Abstract—Fifth generation wireless networks aim at meeting the requirements imposed by the ever increasing demand in capacity. One of the considered proposals is the extension of traditional multiple-input multiple-output (MIMO) systems to the use of a massive number (a few dozens to hundreds) of antennas. Even though promising, this technology brings some challenges that must first be surpassed. One of such problems is the increase in signal processing complexity, which might become unbearable when conventional techniques are used. Hence, one is to find methods that perform as close to optimal as possible, while yielding low complexity. In this work, a special emphasis on large MIMO suitable algorithms to perform detection and precoding is given. Particularly, for the case where the number of antennas at the base station is much larger than the number of served users, an effective linear processing method based on the Neumann series and on the matrix inversion lemma is proposed. Further, randomised algorithms to perform detection in symmetric massive MIMO systems are thoroughly studied. Namely, an optimised Gibbs sampling variant method is suggested and shown to outperform the existing ones in the literature, including state of the art lattice reduction methods. Finally, capitalising on the advantages provided by large-scale arrays, a massive MIMO enabled full-duplex relay setup is evaluated. This includes the design of transmit and receive linear filters that, using a link dependent optimal power allocation scheme, is able to attain high energy efficiencies, whilst satisfying individual throughput requirements. In all of the proposed algorithms, both hardware feasibility and suitability for parallel computation have been taken into consideration. The results herein presented include illustrative examples and insightful numerical simulations.

Index terms - massive MIMO, Neumann series, Gibbs sampling, in-band full-duplex, optimal power allocation.

I. INTRODUCTION

Over the past few decades we have been experiencing a rapid growth in mobile communications traffic. Not only has the number of mobile subscribers been increasing dramatically, but also each user is demanding more speed and reliability, anytime and anywhere. Hitherto network operators have been able to meet the requirements by successively deploying incremental additions to the existing infrastructure. However, this approach will not suffice to deliver the required capacity; thus, a fifth generation of wireless networks (5G) is expected to emerge between 2020 and 2030 [1].

Even though a sheer amount of viable candidate technologies exist, nothing is defined as of yet. Recent studies have been focusing on finding methods that, without compromising on the energy efficiency, are able to attain high spectral efficiencies. On top of that, computational complexity and hardware feasibility should also be taken into account in the design of such new platforms. In summary, the idea is to find methods that perform as close to optimal as possible, whilst yielding the lowest number of operations possible.

Multiple-input multiple-output (MIMO) systems with multiple antennas at both transmitter and receiver sides are already used to achieve higher capacity and link reliability by the exploitation of both spatial multiplexing and diversity. In order to deliver the required data rates, the use of antenna arrays with a very large number (tens to hundreds) of elements might be a reasonable alternative [2]. This technology, denoted as massive MIMO, will allow a much higher spectral efficiency with decreased energy levels, whilst serving a large number of users at the same time. However, with the increase of the number of dimensions, using conventional algorithms may not be suitable any more; hence, new approaches must be looked into.

Alongside the developments in large-dimensional MIMO, in-band full-duplex (IBFD) systems have also attracted a lot of research in the past few years due to their clear advantages. In particular, this feature provides the possibility of transmitting and receiving data at the same time instant and frequency band, which means that with the same amount of energy as in the half-duplex (HD) counterpart, it is possible to double the spectral efficiency [3]. Despite the good progress made so far in these systems, hardware constraints, and the difficulties in modelling the channel and cancelling loopback interference have been hindering the deployment of full-duplex enabled equipment.

In this work, we study and propose techniques suitable for large MIMO systems, where a special emphasis on detection and precoding methods is put. Firstly, the situation where the number of antennas at the base station (BS) is much larger than the number of served users is considered. Exploiting the channel hardening effect [4], we verify the suitability of the Neumann series to compute the required inverse in linear processing techniques, and capitalising on those results, we evaluate procedures based on the matrix inversion lemma to update the inverse when a user is added or removed from the system. Secondly, we consider the symmetric case where the number of users matches the number of antennas at the BS, and assess the performance of randomised algorithms based on Gibbs sampling [5]. In particular, the dependence of these Markov chain Monte Carlo (MCMC) methods with the "temperature" parameter is identified and an optimised approach is proposed.

