

Evolutionary Random Walk Aided Stochastic Sphere Encoder for Broadband G.mgfast

Yangyishi Zhang
Applied Research
British Telecom
Ipswich, UK
yangyishi.zhang@bt.com

Jiankang Zhang
Department of Computer Science
Bournemouth University
Bournemouth, UK
jzhang3@bournemouth.ac.uk

Anas Al Rawi
Department of Physics
University of Cambridge
Cambridge, UK
afm64@cam.ac.uk

Abstract—The next generation digital subscriber line (DSL) standard G.mgfast introduces far stronger co-channel interference termed as far-end crosstalk (FEXT) than the existing ones. Given perfect transmitter-side channel state information (CSIT), it is well known that the lattice-reduction-aided \mathcal{K} -best sphere encoder (LR-KBSE) is a near-optimal transmit precoding (TPC) technique compared to the classic (LR-) depth-first sphere encoder (DFSE), albeit having significantly lower complexity than the latter. However, the decision feedback precoding (DFP) structure and the Schnorr-Euchner enumeration procedure, both perceived as state-of-the-art in the literature, are not provably optimal for solving the closest vector problem (CVP) embedded in sphere encoding. As a counterexample, this paper proposes a stochastic sphere encoder (SSE) relying on differential evolution aided random walk over lattices. The parallel processing complexity, memory efficiency and signal to noise ratio (SNR) improvement of the proposed SSE are all shown to be superior to the LR-KBSE for G.mgfast systems.

Index Terms—Differential evolution, digital subscriber line, G.mgfast, far-end crosstalk, transmit precoding, lattice reduction, sphere encoder, random walk

I. INTRODUCTION

The backhaul support for fifth-generation (5G) mobile access and Internet of Things (IoT) applications requires high throughput, high reliability and low latency. As the globally dominating wireline access technology, digital subscriber line (DSL) systems constitute a widely-deployed type of backhaul solution for indoor and outdoor mobile access. The latest DSL standard G.mgfast aims to support multi-Gigabit nominal throughput by exploiting at least double the baseband bandwidth of its predecessor G.fast. However, due to the propagation loss and mutual coupling between those closely-spaced pairs of twisted copper wires used by DSL networks, the extra bandwidth introduced in G.mgfast systems usually leads to worse channel quality.

Multi-pair DSL systems are conventionally modelled as a family of quasi-static yet strongly frequency-selective multiple-input-multiple-output (MIMO) systems. For multi-pair G.mgfast, it has been recognized that the co-channel interference known as far-end crosstalk (FEXT) is the dominant source of impairment particularly at high frequencies. Classic linear zero forcing precoding (ZFP) is not powerful enough

for the strong FEXT at high frequencies. Such limitations have stimulated research in both non-linear precoding [2] and regularized linear precoding [3]. It is known that vector perturbation (VP) [4] is a near-optimal transmit precoding (TPC) technique in terms of minimising the signal to noise ratio (SNR) penalty caused by ZFP. However, the success of VP predominantly depends on *optimally* solving the *NP-hard* closest vector problem (CVP) over the lattice associated with the channel inverse matrix [5]. Unfortunately, even the best algorithm for solving exactly a general CVP in d dimension has a complexity order of $2^{0.264d+\mathcal{O}(d)}$, while requiring a buffer size of $2^{0.185d+\mathcal{O}(d)}$ [6].

Practical implementations of VP for TPC mainly rely on a sphere encoder based enumeration architecture. The family of sphere encoders generally consist of depth-first and best-first variations, both primarily rely on decision feedback precoding (DFP) for finding an initial point, followed by Schnorr-Euchner enumeration [7] of adjacent points in subspaces of the lattice. Despite the popular belief that DFP-aided Schnorr-Euchner enumeration is the most efficient *deterministic* sphere encoding framework [8], recent literature in cryptanalysis showed that *stochastic* processes may potentially achieve better trade-off between performance and complexity, particularly in high-dimensional spaces [9]. Such discoveries contribute to the escalating research interest in novel MIMO algorithms for the large-scale regime of next-gen networks. Moreover, since DSL channels are widely known to have long coherence time, sphere encoders may achieve their theoretical potential due to the more lenient processing delay constraint.

In this paper, FEXT cancellation for G.mgfast systems is addressed with the aid of a stochastic sphere encoder (SSE) design example. The proposed SSE architecture incorporates a partial *random walk* strategy over lattice grids directed by the *evolutionary algorithm*. This paper proposes the first artificial intelligence (AI) aided SSE framework for G.mgfast systems that disproves the conjectured optimality of Schnorr-Euchner enumeration.

