Received March 19, 2014, accepted April 15, 2014, date of publication May 8, 2014, date of current version May 19, 2014.

Digital Object Identifier 10.1109/ACCESS.2014.2322013

Low-Complexity Soft-Output Quantum-Assisted Multiuser Detection for Direct-Sequence Spreading and Slow Subcarrier-Hopping Aided SDMA-OFDM Systems

PANAGIOTIS BOTSINIS, (Student Member, IEEE), DIMITRIOS ALANIS, (Student Member, IEEE), SOON XIN NG, (Senior Member, IEEE), AND LAJOS HANZO, (Fellow, IEEE)
School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K.

Corresponding author: L. Hanzo (lh@ecs.soton.ac.uk)

This work was supported in part by the RC-UK under the India-U.K. Advanced Technology Centre, in part by the EU under the Concerto Project, in part by the European Research Council, in part by the Advanced Fellow Grant, and in part by the Royal Society's Wolfson Reasearch Merit Award.

ABSTRACT Low-complexity suboptimal multiuser detectors (MUDs) are widely used in multiple access communication systems for separating users, since the computational complexity of the maximum likelihood (ML) detector is potentially excessive for practical implementation. Quantum computing may be invoked in the detection procedure, by exploiting its inherent parallelism for approaching the ML MUDs performance at a substantially reduced number of cost function evaluations. In this contribution, we propose a soft-output (SO) quantum-assisted MUD achieving a near-ML performance and compare it to the corresponding SO ant colony optimization MUD. We investigate rank deficient direct-sequence spreading (DSS) and slow subcarrier-hopping aided (SSCH) spatial division multiple access orthogonal frequency division multiplexing systems, where the number of users to be detected is higher than the number of receive antenna elements used. We show that for a given complexity budget, the proposed SO-Dürr-Høyer algorithm (DHA) QMUD achieves a better performance. We also propose an adaptive hybrid SO-ML/SO-DHA MUD, which adapts itself to the number of users equipped with the same spreading sequence and transmitting on the same subcarrier. Finally, we propose a DSS-based uniform SSCH scheme, which improves the system's performance by 0.5 dB at a BER of 10^{-5}, despite reducing the complexity required by the MUDs employed.

INDEX TERMS Ant colony optimization, computational complexity, Dürr-Høyer algorithm, Grover's quantum search algorithm, multiuser detection, orthogonal frequency division multiplexing, quantum computing, spatial division multiple access, spreading sequences.

NOMENCLATURE

ACO	Ant Colony Optimization.
AE	Antenna Element.
AWGN	Additive White Gaussian-distributed Noise.
BBHT	Boyer-Brassard-Høyer-Tapp.
BICM	Bit Interleaved Coded Modulation.
BS	Base Station.
CD-CFE	Classic Domain Cost Function Evaluation.
CDMA	Code Division Multiple Access.
CF	Cost Function.
CFE	Cost Function Evaluation.
CP	Cyclic Prefix.
DHA	Dürr-Høyer Algorithm.
DSS	Direct-Sequence Spreading.
FBKT	Forward and Backward Knowledge Transfer.
FD	Frequency Domain.
FD-CHTF	Frequency-Domain Channel Transfer Function.
FFT	Fast Fourier Transform.
FKT	Forward Knowledge Transfer.
FW	Free Will.
IFFT	Inverse Fast Fourier Transform.
LLR	Log-Likelihood Ratio.
LUP	Look-Up Table.
MAA	Maximum Approximation.
MIMO	Multiple Input Multiple Output.
ML	Maximum Likelihood.

MMSE	Minimum Mean Square Error.
MUA	Multi-Input Approximation.
MUD	Multi-User Detector.
MUI	Multi-User Interference.
NE	Neighbour Exploitation.
OFDM	Orthogonal Frequency Division Multiplexing.
QCR	Quantum Control Register.
QD-CFE	Quantum Domain Cost Function Evaluation.
QMA	Quantum Mean Algorithm.
QMUD	Quantum Multi-User Detector.
QoS	Quality of Service.
QSA	Quantum Search Algorithm.
QWSA	Quantum Weighted Sum Algorithm.
RSSCH	Random Slow Subcarrier Hopping.
SD	Spatial Domain.
SDMA	Spatial Division Multiple Access.
SFH	Slow Frequency Hopping.
SO	Soft Output.
SSCH	Slow Subcarrier Hopping.
TCC	Turbo Convolutional Code.
TD	Time Domain.
UL	Uplink.
USSCH	Uniform Slow Subcarrier Hopping.
WH	Walsh-Hadamard.
ZF	Zero Forcing.

I. INTRODUCTION

The era when the number of mobile devices will exceed the globe's population is approaching, whilst the requirement of a high Quality of Service (QoS) will still be present. In metropolitan areas and airports, the number of users that has to be supported by a Base Station (BS) may exceed the number of orthogonal, i.e. uninterfered channels. For example, in rank-deficient Code Division Multiple Access (CDMA) [1], [2] systems, where the number of users K is higher than the Spreading Factor SF of the Direct-Sequence Spreading (DSS) code used, the conventional Zero Forcing (ZF) and Minimum Mean Square Error (MMSE) detectors typically experience an inadequate performance [3]. Similar findings are valid also for Spatial Division Multiple Access (SDMA) [4]–[6] arrangements, when the number of receive antenna elements (AE) P used at the BS is exceeded by the number of users. At the same time, the computational complexity of the Maximum Likelihood (ML) Multi-User Detector (MUD) quantified in terms of the number of Cost Function (CF) Evaluations (CFE) might be excessive for practical systems.

An efficient multiple access scheme that combines the principles of CDMA, SDMA and Multiple Input Multiple Output (MIMO) Orthogonal Frequency Division Multiplexing (OFDM) [7]–[9] is constituted by the DSS and Slow Subcarrier-Hopping aided (SSCH) SDMA-OFDM system of [7]. More specifically, each user has been assigned a DSS code, which may or may not be unique among the supported users. Moreover, each user transmits on an identical fraction of the number of available subcarriers, while the specific subcarriers each user transmits on are allocated via a predetermined subcarrier allocation algorithm. Therefore, if two users transmit on different orthogonal subcarriers, their signals are separated in the Frequency Domain (FD). At the same time, when using orthogonal Walsh-Hadamard (WH) codes [2], provided that two users transmitting on the same subcarrier have been allocated different WH codes, their signals are separated in the Time Domain (TD). Finally, if two users transmit on the same subcarrier and have been allocated the same DSS code, they will be separated in the Spatial Domain (SD) by exploiting their channel knowledge obtained with the aid of pilot signals [7].

Focusing on vehicular uplink (UL) communication, low-complexity near-optimal MUDs are required for separating the groups of users who have been allocated the same spreading sequences and transmit over the same subcarriers in the SD. The Soft-Output (SO) Ant Colony Optimization (ACO) MUDs relying on the MUlti-input Approximation (MUA) [10] and on the MAximum Approximation (MAA) [11] have been employed for low-complexity detection, but their performance in rank-deficient systems is unsatisfactory. The complexity of the different systems investigated is quantified in terms of the number of CFEs.

Recent advances in the implementation of quantum algorithms [12]–[14] have expedited the integration of quantum computing with wireless communications [15], [16]. Imre and Balázs have proposed a hard-output Quantum-assisted MUD (QMUD) in [17] based on the Quantum Counting algorithm of [18], [19]. In [20], we presented a soft-input soft-output QMUD based on the Quantum Weighted Sum Algorithm (QWSA), which was inspired by the Quantum Mean Algorithm (QMA) [21]. Both the performance and the complexity of the QWSA QMUD depends on the number of quantum bits or *qubits*[1] employed in the Quantum Control Register (QCR), as well as on the number of MUD-decoder iterations. More specifically, the MUD calculates the Log Likelihood Ratio (LLR) of each bit of the transmitted multi-level symbol. Let us assume that the multi-level symbol $\mathbf{x}$ was transmitted and the signal $\mathbf{y}$ was received. Based on Bayes' theorem, the probability of having transmitted $\mathbf{x}$ given that $\mathbf{y}$ was received $P(\mathbf{x}|\mathbf{y})$, may be characterized by the product of the *a priori* probability $P(\mathbf{x})$ that $\mathbf{x}$ was transmitted and of the probability that the signal $\mathbf{y}$ was received given that $\mathbf{x}$ was transmitted, namely $P(\mathbf{y}|\mathbf{x})$, as encapsulated in [1]

$$P(\mathbf{x}|\mathbf{y}) = \frac{P(\mathbf{y}|\mathbf{x})P(\mathbf{x})}{P(\mathbf{y})}, \tag{1}$$

where $P(\mathbf{y})$ is termed as the model's likelihood, which describes the probability of receiving $\mathbf{y}$ as [1]

$$P(\mathbf{y}) = \sum_{\mathbf{x}} P(\mathbf{y}|\mathbf{x})P(\mathbf{x}). \tag{2}$$

The numerator and denominator of the mth bit's LLR consist of a summation of the above-mentioned specific products, where the mth bit of the multi-level symbol is equal to 0 and 1, respectively. The QWSA proposed in [20] is capable

[1] For a tutorial on Quantum Search Algorithms, please refer to [20].

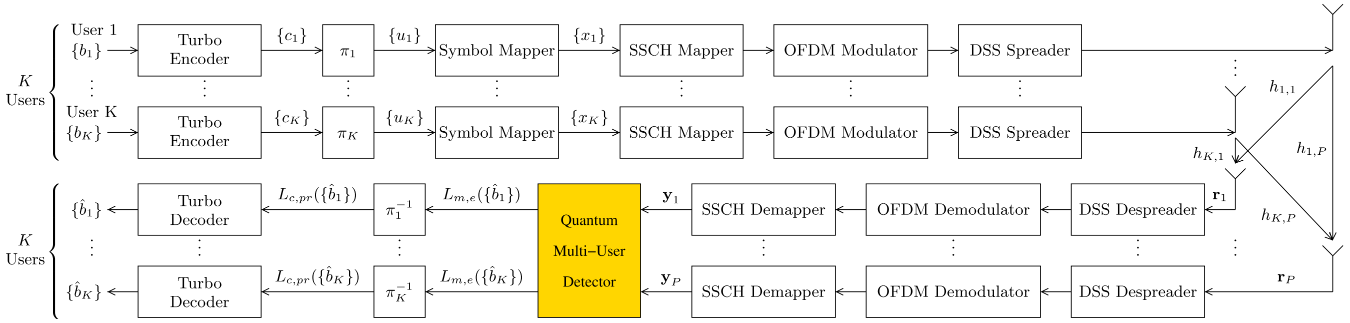


FIGURE 1. Direct Sequence Spreading and Slow Subcarrier-Hopping aided SDMA-OFDM uplink communication system's block diagram supporting K users employing turbo encoding using two convolutional codes as well as non-iterative, soft-output QMUD at the BS.

of computing the necessary summations of both the numerator and of the denominator of each LLR at the cost of requiring fewer CFEs than the SO-ML MUD.

In [22] we presented a fixed-complexity hard-output QMUD based on the Dürr-Høyer Algorithm (DHA) [23] which finds the index of an N-element database's smallest entry after $O(\sqrt{N})$ Look-Up Table (LUP) search steps. The DHA employs the Boyer-Brassard-Høyer-Tapp (BBHT) Quantum Search Algorithm (QSA) [18], which may be considered to be a generalized version of Grover's QSA [24]. The fixed-complexity hard-output QMUD of [22] succeeds in finding the specific multi-level symbol that has the highest CF value by evaluating the CF fewer times than the linear-search ML MUD.

Based on the current state-of-art, our novel contributions are:

1) *We propose a Soft-Output DHA-assisted QMUD relying on the MAA [11] and MUA [10], based on the corresponding SO-ACO MUD philosophy of [10], [11], showing that an improved performance is achieved by the SO-DHA QMUDs at a given fixed number of CFEs.*
2) *We demonstrate that the per iteration complexity of the proposed SO-DHA QMUD is lower than that of the QWSA QMUD [20], while their performance is similar, provided that a sufficient number of qubits is employed in the QCR of the QWSA QMUD.*
3) *We propose three variations of the MAA-based and MUA-based MUDs, where information is exchanged amongst the different detector components for the sake of improving the performance of both the SO-DHA QMUD and of the SO-ACO MUD, while keeping the per iteration complexity the same. The proposed variations have higher memory requirements and in some cases a delay is introduced between the MUD and the channel decoders.*
4) *We employ the SO-DHA QMUD and the SO-ACO MUD in vehicular scenarios of rank-deficient DSS/SSCH SDMA-OFDM systems, performing a three-dimensional detection of the supported users' signals in the FD, TD and SD.*
5) *We apply a mutation probability in the ACO algorithm similar to the mutation probability of [25], termed as the "free will" (FW) of the ants, that never allows them to converge to a local minimum or maximum point, which is otherwise a potential problem in rank-deficient systems. Moreover, the obtained information represented by the best so-far found symbol is exploited for further reducing the complexity of both the SO-DHA-based and SO-ACO-based MUDs.*
6) *We present a novel DSS-based Uniform SSCH (USSCH) algorithm, which takes into consideration the available DSS codes for allocating the users to the subcarriers. We show that a better performance is achieved than that of the USSCH [7], whilst requiring a lower complexity.*

The rest of the paper is structured as follows. The DSS/SSCH SDMA-OFDM system model is presented in Section II. The theoretical background on quantum computing and the QSAs employed is provided in Section III, while Section IV investigates the SO-DHA QMUD and its variations. Furthermore, the FW-aided SO-ACO MUD is analysed in Section V, while our simulation results are discussed in Section VI. Finally, our conclusions are offered in Section VII.

II. SYSTEM MODEL

A DSS/SSCH SDMA-OFDM system supporting K geographically separated users and Bit Interleaved Coded Modulation (BICM) is employed, as seen in the system's block diagram in Fig. 1. The information bit stream $\{b_k\}$ of the kth user is encoded by using Turbo Convolutional Codes (TCC) and the encoded bit stream $\{c_k\}$ is then interleaved. The interleaved encoded bit stream of the kth user $\{u_k\}$ is mapped onto the symbol stream $\{x_k\}$. Let us assume that there are Q available subcarriers, that the length of each user's symbol stream is Γ and that each user occupies W subcarriers during a time frame, where we have $W \leq Q$, $W \leq \Gamma$ and $\mod(\Gamma, W) = 0$. The user-specific subcarrier mapping pattern is periodically generated based on an appropriately selected approach. The SSCH mapper then maps the Q available subcarriers to the users, before the OFDM modulator converts the mapped symbols to OFDM symbols by applying the Q-point Inverse Fast Fourier Transform (IFFT) algorithm. The OFDM symbols are then spread in the time domain based on the user-specific spreading sequences. In our scenario we will assume that each user is allocated one of the $G = SF$ available Walsh-Hadamard (WH) spreading codes, which are orthogonal to each other. Without any loss of generality, the number of users employing the gth DSS code, with $g = 1, 2, \ldots, G$, is equal to

$$\left\lfloor \frac{K}{G} \right\rfloor + \begin{cases} 1 & \text{if } g < \text{mod}(K, G) \\ 0 & \text{if } g \geq \text{mod}(K, G) \end{cases}. \tag{3}$$

The spread signal is then transmitted over multipath channels and it is received by the P receive AEs, where the signals of all the users are added together, assuming the employment of a synchronous system, and Additive White Gaussian Noise (AWGN) is added. At the pth receive AE's chain, with $p = 1, 2, \ldots, P$, the received signal $\mathbf{r}_p$ is despread in the TD and the users employing orthogonal spreading codes are separated. The Fast Fourier Transform (FFT) algorithm is then employed at the OFDM demodulator. After the OFDM demodulated signal is dehopped on the pth receive AE's chain, it is fed to the QMUD along with the signals of the other receive AEs.

Let us focus on the discrete description of the signals that arrived at the pth receive AE on the qth subcarrier. Let us also assume that $1 \leq K_q \leq K$ users have been allocated to the qth subcarrier, while G_q different DSS codes are present on the qth subcarrier, where $1 \leq G_q \leq G$. Two users may or may not have been allocated the same DSS code. For the gth DSS code of the G_q number of present DSS codes on the qth subcarrier, there is a group of $K_{q,g}$ users who have been allocated the gth DSS code, where

$$K_q = \sum_{g=1}^{G_q} K_{q,g}. \tag{4}$$

The signal $\mathbf{r}_{p,q}$ received at the pth received AE, transmitted over the qth subcarrier, $q = 1, 2, \ldots, Q$, during an OFDM symbol duration is [7]

$$\mathbf{r}_{p,q} = \bar{\mathbf{c}}_{G_q} \bar{\mathbf{H}}_{p,q} \bar{\mathbf{x}}_q + \mathbf{n}_{p,q}, \tag{5}$$

where $p = 1, 2, \ldots, P$, $\bar{\mathbf{c}}_{G_q}$ is the $(SF \times K_q)$-element matrix that contains the DSS of each user as in [7]

$$\bar{\mathbf{c}}_{G_q} = [\underbrace{\mathbf{c}_1, \ldots, \mathbf{c}_1}_{k_1}, \underbrace{\mathbf{c}_2, \ldots, \mathbf{c}_2}_{k_2}, \ldots, \underbrace{\mathbf{c}_{G_q}, \ldots, \mathbf{c}_{G_q}}_{k_{G_q}}], \tag{6}$$

where $\mathbf{c}_g = [u_g[1], u_g[2], \ldots, u_g[SF]]^T$ is the gth DSS code and $u_g[i]$ is the value of the ith chip of the gth DSS code. In (5), the Frequency-Domain CHannel Transfer Function (FD-CHTF) of the pth receive AE and on the qth subcarrier $\bar{\mathbf{H}}_{p,q}$ is a $(K_q \times K_q)$-element diagonal matrix, as encapsulated in [7]

$$\begin{aligned} \bar{\mathbf{H}}_{p,q} = \; & \text{diag}[h^{(1)}_{p,1,q}, \ldots, h^{(k_1)}_{p,1,q}, \\ & h^{(1)}_{p,2,q}, \ldots, h^{(k_2)}_{p,2,q}, h^{(1)}_{p,G_q,q}, \ldots, h^{(k_{G_q})}_{p,G_q,q}], \end{aligned} \tag{7}$$

where $h^{(i)}_{p,g,q}$ is the complex-valued channel coefficient of the ith user in the gth DSS code group transmitting at the qth subcarrier and received by the pth receive AE. Finally, the $(1 \times K_q)$-element vector $\bar{\mathbf{x}}_q$ in (5) represents the signal vector on the qth subcarrier and its structure is

$$\bar{\mathbf{x}}_q = [x^{(1)}_{1,q}, \ldots, x^{(k_1)}_{1,q}, x^{(1)}_{2,q}, \ldots, x^{(k_2)}_{2,q}, x^{(1)}_{G_q,q}, \ldots, x^{(k_{G_q})}_{G_q,q}]^T, \tag{8}$$

while $\mathbf{n}_{p,q} = [n_{p,q}[1], \ldots, n_{p,q}[SF]]$ is the $(1 \times SF)$-element complex-valued thermal noise vector at the pth receive AE added to the signal received on the qth subcarrier with zero mean and σ^2 variance.