Lastly, an IBFD relay system equipped with the massive MIMO technology is contemplated. Taking advantage of the increased number of degrees of freedom, we propose jointly designed linear filters to suppress interference and forward
information to an arbitrary number of communication pairs. Additionally, a low complexity linear optimal power allocation scheme based on the individual throughput requirements is detailed.

This paper is organised as follows: in section II a description of the used MIMO model is given; sections III and IV contain an overview of fast methods to invert matrices in the context of MIMO detection; in section VII a method to solve the problem of detection in symmetric large MIMO systems based on MCMC is proposed; sections VIII and IX study linear loopback interference mitigation schemes for full-duplex relays equipped with massive arrays. Finally, comparative numerical results are presented in sections V, VII and X whilst section XI concludes the paper.

II. SYSTEM MODEL

Consider a large scale multiuser MIMO system with $N$ antennas at the base station and $M \leq N$ single antenna users. Each user transmits a symbol from an $m$-QAM constellation set $B = \mathcal{A} + j\mathcal{A}$ with normalised energy equal to one. The resulting transmit vector is denoted by $x = [x_1, x_2, ..., x_M]^T$, has correlation matrix $\mathbb{E}\{xx^H\} = I$ and is filtered by a flat-fading narrow-band channel, represented by the channel matrix $\mathbf{H} \in \mathbb{C}^{N \times M}$. At the receiver, additive white Gaussian noise (AWGN), denoted by $n$, is added and the received vector $y \in \mathbb{C}^{N \times 1}$ can, hence, be expressed as:

$$y = \mathbf{H}x + n,$$  

where $n \sim \mathcal{CN}(0,1)$. Under this model, the signal-to-noise ratio (SNR) is defined by the variance of the entries in $\mathbf{H}$, that is $h_{i,j} \sim \mathcal{CN}(0,\text{SNR}/M)$. Perfect knowledge of $\mathbf{H}$ is assumed at the receiver.

III. LINEAR DETECTION AND NEUMANN SERIES

From the received signal, one is interested in performing hard decisions of $x$, denoted by $\hat{x}$. Note that channel coding techniques will not be considered in this paper; though, the results herein presented are straightforwardly extensible to coded transmissions. Further, the proposed methods in this section are also applicable in linear precoding.

A large variety of massive MIMO detection algorithms exist in the literature [5]; however, under the assumption that $M << N$, linear detection methods are proven to perform close to optimal [7]. Then, the matrix inversion-dependent conventional zero forcing (ZF) and minimum mean-square error (MMSE) filters will be considered for analysis purposes. To this end, define the Gram matrix $\mathbf{Z} = \mathbf{H}^H\mathbf{H} \in \mathbb{C}^{M \times M}$ and compute both ZF and MMSE estimations as:

$$\hat{x}_{ZF} = \mathcal{Q}(\mathbf{H}^H\mathbf{H}^{-1}\mathbf{H}^H y) = \mathcal{Q}(\mathbf{Z}^{-1}\mathbf{H}^H y);$$  

$$\hat{x}_{MMSE} = \mathcal{Q}(\mathbf{H}^H\mathbf{H} + \mathbf{I})^{-1}\mathbf{H}^H y),$$

where $\mathcal{Q}(\cdot)$ is the quantiser operator to the nearest point in the constellation. Note that most of the computational burden in (2) is in computing the inverses of the Hermitian matrix $\mathbf{Z}$. In particular, the computation of $\mathbf{Z}^{-1}$ using exact inversion methods, such as Cholesky decomposition [8], [9], requires $O(M^3)$ operations, which might be cumbersome to implement for the case where a large number of users are being served. Using the fact that in massive MIMO systems $\mathbf{Z}$ is an almost diagonal matrix, a hardware-efficient method based on the Neumann series was first proposed in [10] to approximate the required inverse in (2).

