i)$ .

We define  $\epsilon_n$  as the set of indices for which a decoding error has been made for the steps before

$n$ :  $\epsilon_n = \{i \in \llbracket n + 1; N_u \rrbracket \mid \bar{r}_i = -r_i\}$ . We also note  $\bar{\epsilon}_n$  the complementary set of  $\epsilon_n$ :  $\bar{\epsilon}_n = \llbracket n + 1; N_u \rrbracket \setminus \epsilon_n = \{i \in \llbracket n + 1; N_u \rrbracket \mid \bar{r}_i = r_i\}$ .

Likewise,  $S$  is defined as the set of  $\llbracket 1; N_u \rrbracket$  containing the indices of users for which the encoded bit is equal to 1:  $S = \{i \mid r_i > 0\}$ .  $\bar{S}$  is the complementary set of  $S$ :  $\bar{S} = \llbracket 1; N_u \rrbracket \setminus S = \{i \in \llbracket 1; N_u \rrbracket \mid r_i \leq 0\}$ .

We also define  $\epsilon_n(k)$  such as  $\forall k \in \llbracket n + 1; N \rrbracket$ ,  $\epsilon_n(k) = \epsilon_n \cap \llbracket k + 1; N \rrbracket$ .

Withal,  $I(\epsilon_n, S)$  is defined as the list of indices greater than  $n$  for which the decoded value is negative and it is obtained as  $I(\epsilon_n, S) = (\epsilon_n \cup \{n\}) \oplus S$ . It follows that  $I(\bar{\epsilon}_n, \bar{S}) = (\bar{\epsilon}_n \cup \{n\}) \oplus \bar{S} = \llbracket n; N_u \rrbracket \setminus S$ .

### III. BIT ERROR PROBABILITY OF MULTI-USER NOMA

We present a formula for the bit error probability in a  $N_u$ -user problem with a known attenuation for each user, considering additive white Gaussian noise.

To the best of our knowledge, this formula has been proved for 2 or 3 users [6], [7], but has not been expanded to 4 or more users. We derive a new formula viable regardless of the number of users, and verify that it is identical to that given for the two user case.

In order to derive the error probability, without loss of generality, we only consider the real part of the transmitted symbols. This assumption is particularly pertinent because of the independence between real and imaginary parts of QPSK modulation. This stems from the definition of QPSK as a particular 4-QAM (Quadrature Amplitude Modulation). The error probability for a rectangular  $M$ -QAM (16-QAM, 64-QAM, 256-QAM etc) with size  $L = M^2$  can be calculated by considering two  $M$ -PAM on in-phase and quadrature components. Finally, the error probability of QAM symbol is obtained by the error probability of each branch ( $M$ -PAM). For instance, the QPSK is only a double BPSK; one over each stream (the in-phase stream and the quadrature-phase stream) [14], [15].

Let us now define the real part  $r_{BS}$  of a symbol transmitted by the base station by:

$$r_{BS} = \sum_{i \in S} \sqrt{P_i} - \sum_{j \in \bar{S}} \sqrt{P_j}. \quad (5)$$

where  $S$  is the set of the indexes of the users whose encoded bits are equal to 1. So,  $r_{BS}$  can take  $2^{N_u}$  different values. The expression of the error probability is dependent on  $r_{BS}$  and can change from one  $r_{BS}$  to another.

Besides, when a user receives the signal, we divide the complex plane into several zones named ‘‘decision areas’’. Each area corresponds to a value of  $\epsilon_n$ , i.e. an error on a certain number of symbols while decoding.

Let us consider the basic case where the base station serves two users. When the first user receives the signal, four decision areas exist for the real part, as illustrated in Fig. 2. As shown, the user makes an error on  $r_1$  the real part of the desired symbol in two of these four areas. We can compute the error probability by deriving the probability that the received

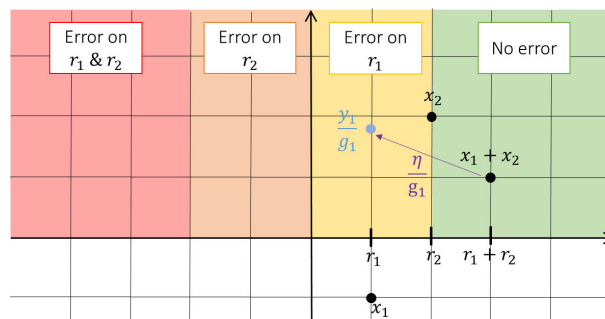


FIGURE 2. Example of decision areas and the associated error for the real part of 2 users’ symbols.

signal is within the limits of the decision areas where the decoder makes an error on  $r_1$ .

In the general case, we obtain the following expression.

*Proposition 1: Considering a base station serving  $N_u$  users. Denoting  $g$  the channel attenuation of the  $n^{th}$  user served by the base station, the expression of its error probability is*

$$P_{e_n} = \frac{1}{2^{N_u}} \sum_S \left( \sum_{\epsilon_n} Q \left( \frac{g}{\sigma} \max_{p \in I_n(\epsilon_n, S)} \left( \sum_{i=1}^p r_i + 2 \sum_{i \in \epsilon_n} r_i \right) \right) - Q \left( \frac{g}{\sigma} \min_{k \in I_n(\bar{\epsilon}_n, S)} \left( \sum_{i=1}^k r_i + 2 \sum_{i \in \bar{\epsilon}_n} r_i \right) \right) \right) \quad (6)$$

with  $S \in \mathcal{P}(\llbracket 1; N_u \rrbracket)$ ,  $\epsilon \in \mathcal{P}(\llbracket n + 1; N_u \rrbracket)$  and Where  $\min(\{\}) = -\infty$ ,  $\max(\{\}) = +\infty$ ,  $I_n(\epsilon_n, S) = (\epsilon_n \cup \{n\}) \oplus S$ ,  $I_n(\bar{\epsilon}_n, S) = \llbracket n; N_u \rrbracket \setminus S$ , and  $Q$  is the  $Q$ -function defined by  $\forall x, Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-u^2/2} du$ .

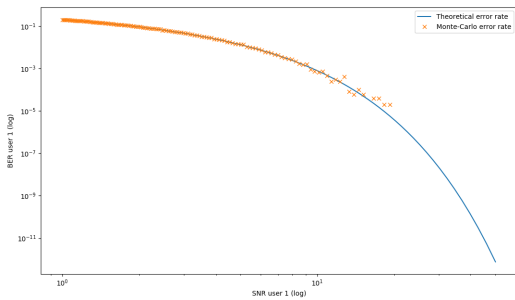
A detailed proof is provided in Appendix VI.

We now express the formula for 2 users in order to check the accuracy of our derivation for this precise use case by comparing it to existing formulas ([6], [7]). For instance,  $\llbracket 1; N \rrbracket = \llbracket 1; 2 \rrbracket = \{\{\}, \{1\}, \{2\}, \{1, 2\}\}$ . The expression of the error probability of user 2 is:

$$P_{e_2} = \frac{1}{2} Q \left( \left( \sqrt{P_1} + \sqrt{P_2} \right) \frac{g_2}{\sigma} \right) + \frac{1}{2} Q \left( \left( -\sqrt{P_1} + \sqrt{P_2} \right) \frac{g_2}{\sigma} \right), \quad (7)$$

The error probability of user 1 is expressed as

$$P_{e_1} = Q \left( \sqrt{P_1} \frac{g_1}{\sigma} \right) + \frac{1}{2} Q \left( \left( -\sqrt{P_1} + \sqrt{P_2} \right) \frac{g_1}{\sigma} \right) - \frac{1}{2} Q \left( \left( \sqrt{P_1} + \sqrt{P_2} \right) \frac{g_1}{\sigma} \right) - \frac{1}{2} Q \left( \left( -\sqrt{P_1} + 2\sqrt{P_2} \right) \frac{g_1}{\sigma} \right) + \frac{1}{2} Q \left( \left( \sqrt{P_1} + 2\sqrt{P_2} \right) \frac{g_1}{\sigma} \right). \quad (8)$$



**FIGURE 3. Comparison between the theoretical error probability and the Monte-Carlo BER simulation for 2 users depending on SNR of user 1. 50000 iterations are considered,  $P_2 = 8500$  and  $P_1 = 1500$ .**

It is worth noting that, making an asymptotic analysis in the case where  $P_1 \ll P_2$ , we obtain the following expression for the error probabilities:

$$P_{e1} \underset{P_1 \ll P_2}{\sim} Q\left(\sqrt{P_1} \frac{g_1}{\sigma}\right), \tag{9}$$

and

$$P_{e2} \underset{P_1 \ll P_2}{\sim} Q\left(\sqrt{P_2} \frac{g_2}{\sigma}\right). \tag{10}$$

As shown by equations (9) and (10), when  $P_1 \ll P_2$ , a perfect SIC is performed and the error probability of each user is identical to that obtained if it would be alone to be served.