The $(G_q \times 1)$-element despread signal $\bar{\mathbf{y}}_{p,q} = [y_{p,1,q}, \ldots, y_{p,G_q,q}]^T$ at the pth receive AE from the qth subcarrier is [7]

$$\bar{\mathbf{y}}_{p,q} = \check{\mathbf{c}}_{G_q} \mathbf{r}_{p,q} = \bar{\mathbf{R}}_{G_q} \bar{\mathbf{H}}_{p,q} \bar{\mathbf{x}}_q + \bar{\mathbf{n}}_{p,q}, \tag{9}$$

where $\bar{\mathbf{n}}_{p,q} = [n_{p,1,q}, n_{p,2,q}, \ldots, n_{p,G_q,q}]^T$ is the effective noise vector, $\check{\mathbf{c}}_{G_q}$ is the DSS codebook that includes the DSS codes that appeared on the qth subcarrier as described in

$$\check{\mathbf{c}}_{G_q} = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{G_q}]^T, \tag{10}$$

and $\bar{\mathbf{R}}_{G_q}$ is the $(SF \times K_q)$-element cross-correlation matrix of the G_q DSS codes that were employed by the K_q users on the qth subcarrier, as given in

$$\bar{\mathbf{R}}_{G_q} = \begin{bmatrix} \omega_{11} \; \cdots \; \omega_{11} & \cdots & \omega_{1G_q} \; \cdots \; \omega_{1G_q} \\ \omega_{22} \; \cdots \; \omega_{22} & \cdots & \omega_{2G_q} \; \cdots \; \omega_{2G_q} \\ \vdots & \vdots & \vdots \\ \underbrace{\omega_{G_q1} \; \cdots \; \omega_{G_q1}}_{k_1} & \cdots & \underbrace{\omega_{G_qG_q} \; \cdots \; \omega_{G_qG_q}}_{k_{G_q}} \end{bmatrix}, \tag{11}$$

where $\omega_{i,j}$ is the cross-correlation between the ith and the jth DSS code [7].

A. MULTI-USER DETECTION

The QMUD will perform a subcarrier-based and DSS-code group-based detection of the users relying on the same DSS code and transmitting over the same subcarrier by combining the signals at the receive AEs, while outputting a soft estimate of each user's transmitted bit sequence in the form of Log-Likelihood Ratios. The hard output of the MMSE detector may be used as the initial input of the SO-DHA QMUDs, as well as for calculating the initial intrinsic affinity in the SO-ACO MUDs. The $(1 \times K_{q,g})$-element signal vector output of the MMSE detector at the qth subcarrier of the gth DSS group linearly combines the signals received by the P receive AEs $\mathbf{y}_{g,q} = [y_{1,g,q}, \ldots, y_{P,g,q}]^T$ as in

$$\mathbf{z}_{\text{MMSE}_{g,q}} = \mathbf{W}^H_{\text{MMSE}_{g,q}} \mathbf{y}_{g,q}, \tag{12}$$

where $\mathbf{z}_{\text{MMSE}_{g,q}} = [z^{(1)}_{\text{MMSE}_{g,q}}, \ldots, z^{(K_{q,g})}_{\text{MMSE}_{g,q}}]^T$, $g = 1, \ldots, G_q$, and $\mathbf{W}^H_{\text{MMSE}_{g,q}}$ is equal to

$$\mathbf{W}^H_{\text{MMSE}_{g,q}} = \begin{cases} \left(\mathbf{H}^H_{g,q}\mathbf{H}_{g,q} + N_0\mathbf{I}\right)^{-1}\mathbf{H}^H_{g,q} & \text{if } K_{q,g} \leq P \\ \mathbf{H}^H_{g,q}\left(\mathbf{H}_{g,q}\mathbf{H}^H_{g,q} + N_0\mathbf{I}\right)^{-1} & \text{if } K_{q,g} > P \end{cases}, \tag{13}$$

where N_0 is the noise's variance and $\mathbf{H}_{g,q}$ is the $(P \times K_{q,g})$-element FD-CHTF matrix of the users associated with the gth DSS code group at the qth subcarrier and it is equal to

$$\mathbf{H}_{g,q} = \begin{bmatrix} h_{1,g,q}^{(1)} & h_{1,g,q}^{(2)} & \cdots & h_{1,g,q}^{(K_{q,g})} \\ \vdots & \vdots & \vdots & \vdots \\ h_{P,g,q}^{(1)} & h_{P,g,q}^{(2)} & \cdots & h_{P,g,q}^{(K_{q,g})} \end{bmatrix}. \tag{14}$$

The LLR of the mth bit of the uth user in the gth DSS code group at the qth subcarrier may be calculated as [1]

$$L_{m,po}\left(b_u^{(m)}\right) = \ln \frac{\sum\limits_{\mathbf{x} \in \chi_{g,q}(u,m,0)} P\left(\mathbf{y}_{g,q}|\mathbf{x}\right) P(\mathbf{x})}{\sum\limits_{\mathbf{x} \in \chi_{g,q}(u,m,1)} P\left(\mathbf{y}_{g,q}|\mathbf{x}\right) P(\mathbf{x})}, \tag{15}$$

where $\chi_{g,q}(u, m, v)$ is the set of multi-level symbols formed by the users in the gth DSS code group at the qth subcarrier, for which the $[(u-1)\log_2 M + m]$th bit's value is equal to v, with $u = 1, \ldots, K_{q,g}$ and $m = 1, \ldots, \log_2 M$. Furthermore, $P(\mathbf{x})$ is the *a priori* probability of the $K_{q,g}$ users transmitting $\mathbf{x}$ and $P(\mathbf{y}_{g,q}|\mathbf{x})$ is the conditional probability of having obtained $\mathbf{y}_{g,q}$, given that $\mathbf{x}$ was transmitted by the $K_{q,g}$ users. The conditional probability $P(\mathbf{y}_{g,q}|\mathbf{x})$ is our CF $f(\mathbf{x})$, as encapsulated in [1]

$$f(\mathbf{x}) = P\left(\mathbf{y}_{g,q}|\mathbf{x}\right) = \exp\left(-\left\|\mathbf{y}_{g,q} - \mathbf{H}_{g,q}\mathbf{x}\right\|^2 / 2\sigma^2\right). \tag{16}$$

When the QMUD is invoked for the first time, the *a priori* probabilities $P(\mathbf{x})$ are equal for all possible vectors of $\mathbf{x}$. After the QMUD, the extrinsic LLR stream of each user is deinterleaved and then passed to the K Max-Log *A Posteriori* Probability (APP) decoders. By making a hard decision on the soft outputs of the kth decoder we may estimate the kth user's information bit stream $\{\hat{b}_k\}$.

B. SLOW FREQUENCY HOPPING

The Slow Frequency Hopping (SFH) methodology [7] divides the Q available subcarriers into Q/W number of subbands having W subcarriers in each subband, as illustrated in Fig. 2(a). Each of the K supported users is allocated the W subcarriers of a subband. A disadvantage of SFH is that if a subcarrier is deeply faded, there is a high probability that its adjacent subcarriers are also experiencing fading during the same OFDM frame. Therefore, the users allocated to that particular subband, which contains the fading subcarriers, experience fading on most of the subcarriers, resulting in a degraded performance. For circumventing this problem, we employ the SSCH methodology.

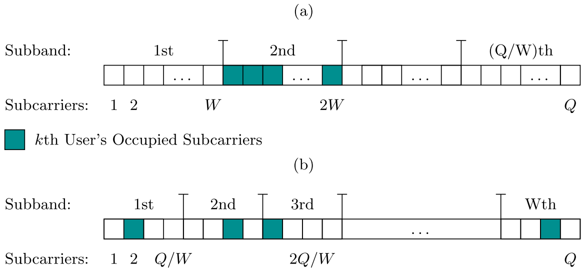


FIGURE 2. (a) The SFH methodology, which allocates one of the Q/W subbands, or, in other words, W adjacent subcarriers to each user. (b) The SSCH methodology, which allocates one subcarrier of each of the W formed subbands to each user. Hence, each user is allocated W non-adjacent subcarriers.

C. DSS-BASED UNIFORM SLOW SUBCARRIER HOPPING

According to the SSCH specifications [7], the Q available subcarriers are divided into W subbands with $SB = Q/W$ subcarriers in each subband, as shown in Fig. 2(b). The SSCH regime allocates a single subcarrier from each of the W subbands to the kth user. As in the SFH methodology, by employing the SSCH methodology each user transmits on W subcarriers, with the difference that the allocated subcarriers are no longer adjacent. Therefore, there is a low probability that most of the subcarriers of a user will be fading.

The Random SSCH (RSSCH) scheme allocates subcarriers to users on a subband basis, by allocating a random subcarrier of each subband to the kth user, with $k = 1, 2, \ldots, K$. By adopting the RSSCH methodology, we may encounter the incident where some subcarriers will be unnecessarily loaded with a high number of users, while others may support a single user or even no users at all.

The USSCH also performs subband-based subcarrier mapping by firstly allocating each subcarrier of the wth subband, $w = 1, 2, \ldots, W$, to a different, randomly selected user, until each of the subcarriers in the wth subband has been allocated to a user. Afterwards, restarting from the first subcarrier of the wth subband, a second user is allocated to each subcarrier. The above procedure is repeated until all the users have been allocated a subcarrier in the wth subband. The USSCH continues by allocating the subcarriers of the $(w + 1)$th subband to the users with the same procedure. The USSCH adopts a specific subcarrier mapping strategy, where the number of users on each subcarrier, and hence the Multi-User Interference (MUI), is similar.

The DSS-based USSCH follows the methodology of the USSCH, apart from the difference that when a user is randomly allocated to a subcarrier, then the rest of the users who belong to the same DSS-code group as the already allocated user will be allocated to subcarriers, before the users who belong to a different DSS-code group. Once all the users of that particular DSS-code group have been allocated to a subcarrier of the current subband, the procedure continues with one of the remaining users being randomly selected to be allocated to the subsequent subcarrier. Afterwards, the rest of the users in the same DSS-code group as the last randomly selected user will be allocated to subcarriers, before a user from a different DSS-code group is randomly selected again. By following the DSS-based USSCH, the users who belong to the same DSS-code group are allocated to as many subcarriers as possible, hence reducing the MUI on each subcarrier. Figure 3 compares the USSCH to the DSS-based USSCH philosophy and shows the advantages of the DSS-based USSCH, such as the achievement of a better system performance, while supporting the same number of users or the capability of supporting more users at the same complexity in the MUD stage.

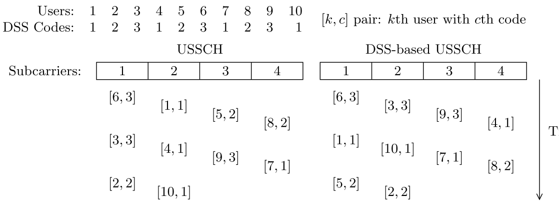


FIGURE 3. Comparison between the USSCH [7] and the proposed DSS-based USSCH in an example with $K = 10$ users, $Q/W = 4$ subcarriers per subband and 3 DSS-codes. If USSCH is employed, the presented scenario may occur and we may have MUI by 2 users at the first subcarrier of that particular subband, as well as MUI by 3 users at the second subcarrier. It should be noted that employing USSCH may result in different distribution of users on each subband, and hence different amount of MUI. On the other hand, if DSS-based USSCH is followed in this scenario, there is no MUI at any of the subcarriers in all subbands with 100% probability, since its methodology keeps MUI to the smallest possible value by distributing users with the same DSS-code to different subcarriers.

The USSCH, as well as the DSS-based USSCH, RSSCH and SFH are periodically performed with a period of T_h for making the communication fairer for the specific users who happen to transmit on deeply fading subcarriers. It is noted that T_h should be higher than the symbol period T_s, for coherent demodulation to be used at the receiver. In practice, T_h will be equal to an integer number of symbol periods.

III. QUANTUM SEARCH ALGORITHMS

The quantum bit or *qubit* is the equivalent of a classic bit in quantum computation [14]–[16]. A qubit $|\psi\rangle$ may assume not only the $\{|0\rangle, |1\rangle\}$ states of the computational basis, but it may also be in a superposition of the states of an orthonormal basis, as in $|\psi\rangle = a|0\rangle + b|1\rangle$ for the computational basis, where $|a|^2$ and $|b|^2$ are the probabilities of observing or measuring the qubit $|\psi\rangle$ in the quantum states $|0\rangle$ and $|1\rangle$, respectively, with $|a|^2 + |b|^2 = 1$ and $a, b \in \mathbb{C}$.

A quantum system may rely on symbols consisting of multiple qubits and the quantum states evolve by applying unitary operators U to them. The most widely used unitary operator is the Hadamard gate H [14]–[16], which evolves the quantum state $|0\rangle$ to $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|1\rangle$ to $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. If two qubits exist in a quantum system, then the state of the quantum system is

$$\begin{aligned}|\varphi\rangle &= |\psi_1\rangle \otimes |\psi_2\rangle = |\psi_1\rangle|\psi_2\rangle = (x|0\rangle + y|1\rangle)(w|0\rangle + z|1\rangle)\\ &= x \cdot w|00\rangle + x \cdot z|01\rangle + y \cdot w|10\rangle + y \cdot z|11\rangle\\ &= a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle, \qquad (17)\end{aligned}$$

where $|a|^2+|b|^2+|c|^2+|d|^2 = 1$. The value $|x|^2 = |a|^2+|b|^2$ is equal to the probability of observing the first qubit in the state $|0\rangle$. If the first qubit is observed in the state $|0\rangle$, then the quantum system in (17) would be in the quantum state $|\varphi\rangle = |0\rangle(w|0\rangle + z|1\rangle) = w|00\rangle + z|01\rangle$. In other words, an observation of the first qubit does not affect the superposition of states the second qubit is in. On the other hand, if a quantum system is in the state $|\varphi\rangle = a|00\rangle + b|11\rangle$, the qubits are *entangled*, since their states cannot be described separately as $|\varphi\rangle = |\psi_1\rangle|\psi_2\rangle$. If the first qubit is observed, then the second qubit's quantum state is also altered and vice versa.

In the rest of this paper we only assume real amplitudes for the quantum states. Furthermore, in a DSS/SSCH SDMA-OFDM system, the MUDs search a pool of $M^{K_{q,g}}$ legitimate inputs to the CF of (16). The symbol index $x \in \{0, \ldots, M^{K_{q,g}} - 1\}$ is the decimal representation of the $M^{K_{q,g}}$-ary symbols. For example, if $M = 4$ and $K_{q,g} = 2$, then $\mathbf{x} = [+1 - j, -1 + j]$ demodulates into $[b_{u=0}^{(m=0)}|b_{u=0}^{(m=1)}|b_{u=1}^{(m=0)}|b_{u=1}^{(m=1)}] = [01|10]$, which corresponds to the index $x = 6$.

A. GROVER'S QUANTUM SEARCH ALGORITHM

In a search problem, given a database or function f and a known value δ, we search for an x so that $f(x) = \delta$. The values of x that satisfy $f(x) = \delta$ are termed as *solutions* to the search problem. When there is only a single solution $S = 1$ in an unsorted search space of $N = M^{K_{q,g}}$ entries, then the optimal classic linear-search algorithm succeeds in finding it after $O(N)$ CFEs. In [24] and [26] Grover proposed a QSA, which finds the position of a user-defined value in a search space of size N after $O(\sqrt{N})$ CFEs with approximately 100% success probability. In [18] Grover's QSA was employed for search problems with $S \geq 1$ and it was shown to be able to find one of the solutions after $O(\sqrt{N/S})$ CFEs.

In greater detail, Grover's QSA commences by preparing $n = \log_2 N$ qubits in an equiprobable superposition of states by passing each of the n qubits initially in the $|0\rangle^{\otimes n}$ state through a Hadamard gate H as in

$$|\gamma\rangle = \sum_{\psi=0}^{N-1} \frac{1}{\sqrt{N}}|\psi\rangle. \qquad (18)$$

Furthermore, the Grover operator $\mathcal{G}$ is applied to the quantum state $|\gamma\rangle$ for $L_{opt} = \lfloor \pi/4\sqrt{N/S} \rfloor$ number of times [24]. The Grover operator is equal to $\mathcal{G} = HP_0H \cdot O$, where P_0 rotates $|x\rangle \rightarrow -|x\rangle$ if and only if $|x\rangle \neq |0\rangle$, H is the Hadamard operator and O is the Oracle detailed in [20] and [24]. To elaborate a little further, the Oracle is capable of marking all the solutions in the search problem by evaluating f in parallel and altering $|q\rangle$ to $-|q\rangle$ if and only if we have $f(q) = \delta$. Since the complexity of a single evaluation of the Oracle operator O will depend on the particular technology employed to create it, let us proceed by assuming that an application of the Oracle, or, equivalently, a CFE in the Quantum Domain (QD-CFE) has the same complexity as a CFE in the Classic Domain (CD-CFE) [18], [24], [26].