II. SYSTEM MODEL

A. DSL Channel Environment

All existing DSL access networks operate over pairs of twisted copper wires, with one or more dedicated copper

This paper is an update to Sec. 2.7 of the thesis [1]. The channel data as shown in Fig. 1 were recorded at Adastral Park, Ipswich, UK.

pairs connecting a customer premise equipment (CPE) and its designated distribution point unit (DPU). When one DPU serves more than one CPE in an area, the pairs belonging to independent CPEs are often bound together as a large DSL binder. It has been widely recognized that such kind of wireline network environment is quasi-static over time, albeit showing strong frequency selectivity. Furthermore, the FEXT between closely spaced pairs relative to the direct channel gain is also an increasing function of the signalling frequency. This is shown in the channel measurements of Fig. 1.

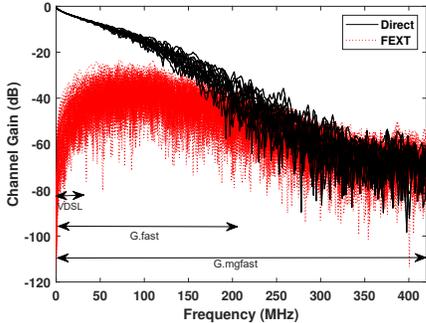


Fig. 1. G.mgfast frequency domain transfer characteristics for directly connected channels and FEXT coupling channels, measured for a 50m 16-pair binder.

B. Vector Perturbation Precoder

G.mgfast employs a multicarrier scheme known as discrete multitone (DMT), which leads to a collection of T independent MIMO subsystems in the frequency domain. MIMO precoding jointly treats multiple independently-transmitted data streams (via n pairs of copper wires) in the escalated high-dimensional signal space as a vector. Such a system is modelled as:

$$\mathbf{y}^t = \mathbf{H}^t \mathbf{x}^t + \mathbf{n}^t, (t = 1, 2, \dots, T). \quad (1)$$

Since each subsystem operates independently, we drop the tone (subcarrier) index t here for the simplicity of lattice analysis. $\mathbf{x}, \mathbf{y}, \mathbf{n} \in \mathbb{R}^n$ are the transmitted data, received data and noise vectors, respectively. $\mathbf{H} \in \mathbb{R}^{n \times n}$ represents the MIMO channel matrix. As the TPC dual counterpart of the optimal maximum likelihood detector, VP precoding [4] performs the following TPC operations:

$$\mathbf{x} = \mathbf{G}(\mathbf{u} - \mathbf{l}'), \quad (2)$$

where $\mathbf{G} = \mathbf{H}^\dagger$ is the (pseudo-)inverse channel matrix. The original message-carrying data vector $\mathbf{u} \in \mathbb{R}^n$ is assumed to be drawn from the square-shaped quadrature amplitude modulated (QAM) constellations. For analytical simplicity, we characterise the multi-stream hyper QAM constellation as the hypercubic set $\mathcal{U} = [-1/2, 1/2]^n$. $\mathbf{l}' \in \mathbb{Z}^n$ is a perturbation vector. Given that \mathbf{x} needs to be scaled by a normalisation factor of $\xi \triangleq \|\mathbf{x}\|$, we have the following formulation of \mathbf{y} by substituting (2) into (1):

$$\mathbf{y} = \frac{\mathbf{u} - \mathbf{l}'}{\xi} + \mathbf{n}. \quad (3)$$

Without receiver cooperation, an estimate \mathbf{z} of the message \mathbf{u} can be found by isolating the fractional part of $\xi \mathbf{y}$ with the aid of the modulo operator:

$$\begin{aligned} \mathbf{z} &\triangleq \text{mod}(\xi \mathbf{y}) \\ &= \mathbf{u} + \text{mod}(\xi \mathbf{n}). \end{aligned} \quad (4)$$

Since \mathbf{n} is Gaussian distributed, (4) has a vanishing probability of error when $E[\|\xi \mathbf{n}\|^2]$ tends to zero, where ξ may also be interpreted as the signal-to-noise ratio (SNR) penalty¹. In order to achieve this, the optimal perturbation vector $\bar{\mathbf{l}}$ is chosen to minimise ξ , which results in the following CVP for $\mathcal{L}(\mathbf{G})$:

$$\bar{\mathbf{l}} = \arg \min_{\mathbf{l}' \in \mathbb{Z}^n} \|\mathbf{x}^* - \mathbf{G}\mathbf{l}'\|^2, \quad (5)$$

where $\mathbf{x}^* = \mathbf{G}\mathbf{u}$ is the transmitted signal as if it were encoded by linear ZFP. Therefore, (5) finds the vector of $\mathcal{L}(\mathbf{G})$ that is the closest to \mathbf{x}^* .