We also compare the result of a Monte-Carlo simulation with the numerical evaluation of this formula. This comparison is given in figure 3 and we can easily check the accuracy of our derivations since the simulated points coincide with the bit error probability curve. The Monte-Carlo simulation here is independent from the formula (we multiply the channel experiences and we only count error events), and uses the same hypothesis as the ones used for the proof of the formula. The Python code used to plot the formula and the Monte-Carlo result of the simulation is provided. A pseudo-code implementation of an algorithm evaluating this formula is also given in Appendix VI.

#### IV. POWER ALLOCATION

##### A. PROBLEM FORMULATION

Using the previously derived theoretical framework, we are able to consider the problem of power allocation in downlink NOMA systems. More precisely, in this section, we aim to minimize the error probability sum (sum of all users error probabilities) with respect to a given power budget. The optimization problem is then described as:

$$\arg \min_{P_i} P_e = \sum_{i=1}^N P_{e_i}, \tag{11a}$$

$$\text{s.t. } \sum_{i=1}^N P_i \leq P_{tot}, \tag{11b}$$

$$\forall (i, j) \in \llbracket 1; N_u \rrbracket^2, \quad i < j \Rightarrow P_i \leq P_j, \tag{11c}$$

$$\forall i \in \llbracket 1; N_u \rrbracket, \quad P_i \geq 0 \tag{11d}$$

where  $P_i$  is the transmit power used for serving user  $i$  and  $P_{tot}$  is the total power budget of the base station. We consider that the power  $P_{tot}$  is allocated for all the users, and that  $g$  and  $\sigma$  are known for each user. The users are ordered in the decreasing order of  $g$ .

Please note that the objective function of the problem studied in this section is the sum and difference of  $Q$ -functions so it is not trivially convex or quasi-convex [16].

In the following, we use numerical simulations so as to show that: when the base station serves two users, the objective function is bowl-shaped and can consequently be solved using tools from convex optimization. Besides, we show that in the three users case, the function has several local minimums.

We propose an algorithm for the numerical resolution of the studied problem. An implementation of the algorithm is available at [https://gitlab-research.centralesupelec.fr/garnier\\_jea/NOMA/tree/master/Python/Find\\_Optimum.py](https://gitlab-research.centralesupelec.fr/garnier_jea/NOMA/tree/master/Python/Find_Optimum.py).

As we could not prove the objective function to be convex or quasi-convex, a convex optimization method cannot be applied to the problem. We propose an algorithm which aims at finding a possible global minimum, through a combination of iterations and a gradient descent. The proposed method cannot ensure a global optimum is found in the general case, but at least a local minimum will be found. Furthermore, the proposed algorithm was compared to the stochastic Basin-Hopping algorithm [17], which is used to optimize non-convex multi-variable problems.

The proposed algorithm assumes an optimal solution uses the whole power available to the base-station. The algorithm then recursively compares the estimated total BER for different power distributions, and keeps the set of powers allowing for the lowest total BER. This finite iteration is then followed by a gradient descent, to ensure at least a local minimum is found.

As will be shown in a following part, this method yields better results comparable to that of the Basin-Hopping algorithm, and often results in a lower BER.

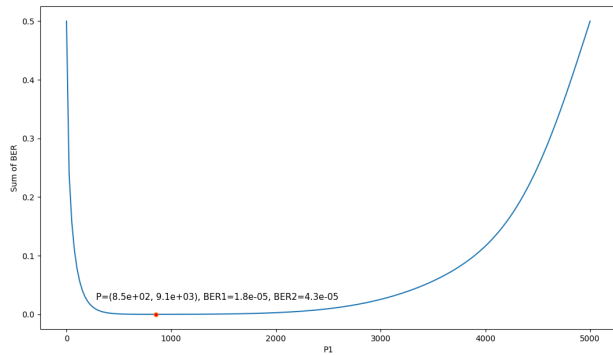
##### B. TWO-USER CASE

We first consider the case of two users, for this precise use case, the objective function has a convex shape. However, as we are unable to provide a general proof of convexity for the studied problem since the objective function is not trivially convex, we try in the following to provide the proof of its convexity in the particular case where  $P_1 \ll P_2$ . This is obviously increasing our understanding of the general problem and provides an additional argument in favor of our solution.

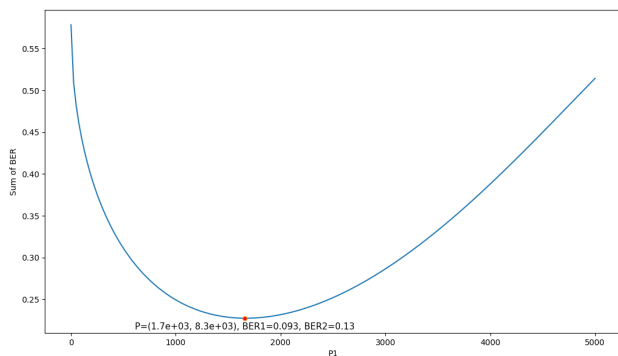
*Proposition 2: In the case where  $P_1 \ll P_2$  the objective function is asymptotically equivalent to a convex function.*

The proof is provided in Appendix VI.

In Figures 4 and 5, we display the bit error probability sum ( $P_e$ ) with respect to the power allocation of user 1 for both high-SNR and low-SNR regimes, respectively.



**FIGURE 4.** Overall bit error probability for  $N_U = 2$  users depending of the power allocation with  $P_1 + P_2 \leq 10000$ ,  $\frac{g_1}{\sigma_1} = 0.2$  (high-SNR regime) and  $\frac{g_2}{\sigma_2} = 0.08$ .



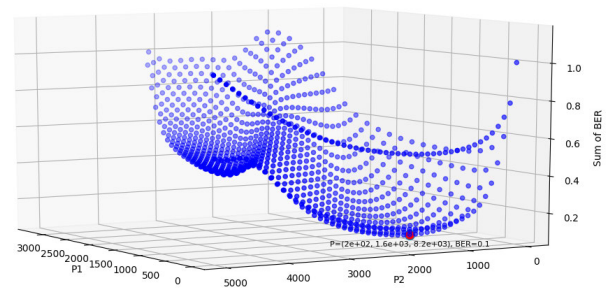
**FIGURE 5.** Overall bit error probability for  $N_U = 2$  users depending of the power allocation with  $P_1 + P_2 \leq 10000$ ,  $\frac{g_1}{\sigma_1} = 0.05$  (low-SNR regime) and  $\frac{g_2}{\sigma_2} = 0.02$ .

For both SNR regimes, when  $P_1$  is zero, all the power is allocated to user 2 and, according to equations (9) and (10), the error probability sum tends towards:

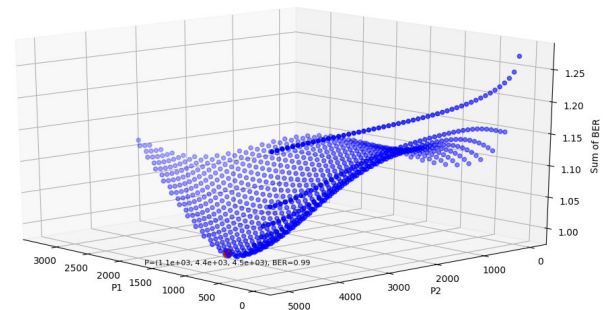
$$P_e \xrightarrow{P_1 \rightarrow 0} \frac{1}{2} + Q\left(\sqrt{P_{tot}} \frac{g_2}{\sigma}\right). \quad (12)$$

In that case, the bit error probability sum is high as all the power is used for the user with the worst channel gain. When  $P_1$  increases, the bit error probability sum reduces as that relative to the user 1 decreases quickly and the one relative to the user 2 increases slowly. While having inverse behaviors at both users, the bit error probability sum continues to get better since the speed of this performance evolution at both users is very different. After reaching the optimum, the performance is deteriorating as both users make more and more mistakes when decoding the symbol  $x_2$ .

The main difference between the results obtained for the low-SNR and high-SNR regimes is that for high SNR regime, the objective function is quite constant over a wide range of  $P_1$  around the optimum; this means that the error probability sum remains quite insensitive to any small-to-medium deviation of the optimal power allocation. This is due to the very slow variations of both users error probabilities.



**FIGURE 6.** Overall bit error probability for  $N_U = 3$  users depending of the power allocation with  $P_1 + P_2 + P_3 \leq 10000$ ,  $\frac{g_1}{\sigma_1} = 0.2$ ,  $\frac{g_2}{\sigma_2} = 0.08$  and  $\frac{g_3}{\sigma_3} = 0.043$ .



**FIGURE 7.** Overall bit error probability for  $N_U = 3$  users depending of the power allocation with  $P_1 + P_2 + P_3 \leq 10000$ ,  $\frac{g_1}{\sigma_1} = 0.04$ ,  $\frac{g_2}{\sigma_2} = 0.016$  and  $\frac{g_3}{\sigma_3} = 0.0087$ .