After applying the Grover operator L_{opt} number of times, the resultant quantum state $\mathcal{G}^{L_{opt}}|\gamma\rangle$ is observed in the computational basis $\{|0\rangle, |1\rangle\}^{\otimes n}$ and an index x is obtained. The

Algorithm 1 BBHT Quantum Search Algorithm [22]

```
1: Set $m \leftarrow 1$, $\lambda \leftarrow 6/5$ and $L_{BBHT}^{QD} \leftarrow 0$, $L_{BBHT}^{CD} \leftarrow 0$.
2: Choose $L$ uniformly from the set $\{0, \dots, \lfloor m \rfloor\}$.
3: Apply the $\mathcal{G}$ operator $L$ times starting from the initial state $|x\rangle$
   in (18), resulting in the final state $|x_f\rangle = \mathcal{G}^L|x\rangle$.
4: Observe $|x_f\rangle$ in the QD and obtain $|j\rangle$.
5: Compute $f(j)$ in the CD.
6: Update $L_{BBHT}^{CD} \leftarrow L_{BBHT}^{CD} + 1$ and $L_{BBHT}^{QD} \leftarrow L_{BBHT}^{QD} + L$.
7: if $f(j) = \delta$ or $L_{BBHT}^{QD} \geq L_{BBHT}^{QD,\max}$ then
8:     Set $x_s \leftarrow j$, output $x_s$, $L_{BBHT}^{CD}$, $L_{BBHT}^{QD}$ and exit.
9: else
10:    Set $m \leftarrow \min\left\{\lambda m, \sqrt{N}\right\}$ and go to Step 2.
11: end if
```

Algorithm 2 Deterministically-Initialized DHA [22]

```
1: Furthermore, set $i \leftarrow x_I$ and $L_{DHA} \leftarrow 0$, $L_{DHA}^{CD} \leftarrow 0$,
   $L_{DHA}^{QD} \leftarrow 0$.
2: The BBHT QSA is employed with $\delta \leftarrow f(i)$, an Oracle that
   marks as solutions the states $|x\rangle$ that obey $f(x) < \delta$ and
   $L_{BBHT}^{QD,\max} \leftarrow 4.5\sqrt{N}$. Obtain $x_s$, $L_{BBHT}^{CD}$ and $L_{BBHT}^{QD}$ from the
   BBHT QSA.
3: $L_{DHA}^{CD} \leftarrow L_{DHA}^{CD} + L_{BBHT}^{CD}$, $L_{DHA}^{QD} \leftarrow L_{DHA}^{QD} + L_{BBHT}^{QD}$ and
   $L_{DHA} \leftarrow L_{DHA} + L_{DHA}^{CD} + L_{DHA}^{QD}$.
4: if $f(x_s) \geq f(i)$ or $L_{DHA} \geq 22.5\sqrt{N}$, then
5:     Set $x_{\min} \leftarrow i$, output $x_{\min}$ and exit.
6: else
7:     Set $i \leftarrow x_s$ and go to Step 2.
8: end if
```

probability of the index x to be a solution is $P_{success} = \sin^2[(2L_{opt} + 1)\theta]$, where $\theta = \arcsin(\sqrt{S/N})$ [24].

B. THE BOYER, BRASSARD, HØYER, TAPP QUANTUM SEARCH ALGORITHM

Boyer *et al.* proposed the BBHT QSA [18] based on Grover's QSA, with the added benefit that it succeeds in finding a solution with ~100% success probability even when the number of solutions S is unknown *a priori*. By contrast, Grover's QSA would be unable to determine the optimal number L_{opt} the Grover operator $\mathcal{G}$ has to be applied to $|\gamma\rangle$ in (18). For circumventing this problem, the BBHT QSA applies $\mathcal{G}$ for a pseudorandom number of times L following a specific pattern, as described in Algorithm 1 of [22]. Afterwards, the resultant quantum state $\mathcal{G}^L|\gamma\rangle$ is observed and $|j\rangle$ is obtained. If $f(j) = \delta$, then a solution has been found, yielding $x_s = j$. Otherwise, another value is selected for L and the above-mentioned process is repeated, until either a solution is found, or, in the case when $S = 0$, the number of QD-CFEs L_{BBHT}^{QD} has exceeded the predetermined upper bound of [18] $L_{BBHT}^{QD,\max} = 4.5\sqrt{N/S}$. In the latter case it is concluded that the search problem has no solutions. The complexity of the BBHT QSA is given by the sum of the number of QD-CFEs and the number of CD-CFEs, as in $L_{BBHT} = L_{BBHT}^{QD} + L_{BBHT}^{CD}$.

C. DETERMINISTICALLY-INITIALIZED DÜRR-HØYER ALGORITHM

The DHA of [23] is capable of finding the specific index $x_{\min}$, which minimizes the CF f of (16), as illustrated in $f(x_{\min}) \leq f(x)$, $x = 0, \dots, N-1$, after $O(\sqrt{N})$ CFEs with ~100% probability, while the optimal classic algorithms need $O(N)$ CFEs for finding $x_{\min}$. The deterministically-initialized DHA was proposed in [22], which is compactly summarized in Algorithm 2. Explicitly, the difference between the deterministically-initialized DHA of [22] and the original DHA [23] is that in Step 1 of the former algorithm, the initial value i is set to a predetermined value x_I, instead of randomly selecting a value for it as in the latter algorithm. The main advantage of the deterministically-initialized DHA of [22] over the randomly-initialized DHA of [23] is that the initial input may correspond to a CF value, which is closer to that of the globally optimal symbol. Therefore, the DHA will find the globally optimal symbol with fewer CFEs [22]. A beneficial predetermined value for our QMUD is the specific index of the $M^{K_{q,g}}$-ary symbol at the output of the MMSE detector.

The BBHT QSA of Section III-B is then employed in Step 2 of the DHA, apart from the slight deviation that the Oracle of the BBHT QSA marks as legitimate solutions those states $|q\rangle$ that correspond to $f(q) < f(i)$. By invoking the BBHT QSA in Step 2 of the DHA multiple times while updating i with an index that corresponds to a lower CF value, the DHA becomes capable of obtaining $x_{\min}$ in less than $22.5\sqrt{N}$ QD-CFEs, as shown in [20] and [23]. If the initial value x_I is equal to $x_{\min}$, then the BBHT QSA times out after $4.5\sqrt{N}$ QD-CFEs since $S = 0$, while allowing us to conclude that $x_{\min}$ was already found before the last call of Step 2. The minimum number of CD-CFEs in the DHA is [22]

$$L_{DHA}^{CD,\min} = \min\left(L_{DHA}^{CD}\right) + 1$$
$$s.t. \left[\sum_{j=0}^{L_{DHA}^{CD}-1} \min\left(\left\lfloor \lambda^j \right\rfloor, \sqrt{N}\right)\right] \geq 4.5\sqrt{N}. \quad (19)$$

IV. SOFT-OUTPUT DHA-AIDED QMUD

The Soft-Output Maximum Likelihood (SO-ML) MUD calculates the CF values of all the legitimate $M^{K_{q,g}}$-ary symbols based on (16) and uses them for calculating each bit's LLR according to (15). It was proposed in [10] and [11] that even if a reduced subset of the CFEs of all the legitimate $M^{K_{q,g}}$-ary symbols is used for the calculation of each bit's LLR, the performance would be near-optimal with respect to that of the SO-ML MUD, provided that the subset of CF values includes the maximum possible CF values of the numerator and the denominator of each bit's LLR. Hence in [10] and [11] an ACO-based search was employed for selecting the particular symbol subset used for the calculation of the LLRs. The main challenges of the ACO invoked for the selection of the symbol subsets were identified in [10] and [11] as being two fold. Firstly, when the ACO converges either to a local or to the global maximum, all the ants follow the same path with ∼100% probability, hence the search is converged and concluded. Moreover, in rank-deficient systems where there are more transmit AEs than receive AEs, the ACO rarely converges to the globally optimal symbol in that particular search space, which naturally leads to an erroneously calculated LLR, since the selected subset of symbols does not include the globally maximum values.

As a beneficial design alternative, the DHA may be employed for performing SO MUD by finding the specific symbol that minimizes $-f(x)$ of (16), or, equivalently, maximizes $f(x)$ of (16), where x is the decimal representation, or index, of $\mathbf{x}$ in (16). Let us denote the true globally optimal symbol as $x_{\max}$ and the best symbol found by the DHA as $\hat{x}_{\max}$, even though we are aware that $\hat{x}_{\max} = x_{\max}$ with ∼100% probability. As illustrated in Fig. 4 for a system associated with QPSK modulation ($M = 4$) and $K_{1,1} = 3$ users relying on the first DSS code on the first subcarrier, the DHA commences its search from an initial symbol and makes its way to the symbol that maximizes $f(x)$, while evaluating numerous other symbols due to the probabilistic nature of the BBHT QSA. Observe in the scenario of Fig. 4, that the symbols with indices of 58, 57, 21, 17 and 61 have been evaluated after the point where the symbol with index 43 has been found, before finally symbol 8 is found, which has a higher CF value than that of symbol 43. The particular symbols that were evaluated during the BBHT QSA calls but were not found to be better than the "best-so-far" symbol may also be used for the calculation of the LLRs, since they have already been evaluated. Again, the DHA has ∼100% success probability in finding the maximum of the $f(x)$ CF of (16) even in rank deficient systems, since it is independent of both the actual CF values and of the "neighbourhood criterion"[2] of the evolutionary algorithms [7]. Furthermore, the probabilistic nature of the BBHT QSA provides the necessary candidate diversity for a potential SO-DHA QMUD.

Let us now proceed conceiving the SO-DHA QMUDs, which are less complex than those presented in [10] and [11]. Moreover, let us assume that the discussions are based on performing MUD at the qth subcarrier, $q = 1, 2, \ldots, Q$ of a DSS/USSCH SDMA-OFDM system, where $K_{q,g}$ users have been allocated the gth DSS code and they transmit M-ary symbols, implying that the database's size is $N_{q,g} = M^{K_{q,g}}$.

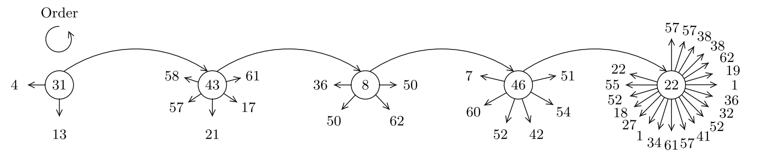
A. SO-DHA QMUD WITH MAXIMUM APPROXIMATION

The SO-DHA QMUD relying on MAA (SO-DHA-MAA QMUD) employs the DHA for finding the specific symbol that maximizes the CF $f(x)$ in (16). Focusing on the mth bit of the $N_{q,g}$-ary sybmol, only the pair of best symbols found for each of the mth bit's 0 and 1 logical values is included in the calculation of that bit's LLR. Since the DHA succeeds in finding the globally best symbol with ∼100% probability, that symbol's CF value will be included in all the LLR calculations, resulting in achieving the same LLR signs as the ML MUD. The unique symbols that were evaluated during the DHA form the evaluated symbol set $\mathcal{X}$. Let us also define the sets $\mathcal{X}_{q,g}^{u,m,\nu}$, which are specific subsets of $\mathcal{X}$, namely those that include the particular symbols for which the $(u \cdot \log_2 M + m)$th bit of the corresponding binary representation is equal to ν, as encapsulated in

$$x \in \mathcal{X}_{q,g}^{u,m,\nu} \Leftrightarrow x{\in}\mathcal{X} \wedge x{\in}\chi_{q,g}(u, m, \nu), \tag{20}$$

[2] The evolutionary algorithms follow a "neighbourhood criterion," where the search space of each generation is created around the best-so-far found symbol.

Symbol / Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Cost Function Value	$9.6 \cdot 10^{-4}$	0.0059	0.086	0.53	$2.4 \cdot 10^{-5}$	$4.9 \cdot 10^{-7}$	0.034	$7 \cdot 10^{-4}$	0.76	0.29	0.23	0.087	0.019	$2.4 \cdot 10^{-5}$	0.092	$1.2 \cdot 10^{-4}$
Symbol / Index	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Cost Function Value	0.0089	0.43	0.0082	0.39	0.064	0.01	0.95	0.16	0.065	0.19	$2 \cdot 10^{-4}$	$6 \cdot 10^{-4}$	0.46	0.0047	0.024	$2.4 \cdot 10^{-4}$
Symbol / Index	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
Cost Function Value	$3.2 \cdot 10^{-8}$	$1.9 \cdot 10^{-5}$	$2.3 \cdot 10^{-5}$	0.013	$8.5 \cdot 10^{-8}$	$1.7 \cdot 10^{-7}$	$9.7 \cdot 10^{-4}$	0.0019	0.0074	0.27	0.018	0.65	0.02	0.0025	0.77	0.095
Symbol / Index	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
Cost Function Value	$3 \cdot 10^{-7}$	0.0014	$2.2 \cdot 10^{-6}$	0.01	$2.3 \cdot 10^{-4}$	0.0037	0.027	0.43	$6.3 \cdot 10^{-4}$	0.18	$1.6 \cdot 10^{-5}$	0.0045	0.49	0.48	0.2	0.19



Unique Symbols Evaluated: 31, 4, 13, 43, 58, 57, 21, 17, 61, 8, 36, 50, 62, 46, 7, 60, 52, 42, 54, 51, 22, 55, 18, 27, 1, 34, 41, 32, 19, 38

FIGURE 4. **Scenario of applying the DHA for finding the $M^{K_{q,g}}$-ary symbol that maximizes the CF in (16) in a rank-deficient DSS/USSCH SDMA-OFDM system where $K_{q,g} = K_{1,1} = 3$ users coexist on the first subcarrier having been allocated the first DSS code and they transmit QPSK symbols associated with $M = 4$ to $P = 1$ receive AE. Top: The database of symbol indices and the corresponding CF values. The rectangular box indicates the optimal symbol index associated with the maximum CF value, while the ellipse encircles the MMSE detector's output which is used as the initial guess in the DHA. Bottom: The DHA process. The circles indicate the best so far found symbols. Starting from the leftmost circle and moving counter-clockwise, the BBHT QSA is invoked for finding a symbol that has a higher CF value than the symbol in that particular circle. Once a better symbol is found, the BBHT QSA is restarted for the new symbol. Once the global best symbol has been found, the BBHT QSA times out after $4.5\sqrt{M^{K_{q,g}}} = 36$ QD-CFEs. The number of QD-CFEs and CD-CFEs in the DHA in this small-scale system were 52 and 41, respectively. Therefore, the total number of CFEs was 93, which is greater than the total number of symbols. The SO-DHA QMUD is designed for large-scale systems where its complexity becomes much lower than that of the ML MUD.**

Algorithm 3 SO-DHA QMUD With MAA and NE

```
$K_{q,g}$ users, $M$-ary modulation, $N_{q,g} \leftarrow M^{K_{q,g}}$.
$\mathcal{X}_{q,g}^{u,m,\nu} \leftarrow \emptyset$, for $u = 1, 2, \ldots, K_{g,q}$, $m = 1, 2, \ldots, \log_2 M$ and $\nu = 0, 1$.
The DHA is employed for finding the symbol $\hat{x}_{\max}$ that maximizes the CF in (16) with $\sim$100% success probability after searching the whole search space $N_{q,g}$. Update the sets $\mathcal{X}$ and $\mathcal{X}_{q,g}^{u,m,\nu}$, for $u = 1, \ldots, K_{g,q}$, $m = 1, \ldots, \log_2 M$, $\nu = 0, 1$, when a unique symbol is evaluated by the DHA.
for $u = 1, 2, \ldots, K_{q,g}$ do
    for $m = 1, 2, \ldots, \log_2 M$ do
        if $\hat{x}_{\max} \in \mathcal{X}_{q,g}^{u,m,0}$ then
            Include symbol $x_{nb}$ in $\mathcal{X}_{q,g}^{u,m,1}$, where the binary representation of $x_{nb}$ is the same as that of $\hat{x}_{\max}$ with the difference that the value of the $[(u-1)\cdot\log_2 M + m]$th bit is equal to 1.
        else if $\hat{x}_{\max} \in \mathcal{X}_{q,g}^{u,m,1}$ then
            Include symbol $x_{nb}$ in $\mathcal{X}_{q,g}^{u,m,0}$, where the binary representation of $x_{nb}$ is the same as that of $\hat{x}_{\max}$ with the difference that the value of the $[(u-1)\cdot\log_2 M + m]$th bit is equal to 0.
        end if
        Calculate the LLR of the $[(u-1)\cdot\log_2 M + m]$th bit according to (21).
    end for
end for
```

where $\wedge$ denotes logical conjunction. Therefore, in our example of Fig. 4, the LLR of the first user's first bit using the MAA would be

$$L_{m,po}\left(b_{u=1}^{(m=1)}\right) = \ln \frac{\max\left(f(x)|x \in \mathcal{X}_{q,g}^{u,m,0}\right)}{\max\left(f(x)|x \in \mathcal{X}_{q,g}^{u,m,1}\right)} = \ln \frac{f(22)}{f(46)}, \tag{21}$$

since the globally best symbol with index $22_{10} = [01|01|10]_2$ corresponds to $b_1^{(1)} = 0$ and the best symbol found with $b_1^{(1)} = 1$ is symbol 46. If the scenario is encountered, where the set $\mathcal{X}_{q,g}^{u,m,\nu}$ is empty for a specific $[q, g, u, m, \nu]$ value, then according to [10], [11] another search process should be initiated for finding the specific symbol that maximizes the CF having fixed the mth bit of the uth user associated with the gth DSS code at the qth subcarrier to the value ν. Reasonably, the new search space has half the size compared to the one in the first DHA operation of the SO-DHA-MAA QMUD, but the resultant complexity may prove to be severe, especially if more than one of the $\mathcal{X}_{q,g}^{u,m,\nu}$ sets are found empty after the first DHA call. Hence, for the sake of circumventing this problem, we propose a Neighbour Exploitation (NE) based complexity reduction technique.

1) NEIGHBOUR EXPLOITATION

If a set $\mathcal{X}_{q,g}^{u,m,\nu}$ is empty, it means that the best symbol $\hat{x}_{\max}$ found by the DHA belongs to the set $\mathcal{X}_{q,g}^{u,m,\bar{\nu}}$, where $\bar{\nu} = \nu \oplus 1$ and $\oplus$ denotes the modulo-2 addition. According to the proposed NE-based modification, the set $\mathcal{X}_{q,g}^{u,m,\nu}$ will include the CF value of the neighbour of the globally best found symbol, where the uth user's mth bit will be equal to $\bar{\nu}$. In our example of Fig. 4, if the indices of all the symbols found by the DHA were smaller than 32, then the set $\mathcal{X}_{q,g}^{0,0,1}$ would be void. Since we have $\hat{x}_{\max} = 22 = [01|01|10]$, the symbol with index $[11|01|10] \Rightarrow x_{nb} = 54$, where nb stands for neighbour, would then be included in the set $\mathcal{X}_{q,g}^{0,0,1}$ according to the NE modification.

The NE contributes an additional CD-CFE to the MUD's complexity every time it is performed, since the new symbol's CF value has to be calculated. It may also be used for improving the MUD's performance even if the set that does not include $\hat{x}_{\max}$ is not empty. In that case x_{nb} is added to that set and it may be used for the calculation of that specific LLR provided that its CF value is the highest one in that set. The steps of the SO-DHA-MAA-NE QMUD are stated in Algorithm 3 and the corresponding flow chart is given in Fig. 5.