III. CLASSIC LR-AIDED SPHERE ENCODERS

Even though approximate solutions of (5) relying on both the LLL preprocessing and Babai's heuristic procedures [11], such as the TPC techniques of [12], are readily powerful, their respective near-optimality was only observed for small-scale wireless and fixed MIMO systems. Moreover, it is a well-known result that approximating the CVP within a constant proximity factor is NP-hard [13]. Therefore, improved CVP approximations over Babai's methods are required as the system gradually becomes large-scale. In this section, we first review Babai's classic CVP approximations, and then the main subcategories of sphere encoders, namely the depth-first (DFSE) and \mathcal{K} -best (KBSE) variations. Since LR is invoked in the remainder of this paper, both (LR-)DFSE and (LR-)KBSE may be perceived as CVP with preprocessing (CVPP).

A. Preprocessing

Babai's rounding-off and nearest-plane algorithms are equivalent to LR-aided ZFP and LR-aided decision feedback precoding (DFP), respectively. They are also widely employed as the initialisation routines of sphere encoders. Given the CVP (5), we may firstly formulate the LR preprocessing as the following joint matrix factorisation:

$$\mathbf{G} = \mathbf{Q}\mathbf{R}\mathbf{Z}^\dagger, \quad (6)$$

where $\mathbf{Q}^H \mathbf{Q} = \mathbf{I}$, and \mathbf{R} is upper triangular. \mathbf{Z} is the transformation matrix as found by the LLL algorithm. Therefore, $\mathbf{Q}\mathbf{R}$ constitutes the LLL-reduced basis. Let $\mathbf{v} = \mathbf{Q}^H \mathbf{x}^*$, $\mathbf{l} = \mathbf{Z}^\dagger \mathbf{l}'$ and \mathbf{R} be that of (6). (5) is now equivalently formulated as:

$$\bar{\mathbf{l}} = \mathbf{Z}(\arg \min_{\mathbf{l}} \|\mathbf{v} - \mathbf{R}\mathbf{l}\|^2). \quad (7)$$

Based on (7), Babai's rounding-off approximation is then formulated as:

$$\mathbf{l}_{\text{RO}} = \mathbf{Z} \lceil \mathbf{Z}^\dagger \mathbf{u} \rceil. \quad (8)$$

¹In fact, $E[\|\xi \mathbf{n}\|^2]$ and the noise variance of (4) were shown to differ by the modulo loss [10], which vanishes in the high-SNR regime.

where $\lceil \cdot \rceil$ represents element-wise rounding off to the nearest integer. Similarly, the nearest-plane approximation is found by performing DFP recursively, defined as follows:

$$l_k = \lceil c_k \rceil = \begin{cases} \lceil \frac{v_k}{r_{k,k}} \rceil & k = n, \\ \lceil \frac{v_k}{r_{k,k}} - \sum_{j=k+1}^n \frac{r_{k,j}}{r_{k,k}} l_j \rceil & k = n-1, \dots, 1. \end{cases} \quad (9)$$

The recursion of (9) is performed top-down sequentially from $k = n$ to $k = 1$, which solves the sub-problem within the parenthesis of (7). The vector $\mathbf{l}_{\text{NP}} = \mathbf{Z}\mathbf{l}$ is the nearest-plane solution of (5).

B. Depth-First Sphere Encoder

The success of DFSE (Fig. 2) relies on bounding the total number of lattice points to be enumerated. Additionally, if a point of the lattice is the closest to the target \mathbf{x}^* within a bounded distance $\sqrt{\beta}$, said lattice point must also be the closest to the reference over the infinite lattice. DFSE attempts to gradually reduce the search radius $\sqrt{\beta}$ until no more lattice points may be found in the interior of the hypersphere. As a result, it is more practical to commence with one of Babai's approximations.

Since (7) is defined for a triangular matrix, the following necessary condition of the hypersphere constraint holds:

$$r_{k,k}^2 (l_k - c_k)^2 < \beta - \sum_{j=k+1}^n r_{j,j}^2 (l_j - c_j)^2. \quad (10)$$

Explicitly, the Euclidean distance metric $\|\mathbf{v} - \mathbf{R}\mathbf{l}\|$ is now decoupled into subspaces, termed as *partial Euclidean distance* (PED), which allows for evaluating the hypersphere constraint at each DFP iteration k . Thus, enumeration is made possible as an intermediate operation during each DFP step. An efficient way of enumerating the closest integer neighbours of a given center $l^{(1)} \in \mathbb{Z}$ is to traverse the *Schnorr-Euchner* series [7]:

$$l^{(n)} = l^{(1)} + (-1)^{n+1} \lfloor \frac{n}{2} \rfloor, \quad n \in \mathbb{Z}^+, \quad (11)$$

where $\lfloor \cdot \rfloor$ and \mathbb{Z}^+ denote the floor operation and the set of positive integers, respectively. The main steps of DFSE are summarised as follows.