Finally, it is worth noting that, in those examples, both users are allocated a significantly non-null different power. It is not always the case, as will be shown in the 3-user case.

### C. THREE-USER CASE

We then study the case where three users are to be served by the base station. This exploits the novelty of our formula (1), which is viable for any number of users. This expands on the existing formulas derived for 2 users only. Once again, we compare both high-SNR and low-SNR environments for our users by plotting the bit error probability sum with respect to the power allocations of users 1 and 2, resulting in 3D-curves. Figure 6 depicts the results for the high-SNR regime while Figure 7 illustrates those for the low-SNR regime.

While in the high-SNR regime, the optimum configuration allocates to each user a unique order of magnitude for their power levels (Figure 6), we observe that in the low-SNR context, an edge-case where a user is allocated very little power can found to be the best regarding our metric. In that case, two users are assigned a very similar power (Figure 7).

A different metric avoiding edge cases and ensuring better equity between users could be found to improve user satisfaction. For example, the selection of edge cases can be prevented by using a non-linear error function rather than simply using the sum of bit error probabilities. It can also be

**TABLE 1.** Average time over 100 rounds for  $N = 2$ ,  $P_{\max} = 10000$ , and different step sizes.

Algorithm \ Step	$P_{\max}/10$	$P_{\max}/25$	$P_{\max}/50$	$P_{\max}/100$
Proposed (C)	4.2 ms	7.9 ms	9.5 ms	15.1 ms
Proposed (Py)	4.4 ms	9.5 ms	9.6 ms	15.8 ms
Basin-Hopping	155.6 ms	169.0 ms	110.9 ms	111.5 ms

**TABLE 2.** Average best BER found over 100 rounds for  $N = 2$ ,  $P_{\max} = 10000$ , and different step sizes.

Algorithm \ Step	$P_{\max}/10$	$P_{\max}/25$	$P_{\max}/50$	$P_{\max}/100$
Proposed	0.08	0.09	0.10	0.08
Basin-Hopping	0.11	0.16	0.18	0.18
Gradient descent	0.15	0.14	0.14	0.14

**TABLE 3.** Average time over 75 rounds for  $N = 3$ ,  $P_{\max} = 10000$ , and different step sizes.

Algorithm \ Step	$P_{\max}/10$	$P_{\max}/25$	$P_{\max}/50$	$P_{\max}/100$
Proposed (C)	40.0 ms	121.9 ms	389.8 ms	1354.8 ms
Proposed (Py)	45.1 ms	137.5 ms	414.5 ms	1404.2 ms
Basin-Hopping	1253.7 ms	998.0 ms	862.7 ms	769.3 ms

achieved by adding an appropriate minimum error probability for each user.

**D. ALGORITHM PERFORMANCE**

The performance of our algorithm was compared to an existing implementation of the Basin-Hopping algorithm [18]. For the sake of comparison, the proposed algorithm was implemented both in C, with a binding in Python, and directly in Python.

We then statistically measured the performance of each algorithm under different noise and attenuation conditions, for different number of users, and with different parameters for each algorithm. Multiple rounds were performed (between 50 and 100) with randomly chosen new noise and attenuation values. The average best BER found was also measured, to ensure the algorithms were performing correctly. In the following tables, the displayed BER stands for the average BER, which is given by sum of BER at each user normalized with respect to the number of users. This normalization is done in order to keep BER values always less than one.

The following measurements were performed on a computer containing a 2.9 GHz Intel Core i7, Dual-Core processor. The Python process run on a single core.

For each round, the average best BER found was measured over many random repartition of power between the users, to evaluate the precision of each algorithm:

As observed, the proposed algorithm finds, on average, better solutions in less time for  $N = 2$  users. Both the Basin-Hopping and the proposed algorithm find, on average, better solution than a simple gradient descent, which may only find a local optimum.

**TABLE 4.** Average best BER found over 75 rounds for  $N = 3$ ,  $P_{\max} = 10000$ , and different step sizes.

Algorithm \ Step	$P_{\max}/10$	$P_{\max}/25$	$P_{\max}/50$	$P_{\max}/100$
Proposed	0.18	0.13	0.16	0.13
Basin-Hopping	0.19	0.21	0.27	0.24
Gradient descent	0.29	0.28	0.28	0.28

**TABLE 5.** Average time (in ms) over 50 rounds for  $N = 4$ ,  $P_{\max} = 10000$ , and different step sizes.

Algorithm \ Step	$P_{\max}/10$	$P_{\max}/25$	$P_{\max}/50$	$P_{\max}/100$
Proposed (C)	515.8	1273.1	8100.1	53440.1 ms
Proposed (Py)	567.2	1813.1	10375.5	61363.2 ms
Basin-Hopping	9176.9	7231.3	6331.1	4744.9 ms

**TABLE 6.** Average best BER found over 50 rounds for  $N = 4$ ,  $P_{\max} = 10000$ , and different step sizes.

Algorithm \ Step	$P_{\max}/10$	$P_{\max}/25$	$P_{\max}/50$	$P_{\max}/100$
Proposed	0.24	0.22	0.19	0.2
Basin-Hopping	0.25	0.28	0.29	0.31
Gradient descent	0.32	0.33	0.32	0.32

For greater numbers of users, the time needed to find the optimal solution greatly increases. Thus, we limited our analysis to  $N = 2$ ,  $N = 3$  and  $N = 4$  users.

For smaller step sizes, the performance of the Basin-hopping algorithm is observed to catch up with that of the proposed algorithm. The impact of the step size on the best BER found was also measured:

Table 4 shows that using a smaller step size does not noticeably increase the best BER found by our proposed algorithm, but improves the average best BER found by the Basin-Hopping method. Indeed, with smaller steps, the Basin-Hopping may sometimes find a local minimum rather than a global one, and thus find a worst solution. Moreover, using a simple gradient descent is once again not sufficient to reach the best BER.

As for  $N = 3$ , when using a very small step size, the proposed algorithm performs poorly. However, for higher step sizes, it remains consistently faster than the Basin-Hopping implementation. It also obtains, on average, better results than the Basin-Hopping algorithm, as is shown in table 6.

For low numbers of users, the proposed algorithm consistently performs better than the Basin-Hopping algorithm, both in regards of execution time and resulting optimal BER. In higher-complexity environments, the Basin-Hopping method may sometimes result in faster optimization times (for  $N = 4$  when the step is fixed to be  $P_{\max}/50$  or  $P_{\max}/100$ ). However, in these cases, the Basin-Hopping method is still far from guaranteeing a precision as good as that offered by our algorithm. Nevertheless, the parameters of the proposed algorithm can be tuned to remain faster than the Basin-Hopping algorithm, while continuing to provide a better BER.



FIGURE 8. Example set-up used for our experiments, involving 3 separate USRPs synchronised through an octoclock.

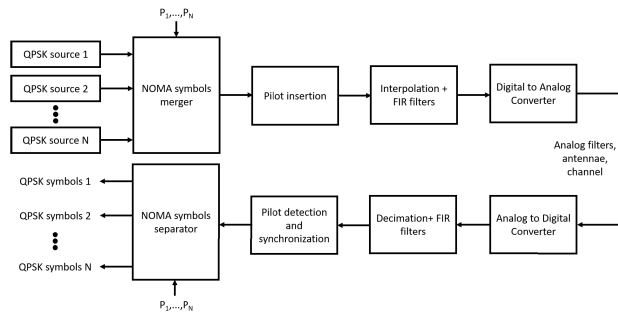


FIGURE 9. Schematic representation of the transmission and reception chains of the SDR based implementation.

Finally, in all of tables 1, 3 and 5, one can observe quite contradictory trends on the evolution of the execution time for our algorithm and for the Basin-hopping method when decreasing the step size. This stems from the major difference on the principle of functioning of the two algorithms. For instance, in the proposed algorithm, decreasing the step size generates more points, so the execution time increases. However, for the Basin-Hopping algorithm, the principle is different and as it is to randomly draw a starting point for a gradient descent then a smaller step will tend to converge the algorithm faster. Meanwhile, when the step becomes too small, we run the risk that the algorithm cannot get out of the local minimum.

## V. EXPERIMENTAL VALIDATION

### A. EXPERIMENTAL SET-UP

We have employed the USRP [9] Software Defined Radio (SDR) platform and GNUradio companion, an open-source software for USRP programming.

In figure 9, we give the schematic representation of the transmission and reception chains so as to improve the following description of our GNU radio implementation.

In our set-up, a first USRP sends the symbols, while another USRP serves as a receiver. The antennas of both USRPs were distant by between 30cm and 1m. In some experiments, we also use a third USRP to generate noise on our channel (see our experimental results V-B for more details).