2) COMPLEXITY

The complexity of the first DHA employment is $O(\sqrt{N_{q,g}})$, since it is applied to the entire available search space, while the complexity of any subsequent DHA employment is $O(\sqrt{N_{q,g}/2})$. More specifically, if the first DHA application encountered symbols for both values of each bit in the SO-DHA-MAA QMUD, the minimum complexity per bit is

$$C_{SO-DHA}^{MAA,\min} = \frac{L_{DHA}^{QD,\min} + L_{DHA}^{CD,\min}}{K_{q,g} \cdot \log_2 M} = \frac{4.5\sqrt{M^{K_{q,g}}} + L_{DHA}^{CD,\min}}{K_{q,g} \cdot \log_2 M} \tag{22}$$

CFEs, where $L_{DHA}^{CD,\min}$ is given in (19) in the specific scenario, where the MMSE detector's output is equal to $\hat{x}_{\max}$. Therefore the DHA will realize that the globally optimal value has already been found after $4.5\sqrt{N_{q,g}}$ QD-CFEs. On the other hand, if no symbols were found for a specific bit's value, the DHA has to be called again for the respective search space, which has half the size of the total search space. The complexity per bit for the SO-DHA-MAA QMUD, when there are two DHA calls in total is

$$\acute{C}_{SO-DHA}^{MAA,\min} = \frac{L_{1,DHA}^{QD,\min} + L_{1,DHA}^{CD,\min} + L_{2,DHA}^{QD,\min} + L_{2,DHA}^{CD,\min}}{K_{q,g} \cdot \log_2 M} = \frac{7.682\sqrt{M^{K_{q,g}}} + L_{1,DHA}^{CD,\min} + L_{2,DHA}^{CD,\min}}{K_{q,g} \cdot \log_2 M} \tag{23}$$

CFEs. The minimum complexity per bit of the SO-DHA-MAA-NE QMUD is

$$C_{SO-DHA}^{MAA,NE,\min} = \frac{L_{DHA}^{QD,\min} + L_{DHA}^{CD,\min} + K_{q,g} \cdot \log_2 M}{K_{q,g} \cdot \log_2 M} = \frac{4.5\sqrt{M^{K_{q,g}}} + L_{DHA}^{CD,\min}}{K_{q,g} \cdot \log_2 M} + 1 \tag{24}$$

CFEs. The third term in (24) corresponds to the CFEs required for finding the neighbours of $\hat{x}_{\max}$, once for each bit of the multi-level symbol and it is independent of the DHA. Note that the complexity per bit of the SO-DHA-MAA-NE QMUD in (24) includes one more CFE compared to the complexity of the SO-DHA-MAA QMUD, if the DHA is applied only once. By contrast, if the DHA is called at least twice in the SO-DHA-MAA QMUD, its complexity formulated in (23) is significantly higher than that of the SO-DHA-MAA-NE QMUD.

B. SO-DHA QMUD WITH MULTI-INPUT APPROXIMATION

The main differences between the SO-DHA-MAA QMUD and the SO-DHA QMUD with MUA (SO-DHA-MUA QMUD) are that multiple symbols take part in the calculation of the LLRs in the MUA. More specifically, commencing from the first user's $u = 1$ first bit's $m = 1$ LLR calculation, the DHA is invoked for its numerator's search space, having fixed the $[(u-1) \cdot \log_2 M + m] = 1$st bit to $\nu = 0$, and then also for the denominator's search space, where only the specific symbols with the 1st bit being $\nu = 1$ are considered. The two sets $\mathcal{X}_{q,g}^{u,m,0}$ and $\mathcal{X}_{q,g}^{u,m,1}$ are filled by the DHAs' outputs in a sorted order. Since the two DHA activations have jointly searched the entire search space, the globally maximal symbol $\hat{x}_{\max}$ corresponds to the maximum CF valued entries $\hat{x}_{\max}^{u,m,0}$ and $\hat{x}_{\max}^{u,m,1}$ of the two sets $\mathcal{X}_{q,g}^{u,m,0}$ and $\mathcal{X}_{q,g}^{u,m,1}$, respectively. Therefore, by comparing the maximum CF valued entries of the two sets we may determine $\hat{x}_{\max}$ as in

$$\hat{x}_{\max} = \begin{cases} \hat{x}_{\max}^{u,m,0} & \text{if } f\left(\hat{x}_{\max}^{u,m,0}\right) > f\left(\hat{x}_{\max}^{u,m,1}\right) \\ \hat{x}_{\max}^{u,m,1} & \text{if } f\left(\hat{x}_{\max}^{u,m,0}\right) \leq f\left(\hat{x}_{\max}^{u,m,1}\right), \end{cases} \tag{25}$$

and hence the sign of the $[(u-1) \cdot \log_2 M + m]$th bit's LLR is given by [10]

$$\varsigma_{u,m} = \text{sign}\left[L_{m,po}\left(b_u^{(m)}\right)\right] = \begin{cases} +1 \text{ if } \hat{x}_{\max} \in \mathcal{X}_{q,g}^{u,m,0} \\ -1 \text{ if } \hat{x}_{\max} \in \mathcal{X}_{q,g}^{u,m,1}. \end{cases} \tag{26}$$

The LLR sign $\varsigma_{u,m}$ will be exploited for selecting the specific entries in the sets $\mathcal{X}_{q,g}^{u,m,0}$ and $\mathcal{X}_{q,g}^{u,m,1}$ that will participate in the calculation of the LLR. Let us denote the size of $\mathcal{X}_{q,g}^{u,m,0}$ by $|\mathcal{X}_{q,g}^{u,m,0}|$ and that of $\mathcal{X}_{q,g}^{u,m,1}$ as $|\mathcal{X}_{q,g}^{u,m,1}|$. The two sets may consist of a different number of entries, therefore $|\mathcal{X}_{q,g}^{u,m,0}|$ may or may not be equal to $|\mathcal{X}_{q,g}^{u,m,1}|$. If $|\mathcal{X}_{q,g}^{u,m,0}| > |\mathcal{X}_{q,g}^{u,m,1}|$ is true, then the $|\mathcal{X}_{q,g}^{u,m,0}| - |\mathcal{X}_{q,g}^{u,m,1}|$ entries with the smallest CF value in $\mathcal{X}_{q,g}^{u,m,0}$ are excluded from the sum. Otherwise, if $|\mathcal{X}_{q,g}^{u,m,0}| < |\mathcal{X}_{q,g}^{u,m,1}|$ is true, then the $|\mathcal{X}_{q,g}^{u,m,1}| - |\mathcal{X}_{q,g}^{u,m,0}|$ symbols that correspond to the smallest CF values in $\mathcal{X}_{q,g}^{u,m,1}$ are deleted. The $\min\{|\mathcal{X}_{q,g}^{u,m,0}|, |\mathcal{X}_{q,g}^{u,m,1}|\}$ number of remaining symbols in $\mathcal{X}_{q,g}^{u,m,0}$ and $\mathcal{X}_{q,g}^{u,m,1}$ form the sets $\bar{\mathcal{X}}_{q,g}^{u,m,0}$ and $\bar{\mathcal{X}}_{q,g}^{u,m,1}$, respectively. Let us now consider the scenario described in Fig. 6, where the above-mentioned two DHA calls created the sorted sets $\mathcal{X}_{q,g}^{u,m,0}$ and $\mathcal{X}_{q,g}^{u,m,1}$ for the calculation of the first user's $u = 1$ first bit's $m = 1$ LLR. The CF values of the symbols in the scenario of Fig. 6 are the same as in Fig. 4. Furthermore, the initial input of the DHA is the MMSE detector's output x_{MMSE}, when the first user's first bit's LLR has to be calculated and the search space does include the MMSE detector's output. By contrast, when the search space does not include x_{MMSE}, as in the second search of Fig. 6, where we want to construct $\mathcal{X}_{q,g}^{1,1,1}$, two options may be used, depending on whether the two DHA calls are running

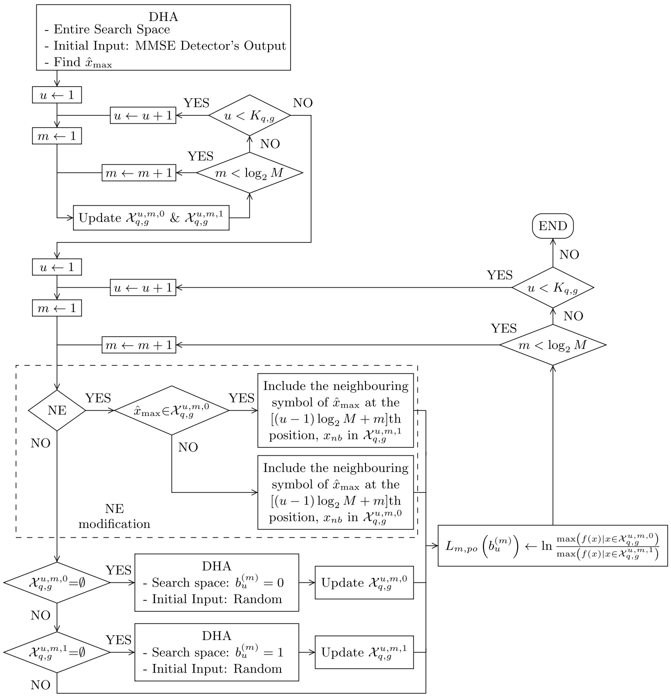

DHA
- Entire Search Space
- Initial Input: MMSE Detector's Output
- Find $\hat{x}_{\max}$
$u \leftarrow 1$
$u \leftarrow u+1$
YES
$u < K_{q,g}$
NO
$m \leftarrow 1$
NO
YES
$m \leftarrow m+1$
$m < \log_2 M$
Update $\mathcal{X}_{q,g}^{u,m,0}$ & $\mathcal{X}_{q,g}^{u,m,1}$
END
NO
YES
$u \leftarrow 1$
$u \leftarrow u+1$
$u < K_{q,g}$
NO
$m \leftarrow 1$
YES
$m \leftarrow m+1$
$m < \log_2 M$
NE
YES
$\hat{x}_{\max} \in \mathcal{X}_{q,g}^{u,m,0}$
YES
Include the neighbouring symbol of $\hat{x}_{\max}$ at the $[(u-1)\log_2 M + m]$th position, x_{nb} in $\mathcal{X}_{q,g}^{u,m,1}$
NO
NO
Include the neighbouring symbol of $\hat{x}_{\max}$ at the $[(u-1)\log_2 M + m]$th position, x_{nb} in $\mathcal{X}_{q,g}^{u,m,0}$
NE
modification
$L_{m,po}\left(b_u^{(m)}\right) \leftarrow \ln \frac{\max\left(f(x)|x \in \mathcal{X}_{q,g}^{u,m,0}\right)}{\max\left(f(x)|x \in \mathcal{X}_{q,g}^{u,m,1}\right)}$
$\mathcal{X}_{q,g}^{u,m,0} = \emptyset$
YES
DHA
- Search space: $b_u^{(m)} = 0$
- Initial Input: Random
Update $\mathcal{X}_{q,g}^{u,m,0}$
NO
$\mathcal{X}_{q,g}^{u,m,1} = \emptyset$
YES
DHA
- Search space: $b_u^{(m)} = 1$
- Initial Input: Random
Update $\mathcal{X}_{q,g}^{u,m,1}$
NO

FIGURE 5. **Flow Chart of the SO-DHA-MAA MUD with and without the NE modification.**

in parallel or consecutively:

1) If the two DHA calls are performed in parallel, the initial point may be the immediate neighbour of x_{MMSE} at the first user's first bit's location. In our scenario of Fig. 6 this may lead to the symbol $x_{init}^{1,1,0} = x_{MMSE} = 31 = [01|11|11] \rightarrow [11|11|11] = 63 = x_{init}^{1,1,1}$.
2) If the two DHA calls are performed serially, we always activate the DHA first on the search space corresponding to $\mathcal{X}_{q,g}^{1,1,\nu_0}$, where the MMSE detector's output is included in the search space as $\nu_0 = b_{1,MMSE}^{(1)}$. The reason behind this choice is that there is a higher probability of finding the global maximum of $x_{\max} = \hat{x}_{\max}^{1,1,0}$ in the same search space that the MMSE detector's output belongs to [7]. After we have found the specific index that maximizes the CF in that search space, which is $\hat{x}_{\max}^{1,1,0} = 22$ in our scenario, we exploit its neighbour at the first user's first bit's location according to $\hat{x}_{\max}^{1,1,0} = 22 = [01|01|10] \rightarrow [11|01|10] = 54 = \hat{x}_{init}^{1,1,1}$ as the initial point of the second DHA activation employed in the search space, which corresponds to $\mathcal{X}_{q,g}^{1,1,\nu_0 \oplus 1}$. The reason of choosing the neighbour of $\hat{x}_{\max}^{1,1,\nu}$ as the initial point of the second DHA is that the neighbour of the "hitherto-optimal" symbol may also be close to the maximum value in that particular search space, especially if $\hat{x}_{\max}^{1,1,\nu}$ turns out to be equal to the globally optimal symbol $x_{\max}$, since the values of the CF in (16) are correlated with the bit locations [3], [7].

In our simulations we assume that the DHA operations are performed serially, therefore we follow the second method of selecting the initial points of the DHA calls.

Once the pair of sets $\bar{\mathcal{X}}_{q,g}^{u,m,0}$ and $\bar{\mathcal{X}}_{q,g}^{u,m,1}$ has been formulated depending on $|\mathcal{X}_{q,g}^{u,m,0}|$ and $|\mathcal{X}_{q,g}^{u,m,1}|$, it has to be ensured that the resultant sign of the LLR is the same as the value $\varsigma_{u,m}$ calculated in (26). It is certain according to (26), that the sign of the difference of the maximum CF values of the two sets, namely $\text{sign}[f(x_1^{u,m,0}) - f(x_1^{u,m,1})]$ is equal to $\varsigma_{u,m}$ [10], where we have $x_1^{u,m,\nu} = x_{\max}^{u,m,\nu}$. For the sake of forcing the LLR to have the same sign as $\varsigma_{u,m}$ with 100% probability, we may only include the symbol pairs of the two sets of which the sign of the CF-difference is equal to $\varsigma_{u,m}$ of (26), as encapsulated in

$$\begin{aligned} x_i^{u,m,0} &\in \hat{\mathcal{X}}_{q,g}^{u,m,0} \wedge x_i^{u,m,1} \in \hat{\mathcal{X}}_{q,g}^{u,m,1} \\ \Leftrightarrow \text{sign}&\left[f\left(x_i^{u,m,0}\right) - f\left(x_i^{u,m,1}\right)\right] = \varsigma_{u,m} \\ &\wedge x_i^{u,m,0} \in \bar{\mathcal{X}}_{q,g}^{u,m,0} \wedge x_i^{u,m,1} \in \bar{\mathcal{X}}_{q,g}^{u,m,1} \\ &i = 2, \ldots, \min\{|\mathcal{X}_{q,g}^{1,1,0}|, |\mathcal{X}_{q,g}^{1,1,1}|\}, \end{aligned} \tag{27}$$

where the sets $\hat{\mathcal{X}}_{q,g}^{u,m,0}$ and $\hat{\mathcal{X}}_{q,g}^{u,m,1}$ consist of the filtered symbols. In our scenario of Fig. 6, three of the symbol pairs in $\bar{\mathcal{X}}_{q,g}^{u,m,0}$ and $\bar{\mathcal{X}}_{q,g}^{u,m,1}$ satisfy the constraints introduced in (27), therefore they are the specific symbols that will be included in the calculation of the LLR.

After the pair of sets $\hat{\mathcal{X}}_{q,g}^{u,m,0}$ and $\hat{\mathcal{X}}_{q,g}^{u,m,1}$ have been determined for the uth user's mth bit, its LLR in the SO-DHA-MUA QMUD may be calculated according to

$$L_{m,po}\left(b_u^{(m)}\right) = \ln \frac{\sum\limits_{x \in \hat{\mathcal{X}}_{q,g}^{u,m,0}} f(x)}{\sum\limits_{x \in \hat{\mathcal{X}}_{q,g}^{u,m,1}} f(x)}. \tag{28}$$

In Fig. 6 the first user's first bit's LLR would be equal to

$$L_{m,po}\left(b_1^{(1)}\right) = \ln \frac{f(22) + f(8) + f(9)}{f(46) + f(43) + f(62)} = 0.21. \tag{29}$$

Recall from [10] that when we proceed to the calculation of the second bit of the $M^{K_{q,g}}$-ary symbol, the ACO algorithm is called again for performing the same operation as for the first bit, since the true globally optimal symbol $x_{\max}$ may not have been found during the first two ACO operations, especially in rank-deficient systems. By contrast, this is unnecessary in the SO-DHA-MUA QMUD, since we have $\hat{x}_{\max} = x_{\max}$ with ~100% probability. Therefore, there is no need for calling the DHA for finding the best symbol in the subsequent search spaces where $\hat{x}_{\max}$ belongs to, resulting in a substantially

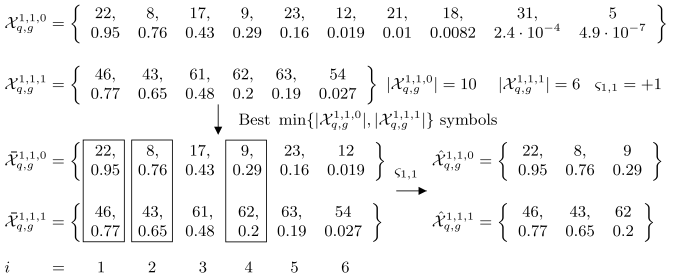

$\mathcal{X}_{q,g}^{1,1,0} = \left\{ \begin{matrix} 22, & 8, & 17, & 9, & 23, & 12, & 21, & 18, & 31, & 5 \\ 0.95 & 0.76 & 0.43 & 0.29 & 0.16 & 0.019 & 0.01 & 0.0082 & 2.4 \cdot 10^{-4} & 4.9 \cdot 10^{-7} \end{matrix} \right\}$
$\mathcal{X}_{q,g}^{1,1,1} = \left\{ \begin{matrix} 46, & 43, & 61, & 62, & 63, & 54 \\ 0.77 & 0.65 & 0.48 & 0.2 & 0.19 & 0.027 \end{matrix} \right\}$
$|\mathcal{X}_{q,g}^{1,1,0}| = 10$
$|\mathcal{X}_{q,g}^{1,1,1}| = 6$
$\varsigma_{1,1} = +1$
Best $\min\{|\mathcal{X}_{q,g}^{1,1,0}|, |\mathcal{X}_{q,g}^{1,1,1}|\}$ symbols
$\bar{\mathcal{X}}_{q,g}^{1,1,0} = \left\{ \begin{matrix} 22, & 8, & 17, & 9, & 23, & 12 \\ 0.95 & 0.76 & 0.43 & 0.29 & 0.16 & 0.019 \end{matrix} \right\}$
$\bar{\mathcal{X}}_{q,g}^{1,1,1} = \left\{ \begin{matrix} 46, & 43, & 61, & 62, & 63, & 54 \\ 0.77 & 0.65 & 0.48 & 0.2 & 0.19 & 0.027 \end{matrix} \right\}$
$i = 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6$
$\varsigma_{1,1}$
$\hat{\mathcal{X}}_{q,g}^{1,1,0} = \left\{ \begin{matrix} 22, & 8, & 9 \\ 0.95 & 0.76 & 0.29 \end{matrix} \right\}$
$\hat{\mathcal{X}}_{q,g}^{1,1,1} = \left\{ \begin{matrix} 46, & 43, & 62 \\ 0.77 & 0.65 & 0.2 \end{matrix} \right\}$

FIGURE 6. Scenario of the SO-DHA QMUD with MUA. The set $\mathcal{X}_{q,g}^{1,1,0}$ was created based on the first DHA call with the MMSE detector's output 31 as the initial input, where only symbols with $b_1^{(1)} = 0$ were searched. The set $\mathcal{X}_{q,g}^{1,1,1}$ was constructed based on the second DHA call with the neighbour of $\hat{x}_{\max}^{1,1,0}$ at the first user's first bit's position 22 → 54 as initial input, where only symbols with $b_1^{(1)} = 1$ were searched. Top to bottom left: Formulation of the sets $\bar{\mathcal{X}}_{q,g}^{1,1,0}$ and $\bar{\mathcal{X}}_{q,g}^{1,1,1}$ based on the set sizes $|\mathcal{X}_{q,g}^{1,1,0}| = 10$ and $|\mathcal{X}_{q,g}^{1,1,1}| = 6$ of the sets $\mathcal{X}_{q,g}^{1,1,0}$ and $\mathcal{X}_{q,g}^{1,1,1}$, respectively. The last $|\mathcal{X}_{q,g}^{1,1,0}| - |\mathcal{X}_{q,g}^{1,1,1}| = 4$ symbols are deleted from $\bar{\mathcal{X}}_{q,g}^{1,1,0}$, since $|\mathcal{X}_{q,g}^{1,1,0}| > |\mathcal{X}_{q,g}^{1,1,1}|$. Bottom left to bottom right: Construction of the sets $\hat{\mathcal{X}}_{q,g}^{1,1,0}$ and $\hat{\mathcal{X}}_{q,g}^{1,1,1}$ based on the signs of the difference of the CF values of the sorted symbol pairs in $\bar{\mathcal{X}}_{q,g}^{1,1,0}$ and$\bar{\mathcal{X}}_{q,g}^{1,1,1}$. The symbols in the positions $i = 3, 5, 6$ are deleted since $\text{sign}(f(x_i^{1,1,0}) - f(x_i^{1,1,1})) \neq \varsigma_{1,1} = +1$ for $i = 3, 5, 6$.