Assuming that (10) is satisfied at layer $k+1$, DFP is immediately performed at layer k as formulated in (9). This finds the first term $l_k^{(i=1)}$ in the Schnorr-Euchner series (11) associated with layer k , as well as overwriting the pre-existing Schnorr-Euchner process of the same layer. If (10) is not satisfied at layer k and $k < n$, DFSE retreats to layer $k+1$ and enumerates the next term in its associated Schnorr-Euchner series. If (10) is satisfied at layer k and $k = 1$, a new integer vector representing a lattice point closer to the target \mathbf{v} is found and written to the output buffer. β is reduced accordingly to be the new squared distance. If (10) is not satisfied at layer k and $k = n$, it is no longer possible to find a new integer vector representing a lattice point closer to the target \mathbf{v} . Therefore, DFSE terminates. The most recently cached integer vector is the final output of DFSE.

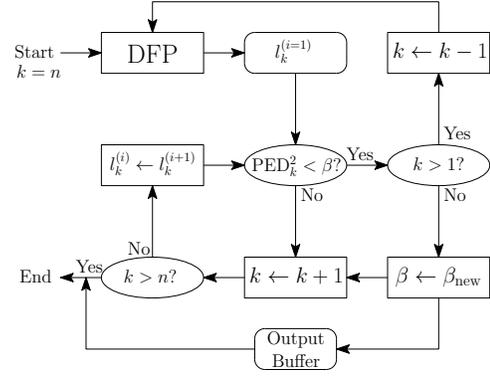


Fig. 2. Depth-First Sphere Encoder.

C. \mathcal{K} -Best Sphere Encoder

In contrast to the bidirectional search tree traversing strategy of DFSE, KBSE (Fig. 3) only progresses through the search tree (from the root $k = n$ to the leaf $k = 1$) once, dispensing with any backwards travel. In general, this reduces KBSE's complexity order from exponential to polynomial, at the risk of prematurely pruning the globally optimal path. However, KBSE needs more buffer than DFSE, while also having to sort its memory iteratively.

If no hypersphere constraint β is invoked, KBSE performs a fixed number of operations. Conventionally, a KBSE implementation that generates \mathcal{K} output candidates requires a processing buffer of size \mathcal{K}^2 . The main steps of KBSE are summarised as follows:

At the beginning of KBSE, DFP is performed once at layer n as formulated in (9). This finds the first term $l_n^{(i=1)}$ in the Schnorr-Euchner series associated with layer n . The next $\mathcal{K} - 1$ terms until $l_n^{(\mathcal{K})}$ are also extracted. Each of the \mathcal{K} cached (partially-filled) integer vectors generated from layer $k+1$ spawns \mathcal{K} candidates at layer k in the same way as in initialization. Thus, the processing buffer needs to cache a total number of \mathcal{K}^2 partially-filled integer vectors for any given layer except for the n th. The \mathcal{K}^2 candidate vectors (\mathcal{K} for layer n) generated in the previous step are sorted by their respective *partial squared distance* to the target \mathbf{v} in ascending order, as defined by the summation on the RHS of (10). The first \mathcal{K} candidates in the sorted list of layer k are admitted to layer $k-1$. The integer vectors enumerated at layer 1 are no longer partially-filled. Therefore, KBSE only needs to find the optimal candidate having the shortest Euclidean distance to \mathbf{v} instead of performing the standard \mathcal{K} -min operation. KBSE terminates when the optimal candidate is found.

Due to its DFP-based structure, the worst-case performance of KBSE is given by the nearest-plane approximation \mathbf{l}_{NP} formulated in (9). Therefore, if we set a hypersphere constraint of $\sqrt{\beta} < \|\mathbf{x}^* - \mathbf{G}\mathbf{l}_{\text{NP}}\|$, KBSE may need to enumerate more entries in the Schnorr-Euchner series than the unconstrained version does in order to find all legitimate candidates. This paper will mainly focus on the unconstrained version which has a deterministic complexity.

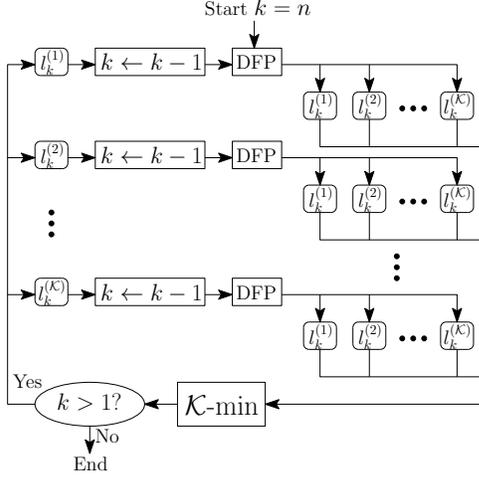


Fig. 3. \mathcal{K} -Best Sphere Encoder.