We built custom blocks to generate a random NOMA-encoded signal for an arbitrary number of users. We then developed blocks to automatically generate the phase-shift induced by the transmission. During the transmission,

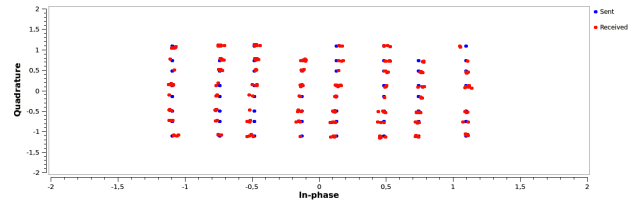


FIGURE 10. Example of received constellation for 3 users using NOMA and our GNU Radio flowgraph.

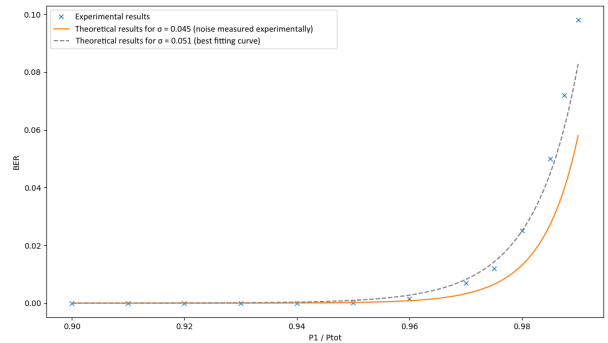


FIGURE 11. Theoretical and experimental BER for 2 users depending of the power allocation with  $P_{tot} = 1$ ,  $g = 1$  and without adding supplementary gaussian noise.

each symbol is repeated multiple times (oversampling process), and FIR filters are used to smooth out the signal. A single sample is then selected for each symbol at the receiving end (downsampling process). The blocks used in GNUradio are described in detail in Appendix VI. The code is also available at [https://gitlab-research.centralesupelec.fr/garnier\\_jea/NOMA](https://gitlab-research.centralesupelec.fr/garnier_jea/NOMA).

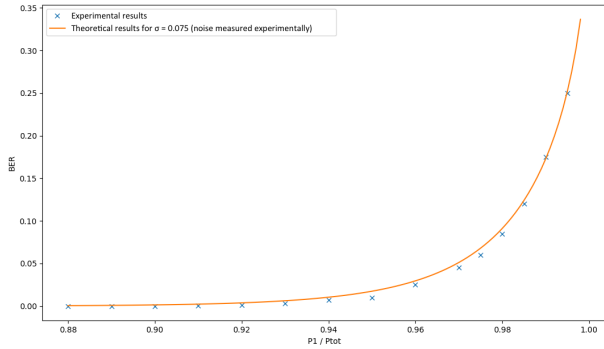
In our formula, we assume the noise to be AWGN (Additive White Gaussian Noise). Actually, the received noise is far from being ideal, and we believe that this ambient noise was the reason why the experimental results were worse than the predictions, using the theoretical formula or simulations. In order to verify that theory, we added a third USRP with the purpose of generating an AWGN that was significantly more powerful than the real noise, i.e, this artificial noise is strong enough to drown out the real noise and consequently to be able to only consider that one. The ambient noise would be overlooked by this artificial AWGN, thus getting the experimental results closer to the theoretical results, as we will see in the following subsection (V-B).

The noise was measured once at the start of each experiment and considered identical for the duration of our measures. An otherwise unused frequency in the lab (445 MHz) was chosen to communicate between USRPs.

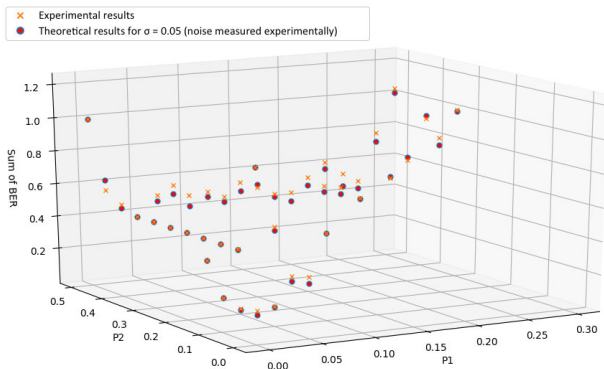
### B. EXPERIMENTAL RESULTS

The experiments have been conducted in real transmission conditions: the signals were sent and received by different USRPs. We implemented a NOMA modulation, as well as





**FIGURE 12.** Theoretical and experimental BER for 2 users depending of the power allocation with  $P_{tot} = 1, g = 1$  and added gaussian noise.



**FIGURE 13.** Theoretical and experimental BER for 3 users depending of the power allocation with  $P_{tot} = 1, g = 1$  and added gaussian noise.

successive interference cancellation (SIC) using GNU Radio Companion.

In order to estimate the BER evolution for two users, we implemented a transmission scenario where the decoded symbols were compared with the sent symbols (see our GNU Radio flowgraph in appendix VI). This allowed us to estimate the BER for a given power distribution, resulting in multiple measures points that we scattered on the theoretical curve for these conditions and the estimated noise.

From figure 11, where we only consider the real noise, the measures and the theoretical functions have a similar trend. We hypothesize that the difference arises from the non-Gaussian nature of ambient noise, as explained before.

In a second time, we added a high power Gaussian noise from a third USRP, dedicated to overlook the real noise, bringing the transmission closer to the ideal condition. In these conditions, we can see, from figure 12, that the gap between the two (theoretical and experimental BER) curves have been significantly reduced.

Combining all these results, we can infer that the difference between theory and experiments is mostly an effect of the non-Gaussian nature of real noise.

This same experiment was reproduced for 3 users. From figure 13, we can then see that the experimental results follow

very closely what is predicted by proposition 1. This corroborates again the assumption made at the beginning, stipulating that the ambient noise is not AWGN

## VI. CONCLUSION

We proposed in this paper an original formula of the bit error probability of a multi-user NOMA scenario where all the users are served using a QPSK modulation. This theoretical derivation is valid regardless of the number of involved users. Thanks to this new theoretical framework, we explored and solved the problem of power allocation in the two and three users cases. Moreover, we verified the accuracy of our derivations by both simulation and experimental validation. However, all these results are given based on the assumption of a perfect SIC. Therefore, future works will pay more attention to develop the best processing at the receiver side to handle possible imperfections.

## AVAILABILITY OF MATERIAL

Our GNU Radio project, as well as the Python sources of our algorithms, are available at [https://gitlab-research.centralesupelec.fr/garnier\\_jea/NOMA](https://gitlab-research.centralesupelec.fr/garnier_jea/NOMA).

## APPENDICES

### APPENDIX A PROOF OF PROPOSITION 1

Without loss of generality, we only consider the real part of the received signal. Let us consider the  $n^{th}$  user served by the base station. The probability to make an error on the real part is dependent on:

- $S$  the set of QPSK symbol with a positive real part.
- $\epsilon_n$  the set of indices for which an error has been made.

For a given  $S$  and  $\epsilon_n$ , in the case where  $r_n > 0$ , the probability to make an error while decoding is:

$$P_{\epsilon_n}^+(S, \epsilon_n) = P \left( \left( y - g_n \sum_{i=n+1}^{N_u} \bar{r}_i < 0 \right) \cap (r_n > 0) \cap \epsilon_n \mid S \right), \tag{13}$$

where,  $\bar{r}_i = r_i$  if  $i \in \epsilon_n$  and  $\bar{r}_i = -r_i$  otherwise. So the error probability can be expressed as:

$$P_{\epsilon_n}^+(S, \epsilon_n) = P \left( \left( \frac{\eta}{g_n} < - \left( \sum_{i=1}^n r_i + 2 \sum_{i \in \epsilon_n} r_i \right) \right) \cap (r_n > 0) \cap \epsilon_n \mid S \right). \tag{14}$$

Similarly, in the case where  $r_n < 0$ , we derive:

$$P_{\epsilon_n}^-(S, \epsilon_n) = P \left( \left( \frac{\eta}{g_n} > - \left( \sum_{i=1}^n r_i + 2 \sum_{i \in \epsilon_n} r_i \right) \right) \cap (r_n < 0) \cap \epsilon_n \mid S \right). \tag{15}$$

Using the formula of total probability :

$$P_{e_n} = \sum_{S \in \mathcal{P}(\llbracket 1; N_u \rrbracket)} \left( P(S) \sum_{\epsilon \in \mathcal{P}(\llbracket n+1; N_u \rrbracket)} P_{e_n}(S, \epsilon_n) \right) \quad (16)$$

where

$$P_{e_n}(S, \epsilon_n) = \begin{cases} P_{e_n}^+(S, \epsilon_n) & \text{if } n \in S \\ P_{e_n}^-(S, \epsilon_n) & \text{otherwise} \end{cases} \quad (17)$$

and  $P(S) = \frac{1}{2^N}$ .