$u =$	1	1	2	2	3	3
$m =$	1	2	1	2	1	2
$b_u^{(m)} = 0$	**22**					
$b_u^{(m)} = 1$	**46**	31	31	31	31	31

DHA: $\mathcal{X}^{1,1,0}_{g,q} = \{\ 22,\ 8,\ 17,\ 9,\ 23,\ 12,\ 21,\ 18,\ \enclose{circle}{31},\ 5\ \}$

DHA: $\mathcal{X}^{1,1,1}_{g,q} = \{\ 46,\ 43,\ 61,\ 62,\ 63,\ \enclose{circle}{54}\}$

$\hat{\mathcal{X}}^{1,1,0}_{g,q} = \{\ 22,\ 8,\ 9\ \}$ $\quad\hat{\mathcal{X}}^{1,1,1}_{g,q} = \{\ 46,\ 43,\ 62\ \}$

$b_u^{(m)} = 0$	22	**46**				
$b_u^{(m)} = 1$	46	**22**	31	31	31	31

DHA: $\mathcal{X}^{1,2,0}_{g,q} = \{\ 46,\ 8,\ 43,\ \enclose{circle}{6},\ 15,\ 13\ \}$

$\mathcal{X}^{1,2,1}_{g,q} = \{\ 22,\ 31\ \}$

$\hat{\mathcal{X}}^{1,2,0}_{g,q} = \{\ 46\ \}$ $\quad\hat{\mathcal{X}}^{1,2,1}_{g,q} = \{\ 22\ \}$

$\vdots$

$b_u^{(m)} = 0$	22	46	22	8	8	**22**
$b_u^{(m)} = 1$	46	22	46	22	22	**43**

$\mathcal{X}^{3,2,0}_{g,q} = \{\ 22,\ 31\ \}$

DHA: $\mathcal{X}^{3,2,1}_{g,q} = \{\ 43,\ 61,\ 55,\ 9,\ \enclose{circle}{23},\ 35,\ 51,\ 33\ \}$

$\hat{\mathcal{X}}^{3,2,0}_{g,q} = \{\ 22\ \}$ $\quad\hat{\mathcal{X}}^{3,2,1}_{g,q} = \{\ 43\ \}$

FIGURE 7. Construction of the sets $\hat{\mathcal{X}}^{u,m,\nu}_{q,g}$ for $u = 1, 2, 3$, $m = 1, 2$, $\nu = 0, 1$ in the SO-DHA QMUD with MUA scenario described in Fig. 4. The symbol indices in the boxes represent the best so far found symbols for that particular bit value, while the bold numbers indicate the bit of which the sets have just been constructed. The circled indices denote the initial point of a DHA procedure. Since we assume that the DHA calls are performed serially, the initial point will be the MMSE output for one of the first bit's values and the neighbours of the best found symbol for the other value of the first bit, as well as for the rest of the bits.

reduced complexity. The DHA is called only for the particular bit value that is not the same as that bit's value in $\hat{x}_{\max}$. In our scenario of Fig. 4 and Fig. 6, the binary representation of the optimal symbol is $\hat{x}_{\max} = 22 = [01|01|10]$, therefore the set $\hat{\mathcal{X}}^{1,2,1}_{q,g}$ will only consist of $\hat{\mathcal{X}}^{1,2,1}_{q,g} = \{22, 31\}$. The MMSE detector's output of $x_{MMSE} = 31$ is also included in that set, since the second bit of its binary representation is equal to 1. For filling the set $\hat{\mathcal{X}}^{1,2,0}_{q,g}$, we employ the DHA for searching the symbols associated with $b_1^{(2)} = 0$, as in Fig. 6. The same procedure applies for the rest of the bits in the $M^{K_{q,g}}$-ary symbol, as described in the example of Fig. 7, where the sets of the bits are presented along with their method of construction. The performance of the SO-DHA-MUA QMUD will be shown to be substantially better than that of the SO-DHA-MAA QMUD, since the maximum CF value of each LLR's numerator and denominator is found with a ~100% probability. The SO-DHA-MUA QMUD is described in Algorithm 4.

Algorithm 4 SO-DHA QMUD With MUA and NE

```
$K_{q,g}$ users, $M$-ary modulation, $N \leftarrow M^{K_{q,g}}$.
$\mathcal{X}^{u,m,\nu}_{q,g} \leftarrow \emptyset$, $\bar{\mathcal{X}}^{u,m,\nu}_{q,g} \leftarrow \emptyset$, $\hat{\mathcal{X}}^{u,m,\nu}_{q,g} \leftarrow \emptyset$, for $u = 1, \ldots, K_{g,q}$, $m = 1, \ldots, \log_2 M$ and $\nu = 0, 1$.
if $b^{(1)}_{1,\ MMSE} = 0$ then
    The DHA is employed for finding $\hat{x}^{1,1,0}_{\max}$ and for filling the set $\mathcal{X}^{1,1,0}_{q,g}$, by exploiting the search space where $b^{(1)}_1 = 0$. Initial input: MMSE detector's output $x_{MMSE}$.
    The DHA is employed for finding $\hat{x}^{1,1,1}_{\max}$ and for filling the set $\mathcal{X}^{1,1,1}_{q,g}$, by exploiting the search space where $b^{(1)}_1 = 1$. Initial input: neighbour of $\hat{x}^{1,1,0}_{\max}$ at the first bit.
else if $b^{(1)}_{1,\ MMSE} = 1$ then
    The DHA is employed for finding $\hat{x}^{1,1,1}_{\max}$ and for filling the set $\mathcal{X}^{1,1,1}_{q,g}$, by searching the search space where $b^{(1)}_1 = 1$. Initial input: MMSE detector's output $x_{MMSE}$.
    The DHA is employed for finding $\hat{x}^{1,1,0}_{\max}$ and for filling the set $\mathcal{X}^{1,1,0}_{q,g}$, by searching the search space where $b^{(1)}_1 = 0$. Initial input: neighbour of $\hat{x}^{1,1,1}_{\max}$ at the first bit.
end if
Determine $\hat{x}_{\max}$ according to (25).
for $u = 1, 2, \ldots, K_{q,g}$ do
    for $m = 1, 2, \ldots, \log_2 M$ do
        if $u \neq 1$ or $m \neq 1$ then
            if $\hat{x}_{\max} \in \mathcal{X}^{u,m,0}_{q,g}$ then
                The DHA is employed for finding $\hat{x}^{u,m,1}_{\max}$ and for filling the set $\mathcal{X}^{u,m,1}_{q,g}$, by exploiting the search space where $b^{(m)}_u = 1$. Initial input: neighbour of $\hat{x}_{\max}$ at the $[(u-1)\cdot\log_2 M + m]$th bit.
            else if $\hat{x}_{\max} \in \mathcal{X}^{u,m,1}_{q,g}$ then
                The DHA is employed for finding $\hat{x}^{u,m,0}_{\max}$ and for filling the set $\mathcal{X}^{u,m,0}_{q,g}$, by exploiting the search space where $b^{(m)}_u = 0$. Initial input: neighbour of $\hat{x}_{\max}$ at the $[(u-1)\cdot\log_2 M + m]$th bit.
            end if
        end if
        Determine the sign of the LLR $\varsigma_{u,m}$ based on (26).
        Create $\bar{\mathcal{X}}^{u,m,0}_{q,g}$ and $\bar{\mathcal{X}}^{u,m,1}_{q,g}$ depending on the sets' sizes $|\mathcal{X}^{u,m,0}_{q,g}|$ and $|\mathcal{X}^{u,m,1}_{q,g}|$, by keeping the best $\min\{|\mathcal{X}^{u,m,0}_{q,g}|, |\mathcal{X}^{u,m,1}_{q,g}|\}$ values from $\mathcal{X}^{u,m,0}_{q,g}$ and $\mathcal{X}^{u,m,1}_{q,g}$, respectively.
        Create $\hat{\mathcal{X}}^{u,m,0}_{q,g}$ and $\hat{\mathcal{X}}^{u,m,1}_{q,g}$ according to (27).
        Calculate the $[(u-1)\cdot\log_2 M + m]$th bit's LLR based on (28).
    end for
end for
```

1) COMPLEXITY

The complexity of each DHA call in the SO-DHA-MUA QMUD is $O(\sqrt{N_{q,g}/2})$, since the searched space is half of the total search space. Therefore, the minimum complexity per bit for the SO-DHA-MUA-NE QMUD is

$$\begin{aligned}
&C^{MUA,NE,\min}_{SO-DHA}\\
&= \left[\left(K_{q,g}\cdot\log_2 M + 1\right)\left(4.5\sqrt{M^{K_{q,g}}/2} + L^{CD,\min}_{DHA}\right)\right.\\
&\qquad\left. + K_{q,g}\cdot\log_2 M\right] / \left(K_{q,g}\cdot\log_2 M\right)\\
&= \left(4.5\sqrt{M^{K_{q,g}}/2} + L^{CD,\min}_{DHA}\right)\left(1 + \frac{1}{K_{q,g}\cdot\log_2 M}\right) + 1
\end{aligned} \tag{30}$$

CFEs, where $L_{DHA}^{CD,\min}$ is computed according to (19). The first additive term of the lower complexity bound of the SO-DHA-MUA-NE QMUD in (30) corresponds to the scenario, where the initial inputs of all $(K_{q,g} \cdot \log_2 M + 1)$ number of DHA calls are equal to the respective optimal symbols of the search space. The second additive term corresponds to the CFE of the neighbour of $\hat{x}_{\max}$ for each bit of the multi-level symbol. The complexity of the SO-DHA-MUA-NE QMUD in (30) is higher than that of the SO-DHA-MAA-NE QMUD in (24).

Let us now proceed by proposing two modifications to the SO-DHA-MUA QMUD, which further improve the attainable performance, while reducing the complexity even further, but may impose delay and additional memory requirements on our system.

2) FORWARD KNOWLEDGE TRANSFER

In the Forward Knowledge Transfer (FKT) based modification, the symbols evaluated by the DHA performed for the ith bit of the $N_{q,g}$-ary symbol, with $i = 1, 2, \ldots, N_{q,g}$, are included not only in the set $\mathcal{X}_{q,g}^{u,m,v}$, for which set we have $i = (u-1) \cdot \log_2 M + m$, but also in the appropriate sets of the subsequent bits j, for $j = i+1, i+2, \ldots, N_{q,g}$. By invoking the proposed FKT the performance of the SO-DHA QMUD is expected to improve, since the sets that only consisted of $\hat{x}_{\max}$ and x_{MMSE} as in Fig. 7 will now include more symbols, which will have been evaluated during previous calls of the DHA, leading to LLR values closer to those of the ML MUD. No additional delay is introduced, because once the DHA has finished the search related to the ith bit, that bit's LLR is computed and passed to the deinterleaver. However, additional memory is required for storing the found symbols in the multiple appropriate sets.

When the FKT is applied, the initial input of the DHA of the $i = [(u-1) \cdot \log_2 M + m]$th bit, with $i \geq 2$, is determined by a comparison between the "best-so-far" symbol included in $\mathcal{X}_{q,g}^{u,m,v}$ by the previous DHA operations, and the specific neighbour of the globally optimal symbol $\hat{x}_{\max}$ at the ith bit position $x_{nb,\max}^{u,m}$, as formulated in

$$x_{init}^{u,m,v} = \begin{cases} \hat{x}_{nb,\max}^{u,m}, & \text{if } f\left(\hat{x}_{nb,\max}^{u,m}\right) > f\left(\hat{x}_{\max}^{u,m,v}\right) \\ \hat{x}_{\max}^{u,m,v}, & \text{if } f\left(\hat{x}_{nb,\max}^{u,m}\right) \leq f\left(\hat{x}_{\max}^{u,m,v}\right) \end{cases}. \quad (31)$$

By selecting $x_{init}^{u,m,v}$ according to (31) we introduce an extra CFE for calculating $f\left(\hat{x}_{nb,\max}^{u,m}\right)$, but the total complexity is expected to be reduced, since the initial point in the DHA will be closer to the symbol having the maximum CF value in that particular search space, hence saving multiple CFEs during the search. An example of the FKT is illustrated in Fig. 8a and its operation is formally described in Algorithm 5.

3) FORWARD AND BACKWARD KNOWLEDGE TRANSFER

In the Forward and Backward Knowledge Transfer (FBKT) modification both sets of every bit of the $N_{q,g}$-ary symbol are updated after every DHA application, regardless of which user's bit's search space the DHA was employed for. The FBKT process is formally stated in Algorithm 5. When using the FBKT, all the LLR values will be calculated based on the same symbol pool containing all the symbols evaluated during the $(K_{q,g} \cdot \log_2 M + 1)$ different DHA calls. For this reason an additional memory requirement and delay are imposed on the system, since the bits' LLRs are calculated after all the DHA activations have been completed, which is in contrast to the FKT, where the LLR of a bit is calculated after the DHA has been applied to its own search spaces.

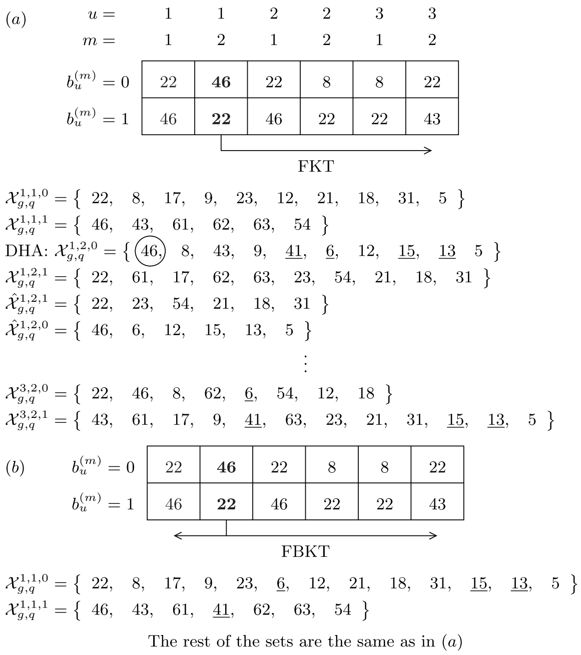

(a)
$u =$ 1 1 2 2 3 3
$m =$ 1 2 1 2 1 2
$b_u^{(m)} = 0$ 22 **46** 22 8 8 22
$b_u^{(m)} = 1$ 46 **22** 46 22 22 43
FKT
$\mathcal{X}_{g,q}^{1,1,0} = \{\ 22,\ 8,\ 17,\ 9,\ 23,\ 12,\ 21,\ 18,\ 31,\ 5\ \}$
$\mathcal{X}_{g,q}^{1,1,1} = \{\ 46,\ 43,\ 61,\ 62,\ 63,\ 54\ \}$
DHA: $\mathcal{X}_{g,q}^{1,2,0} = \{\ \bigcirc\!\!\!\!\!\!\!\;46,\ 8,\ 43,\ 9,\ \underline{41},\ \underline{6},\ 12,\ \underline{15},\ \underline{13}\ 5\ \}$
$\mathcal{X}_{g,q}^{1,2,1} = \{\ 22,\ 61,\ 17,\ 62,\ 63,\ 23,\ 54,\ 21,\ 18,\ 31\ \}$
$\hat{\mathcal{X}}_{g,q}^{1,2,1} = \{\ 22,\ 23,\ 54,\ 21,\ 18,\ 31\ \}$
$\hat{\mathcal{X}}_{g,q}^{1,2,0} = \{\ 46,\ 6,\ 12,\ 15,\ 13,\ 5\ \}$
$\vdots$
$\mathcal{X}_{g,q}^{3,2,0} = \{\ 22,\ 46,\ 8,\ 62,\ \underline{6},\ 54,\ 12,\ 18\ \}$
$\mathcal{X}_{g,q}^{3,2,1} = \{\ 43,\ 61,\ 17,\ 9,\ \underline{41},\ 63,\ 23,\ 21,\ 31,\ \underline{15},\ \underline{13},\ 5\ \}$
(b)
$b_u^{(m)} = 0$ 22 **46** 22 8 8 22
$b_u^{(m)} = 1$ 46 **22** 46 22 22 43
FBKT
$\mathcal{X}_{g,q}^{1,1,0} = \{\ 22,\ 8,\ 17,\ 9,\ 23,\ \underline{6},\ 12,\ 21,\ 18,\ 31,\ \underline{15},\ \underline{13},\ 5\ \}$
$\mathcal{X}_{g,q}^{1,1,1} = \{\ 46,\ 43,\ 61,\ \underline{41},\ 62,\ 63,\ 54\ \}$
The rest of the sets are the same as in (a)

FIGURE 8. Scenario of the SO-DHA QMUD with MUA and (*a*) FKT or (*b*) FBKT in the rank-deficient DSS/USSCH SDMA-OFDM system of Fig. 4. Focusing on the DHA application for the *i* = 2nd bit, the underlined indices represent the new symbols found during the current DHA search. The circled index 46 is selected to be the initial input of the DHA, because it has a higher CF value than that of the neighbour of the optimal symbol at the *i* = 2nd bit, corresponding to the decimal index 6, according to Fig. 4. When FKT is applied in (*a*) the sets of the subsequent bits *i* = 3, 4, 5, 6 are also updated with the new symbols found. Similarly, when FBKT is applied in (*b*), the sets of the preceding and the subsequent bits are updated along with the sets of the *i* = 2nd bit for which the search was performed.