IV. STOCHASTIC SPHERE ENCODER

A. Directional Walk over Lattices

The geometric nature of most existing sphere encoding algorithms is to traverse the lattice grids under the constraint of (partial) Euclidean distance. Unlike the case of gradient descent over continuous surfaces, there is no trivial algorithm that finds the most efficient path over general lattices when solving the CVP. Namely, the main bottleneck of enumeration-based algorithms is the inability of determining the best ‘direction of travel’ per step. In this section, we propose a novel SSE framework relying on evolutionary algorithm assisted random walk for FEXT cancellation in G.mgfast. The evolutionary MIMO detector conceived in [14] may be viewed as an upstream dual counterpart of the presented SSE.

B. Evolutionary Random Walk Algorithm

Using a format similar to gradient descent, the proposed evolutionary random walk over a lattice $\mathcal{L}(\mathbf{R})$ is iteratively formulated in the context of solving (7) as follows:

$$\mathbf{w}_{g+1} = \mathbf{w}_g + \mathbf{R}\hat{\nabla}(\mathbf{w}_g), \quad (12)$$

where $\mathbf{w}_g = \mathbf{R}\mathbf{l}_g$ represents the starting lattice point of the g th step. To simulate the fast converging property of gradient descent, we define the random walk operator $\hat{\nabla}(\mathbf{w})$ as taking a step (over the lattice basis grid) that *approximately* results in the minimum distance to the target \mathbf{v} of (7):

$$\hat{\nabla}(\mathbf{w}) \triangleq \arg \min_{\xi \in \mathbb{B}^n} \|\mathbf{w} + \mathbf{R}\xi - \mathbf{v}\|^2, \quad (13)$$

where $\mathbb{B} \triangleq \{-1, 0, 1\}$. The complexity of solving (13) by exhaustively searching the entire set \mathbb{B}^n increases exponentially in n . Therefore, it is implicitly solved by performing discrete differential evolution over a subset of \mathbb{B}^n . More specifically, (13) is characterised as the mutation and crossover of a population $\{\mathbf{l}\}$ of candidates $\mathbf{l} \in \mathbb{Z}^n$ for (7).

For simplicity of notation, let $\gamma(\mathbf{l}) \triangleq \|\mathbf{v} - \mathbf{R}\mathbf{l}\|^2$ be the fitness of a member \mathbf{l} during evolutionary random walking for

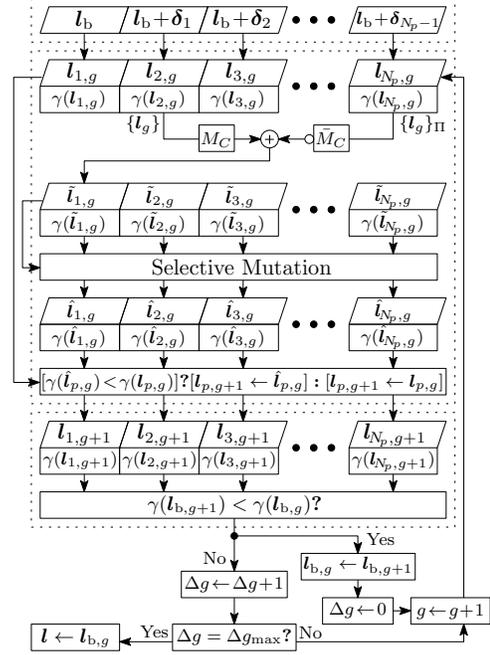


Fig. 4. Evolutionary random walk aided SSE. The initial population is found by Babai’s rounding off algorithm [11] plus a random sparse binary vector δ_p for $p = 1, 2, \dots, N_p - 1$.

solving (7), given a fixed target \mathbf{v} and the reduced lattice basis \mathbf{R} . The detailed steps of evolutionary random walk (Fig. 4) are summarised as follows:

1) **Initialization:** Babai’s rounding-off algorithm is performed, which finds an initial reference coolant vector \mathbf{l}_b . If the QoS target cannot be satisfied with the aid of \mathbf{l}_b , then the members of the initial population $\{\mathbf{l}_1\}$ are initialized according to:

$$\mathbf{l}_{p,1} = \begin{cases} \mathbf{l}_b & p = 1, \\ \mathbf{l}_b + \delta_{p-1} & p = 2, 3, \dots, N_p, \end{cases} \quad (14)$$

where N_p is the size of the population, while δ_p represents randomly-generated sparse vectors whose non-zero entries are drawn from ± 1 . This initialization approach ensures the near-optimality of the initial population. If the buffer supports a large N_p , then δ_p is allowed to draw uniformly distributed random integers from a wider symmetric interval such as $[-2, 0) \cup (0, 2]$. Due to the lack of ergodicity of lattice vectors, allowing a relatively dense fluctuation vector δ_p over small N_p may increase the risk of significantly reducing the quality of the initial population $\{\mathbf{l}_1\}$. This drawback outweighs the small rise of diverging from a locally optimal lattice vector.