The probability  $P(E_n \cap \epsilon_n | S)$  should now be explicitly expressed. We shall use a recursive expression:

$$P(E_n \cap \epsilon_n | S) = \begin{cases} P(E_n) & \text{if } n = N_u, \text{ since } \epsilon_{N_u} = \{\} \\ P(E_n \cap E_{n+1} \cap \epsilon_{n+1} | S) & \text{if } (n+1) \in \epsilon_n \\ P(E_n \cap \overline{E_{n+1}} \cap \epsilon_{n+1} | S) & \text{otherwise} \end{cases} \quad (18)$$

By developing the recursive formula, the following expression can be obtained:

$$P(E_n \cap \epsilon_n | S) = \left( E_n \bigcap_{k \in \epsilon_n} E_k \bigcap_{p \in \overline{\epsilon_n}} \overline{E_p} \mid S \right)$$

where  $S$  is the set of symbols with a positive real part.

Using (14) and (15), this can be simplified. Indeed:

$$\begin{aligned} & (E_n \cap \epsilon_n | S) \\ &= \begin{cases} \left( \frac{\eta}{g} < -\sum_{i=1}^{N_u} r_i \right) & \text{if } n = N_u \text{ and } n \in S \\ \left( \frac{\eta}{g} > -\sum_{i=1}^{N_u} r_i \right) & \text{if } n = N_u \text{ and } n \notin S \\ \left( \left( \frac{\eta}{g} < -\sum_{i=1}^n r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \cap \epsilon_n \mid S \right) & \text{if } n \in S \\ \left( \left( \frac{\eta}{g} > -\sum_{i=1}^n r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \cap \epsilon_n \mid S \right) & \text{otherwise} \end{cases} \end{aligned}$$

and

$$\begin{aligned} & (\overline{E_n} \cap \epsilon_n | S) \\ &= \begin{cases} \left( \frac{\eta}{g} > -\sum_{i=1}^{N_u} r_i \right) & \text{if } n = N_u \text{ and } n \in S \\ \left( \frac{\eta}{g} < -\sum_{i=1}^{N_u} r_i \right) & \text{if } n = N_u \text{ and } n \notin S \\ \left( \left( \frac{\eta}{g} > -\sum_{i=1}^n r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \cap \epsilon_n \mid S \right) & \text{if } n \in S \\ \left( \left( \frac{\eta}{g} < -\sum_{i=1}^n r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \cap \epsilon_n \mid S \right) & \text{otherwise} \end{cases} \end{aligned}$$

By using the symmetry of the equations, (18) can be simplified into:

$$\begin{aligned} & P(E_n \cap \epsilon_n | S) \\ &= P(E_n \bigcap_{k \in (\epsilon_n \cap S) \cup (\overline{\epsilon_n} \cap \overline{S})} \left( \frac{\eta}{g} < -\sum_{i=1}^k r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \\ & \quad \times \bigcap_{p \in (\overline{\epsilon_n} \cap S) \cup (\epsilon_n \cap \overline{S})} \left( \frac{\eta}{g} > -\sum_{i=1}^p r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \end{aligned}$$

$$\begin{aligned} &= P \left( \bigcap_{k \in ((\epsilon_n \cup \{n\}) \cap S) \cup (\overline{\epsilon_n} \cap \overline{S})} \left( \frac{\eta}{g} < -\sum_{i=1}^k r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \right. \\ & \quad \times \left. \bigcap_{p \in (\overline{\epsilon_n} \cap S) \cup ((\epsilon_n \cup \{n\}) \cap \overline{S})} \left( \frac{\eta}{g} > -\sum_{i=1}^p r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \right) \quad (19) \end{aligned}$$

Once can notice that  $(\overline{\epsilon_n} \cap S) \cup (\epsilon_n \cup \{n\} \cap \overline{S})$  is the symmetric difference between  $\epsilon_n \cup \{n\}$  and  $\cap S$ , and can thus be noted as  $(\epsilon_n \cup \{n\}) \oplus S$ .

Let  $I_n(\epsilon_n, S) = (\epsilon_n \cup \{n\}) \oplus S$  and  $\overline{I_n(\epsilon_n, S)} = \llbracket n; N_u \rrbracket \setminus S$ . Then:

$$\begin{aligned} & P(E_n \cap \epsilon_n | S) = P \left( \bigcap_{k \in \overline{I_n(\epsilon_n, S)}} \left( \frac{\eta}{g} < -\sum_{i=1}^k r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \right. \\ & \quad \times \left. \bigcap_{p \in I_n(\epsilon_n, S)} \left( \frac{\eta}{g} > -\sum_{i=1}^p r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \right) \quad (20) \end{aligned}$$

All these expressions being intersections of boundaries for  $\frac{\eta}{g}$ , only the strictest should be considered. Indeed:

$$\begin{aligned} & \bigcap_{k \in \overline{I_n(\epsilon_n, S)}} \left( \frac{\eta}{g} < -\sum_{i=1}^k r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \\ &= \frac{\eta}{g} < -\max_{k \in \overline{I_n(\epsilon_n, S)}} \left( \sum_{i=1}^k r_i + 2 \sum_{i \in \epsilon_n} r_i \right) \quad (21) \end{aligned}$$

and

$$\begin{aligned} & \bigcap_{p \in I_n(\epsilon_n, S)} \left( \frac{\eta}{g} > -\sum_{i=1}^p r_i - 2 \sum_{i \in \epsilon_n} r_i \right) \\ &= \frac{\eta}{g} > -\min_{p \in I_n(\epsilon_n, S)} \left( \sum_{i=1}^p r_i + 2 \sum_{i \in \epsilon_n} r_i \right) \quad (22) \end{aligned}$$

considering that  $\min(\{\}) = -\infty$  and  $\max(\{\}) = \infty$ .

Thus, by combining (20), (21) and (22):

$$\begin{aligned} & P(E_n \cap \epsilon_n | S) \\ &= P \left( \left( \frac{\eta}{g} < -\max_{k \in \overline{I_n(\epsilon_n, S)}} \left( \sum_{i=1}^k r_i + 2 \sum_{i \in \epsilon_n} r_i \right) \right) \right. \\ & \quad \left. \cap \left( \frac{\eta}{g} > -\min_{p \in I_n(\epsilon_n, S)} \left( \sum_{i=1}^p r_i + 2 \sum_{i \in \epsilon_n} r_i \right) \right) \right) \\ &= P \left( -\min_{k \in \overline{I_n(\epsilon_n, S)}} \left( \sum_{i=1}^k r_i + 2 \sum_{i \in \epsilon_n} r_i \right) < \frac{\eta}{g} \right. \\ & \quad \left. < -\max_{p \in I_n(\epsilon_n, S)} \left( \sum_{i=1}^p r_i + 2 \sum_{i \in \epsilon_n} r_i \right) \right) \quad (23) \end{aligned}$$

We then rewrite this formula using the  $Q$ -function.

Since  $\eta \sim \mathcal{N}(0, \sigma^2)$ ,  $\frac{\eta}{\sigma} \sim \mathcal{N}(0, 1)$ ,  $\forall(a, b) \in \mathbb{R}^{\neq}$  where  $a < b$ :

$$\begin{aligned}
 &P\left(a < \frac{\eta}{\sigma} < b\right) \\
 &= P\left(\frac{\eta}{\sigma} > a \cap \frac{\eta}{\sigma} < b\right) \\
 &= P\left(\frac{\eta}{\sigma} < b\right) - P\left(\frac{\eta}{\sigma} < a\right) \\
 &= P\left(\frac{\eta}{\sigma} < b\right) - P\left(\frac{\eta}{\sigma} < a\right) \\
 &= \frac{1}{2} \left(1 + \operatorname{erf}\left(\frac{b}{\sqrt{2}}\right)\right) - \frac{1}{2} \left(1 + \operatorname{erf}\left(\frac{a}{\sqrt{2}}\right)\right) \\
 &= \frac{1}{2} \left(1 - \operatorname{erf}\left(\frac{a}{\sqrt{2}}\right)\right) - \frac{1}{2} \left(1 - \operatorname{erf}\left(\frac{b}{\sqrt{2}}\right)\right) \\
 &= Q(a) - Q(b) \tag{24}
 \end{aligned}$$

Thus, we deduce the following inequalities:

$$\begin{cases}
 P\left(a < \frac{\eta}{g} < b\right) = Q\left(a\left(\frac{g}{\sigma}\right)\right) - Q\left(b\left(\frac{g}{\sigma}\right)\right) \\
 P\left(a < \frac{\eta}{g}\right) = Q\left(a\left(\frac{g}{\sigma}\right)\right) \\
 P\left(\frac{\eta}{g} < b\right) = 1 - Q\left(b\left(\frac{g}{\sigma}\right)\right)
 \end{cases} \tag{25}$$