Furthermore, the FBKT is expected to have a substantial impact on sub-optimal heuristic search algorithms used in rank-deficient systems. For example, let us assume that the true global optimal symbol $x_{\max}$ is found during an ACO search in the search space of the last bit of the multi-level symbol $i = K_{q,g} \cdot \log_2 M$. Based on the FBKT, all the bit-based LLRs of that particular $N_{q,g}$-ary symbol will be computed based on $x_{\max}$, as if it was found during the first search. If no knowledge transfer was employed, or even if the above-mentioned FKT was used in the same scenario, only the last bit's LLR would exploit $x_{\max}$ and hence calculate an LLR close to that of the ML MUD. An example of the FBKT is portrayed in Fig. 8b. The flowchart of the SO-DHA-MUA MUD with and without the aid of the FKT or the FBKT is depicted in Fig. 9.

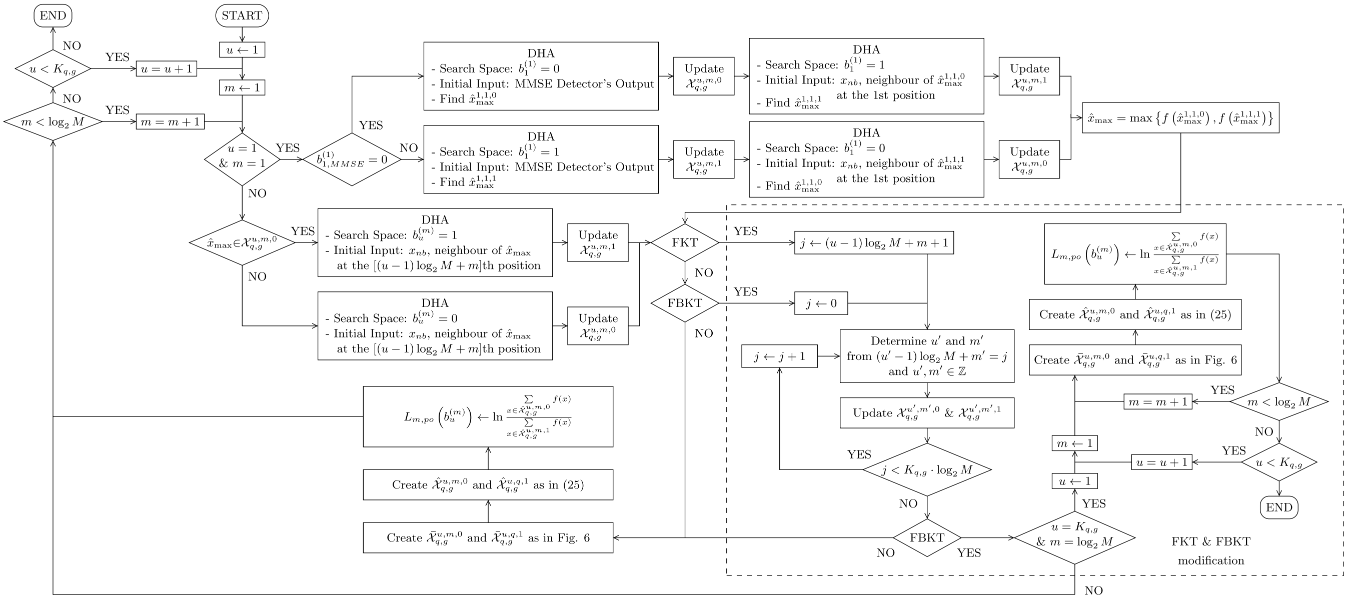

END
START
$u \leftarrow 1$
$u < K_{q,g}$
YES
NO
$u = u+1$
$m \leftarrow 1$
$m < \log_2 M$
$m = m+1$
$u = 1$ & $m = 1$
$b^{(1)}_{1,MMSE} = 0$
DHA
- Search Space: $b_1^{(1)} = 0$
- Initial Input: MMSE Detector's Output
- Find $\hat{x}^{1,1,0}_{\max}$
DHA
- Search Space: $b_1^{(1)} = 1$
- Initial Input: MMSE Detector's Output
- Find $\hat{x}^{1,1,1}_{\max}$
Update $\mathcal{X}^{u,m,0}_{q,g}$
Update $\mathcal{X}^{u,m,1}_{q,g}$
DHA
- Search Space: $b_1^{(1)} = 1$
- Initial Input: x_{nb}, neighbour of $\hat{x}^{1,1,0}_{\max}$ at the 1st position
- Find $\hat{x}^{1,1,1}_{\max}$
DHA
- Search Space: $b_1^{(1)} = 0$
- Initial Input: x_{nb}, neighbour of $\hat{x}^{1,1,1}_{\max}$ at the 1st position
- Find $\hat{x}^{1,1,0}_{\max}$
$\hat{x}_{\max} = \max\left\{f\left(\hat{x}^{1,1,0}_{\max}\right), f\left(\hat{x}^{1,1,1}_{\max}\right)\right\}$
$\hat{x}_{\max} \in \mathcal{X}^{u,m,0}_{q,g}$
DHA
- Search Space: $b_u^{(m)} = 1$
- Initial Input: x_{nb}, neighbour of $\hat{x}_{\max}$ at the $[(u-1)\log_2 M + m]$th position
DHA
- Search Space: $b_u^{(m)} = 0$
- Initial Input: x_{nb}, neighbour of $\hat{x}_{\max}$ at the $[(u-1)\log_2 M + m]$th position
FKT
FBKT
$L_{m,po}\left(b_u^{(m)}\right) \leftarrow \ln \frac{\sum_{x \in \hat{\mathcal{X}}^{u,m,0}_{q,g}} f(x)}{\sum_{x \in \hat{\mathcal{X}}^{u,m,1}_{q,g}} f(x)}$
Create $\hat{\mathcal{X}}^{u,m,0}_{q,g}$ and $\hat{\mathcal{X}}^{u,q,1}_{q,g}$ as in (25)
Create $\bar{\mathcal{X}}^{u,m,0}_{q,g}$ and $\bar{\mathcal{X}}^{u,q,1}_{q,g}$ as in Fig. 6
$j \leftarrow (u-1)\log_2 M + m + 1$
$j \leftarrow 0$
$j \leftarrow j+1$
Determine u' and m' from $(u'-1)\log_2 M + m' = j$ and $u', m' \in \mathbb{Z}$
Update $\mathcal{X}^{u',m',0}_{q,g}$ & $\mathcal{X}^{u',m',1}_{q,g}$
$j < K_{q,g} \cdot \log_2 M$
$u = K_{q,g}$ & $m = \log_2 M$
FKT & FBKT modification

FIGURE 9. Flow Chart of the SO-DHA-MUA MUD with the NE modification, as well as the FKT and FBKT methodologies.

Algorithm 5 Forward Knowledge Transfer & Forward and Backward Knowledge Transfer

```
Applicable immediately after the DHA application in steps 4, 5, 7, 8, 15 and 17 of Algorithm 4, or after any search algorithm used in the MUA. Let us assume the DHA was employed for the $u$th user's $m$th bit with $i=(u-1)\cdot\log_2 M+m$, $u=1,\ldots,K_{q,g}$ and $m=1,\ldots,\log_2 M$.
if Forward Knowledge Transfer then
    $d \leftarrow i$
else if Forward and Backward Knowledge Transfer then
    $d \leftarrow 0$
end if
for $j = d, d+1, \ldots, K_{q,g} \cdot \log_2 M$ do
    Determine $u'$ and $m'$ from $u' \cdot \log_2 M + m' = j$ and $u', m' \in \mathbb{Z}$.
    Update $\mathcal{X}^{u',m',\nu}_{q,g}$ for $\nu = 0, 1$ with the new symbols evaluated by the DHA for the $i$th bit of the $M^{K_{q,g}}$-ary symbol.
end for
```

V. SOFT-OUTPUT ANT COLONY OPTIMIZATION WITH FREE WILL

The ACO algorithm of [27] employs ζ ants during Ξ generations. In the ξth generation, with $\xi = 1, 2, \ldots, \Xi$, every ant chooses a $(K_{q,g} \cdot \log_2 M)$-bit long binary path, having a decimal representation corresponding to a specific symbol index. According to the ACO algorithm [27], if all ζ ants follow the same path during a single generation, the ACO algorithm is deemed to have converged and it is stopped at that particular generation. When the SO-ACO MUDs proposed in [10] and [11] are employed in heavily rank-deficient systems, they may converge with a high probability to a local optimum, rather than to the globally optimal symbol $x_{\max}$. Therefore, the performance of the SO-ACO MUD may be degraded in rank-deficient systems and its complexity may vary, depending on the number of generations that were evaluated before the ACO algorithm converged.

For the sake of circumventing these problems and for improving the performance, while fixing the complexity to a predetermined number of CFEs, we introduce a mutation probability [25], which may be interpreted as the *free will*[3] (FW) of the ants. According to the FW, after each ant has chosen its path for the current generation based on both the pheromone and on the intrinsic affinity [11], we introduce the concept of mutation probability, namely the probability that a bit of the selected path will change its value. For example, if an ant has chosen the [01|10|11] path in a system where $K_{q,g} = 3$ users transmit QPSK ($M = 4$) symbols, and the FW probability is chosen to be $p_{fw} = 0.1$, then there is a $0.1 \cdot 0.9^5 = 0.06$ probability of the final path to be [11|10|11]. Furthermore, there also is a 0.06 probability for the final path to be [01|10|01], while there is a $0.9^6 = 0.53$ probability of the selected path to remain unchanged.

By introducing the concept of FW, we deliberately expand the search space by introducing new symbols that might not have been visited otherwise, and the attainable performance is therefore expected to be improved. At the same time, the ACO algorithm has a modest, but non-negligible probability of convergence given by $(1-p_{fw})^{\zeta \cdot K_{q,g} \cdot \log_2 M}$, which corresponds to the unlikely scenario, where none of the ants changed the initially selected paths. Therefore, the complexity may be considered as being fixed to $(\zeta \cdot \Xi)$ number of CFEs.

The NE modification proposed for the SO-DHA QMUD may also be used for the SO-ACO MUDs. More specifically, during the SO-ACO-MAA-NE MUD the neighbours of the

[3] The free will probability of an ant in the ACO algorithm may be considered as the equivalent of the mutation probability of a chromosome in the genetic algorithm.

best found symbol $\hat{x}_{\max}$ of the first ACO search applied to the total search space are also considered as candidates for the calculations of the LLRs. Moreover, when the ACO is employed twice in the SO-ACO-MUA-NE MUD for the ith bit, the neighbour of the best found symbol in these two ACO calls is included for the ith bit's LLR calculation. On the other hand, if the SO-ACO-MUA-FKT-NE MUD is employed, the neighbour of the "best-so-far" symbol of all the ACO searches up to the ith bit's searches is included in the ith bit's LLR calculation. Finally, in the SO-ACO-MUA-FBKT-NE MUD the neighbours of the best found symbol after all the ACO searches have been completed for the current multi-level symbol are included in all the LLR calculations.

The corresponding complexity of the SO-ACO-MAA-NE-FW MUD becomes

$$C_{SO-ACO}^{MAA,NE,FW,\min} = \frac{\zeta \cdot \Xi}{K_{q,g} \cdot \log_2 M} + 1 \tag{32}$$

CFEs per bit, where the first additive term corresponds to the number of CFEs of the sole ACO based search in the total search space divided by the number of bits in the $M^{K_{q,g}}$-ary symbol, while the second term represents the extra CFE required for the calculation of the best found symbol's neighbour for each bit. If no FW is used in the ACO, then the complexity of the SO-ACO-MAA-NE is expected to be lower, since the ACO will converge sooner with a high probability. Nonetheless, it might converge to a local optimum. The total complexity of the SO-ACO-MUA-NE-FW MUD is equal to

$$C_{SO-ACO}^{MUA,NE,FW,\min} = 2 \cdot \zeta \cdot \Xi + 1 \tag{33}$$

CFEs per bit, since two ACO activations take place for each bit and a neighbour's CF value is calculated.

VI. SIMULATION RESULTS

In this section we evaluate the BER performance versus complexity performance of our proposed SO-DHA QMUDs in the uplink of DSS/USSCH SDMA-OFDM systems. The first communication system investigated supports $K = 8$ users transmitting QPSK ($M = 4$) symbols and the BS station is equipped with $P = 4$ AEs. We adopt a channel model used in the Long Term Evolution (LTE) standard, termed as the Extended Vehicular A (EVA) model [28]. We assume an urban vehicular scenario, where the velocity of each mobile user is $v = 30$ km/h, the carrier frequency is $f_c = 2.5$ GHz, the sampling frequency is equal to $f_s = 15.36$ MHz and hence the Doppler frequency is $f_d \simeq 70$ Hz. The users transmit on 1024 OFDM subcarriers and the length of the cyclic prefix (CP) is 128. For comparison between the SO-DHA QMUDs and the SO-ACO MUDs, let us firstly investigate a scenario where all users transmit on all the subcarriers. Moreover, we will also temporarily assume that all the users have been allocated the same DSS code, therefore they all interfere with each other, hence the resultant system becomes an SDMA-OFDM system. The interleaver length is equal to 10 240 symbols per user and TCC associated with a rate of $R = 1/2$, 8 trellis states and $I_{\text{inner}} = 4$ inner iterations are employed. The specifications of our SDMA-OFDM system are summarized in Table 1.

TABLE 1. Parameters of the SDMA-OFDM uplink scenario.

Number of Users	$K = 8$
Number of AEs per User	$N_{T_x} = 1$
Number of AEs at the BS	$P = 4$
Normalized User-Load	$U_L = K_{q,g}/P = 2$
Modulation	QPSK $M = 4$
Channel Code	TCC, $R = 1/2$, 8 trellis states
	$I_{\text{inner}} = 4$ iterations
Mobile Velocity	$v = 30$ km/h
Carrier Frequency	$f_c = 2.5$ GHz
Sampling Frequency	$f_s = 15.36$ GHz
Doppler Frequency	$f_d = 70$ Hz
Channel Model	Extended Vehicular A [28]
Number of Subcarriers	$Q = 1024$
Cyclic Prefix	CP $= 128$
Number of Subcarriers per User	$W = 1024$
Number of DSS codes	$G = 1$
Bit Interleaver Length	10 240 symbols per user

Observe in Fig. 10, that the SO-ACO-MAA MUD and SO-ACO-MUA MUD are exactly as proposed in [11] and [10], respectively, when employed in the context of the SDMA-OFDM system of Table 1. These may be compared to the BER curves of the SO-ML MUD, as well as to those of the proposed SO-ACO-MAA-FW, SO-ACO-MAA-NE, SO-ACO-MAA-NE-FW, SO-ACO-MUA-NE-FW, SO-ACO-MUA-FKT, SO-ACO-MUA-FKT-NE-FW, SO-ACO-MUA-FBKT and SO-ACO-MUA-FBKT-NE-FW MUDs, which are gathered in Table 2. After appropriate tuning of the ACO for minimizing the BER, we have opted for the pheromone weight of $a = 1$, the intrinsic affinity weight of $\beta = 6$, while the weight of the elite ant was set to $\sigma = 8$ and the evaporation rate to $\rho = 0.5$ [10] for the ACO-aided searches in all the ACO-based MUDs. The number of ants in the SO-ACO-MAA MUDs is equal to $\zeta = 44$, and the number of generations is also $\Xi = 44$. Similarly, in the SO-ACO-MUA MUDs invoked for our SDMA-OFDM system of Fig. 10 we employ $\zeta = 27$ ants during $\Xi = 27$ generations, while $\zeta = 25$ ants and $\Xi = 25$ generations are used in the SO-ACO-MUA-FKT and SO-ACO-MUA-FBKT MUDs. Explicitly, we have selected these values for achieving a similar average number of CFEs per bit between the ACO-FW-based MUDs and the DHA-based QMUDs. The FW probability, when used, is chosen to be equal to $p_{FW} = 0.1$. The parameters of the ACO instances are gathered in Table 3.

According to Fig. 10, the BER curves of the SO-ACO-MAA and SO-ACO-MAA-NE MUDs of Table 2 are approximately 7.25 dB and 8.25 dB away from the ML MUD's

TABLE 2. Algorithmic elements of the prototype detectors.

Search Algorithm:	• Dürr-Høyer Algorithm (DHA). • Ant Colony Optimization (ACO) Algorithm.
Approximation:	• MAximum Approximation (MAA): Each LLR is computed based on the pair of symbols that maximize the CF both in its numerator and denominator, respectively, as in (21). • MUlti-input Approximation (MUA): Each LLR is computed based on a carefully selected subset of symbols, as in (26), (27) and (28).
Knowledge Transfer:	After the completion of the selected search algorithm for the $i = [(u-1)\log_2 M + m]$th bit position • OFF: Only $\mathcal{X}_{q,g}^{u,m,0}$ and $\mathcal{X}_{q,g}^{u,m,1}$ are updated with the search results. • Forward Knowledge Transfer (FKT): The sets $\mathcal{X}_{q,g}^{u',m',0}$ and $\mathcal{X}_{q,g}^{u',m',1}$ are updated with the search results, for all $[u', m']$ values satisfying $(u'-1)\log_2 M + m' \geq i$ with $u' = 0, 1, \ldots, K_{q,g}$, $m' = 0, 1, \ldots, \log_2 M$. • Forward & Backward Knowledge Transfer (FBKT): All the sets $\mathcal{X}_{q,g}^{u,m,0}$ and $\mathcal{X}_{q,g}^{u,m,1}$ are updated with the search results, for $u = 0, 1, \ldots, K_{q,g}$, $m = 0, 1, \ldots, \log_2 M$.
Neighbour Exploitation (NE):	• OFF: If a set $\mathcal{X}_{q,g}^{u,m,\nu}$ is empty for a specific $[u, m, \nu]$ value, the search algorithm employed is initiated again for the search space corresponding to the $[u, m, \nu]$ value. • ON: The neighbour x_{nb} of the best-so-far found symbol $\hat{x}_{\max}$ at each bit position is included as a candidate for the calculation of the respective LLR. If the DHA is used, x_{nb} is the initial input when possible. No set $\mathcal{X}_{q,g}^{u,m,\nu}$, for $u = 1, 2, \ldots, K_{q,g}$, $m = 1, 2, \ldots, \log_2 M$, $\nu = 0, 1$ will be empty with NE.
Free Will (FW):	Applicable only when the ACO algorithm is chosen as the search algorithm. • OFF: Conventional ACO algorithm employed in [10], [11]. • ON: After an ant has selected a step along its path, which corresponds to a specific bit position, during each generation, there is a predetermined FW mutation probability to flip that bit's value for that specific ant's path.