2) **Crossover:** Also known as *discrete recombination*, this specific operation uses a crossover mask M_C to select and recombine the component vectors, i.e. subspace projections, from a randomly-selected pair of members of the current generation g . Specifically, the ‘child set’ in the g th generation $\{\mathbf{l}_g\}$ is produced as follows:

$$\tilde{\mathbf{l}}_{p,g} \supset \tilde{\mathbf{l}}_{k,p,g} = \begin{cases} \mathbf{l}_{k,p,g}, & \text{if } \text{rand}(0, 1) < M_C, \\ \mathbf{l}_{k,p,g}^\Pi, & \text{otherwise,} \end{cases} \quad (15)$$

where $\{\mathbf{l}_g\}_\Pi \supset \mathbf{l}_{p,g}^\Pi \supset \mathbf{l}_{k,p,g}^\Pi$ represents a random permutation Π of the ‘parent set’ $\{\mathbf{l}_g\}$, and $\text{rand}(0,1)$ represents a uniformly distributed random variable within the range $(0,1)$. If all of the member vectors in the ‘parent set’ are sufficiently fit and non-overlapping, they should be close to each other in the lattice. As a result, the probability of producing significantly worse member vectors for the ‘child set’ becomes low, if we simply recombine the component vectors.

3) **Mutation:** In contrast to the regular definition of mutation in the differential evolution algorithm, we define the specific operation as a random jump over the lattice grid, which is characterized by the LLL-reduced basis obtained during initialization. This modification addresses a potential problem associated with using the regular mutation strategy in our context. More specifically, the difference of two close lattice vectors may be very far away from either of them, which does not constitute a good ‘trial set’ $\{\hat{\mathbf{l}}_{p,g}\}$. In particular, the mutation operation is defined for a random movement vector $\tilde{\delta}_p$ as follows:

$$\hat{\mathbf{l}}_{p,g} = \begin{cases} \tilde{\mathbf{l}}_{p,g} + \tilde{\delta}_p, & \text{if } \gamma(\tilde{\mathbf{l}}_{p,g} + \tilde{\delta}_p) - \gamma(\tilde{\mathbf{l}}_{p,g}) < 0, \\ \tilde{\mathbf{l}}_{p,g}, & \text{otherwise,} \end{cases} \quad (16)$$

Unlike in (14), the random jump $\tilde{\delta}_p$ in (16) does not have to be based on a sparse vector, but its entries must be selected strictly from $\{-1, 0, 1\}$ in order to guarantee that the change is only by a single grid position. Moreover, the mutation strategy in (16) does not rely on calculating the difference of the member vectors. Instead, it uses the difference of the cost function values, i.e. the Euclidean distance from the target vector, in order to produce a ‘trial set’ that is at least as good as the ‘child set’. This is the principal stage that characterizes the ‘progressive sieving’ concept. However, this operation incurs an additional N_p cost function evaluations compared to the conventional mutation.

4) **Selection:** In order to determine the member vectors to be retained for the $(g+1)$ st generation, the cost of the ‘trial set’ is compared against that of the ‘parent set’. Because the costs of both the ‘trial set’ and of the ‘parent set’ are known, there are no additional cost function evaluations at this stage. Similar to (16), each member vector of the $(g+1)$ st generation may be formulated as:

$$\mathbf{l}_{p,g+1} = \begin{cases} \hat{\mathbf{l}}_{p,g}, & \text{if } \gamma(\hat{\mathbf{l}}_{p,g}) - \gamma(\mathbf{l}_{p,g}) < 0, \\ \mathbf{l}_{p,g}, & \text{otherwise.} \end{cases} \quad (17)$$

When the new ‘parent set’ of the $(g+1)$ st generation is filled, the cost of the member vectors is compared to that of the current best member. The best member vector is updated if the best member in the new ‘parent set’ has a lower cost than the existing one, where the ‘converged generation counter’ Δg is reset to zero. If no new best member can be found, then Δg is increased by one. The algorithm terminates when Δg reaches its designated limit Δg_{\max} .

V. PERFORMANCE EVALUATION

In this section, the performances of our proposed SSE will be compared to the KBSE benchmark, in terms of their impact

on the received signal’s average SNR, parallel processing complexity and lattice theoretical characteristics. We define these performance metrics as follows.