Using these expressions, (23) can be simplified:

$$\begin{aligned}
 &P(E_n \cap \epsilon_n | S) \\
 &= P\left(-\min_{k \in I_n(\epsilon_n, S)} \left(\sum_{i=1}^k r_i + 2 \sum_{i \in \epsilon_n} r_i\right) < \frac{\eta}{g}\right) \\
 &< -\max_{p \in I_n(\epsilon_n, S)} \left(\sum_{i=1}^p r_i + 2 \sum_{i \in \epsilon_n} r_i\right) \\
 &= Q\left(-\frac{g}{\sigma} \min_{k \in I_n(\epsilon_n, S)} \left(\sum_{i=1}^k r_i + 2 \sum_{i \in \epsilon_n} r_i\right)\right) \\
 &\quad - Q\left(-\frac{g}{\sigma} \max_{p \in I_n(\epsilon_n, S)} \left(\sum_{i=1}^p r_i + 2 \sum_{i \in \epsilon_n} r_i\right)\right) \\
 &= Q\left(\frac{g}{\sigma} \max_{p \in I_n(\epsilon_n, S)} \left(\sum_{i=1}^p r_i + 2 \sum_{i \in \epsilon_n} r_i\right)\right) \\
 &\quad - Q\left(\frac{g}{\sigma} \min_{k \in I_n(\epsilon_n, S)} \left(\sum_{i=1}^k r_i + 2 \sum_{i \in \epsilon_n} r_i\right)\right) \tag{26}
 \end{aligned}$$

Finally:

$$\begin{aligned}
 P_{e_n} &= \frac{1}{2^{N_u}} \sum_S \left( \sum_{\epsilon_n} Q\left(\frac{g}{\sigma} \max_{p \in I_n(\epsilon_n, S)} \left(\sum_{i=1}^p r_i + 2 \sum_{i \in \epsilon_n} r_i\right)\right) \right. \\
 &\quad \left. - Q\left(\frac{g}{\sigma} \min_{k \in I_n(\epsilon_n, S)} \left(\sum_{i=1}^k r_i + 2 \sum_{i \in \epsilon_n} r_i\right)\right) \right) \tag{27}
 \end{aligned}$$

with  $S \in \mathcal{P}(\llbracket 1; N_u \rrbracket)$ ,  $\epsilon_n \in \mathcal{P}(\llbracket n+1; N_u \rrbracket)$  and where  $\min(\{\}) = -\infty$  and  $\max(\{\}) = \infty$ ,

$I_n(\epsilon_n, S) = (\epsilon_n \cup \{n\}) \oplus S$  and  $\overline{I_n(\epsilon_n, S)} = \llbracket n; N_u \rrbracket \setminus S$ ,  $\sigma$  is the standard deviation of the real part of the noise and

$$\forall i \in \llbracket 1; N_u \rrbracket, r_i = \begin{cases} \sqrt{P_i} & \text{if } i \in S \\ -\sqrt{P_i} & \text{otherwise} \end{cases}$$

**APPENDIX B PROOF OF PROPOSITION 2**

In this proof, we derive a first order approximation of the objective function. To do so, we use the Bachmann-Landau Little-o notation. Besides, we denote  $\epsilon = \frac{P_1}{P_{\text{tot}}} \ll 1$ . As all the power is allocated, we write:

$$\frac{P_2}{P_t} = 1 - \epsilon. \tag{28}$$

As  $\epsilon$  is very small, we write:

$$\sqrt{1 - \epsilon} = 1 - \frac{\epsilon}{2} + o(\epsilon) \tag{29}$$

As a consequence,

$$Q\left(\left(\sqrt{P_1} + \sqrt{P_2}\right) \frac{g_2}{\sigma}\right) = Q\left(\frac{\sqrt{P_{\text{tot}} g_2}}{\sigma} \left(1 - \frac{\epsilon}{2} + \sqrt{\epsilon} + o(\epsilon)\right)\right). \tag{30}$$

We now employ the second order Taylor expansion:

$$\begin{aligned}
 Q(\alpha(x+h)) &\underset{h \approx 0}{=} Q(\alpha x) - \frac{\alpha}{\sqrt{2\pi}} e^{-\frac{\alpha^2 x^2}{2}} h \\
 &\quad + \frac{\alpha^3 x}{\sqrt{2\pi}} e^{-\frac{\alpha^2 x^2}{2}} \frac{h^2}{2} + o(h^2). \tag{31}
 \end{aligned}$$

For  $\epsilon \approx 0$ , we obtain the following equality:

$$\begin{aligned}
 Q\left(\left(\sqrt{P_1} + \sqrt{P_2}\right) \frac{g_2}{\sigma}\right) &= Q\left(\frac{\sqrt{P_{\text{tot}} g_2}}{\sigma}\right) \\
 &\quad - \frac{\sqrt{P_{\text{tot}} g_2}}{\sigma \sqrt{2\pi}} e^{-\frac{P_{\text{tot}} g_2^2}{2\sigma^2}} \left(-\frac{\epsilon}{2} + \sqrt{\epsilon}\right) \\
 &\quad + \left(\frac{\sqrt{P_{\text{tot}} g_2}}{\sigma}\right)^3 \frac{e^{-\frac{P_{\text{tot}} g_2^2}{2\sigma^2}}}{2\sqrt{2\pi}} \epsilon + o(\epsilon) \tag{32}
 \end{aligned}$$

Similarly,

$$\begin{aligned}
 Q\left(\left(\sqrt{P_1} - \sqrt{P_2}\right) \frac{g_2}{\sigma}\right) &= Q\left(\frac{\sqrt{P_{\text{tot}} g_2}}{\sigma}\right) \\
 &\quad - \frac{\sqrt{P_{\text{tot}} g_2}}{\sigma \sqrt{2\pi}} e^{-\frac{P_{\text{tot}} g_2^2}{2\sigma^2}} \left(-\frac{\epsilon}{2} - \sqrt{\epsilon}\right) \\
 &\quad + \left(\frac{\sqrt{P_{\text{tot}} g_2}}{\sigma}\right)^3 \frac{e^{-\frac{P_{\text{tot}} g_2^2}{2\sigma^2}}}{2\sqrt{2\pi}} \epsilon + o(\epsilon). \tag{33}
 \end{aligned}$$

As a consequence, in the particular case where  $\epsilon \approx 0$ , we have the following equality:

$$\begin{aligned}
 P_{e_2} &= Q\left(\frac{\sqrt{P_{\text{tot}} g_2}}{\sigma}\right) \\
 &\quad + \frac{\sqrt{P_{\text{tot}} g_2}}{\sigma \sqrt{2\pi}} e^{-\frac{P_{\text{tot}} g_2^2}{2\sigma^2}} \left(1 + \frac{P_{\text{tot}} g_2^2}{\sigma^2}\right) \frac{\epsilon}{2} + o(\epsilon) \tag{34}
 \end{aligned}$$

Using the same approximation, we obtain the following approximation when  $\epsilon \ll 1$ :

$$\begin{aligned} & \frac{1}{2}Q\left(\left(\sqrt{P_1} + 2\sqrt{P_2}\right)\frac{g_1}{\sigma}\right) - \frac{1}{2}Q\left(\left(-\sqrt{P_1} + 2\sqrt{P_2}\right)\frac{g_1}{\sigma}\right) \\ &= -\frac{\sqrt{P_{\text{tot}}g_1}}{\sigma\sqrt{2\pi}}e^{-\frac{2P_{\text{tot}}g_1^2}{\sigma^2}}\sqrt{\epsilon} + o(\epsilon) \end{aligned} \quad (35)$$

Moreover,

$$\begin{aligned} & Q\left(\sqrt{P_1}\frac{g_1}{\sigma}\right) + \frac{1}{2}Q\left(\left(-\sqrt{P_1} + \sqrt{P_2}\right)\frac{g_1}{\sigma}\right) \\ & \quad - \frac{1}{2}Q\left(\left(\sqrt{P_1} + \sqrt{P_2}\right)\frac{g_1}{\sigma}\right) \\ &= \frac{1}{2} - \frac{\sqrt{P_{\text{tot}}g_1}}{\sigma\sqrt{2\pi}}\left(1 - e^{-\frac{P_{\text{tot}}g_1^2}{2\sigma^2}}\right)\sqrt{\epsilon} + o(\epsilon) \end{aligned} \quad (36)$$

As a consequence, in the case where the base station serves two users and where  $P_1 \ll P_2$ , the objective function has the following equivalent:

$$\begin{aligned} P_e \underset{\epsilon \approx 0}{\sim} & \frac{1}{2} + Q\left(\frac{\sqrt{P_{\text{tot}}g_2}}{\sigma}\right) - \frac{\sqrt{P_{\text{tot}}g_1}}{\sigma\sqrt{2\pi}}e^{-\frac{2P_{\text{tot}}g_1^2}{\sigma^2}}\sqrt{\epsilon} \\ & - \frac{\sqrt{P_{\text{tot}}g_1}}{\sigma\sqrt{2\pi}}\left(1 - e^{-\frac{P_{\text{tot}}g_1^2}{2\sigma^2}}\right)\sqrt{\epsilon} \\ & + \frac{\sqrt{P_{\text{tot}}g_2}}{\sigma\sqrt{2\pi}}e^{-\frac{P_{\text{tot}}g_2^2}{2\sigma^2}}\left(1 + \frac{P_{\text{tot}}g_2^2}{\sigma^2}\right)\frac{\epsilon}{2}. \end{aligned} \quad (37)$$

This expression is convex as the sum of convex functions. So, in the particular case  $P_1 \ll P_2$ , the objective function is asymptotically equivalent to a convex function.