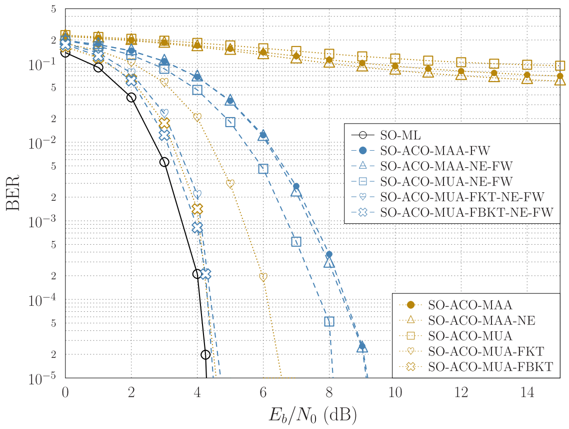


FIGURE 10. BER versus E_b/N_0 performance in the rank-deficient SDMA-OFDM system of Table 1, supporting a normalized user-load of $U_L = K/P = 2$ when the SO-ML MUD and various modifications of the SO-ACO-MAA and SO-ACO-MUA MUDs with neighbour exploitation, free will, forward knowledge transfer, as well as forward and backward knowledge transfer are employed.

TABLE 3. Parameters of the ACO in the SDMA-OFDM scenario.

Number of Ants	MAA:	$\zeta = 44$ ants
	MAA-NE:	$\zeta = 44$ ants
	MUA:	$\zeta = 27$ ants
	MUA-FKT:	$\zeta = 25$ ants
	MUA-FBKT:	$\zeta = 25$ ants
Number of Generations	MAA:	$\Xi = 44$ generations
	MAA-NE:	$\Xi = 44$ generations
	MUA:	$\Xi = 27$ generations
	MUA-FKT:	$\Xi = 25$ generations
	MUA-FBKT:	$\Xi = 25$ generations
Pheromone Weight	$a = 1$	
Intrinsic Affinity Weight	$\beta = 6$	
Weight of Elite Ant	$\sigma = 8$	
Evaporation Rate	$\rho = 0.5$	
Free Will Probability	$p_{FW} = 0.1$	

limit for a BER of 10^{-1}, respectively, indicating that the SO-ACO-MAA MUDs' performances are degraded in rank-deficient systems supporting a normalized user-load of $U_L = K/P = 2$. The SO-ACO-MAA-NE MUDs attain a modest performance improvement in rank-deficient systems supporting $U_L = 2$ over the respective SO-ACO-MAA MUDs, and they also have a lower complexity than the SO-ACO-MAA MUD when no FW is enabled, since only a single ACO search is performed, when the NE modification is applied. In our scenario of Fig. 10 the average number of CFEs per bit is approximately three times lower in the SO-ACO-MAA-NE MUD compared to the SO-ACO-MAA MUD.

When the FW-based mutation is enabled in the SO-ACO-MAA-FW and SO-ACO-MAA-NE-FW MUDs of Table 2, the BER performances become better than those of the SO-ACO-MAA or SO-ACO-MAA-NE MUDs. More precisely, the SO-ACO-MAA-FW and SO-ACO-MAA-NE-FW MUDs reach a BER of 10^{-5} when operating 4.9 dB and 4.85 dB away from the ML MUD, respectively. Still referring to Fig. 10, the performances of the SO-ACO-MUA and SO-ACO-MUA-NE-FW MUDs of Table 2 are also seen to be close to those of the SO-ACO-MAA and SO-ACO-MAA-FW MUDs, respectively. This is because in rank-deficient systems the ACO algorithm fails in finding the true globally maximal symbol $x_{\max}$ for the numerator and the denominator of each bit of a multi-level symbol, even if FW-based mutation is adopted. Note that when the FW-based mutation is not used, the performance of the SO-ACO-MUA MUD becomes worse than those of the SO-ACO-MAA and of the SO-ACO-MAA-NE MUDs. The reason for this observation is that according to the MUA approach of [10], when $x_{\max}$ is found during the two ACO activations for the ith bit, it will only be used for the LLR calculation of the ith bit. This is in contrast to the MAA approach of [11], where $x_{\max}$ is exploited for the calculation of all the LLRs provided that it has been found. The search space of the first ACO employment in the MAA approach has twice the size with respect to the search spaces of the ACO calls in the MUA approach. However, even in conjunction with the provision of a larger number of ants and generations, $x_{\max}$ may not be found by the ACO search in the MAA approach. When FW-based mutation is used, the ant population does not converge, therefore the incident of evaluating the true globally maximal symbol $x_{\max}$ becomes more frequent in the FW-aided MUA approach. The increased probability of finding $x_{\max}$, in conjunction with the use of a subset of legitimate symbols for calculating the LLR in the SO-ACO-MUA-NE-FW, results in a better performance than that of the SO-ACO-MAA-FW and SO-ACO-MAA-NE-FW MUDs of Table 2.

When the proposed FKT of Section IV-B.2 is adopted in the MUA approach, the performance is improved, as it may be verified by observing the SO-ACO-MUA-FKT MUD's and SO-ACO-MUA-FKT-NE-FW MUD's curves in Fig. 10. Upon invoking the FKT of Table 2 in the ACO, if the true globally maximal symbol $x_{\max}$ is found during one of the ith bit's ACO searches, then it will also be included in the LLR calculations of the subsequent bits, even if those bits' ACO calls failed to find $x_{\max}$ on their own. Hence, by using the FKT of Table 2, the accuracy of the LLR values is greatly improved provided that $x_{\max}$ is found, hence gradually approaching the LLR values of the SO-ML MUD. Similarly, the BER performances of the SO-ACO-MUA-FBKT and SO-ACO-MUA-FBKT-NE-FW MUDs of Table 2 become even better than that when the FKT is used, since all the LLRs exploit $x_{\max}$, provided it is indeed found during any of the ACO searches employed for that particular $N_{q,g}$-ary symbol. Upon invoking the FBKT of Table 2 there is a trade-off between the attainable performance and the delay imposed, since the calculation of the LLR of each bit in a multi-level symbol has to be stalled until after all the searches have been completed. In all the ACO-based MUDs of Table 2 the performance is improved, when the FW-based mutation is adopted. When the FKT or the FBKT is used, the memory requirements are increased.

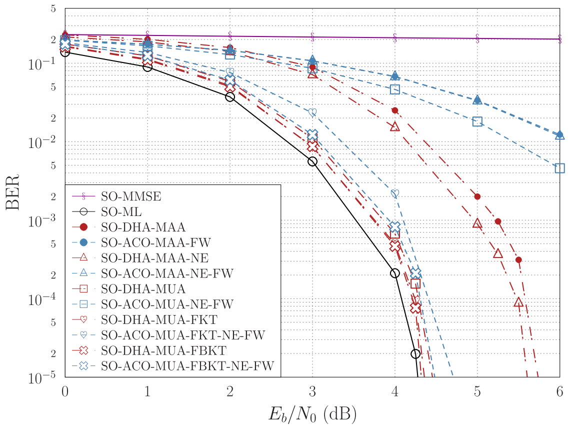


FIGURE 11. **BER performance with respect to E_b/N_0 in the rank-deficient SDMA-OFDM system of Table 1 when the SO-ML MUD and various SO-DHA QMUDs, as well as SO-ACO MUDs with free will are employed. The effects of the modifications of neighbour exploitation, forward knowledge transfer, as well as forward and backward knowledge transfer on the BER of the SO-ACO-FW MUDs and the SO-DHA QMUDs are illustrated.**

The SO-DHA QMUDs of Fig. 5 and Fig. 9, which are also summarized in Table 2, perform better than their classic SO-ACO MUDs counterparts, as shown in Fig. 11, where we compare the proposed SO-DHA QMUDs to our novel SO-ACO MUDs using both the NE and FW, which outperform the SO-ACO MUDs presented in [10] and [11], as evidenced in Fig. 10. Since the DHA succeeds in finding $x_{\max}$ with ∼100% probability, the SO-DHA-MAA, SO-DHA-MAA-NE and SO-DHA-MUA QMUDs of Fig. 5, Fig. 9 and Table 2 experience improved BER curves in contrast to the SO-ACO-MAA-FW, SO-ACO-MAA-NE-FW and SO-ACO-MUA-FW MUDs. The SO-DHA-MAA and SO-DHA-MAA-NE QMUDs perform within about 1.35 dB and 1.45 dB, respectively, of the SO-ML MUD's limit at a BER of 10^{-5}. The SO-DHA-MUA QMUD performs better than the SO-ACO-MUA-FBKT-NE-FW MUD, which is the best-performing MUD from the family of SO-ACO MUDs, demonstrating the attractive performance versus complexity potential of the QMUDs. As expected, the SO-DHA-MUA-FKT and SO-DHA-MUA-FBKT QMUDs have an improved performance than that of the SO-DHA-MUA QMUD, despite their lower complexity. Since information about the searched symbols is transferred between the relevant bits after each DHA application in the FKT and FBKT, there is a higher probability that the next DHA call will have as an initial input a symbol with a closer CF value to that specific search space's optimal symbol's CF value. This leads to fewer CFEs before that optimal symbol is found by the DHA. The BER performances of the SO-DHA-MUA QMUDs closely match the SO-ML limit of Fig. 11. The specific effect that the FKT and FBKT modifications have on the SO-DHA-MUA QMUD is minor compared to their effect on the SO-ACO-MUA MUDs, since all the SO-DHA-MUA QMUDs find $x_{\max}$ during the first two DHA calls.

The complexity expressed in terms of the total number of CFEs performed per bit both in the CD and the QD of the analysed SO-DHA QMUDs and SO-ACO MUDs is depicted in Fig. 12 versus the E_b/N_0 values. The total number of CFEs of all the SO-ACO MUDs that use FW is fixed and independent of the E_b/N_0 values. More specifically, the exact number of CFEs per bit for the SO-ACO-MAA-FW MUDs is 122 CFEs according to both (32) and Fig. 12, while the com-

TABLE 4. Comparison of the MUDs in the SDMA-OFDM system of Table 1.

MUD	Distance from SO-ML MUD at BER $= 10^{-5}$ (dB)	Complexity at BER $= 10^{-5}$ (CFEs / bit)	Increased Delay	Increased Memory Requirements
SO-ML	0	4096	✓	✓✓✓
SO-DHA-MAA	1.46	119.4		
SO-ACO-MAA-FW	4.90	121		
SO-ACO-MAA	(BER $= 0.1$) 8.25	(BER $= 0.1$) 218.77		
SO-DHA-MAA-NE	1.32	120.4		
SO-ACO-MAA-NE-FW	4.85	122		
SO-ACO-MAA-NE	(BER $= 0.1$) 7.25	(BER $= 0.1$) 75.06		
SO-DHA-MUA	0.18	1409		
SO-ACO-MUA-NE-FW	3.81	1459		
SO-ACO-MUA	(BER $= 0.1$) 13.25	(BER $= 0.1$) 1060		
SO-DHA-MUA-FKT	0.08	1176.8		✓
SO-ACO-MUA-FKT-NE-FW	0.44	1251		✓
SO-ACO-MUA-FKT	2.29	955.2		✓
SO-DHA-MUA-FBKT	0.045	1177.42	✓	✓✓
SO-ACO-MUA-FBKT-NE-FW	0.2	1251	✓	✓✓
SO-ACO-MUA-FBKT	0.67	967.73	✓	✓✓

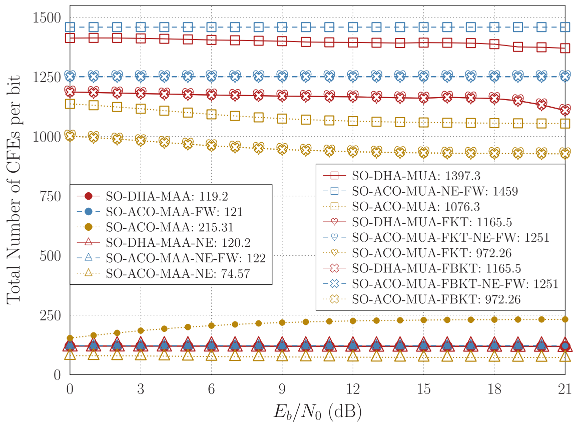


FIGURE 12. Complexity in terms of total number of CFEs per bit of each MUD and QMUD employed in the SDMA-OFDM system of Table 1 with respect to the E_b/N_0 values. The BER performances of the SO-ACO-based MUDs and SO-DHA-based QMUDs are depicted in Fig. 10 and Fig. 11, respectively. The average number of CFEs per bit of each MUD is stated at the legend. The complexity of the SO-ML MUD is 4096 CFEs per bit.

plexity of the SO-ACO-MUA-FW MUD is 1459 CFEs and that of both the SO-ACO-MUA-FKT-NE-FW and SO-ACO-MUA-FBKT-NE-FW MUDs is 1251 CFEs based on (33) and Fig. 12. The complexities of the SO-DHA QMUDs are reduced only marginally versus the transmit power. Therefore they may be practically considered as being fixed even in the context of rank-deficient systems. This phenomenon is observed due to the E_b/N_0-related improvement of the MMSE detector's output, which is the input of the first DHA call, as investigated in [22]. If the DHA-QWSA QMUD [20] was employed in our SDMA-OFDM system in conjunction with 11 qubits in the Quantum Control Register (QCR) of the QWSA circuit described in [20], its minimum complexity would be 32 792 CFEs per bit for $K_{q,g} = 6$ and $M = 4$ [20], which is significantly higher than both that of the SO-ML MUD and the SO-DHA QMUDs. The DHA-QWSA QMUD aims to operate in systems supporting many users and multiple MUD-decoder iterations, while its performance matches that of the ML MUD, provided that the number of qubits in the QCR is sufficiently high. Observe in Fig. 11 that the SO-DHA QMUDs with no MUD-decoder iterations perform close to the ML MUD's limit and they only impose a fraction of the complexity that DHA-QWSA QMUD does.

The specific choice of the number of ants ζ and generations Ξ in the SO-ACO MUDs was specifically arranged to make the complexities of the SO-DHA QMUDs and of the respective SO-ACO-FW MUDs comparable. The ML MUD performs 4096 CFEs per bit in our SDMA-OFDM system, which is higher than that of all the investigated MUDs. The complexities of a system that employs FKT and one that uses FBKT are the same, since they have the same effect on the subsequent searches. Furthermore, the difference in the number of CFEs per bit between the SO-DHA-MAA and the SO-DHA-MAA-NE QMUDs is only 1 CFE, corresponding to the CFE of the best found symbol's neighbour for each bit.

The complexity of the SO-ACO-MAA MUD is seen to be higher in Fig. 12 than those of the SO-ACO-MAA-NE, SO-ACO-MAA-FW and SO-ACO-MAA-NE-FW MUDs, as well

TABLE 5. Parameters of the DSS/USSCH SDMA-OFDM uplink scenario.

Number of Users	$K = 24$
Number of AEs at the BS	$P = 2$
Number of Subcarriers per User	$W = 512$
Number of DSS codes	$G = 2$
Normalized User-Load	
↪ DSS-based USSCH	$U_L = K_{q,g}/P = 3$
↪ USSCH	$U_L = K_{q,g}/P \in \{0, 0.5, \ldots, 6\}$
Subcarrier Allocation Period	$T_h = 2$ OFDM frames

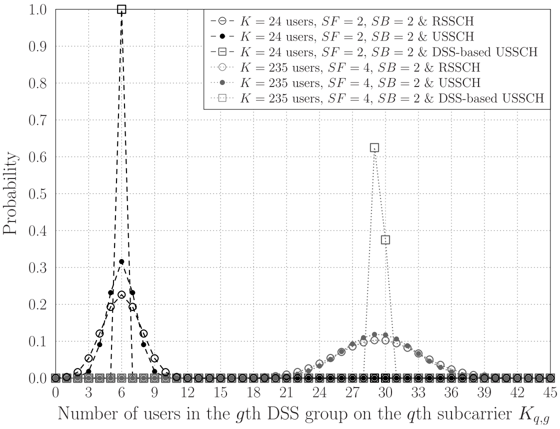


FIGURE 13. Histograms of the number of users who belong to the same *g*th DSS code group while transmitting on the *q*th subcarrier when the RSSCH, USSCH and the DSS-based USSCH are employed. The two subcarrier allocation schemes are compared in a system where *SF* = 2 WH DSS codes are allocated to *K* = 24 users who transmit on half of the available subcarriers *SB* = *Q*/*W* = 2, as well as in a system where *SF* = 4 WH DSS codes are allocated to *K* = 235 users who also transmit on half of the available subcarriers.

as than those of the SO-DHA-MAA QMUDs, since the ACO-aided search is re-employed, unless a symbol corresponding to each bit's value is found. This is not an issue in the SO-ACO-MAA-NE MUD, where the ACO is called only once. Based on this fact, it may be observed in Fig. 12 that the complexity of the SO-ACO-MAA-NE MUD is the lowest of the MUDs based on MAA, as are the complexities of the SO-ACO-MUA MUDs of Table 2 operating without FW and NE with respect to the rest of the MUDs based on MUA.

Table 4 summarizes both the performance and complexity, as well as the delay and memory requirements of the MUDs employed. The check mark (✓) in Table 4 indicates the existence of delay or increased memory requirement, when compared to the MAA and MUA methodologies. The number of check marks represents the size of the increase. Even though the SO-ACO-MUA-FBKT needs fewer CFEs per bit on average than the SO-DHA-MUA, SO-DHA-MUA-FKT and SO-DHA-MUA-FBKT QMUDs according to Fig. 12, its performance is slightly worse than that of the above-mentioned QMUDs according to Fig. 10 and Fig. 11. Furthermore, the comparison of the SO-DHA-MUA-FKT and SO-DHA-MUA-FBKT QMUDs indicates that there is no substantial benefit in tolerating an increased delay, since the performance of the higher-delay systems is almost identical. Finally, when we compare the SO-DHA-MUA QMUD to the SO-ACO-MUA-FBKT MUD, a trade-off between the complexity versus the extra delay - and memory - requirements has to be struck, with the SO-DHA-MUA QMUD offering a better performance.