Assuming the transmit power spectral density limit is $E\{\|\mathbf{x}^t\|^2\} = 1$ for a given tone t , then the average received SNR is reduced by a penalty factor due to the normalization of DPU output (3) based on the optimization result of the cost function (5):

$$\text{SNR Penalty} \triangleq \frac{1}{T} \sum_{t=1}^T \|\mathbf{G}^t(\mathbf{u}^t - \mathbf{l}^t)\|^2. \quad (18)$$

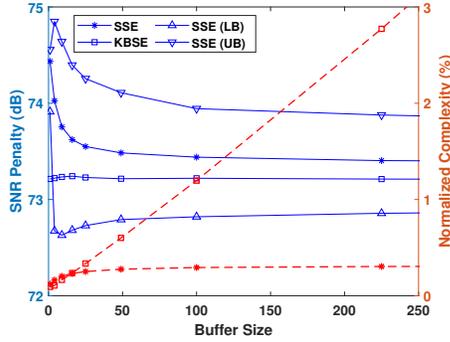
The complexity of an algorithm solving the CVP is classically characterised by the size of its search space, namely, the total number of lattice points to be processed. In order to emphasise the trade-off between processing delay and buffer usage, we will evaluate a closely related metric termed as the *parallel processing complexity*. For fair comparisons, the parallel processing complexity of a sphere encoder is defined as the *number of concurrent-enumeration and swap (CES) operations*. For a *concurrent enumeration* step, each free memory address of the parallel buffer is assigned an element from a list of candidates (e.g. the Schnorr-Euchner list) concurrently. Meanwhile, a *swap* is a step where the memory addresses of two buffered candidates are exchanged to satisfy the ordering criterion.

The performance comparisons between KBSE and SSE are collectively portrayed in Fig. 5 for full bandwidth operation (Fig. 5(a)) and partial bandwidth operation above G.fast frequencies (Fig. 5(b)), assuming that they both have access to an ideal buffer of the same size. Since SSE is randomized, its upper and lower bound performances are also shown.

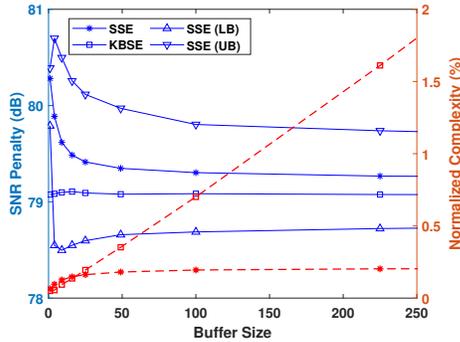
From a lattice-theoretical perspective, the performance measure equivalent to the SNR penalty (18) becomes the minimum Euclidean distance in the multi-user signal space. In this respect, a pair of closely related performance metrics, namely the *proximity factor* [15] \mathcal{F}_0 and the *success probability* P_s [16], is selected for evaluating the algorithms. Here, we define the success probability of SSE in terms of its \mathcal{F}_0 when compared to the KBSE benchmark:

$$P_s \triangleq P\left(\frac{\|\mathbf{G}(\mathbf{u} - \mathbf{l}'_{\text{SSE}})\|}{\|\mathbf{G}(\mathbf{u} - \mathbf{l}'_{\text{KBSE}})\|} \leq \mathcal{F}_0\right). \quad (19)$$

The trade-off is portrayed in Fig. 6. Bearing in mind that in full bandwidth operation, the lower half 0 – 212 MHz spectrum does not cause nearly the same degree of SNR penalty as the higher half, and therefore the performance gap between KBSE and SSE is negligible. Such behaviours are demonstrated by the pair of success probability curves in Fig. 6 showing $\mathcal{F}_0 = 1$. When compared to Fig. 5, it may be observed that the optimal operating point of SSE needs a buffer size of 50. Assuming that $\mathcal{F}_0 < 1$, we have $P_s \approx 1/3$ for the above 212 MHz band, and $P_s \approx 1/5$ for the full band. Therefore, in automatic repeat request (ARQ) aided systems, SSE is expected to outperform KBSE when at least four extra retransmissions are allowed for each user data vector. With at



(a) 0 – 424 MHz full bandwidth



(b) 212 – 424 MHz half bandwidth

Fig. 5. Average SNR penalty and normalized complexity of KBSE and SSE, assuming that the parallel processing buffers have the same size in both cases. Complexity is normalized against the number of sequential enumeration steps of DFSE, requiring no parallel processing buffer.

least one extra transmission allowed, the performance of SSE is likely to be at least as good as KBSE.

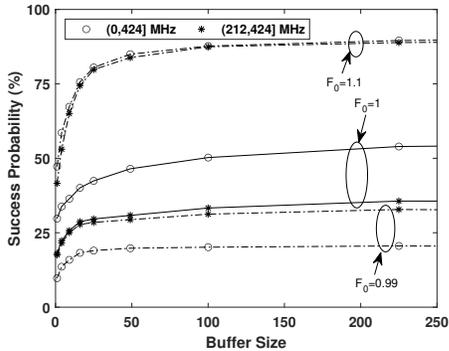


Fig. 6. The success probability of SSE when operating on G.mgfast’s full bandwidth and the above 212 MHz spectrum, respectively.