### APPENDIX C PSEUDO-CODE OF THE THEORETICAL MODEL

Given  $N_u$ ,  $g$  and  $\sigma$ , this algorithm computes the numerical value of the error probability.

**Require:**  $g \geq 0, \sigma > 0, N_u > 0, P \in (\mathbb{R}^+)^N$   
 {Generate all possibilities for the signs of symbols}  
 $S \leftarrow \text{parts}(\{0, 1\}, N)$   
 {Computed probabilities are stored in “output”}  
 output  $\leftarrow \{\}$   
**for**  $n \in \{N, N - 1, \dots, 1\}$  **do**  
   {List of possible error combinations for the step  $n$ }  
    $E_n \leftarrow \text{parts}(\{0, 1\}, N - n)$   
   {List of probabilities computed at the step  $n$ }  
   Probas $_n \leftarrow \{\}$   
   **for**  $s \in S$  **do**  
     {Amplitude of each symbol based on allocated power and signs}  
      $r \leftarrow \{s_i\sqrt{P_i} | i \in \{N, N - 1, \dots, 1\}\}$   
     **for**  $e \in E_n$  **do**  
       {Constraints for the upper boundary of  $\frac{\eta}{g}$ }  
        $C_{\text{sup}} \leftarrow \{\}$   
       {Constraints for the lower boundary of  $\frac{\eta}{g}$ }  
        $C_{\text{inf}} \leftarrow \{\}$

**for**  $i \in ((e \cup \{n\}) \cap s) \cup (\bar{e} \cap \bar{s})$  **do**  
    $C_{\text{sup}} \leftarrow C_{\text{sup}} + \left\{ - \left( \sum_{k=1}^i r_k + 2 \sum_{k \in \epsilon_n(i)} r_k \right) \right\}$   
**end for**  
**for**  $i \in (\bar{e} \cap s) \cup ((e \cup \{n\}) \cap \bar{s})$  **do**  
    $C_{\text{inf}} \leftarrow C_{\text{inf}} + \left\{ - \left( \sum_{k=1}^i r_k + 2 \sum_{k \in \epsilon_n(i)} r_k \right) \right\}$   
**end for**  
 {Compute the error probability for this step depending on the constraints we found}  
**if**  $\max(C_{\text{inf}}) > \min(C_{\text{sup}})$  **then**  
    $P_e \leftarrow 0$   
**else if**  $C_{\text{inf}} = \emptyset$  **and**  $C_{\text{sup}} = \emptyset$  **then**  
    $P_e \leftarrow 1$   
**else if**  $C_{\text{inf}} = \emptyset$  **then**  
    $P_e \leftarrow 1 - Q\left(\frac{g}{\sigma} \min(C_{\text{sup}})\right)$   
**else if**  $C_{\text{sup}} = \emptyset$  **then**  
    $P_e \leftarrow Q\left(\frac{g}{\sigma} \max(C_{\text{inf}})\right)$   
**else**  
    $P_e \leftarrow Q\left(\frac{g}{\sigma} \max(C_{\text{inf}})\right) - Q\left(\frac{g}{\sigma} \min(C_{\text{sup}})\right)$   
**end if**  
 Probas $_n \leftarrow \text{Probas}_n + \{P_e\}$   
**end for**  
**end for**  
 output  $\leftarrow \text{output} + \left\{ \frac{1}{2^N} \sum_{p \in \text{Probas}_n} P \right\}$   
**end for**  
**return** output

### APPENDIX D PSEUDO-CODE OF THE DECODING ALGORITHM

Given  $N_u$ ,  $g$ ;  $P_i$  for each user, and  $y$ , this algorithm decodes each user’s symbol.

**Require:**  $g \geq 0, N_u > 0, P \in (\mathbb{R}^+)^N$   
 {The decoded symbols will be stored in “output”}  
 output  $\leftarrow \{\}$   
**for**  $n \in \{N, N - 1, \dots, 1\}$  **do**  
   {Decode the perceived bit for user  $n$ }  
   **if**  $y \leq 0$  **then**  
      $y_n \leftarrow 1$   
   **else**  
      $y_n \leftarrow 0$   
   **end if**  
   output  $\leftarrow \text{output} + \{y_i\}$   
   {Remove the value which was just decoded from the received symbol}  
    $y \leftarrow y - g\sqrt{P_n}y_n$   
**end for**  
**return** output

### APPENDIX E GNU RADIO FLOWGRAPH FOR BER ESTIMATION

Here, we present the flowgraph we used to evaluate the BER for different power distributions. For the sake of readability, we present the 2-user flowgraph, though it can be expanded to any number of users (see [https://gitlab-research.centralesupelec.fr/garnier\\_jea/NOMA](https://gitlab-research.centralesupelec.fr/garnier_jea/NOMA) for the sources to our project, the flowgraph, and our custom blocks).

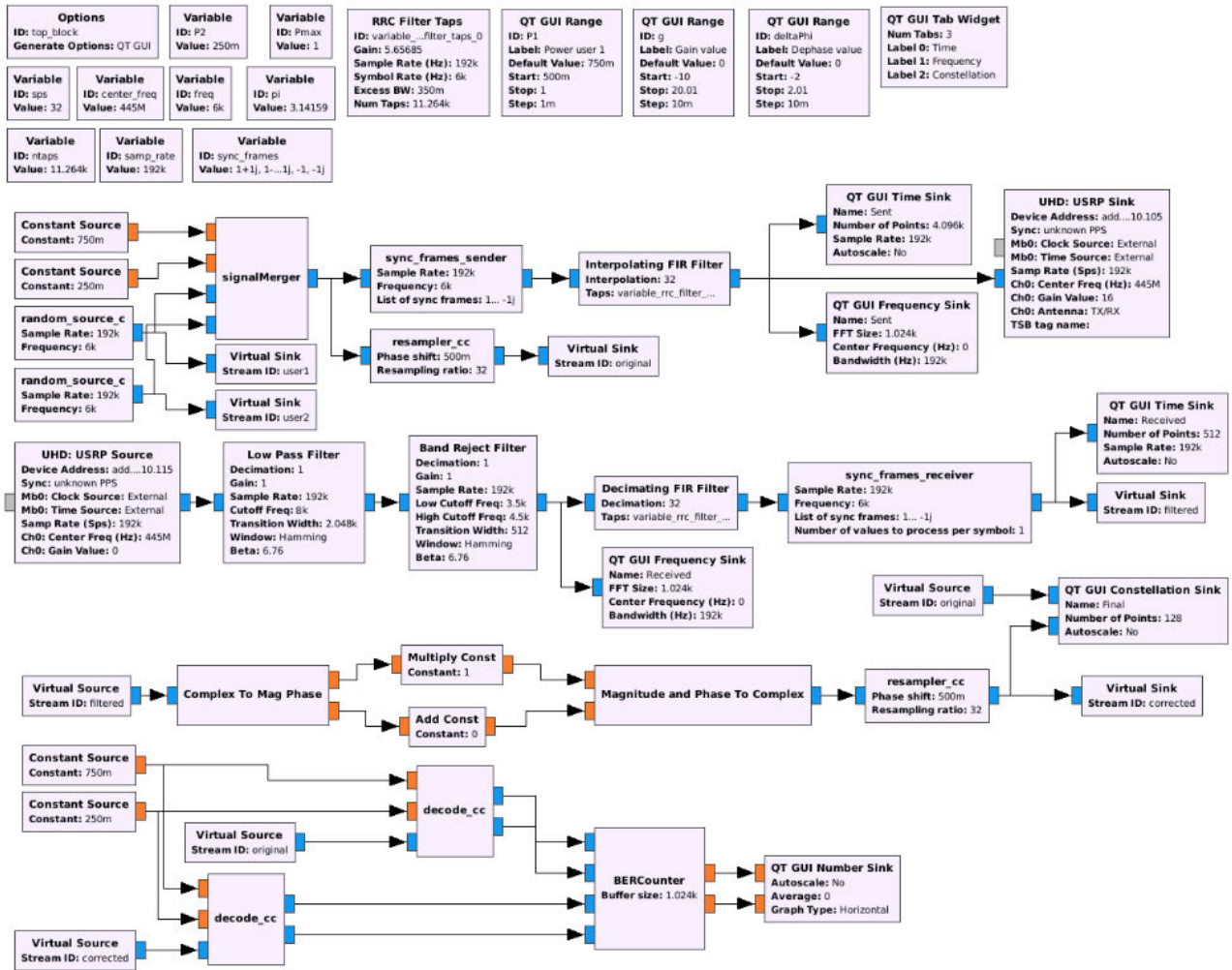


FIGURE 14. GNU Radio flowgraph sending data through 2 USRPs and measuring the BER at the receiving end.