Let us now proceed to the uplink scenario of the DSS/SSCH SDMA-OFDM system of Fig. 1, where $K = 24$ users transmitting QPSK ($M = 4$) symbols are supported, there are $P = 2$ receive AEs at the BS and $G = 2$ WH DSS codes are allocated to the $K = 24$ users. Furthermore, each user transmits on $W = 512$ out of the $Q = 1024$ subcarriers. The subcarrier allocation algorithm is performed every $T_h = 2$ OFDM frames. The new parameters are summarized in Table 5, while the rest of the parameters are the same as in Table 1. Since there are $K = 24$ users and $G = 2$ DSS codes, $K/G = 12$ users will be allocated the $g = 1$st code and the remaining 12 users will have the code associated with $g = 2$. Similarly, since each user transmits on W subcarriers, the number of subbands is also $W = 512$. The number of subcarriers per subband is $SB = Q/W = 2$. According to the DSS-based USSCH and the example presented in Fig. 3, $K_1 = 6$ users with the $g = 1$st code will be allocated to one of the $SB = 2$ subcarriers of a subband and the other $K_2 = 6$ users will be allocated to the remaining subcarrier in the same subband. Therefore, we expect to support $K_{q,g} = 6$ users for $q = 1, 2, \ldots, 1024$ and $g = 1, 2$. Indeed, by observing Fig. 13, it is verified that $K_{q,g} = 6$ users can be supported for all values of q and g in the DSS-based USSCH, where we compare the proposed DSS-based USSCH to the USSCH and the RSSCH [7] for the same DSS code allocated to the same subcarrier. On the other hand, when the USSCH is used in our system supporting $K = 24$ users, the allocation of the $K_1 = 12$ users which have been allocated the same DSS code on the two subcarriers of each subband varies from 6 users to each subcarrier up to the rare incident, where all 12 users are allocated to one of the two subcarriers causing an excessive amount of unnecessary MUI and simultaneous increase in complexity, albeit this only occurs with a probability of 10^{-6}. Quantitatively, if the SO-ML MUD is employed in the DSS-based USSCH system the average complexity would be

$$C_{SO-ML}^{DSS/USSCH} = \sum_{K_{q,g}=1}^{\infty} \frac{p_{K_{q,g}} \cdot M^{K_{q,g}}}{K_{q,g} \cdot \log_2 M} = \frac{1 \cdot 4^6}{6 \cdot 2} = 341.33 \tag{34}$$

CFEs per bit, where $p_{K_{q,g}}$ is the probability of having $K_{q,g}$ users associated with the gth DSS code on the qth subcarrier. On the other hand, the average complexity in the USSCH system would be 1135.3 CFEs per bit according to Fig. 13, while the complexity of the RSSCH was 3188.3 CFEs per bit for the system supporting 24 users.

Let us now also investigate a system, where $SF = 4$ WH DSS codes are allocated to $K = 235$ users. Let us assume furthermore that each user transmits on half of the available subcarriers, hence we have $SB = Q/W = 2$. Since there are $K = 235$ users and 4 DSS codes, 59 users will be allocated to the first 3 DSS codes and 58 will be allocated to the last one according to (3). When the DSS-based USSCH is used, exploiting the fact that $SB = 2$, the first DSS code group's users will be split as uniformly as possible between the two subcarriers. In other words, for each of the first three DSS code groups 29 users will be allocated to one subcarrier and the remaining 30 users will be allocated to the other subcarrier. For the last DSS code group, 29 users will be allocated to each subcarrier. As a result, 5 out of 8 times, or 62.5% of the time, we will have 29 users of the same DSS code group on the same subcarrier and 37.5% of the instances 30 users of the same DSS code group will transmit on the same subcarrier. The simulations of Fig. 13 verify these expectations, resulting in an average complexity for the DSS-based USSCH which is ∼736 times lower than that of the USSCH system, when the SO-ML MUD is used.

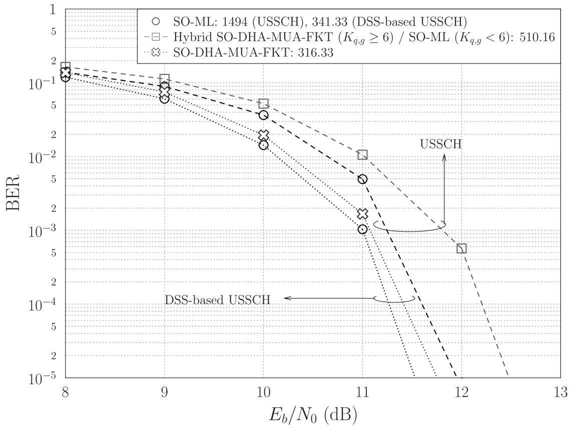


FIGURE 14. BER performance with respect to E_b/N_0 in the rank-deficient DSS/USSCH SDMA-OFDM system of Table 5, when the USSCH and DSS-based USSCH subcarrier allocation algorithms are used. The hybrid SO-DHA-MUA-FKT/SO-ML MUD employs the SO-DHA-MUA-FKT QMUD when the number of users on the *q*th subcarrier using the *g*th DSS code is higher than $K_{q,g} \geq 6$, otherwise the SO-ML MUD is employed. The numbers in the legend denote the average number of CFEs per bit for each MUD.

In Fig. 14 the BER curves of the DSS/USSCH SDMA-OFDM system employing the two SSCH procedures of Section II-C are combined with different MUDs. More specifically, the SO-ML MUD is used in both scenarios and the BER performance is improved by the system employing DSS-based USSCH by approximately 0.5 dB, as a benefit of its reduced MUI. Moreover, the average complexity of the SO-ML MUD is reduced from 1494 CFEs per bit in the USSCH system to 341.33 CFEs per bit in the DSS-based USSCH system, which is due to the fact that we have $K_{q,g} = 6$ for all values of g and q. Furthermore, in Fig. 14 we employ the SO-DHA-MUA-FKT QMUD of Fig. 9 and Table 2 in our DSS-based USSCH scenario achieving a near-optimal performance, which is also better than that of the SO-ML MUD, when USSCH is used. As an added benefit, it has a lower complexity than both the SO-ML MUDs investigated in the context of the DSS/USSCH SDMA-OFDM system.

In Fig. 14 we also present a hybrid SO-DHA-MUA-FKT/SO-ML MUD, when USSCH is employed. According to Fig. 13, the value of $K_{q,g}$ varies with respect to the specific subcarrier and DSS code group when USSCH is used. The average complexity of the SO-DHA-MUA-FKT QMUD used by DSS-based USSCH system associated with $K_{q,g} = 6$ and $M = 4$ is 316.33 CFEs per bit. According to (30), the minimum complexity of pure QD-CFEs employed in isolation if $K_{q,g} = 5$ and QPSK ($M = 4$) is used when the SO-DHA-MUA-FKT QMUD is employed becomes

$$4.5 \cdot \sqrt{M^{K_{q,g}}/2} \cdot (K_{q,g} \log_2 M + 1)/(K_{q,g} \cdot \log_2 M) = 112.01$$

CFEs per bit, which is slightly higher than the SO-ML MUD's complexity of $4^5/10 = 102.4$ CD-CFEs per bit. Therefore, we introduce a threshold based on $K_{q,g}$ for activating either the use of the SO-DHA-MUA-FKT QMUD or that of the SO-ML MUD. If $K_{q,g} \geq 6$ is true on the qth subcarrier, the SO-DHA-MUA-FKT QMUD is employed, otherwise the SO-ML MUD performs the detection. The resultant SNR performance of the hybrid MUD is approximately 0.5 dB away from the SO-ML MUD's performance limit, while performing 34.1% of the CFEs that the SO-ML MUD requires.

VII. CONCLUSIONS

Diverse SO-DHA QMUDs were designed, which were applied in rank-deficient vehicular DSS/SSCH SDMA-OFDM systems, where the conventional detectors experience an error floor. A range of further measures were also applied for improving the SO-ACO MUDs of [10], [11], while introducing the FW-based mutation. The BER performances of the SO-DHA QMUDs were better than those of the corresponding SO-ACO MUDs, while requiring fewer CFEs per bit than the DHA-QWSA QMUD of [20]. A novel DSS-based USSCH was also proposed for both reducing the complexity of the MUD and for improving the BER performance of the system, when compared to a system where USSCH [7] is used. Finally, a hybrid SO-DHA/SO-ML MUD was proposed for the DSS/SSCH SDMA-OFDM systems.

REFERENCES

[1] L. Hanzo, T. H. Liew, B. Yeap, R. Y. S. Tee, and S. X. Ng, *Turbo Coding, Turbo Equalisation and Space-Time Coding: EXIT-Chart Aided Near-Capacity Designs for Wireless Channels*. New York, NY, USA: Wiley, 2010.

[2] L. Hanzo, L.-L. Yang, E.-L. Kuan, and K. Yen, *Single and Multi-Carrier DS-CDMA: Multi-User Detection, Space-Time Spreading, Synchronisation, Networking, and Standards*. New York, NY, USA: Wiley, 2003.

[3] L. Hanzo, O. Alamri, M. El-Hajjar, and N. Wu, *Near-Capacity Multi-Functional MIMO Systems: Sphere-Packing, Iterative Detection and Cooperation*. Chichester, U.K.: Wiley, May 2009.

[4] S. Chen, A. Livingstone, and L. Hanzo, "Minimum bit-error rate design for space-time equalization-based multiuser detection," *IEEE Trans. Commun.*, vol. 54, no. 5, pp. 824–832, May 2006.

[5] C.-Y. Wei, J. Akhtman, S. X. Ng, and L. Hanzo, "Iterative near-maximum-likelihood detection in rank-deficient downlink SDMA systems," *IEEE Trans. Veh. Technol.*, vol. 57, no. 1, pp. 653–657, Jan. 2008.

[6] C.-Y. Wei, L. Wang, and L. Hanzo, ''Iterative irregular sphere detection in high-rate downlink SDMA systems,'' *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3855–3861, Sep. 2009.

[7] L. Hanzo, Y. Akhtman, M. Jiang, and L. Wang, *MIMO-OFDM for LTE, WIFI and WIMAX: Coherent Versus Non-Coherent and Cooperative Turbo-Transceivers*. New York, NY, USA: Wiley, 2010.

[8] Y. Li, J. H. Winters, and N. R. Sollenberger, ''MIMO-OFDM for wireless communications: Signal detection with enhanced channel estimation,'' *IEEE Trans. Commun.*, vol. 50, no. 9, pp. 1471–1477, Sep. 2002.

[9] M. Jiang and L. Hanzo, ''Multiuser MIMO-OFDM for next-generation wireless systems,'' *Proc. IEEE*, vol. 95, no. 7, pp. 1430–1469, Jul. 2007.

[10] C. Xu, R. G. Maunder, L.-L. Yang, and L. Hanzo, ''Near-optimum multiuser detectors using soft-output ant-colony-optimization for the DS-CDMA uplink,'' *IEEE Signal Process. Lett.*, vol. 16, no. 2, pp. 137–140, Feb. 2009.

[11] C. Xu, B. Hu, L.-L. Yang, and L. Hanzo, ''Ant-colony-based multiuser detection for multifunctional-antenna-array-assisted MC DS-CDMA systems,'' *IEEE Trans. Veh. Technol.*, vol. 57, no. 1, pp. 658–663, Jan. 2008.

[12] E. Martín-López, A. Laing, T. Lawson, R. Alvarez, X.-Q. Zhou, and J. L. O'Brien, ''Experimental realization of Shor's quantum factoring algorithm using qubit recycling,'' *Nature Photon.*, vol. 6, no. 11, pp. 773–776, Nov. 2012.

[13] X.-Q. Zhou, P. Kalasuwan, T. C. Ralph, and J. L. O'Brien, ''Calculating unknown eigenvalues with a quantum algorithm,'' *Nature Photon.*, vol. 7, no. 3, pp. 223–228, Mar. 2013.

[14] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[15] S. Imre and F. Balázs, *Quantum Computing and Communications: An Engineering Approach*. New York, NY, USA: Wiley, 2005.

[16] S. Imre and L. Gyongyosi, *Advanced Quantum Communications: An Engineering Approach*. New York, NY, USA: Wiley, 2013.

[17] S. Imre and F. Balázs, ''Non-coherent multi-user detection based on quantum search,'' in *Proc. IEEE ICC*, vol. 1. May 2002, pp. 283–287.

[18] M. Boyer, G. Brassard, P. Høyer, and A. Tapp, ''Tight bounds on quantum searching,'' *Fortschritte der Physik*, vol. 46, nos. 4–5, pp. 493–506, 1998.

[19] G. Brassard, P. Høyer, and A. Tapp, ''Quantum counting,'' *Autom., Lang. Program.*, vol. 1443, pp. 820–831, May 1998.

[20] P. Botsinis, S. X. Ng, and L. Hanzo. (2013, May). Quantum search algorithms, quantum wireless, and a low-complexity maximum likelihood iterative quantum multi-user detector design. *IEEE Access* [Online]. *1(1)*, pp. 94–122. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6515077

[21] G. Brassard, F. Dupuis, S. Gambs, and A. Tapp, ''An optimal quantum algorithm to approximate the mean and its application for approximating the median of a set of points over an arbitrary distance,'' Jun. 2011.

[22] P. Botsinis, S. X. Ng, and L. Hanzo, ''Fixed-complexity quantum-assisted multi-user detection for CDMA and SDMA,'' *IEEE Trans. Commun.*, vol. 62, no. 3, pp. 990–1000, Mar. 2014.

[23] C. Durr and P. Høyer, ''A quantum algorithm for finding the minimum,'' Jul. 1996.

[24] L. K. Grover, ''A fast quantum mechanical algorithm for database search,'' in *Proc. 28th Annu. ACM Symp. Theory Comput.*, May 1996, pp. 212–219.

[25] N. Zhao, Z. Wu, Y. Zhao, and T. Quan, ''A population declining mutated ant colony optimization multiuser detector for MC-CDMA,'' *IEEE Commun. Lett.*, vol. 14, no. 6, pp. 497–499, Jun. 2010.

[26] L. K. Grover, ''Quantum mechanics helps in searching for a needle in a haystack,'' *Phys. Rev. Lett.*, vol. 79, no. 9, pp. 325–328, Jul. 1997.

[27] M. Dorigo and L. M. Gambardella, ''Ant colony system: A cooperative learning approach to the traveling salesman problem,'' *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, Apr. 1997.

[28] S. Sesia, I. Toufik, and M. Baker, *LTE—The UMTS Long Term Evolution: From Theory to Practice*, 2nd ed. New York, NY, USA: Wiley, 2011.

PANAGIOTIS BOTSINIS (S'12) received the Diploma (M.Eng.) degree from the School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece, in 2010, and the M.Sc. (Hons.) degree in wireless communications from the University of Southampton, Southampton, U.K., in 2011, where he is currently pursuing the Ph.D. degree with the Communications, Signal Processing and Control Group, School of Electronics and Computer Science. Since 2010, he has been a member of the Technical Chamber of Greece.

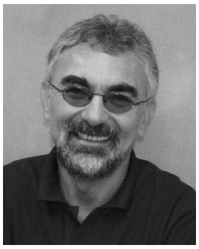
His research interests include quantum-assisted communications, iterative detection, OFDM, MIMO, multiple access systems, coded modulation, channel coding, cooperative communications, and combinatorial optimization.

DIMITRIOS ALANIS (S'13) received the M.Eng. degree in electrical and computer engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2011, and the M.Sc. degree in wireless communications from the University of Southampton, Southampton, U.K., in 2012, where he is currently pursuing the Ph.D. degree with the Communications, Signal Processing and Control Group, School of Electronics and Computer Science.

His research interests include quantum computation and quantum information theory, quantum search algorithms, cooperative communications, resource allocation for self-organizing networks, bioinspired optimization algorithms, and classical and quantum game theory.

SOON XIN NG (S'99–M'03–SM'08) received the B.Eng. (Hons.) degree in electronics engineering and the Ph.D. degree in wireless communications from the University of Southampton, Southampton, U.K., in 1999 and 2002, respectively. From 2003 to 2006, he was a Post-Doctoral Research Fellow, involved in collaborative European research projects known as SCOUT, NEWCOM, and PHOENIX. Since 2006, he has been a member of the academic staff with the School of Electronics and Computer Science, University of Southampton. He is involved in the OPTIMIX and CONCERTO European projects, and the IU-ATC and UC4G projects. He is currently a Senior Lecturer with the University of Southampton.

His research interests include adaptive coded modulation, coded modulation, channel coding, space-time coding, joint source and channel coding, iterative detection, OFDM, MIMO, cooperative communications, distributed coding, quantum error correction codes, and joint wireless and optical-fiber communications. He has authored over 150 papers and co-authored two John Wiley/IEEE Press books in this field. He is a Chartered Engineer and a fellow of the Higher Education Academy in the U.K.



LAJOS HANZO is a fellow of the Royal Academy of Engineering, the Institution of Engineering and Technology, and the European Association for Signal Processing. He received the D.Sc. degree in electronics and the Ph.D. degree in 1976 and 1983, respectively. He was a recipient of the Doctor Honoris Causa Award from the Technical University of Budapest in 2009. During his 35-year career in telecommunications, he has held various research and academic positions in Hungary, Germany, and the U.K. Since 1986, he has been with the School of Electronics and Computer Science, University of Southampton, Southampton, U.K., where he is the Chair of Telecommunications. He has supervised 80 Ph.D. students, co-authored 20 John Wiley/IEEE Press books in mobile radio communications totaling in excess of 10 000 pages, authored more than 1 300 research entries on IEEE Xplore, was the TPC and General Chair of the IEEE conferences, presented keynote lectures, and has been awarded a number of distinctions. He is currently directing 100-strong academic research team, working on a range of research projects in the field of wireless multimedia communications sponsored by the Engineering and Physical Sciences Research Council, U.K., the European IST Program, and the Mobile Virtual Centre of Excellence, U.K. He is an enthusiastic supporter of industrial and academic liaison, and he offers a range of industrial courses. He is also a Governor of the IEEE Vehicular Technology Society. From 2008 to 2012, he was the Editor-in-Chief of the IEEE Press and also a Chaired Professor at Tsinghua University, Beijing, China. His research is funded by the European Research Council's Senior Research Fellow Grant. He has more than 16 000 citations.