VI. CONCLUSIONS

This paper investigated LR-aided TPC techniques for mitigating the hostile FEXT environment of broadband multi-pair G.mgfast systems. In particular, we assessed the family of conventional sphere encoders, and proposed a fully parallel evolutionary random walk assisted SSE. By comparing to

KBSE in simulations, we found that the parallel processing complexity of SSE scales much more favourably with the buffer size than KBSE, whereas the average SNR penalty incurred by SSE converges to approximately 0.2 dB higher than KBSE. However, the best case SNR improvement of SSE over KBSE is above 0.4 dB for all buffer sizes. Additionally, SSE is likely to outperform KBSE in ARQ aided systems that only needs a few retransmissions. Finally, the design of SSE is also applicable to a wider range of MIMO systems beyond multi-pair G.mgfast, such as massive MIMO or fixed wireless systems where there is little CSI variations.

REFERENCES

- [1] Y. Zhang, “Signal processing for ultra fast next generation metallic access network,” Ph.D. dissertation, University of Southampton, Jan. 2020.
- [2] W. Lanneer, C. Nuzman, Y. Lefevre, P. Tsiaflakis, W. Coomans, and M. Moonen, “Lattice reduction aided precoding design in downstream G.fast DSL networks,” *IEEE Access*, vol. 8, pp. 19 208–19 220, 2020.
- [3] W. Lanneer, P. Tsiaflakis, J. Maes, and M. Moonen, “Linear and non-linear precoding based dynamic spectrum management for downstream vectored G.fast transmission,” *IEEE Trans. Commun.*, vol. 65, no. 3, pp. 1247–1259, March 2017.
- [4] B. M. Hochwald, C. B. Peel, and A. L. Swindlehurst, “A vector-perturbation technique for near-capacity multi-antenna multiuser communication-part II: perturbation,” *IEEE Trans. Commun.*, vol. 53, no. 3, pp. 537 – 544, Mar 2005.
- [5] D. J. Ryan, I. B. Collings, I. V. L. Clarkson, and R. W. Heath, “Performance of vector perturbation multiuser MIMO systems with limited feedback,” *IEEE Trans. Commun.*, vol. 57, no. 9, pp. 2633 – 2644, Oct 2009.
- [6] L. Ducas, T. Laarhoven, and W. P. J. van Woerden, “The Randomized Slicer for CVPP: Sharper, Faster, Smaller, Batchier,” in *Public-Key Cryptography – PKC 2020*, A. Kiayias, M. Kohlweiss, P. Wallden, and V. Zikas, Eds. Cham: Springer International Publishing, 2020, pp. 3–36.
- [7] C. P. Schnorr and M. Euchner, “Lattice basis reduction: Improved practical algorithms and solving subset sum problems,” *Mathematical Programming*, vol. 66, no. 1, pp. 181–199, 1994.
- [8] A. Ghasemmehdi and E. Agrell, “Faster recursions in sphere decoding,” *IEEE Trans. Inf. Theory*, vol. 57, no. 6, pp. 3530–3536, 2011.
- [9] M. R. Albrecht, L. Ducas, G. Herold, E. Kirshanova, E. W. Postlethwaite, and M. Stevens, “The general sieve kernel and new records in lattice reduction,” in *Advances in Cryptology – EUROCRYPT 2019*. Cham: Springer International Publishing, 2019, pp. 717–746.
- [10] M. L. Honig, *Capacity-approaching multiuser communications over multiple input/multiple output broadcast channels*. Wiley-IEEE Press, 2009, pp. 384–.
- [11] L. Babai, “On Lovász’ lattice reduction and the nearest lattice point problem,” *Combinatorica*, vol. 6, no. 1, pp. 1–13, 1986.
- [12] Y. Zhang, R. Zhang, A. F. A. Rawi, and L. Hanzo, “Approximate perturbation aided lattice encoding (APPLE) for G.fast and beyond,” *IEEE Access*, vol. 6, pp. 53 438–53 451, 2018.
- [13] J. Conway and N. Sloane, *Sphere packings, lattices, and groups*, ser. Grundlehren der mathematischen Wissenschaften. Springer-Verlag, 1993.
- [14] J. Zhang, S. Chen, R. Zhang, A. F. A. Rawi, and L. Hanzo, “Differential evolution algorithm aided turbo channel estimation and multi-user detection for G.fast systems in the presence of FEXT,” *IEEE Access*, vol. 6, pp. 33 111–33 128, 2018.
- [15] C. Ling, “On the proximity factors of lattice reduction-aided decoding,” *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2795–2808, 2011.
- [16] X. W. Chang, J. Wen, and X. Xie, “Effects of the LLL reduction on the success probability of the Babai point and on the complexity of sphere decoding,” *IEEE Trans. Inf. Theory*, vol. 59, no. 8, pp. 4915–4926, Aug 2013.