Our USRPs have synchronized clocks thanks to an external OctoClock setup. We used custom blocks, in C++, and pre-existing GNU Radio blocks:

- **random\_source\_c** Generates a random complex signal at a given frequency (each value is taken at random in the set  $\{1 + j, 1 - j, -1 - j, -1 + j\}$ ),
- **signalMerger** Converts the complex inputs of N users to a single output using NOMA,
- **sync\_frames\_sender** Send an “initialization” frame, and then acts as a simple passthrough block. This frame is used to correct the phase and amplitude of received symbols,
- **FIR Filters** We use raised cosine filters to shape our symbols so they have a finite frequency usage,
- **Low Pass Filter** and **Band Reject Filter** Filter-out noise on frequencies that are unused by our FIR Filters,
- **Low Pass Filter** and **Band Reject Filter** Filter-out noise on frequencies that are unused by our FIR Filters,
- **sync\_frames\_receiver** The complementary block to **sync\_frames\_sender**. It uses the known initial frames

sent to compute the initial phase and gain correction to apply, and corrects it for all the following received symbols,

- **Complex To Mag Phase to Magnitude and Phase To Complex** is a simple contraption to manually tweak the gain and phase correction if it’s a bit off,
- **resampler\_cc** Since there are multiple samples per symbol ( $sps = 32$  in this case), this block keeps a single sample representing the symbol (the one with the most energy),
- **decode\_cc** The complementary block to **signalMerger**. It decodes the received complex signal into the signal for each user,
- **decode\_cc** Measures the BER for each user in a sliding window (of 1024 values in our case).

The variables defined in our flowgraph have the following values:

- $P2 = Pmax - P1$ ,
- $Pmax = 1$ ,
- $sps = 32$ ,

- `center_freq = 445000000`,
- `freq = samp_rate/psps`: frequency used by `random_source_c`,
- `pi = math.pi`,
- `ntaps = 11 * pow(psps, 2)`,
- `samp_rate = 512000`,
- `sync_frames = [1 + 1j, 1 - 1j, -1 - 1j, -1 + 1j, 1, 1j, -1, -1j]`: List of frames used by `sync_frames_sender` and `sync_frames_receiver`,
- **RRC Filter taps**: `Gain = math.sqrt(psps)` and `Num Taps = ntaps`.

## REFERENCES

- [1] Y. Saito, Y. Kishiyama, A. Benjebbour, T. Nakamura, A. Li, and K. Higuchi, "Non-orthogonal multiple access (NOMA) for cellular future radio access," in *Proc. IEEE 77th Veh. Technol. Conf. (VTC Spring)*, Jun. 2013, pp. 1–5.
- [2] Z. Ding, Y. Liu, J. Choi, Q. Sun, M. Elkashlan, C.-L. I, and H. V. Poor, "Application of non-orthogonal multiple access in LTE and 5G networks," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 185–191, Feb. 2017.
- [3] Z. Ding, Z. Yang, P. Fan, and H. V. Poor, "On the performance of non-orthogonal multiple access in 5G systems with randomly deployed users," *IEEE Signal Process. Lett.*, vol. 21, no. 12, pp. 1501–1505, Dec. 2014.
- [4] Y. Zhang, H.-M. Wang, Q. Yang, and Z. Ding, "Secrecy sum rate maximization in non-orthogonal multiple access," *IEEE Commun. Lett.*, vol. 20, no. 5, pp. 930–933, May 2016.
- [5] Q. C. Li, H. Niu, A. T. Papatthanasious, and G. Wu, "5G network capacity: Key elements and technologies," *IEEE Veh. Technol. Mag.*, vol. 9, no. 1, pp. 71–78, Mar. 2014.
- [6] X. Wang, F. Labeau, and L. Mei, "Closed-form BER expressions of QPSK constellation for uplink non-orthogonal multiple access," *IEEE Commun. Lett.*, vol. 21, no. 10, pp. 2242–2245, Oct. 2017.
- [7] F. Kara and H. Kaya, "BER performances of downlink and uplink NOMA in the presence of SIC errors over fading channels," *IET Commun.*, vol. 12, no. 15, pp. 1834–1844, Sep. 2018.
- [8] T. Assaf, A. Al-Dweik, M. Moursi, and H. Zeineldin, "Exact BER performance analysis for downlink NOMA systems over Nakagami- $m$  fading channels," *IEEE Access*, vol. 7, pp. 134539–134555, 2019.
- [9] M. Ettus and M. Braun, *The Universal Software Radio Peripheral (USRP) Family of Low-Cost SDRs*, Hoboken, NJ, USA: Wiley, 2015, ch. 1, pp. 3–23.
- [10] X. Xiong, W. Xiang, K. Zheng, H. Shen, and X. Wei, "An open source SDR-based NOMA system for 5G networks," *IEEE Wireless Commun.*, vol. 22, no. 6, pp. 24–32, Dec. 2015.
- [11] X. Wei, H. Liu, Z. Geng, K. Zheng, R. Xu, Y. Liu, and P. Chen, "Software defined radio implementation of a non-orthogonal multiple access system towards 5G," *IEEE Access*, vol. 4, pp. 9604–9613, 2016.
- [12] E. Khorov, A. Kureev, and I. Levitsky, "NOMA testbed on Wi-Fi," in *Proc. IEEE 29th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2018, pp. 1153–1154.
- [13] S. Abeywickrama, L. Liu, Y. Chi, and C. Yuen, "Over-the-Air implementation of uplink NOMA," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–6.
- [14] J. Proakis and M. Salehi, *Digital Communications*, 5th ed. New York, NY, USA: McGraw-Hill, 2007.
- [15] M. P. Fitz, "Further results in the unified analysis of digital communication systems," *IEEE Trans. Commun.*, vol. 40, no. 3, pp. 521–532, Mar. 1992.
- [16] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge Univ. Press, 2004.
- [17] D. J. Wales and J. P. K. Doye, "Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms," *J. Phys. Chem. A*, vol. 101, no. 28, pp. 5111–5116, Jul. 1997.
- [18] Accessed: 2019. [Online]. Available: <https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.basinhopping.html>

**JEAN-ROMAIN GARNIER** attended a competitive two-year course specialized in physics and mathematics, before enrolling in a generalist engineering degree with a specialization in cybersecurity, from 2017 to 2020. Along cybersecurity, he worked on projects ranging from telecommunications to computer vision, and applied this experience to professional and commercial projects.

**ALEXIS FABRE** received the Technical Diploma degree in electrical engineering, telecommunications, and industrial computing, in 2017. He is currently pursuing a generalist engineering degree with a specialization in cybersecurity, for 2020. He worked twice as a Research Assistant to help the understanding of spiking neural networks, as well as non-orthogonal multiple access in telecommunications.

**HAÏFA FARÈS** received the bachelor's and M.Sc. degrees in telecommunication engineering from the Higher School of Communications of Tunis (Sup'Com), in 2007 and 2008, respectively.

Since October 2008, she has been working in the Electronics Department of Telecom Bretagne, in order to obtain her Ph.D. degree, in 2011. Her research interests are in the areas of communication theory, including Hybrid Automatic Repeat reQuest (HARQ) protocols for cooperative systems, iterative decoding algorithms, and MIMO systems. From January 2012 to August 2013, she was holding a postdoctoral position at the Electronics Department, IMT Atlantique Brest. She was working on several topics such as Iterative joint source-relay decoding, decoding schemes of correlated sources in a relay network, and software simulation chain for the physical layer of the uplink LTE. During her experience as a temporary Associate Professor with ENSSAT Lannion, she worked on the timely topic of "green communications for sensor networks. From September 2016 to June 2017, she worked with CentraleSupélec for a joint project with the French Atomic Energy Commission, investigating a new waveform of the family of continuous phase modulations (CPM), directly inspired from quantum physics. Since January 2018, she has been with CentraleSupélec of Rennes as an Associate Professor and a member of the SCEE Team of IETR. She pursues work on CPM schemes and is doing research on non-orthogonal multiple access in telecommunications.

Dr. Farès was awarded the Best Student Engineering Project from the Higher School of Communications of Tunis, in 2007.



**RÉMI BONNEFOI** received an engineering degree in electronics and telecommunications, in 2015, and the Ph.D. degree in telecommunications from CentraleSupélec, in 2018. He is currently working as a Telecommunications Research Engineer at Edison Ways, Montauban, France.

• • •