It has been proven in [11] that if one applies the decomposition of $\mathbf{Z}$ as $\mathbf{Z} = \mathbf{D} + \mathbf{E}$, where $\mathbf{D}$ is a diagonal matrix with the diagonal entries of $\mathbf{Z}$ and $\mathbf{E}$ the corresponding hollow, then the Neumann series to compute its inverse can be expressed as:

$$\mathbf{Z}^{-1} - 1 = \sum_{n=0}^{K-1} (-\mathbf{D}^{-1}\mathbf{E})^n\mathbf{D}^{-1},$$

where $K$ is the number of terms to be computed in the series and $\mathbf{Z}^{-1}$ is the $K$-term approximation of $\mathbf{Z}^{-1}$. Convergence of (4) is only guaranteed if the maximum modulus of eigenvalues of matrix $(\mathbf{I} - \mathbf{D}^{-1}\mathbf{E})$ is less than one and, if this condition is satisfied, then approximation approaches equality as $K \rightarrow \infty$ [11]. Moreover, the lower the eigenvalues, the faster the convergence; which holds true when the ratio $\beta = N/M$ is high [12].

Neumann series is a low complexity iterative method; hence it is hardware friendly, unlike conventional inversion methods [13]. As an example, consider the approximation when $K = 3$:

$$\hat{x}_{MMSE} = \sum_{n=0}^{2} (-\mathbf{D}^{-1}\mathbf{E})^n\mathbf{D}^{-1}

\mathbf{A}_0 \mathbf{A}_1 \mathbf{A}_2 \mathbf{D}^{-1} = \mathbf{D}^{-1} - (\mathbf{D}^{-1}\mathbf{E})\mathbf{D}^{-1} + (\mathbf{D}^{-1}\mathbf{E})(\mathbf{D}^{-1}\mathbf{ED}^{-1}),$$

where $\mathbf{A}_0, \mathbf{A}_1$ and $\mathbf{A}_2$ are the $n = 0, 1, 2$ terms in (4). The complexity involved in computing $\mathbf{A}_0$ is $M$ divisions, whilst calculating $\mathbf{A}_1$ and $\mathbf{A}_2$ yields $3M^2 - 3M$ and $16M^2 - 2M$ real-valued multiplications, respectively (these values exploit the existence of zeros in the diagonal and the fact that each of the partial results $\mathbf{A}_i$ has to be Hermitian). Even though the complexity of (5) scales with $O(M^3)$ for $K \geq 3$ (same as exact inverse), its recursive nature is of much practical interest.

IV. INVERSE UPDATE

Algorithms to efficiently update the inverse of a matrix after a small perturbation are proposed here. The reason behind this study is simple: when a user is added or removed from the system, one is interested in recomputing the new inverse $\mathbf{Z}^{-1}$ in the least time possible in order to maximise data throughput. Having this in mind, it is possible, using the matrix inversion lemma, to decrease the number of computations from $O(M^3)$ to $O(M^2)$, hence increasing speed. Additionally, the update of $\mathbf{Z}^{-1}$ after a new channel estimation from a single user is obtained (which corresponds to a rank-1 perturbation in $\mathbf{H}$ and a rank-2 perturbation in $\mathbf{Z}$) is evaluated. Without recomputing the entire inverse, this can be achieved using the Sherman-Morrison formula (a special case of the matrix inversion lemma).

A. Adding and Removing a User

Assume that at a given time instant, the inverse $\mathbf{Z}^{-1} = (\mathbf{H}^H\mathbf{H})^{-1}$ is already computed (via exact inversion or Neumann series) and that a user is added to the system with
Algorithm 1 Update $Z^{-1}$, when a user $h_p$ is added to $H$ at position $p$

**Input:** $Z^{-1}, H, h_p$

1. $t_1 \leftarrow H^H h_p$
2. $t_2 \leftarrow Z^{-1} t_3$
3. $c \leftarrow 1/(h_p^H h_p - t_1^H t_2)$
4. $t_3 \leftarrow c t_2$
5. $F_{11} \leftarrow Z^{-1} + c t_3 t_3^H$
6. $Z_{e}^{-1} \leftarrow \begin{bmatrix} F_{11}^{-1} & -t_3 \\ -t_3^H & c \end{bmatrix}$

Change last column and last row of $Z_{e}^{-1}$ to column $p$ and row $p$

**Output:** $Z_{e}^{-1}$

Estimated channel denoted by the column vector $h_n$. Define the new inflated matrix as $H = [H, h_n]$ (the user is added in the last column but in fact it could have been added in any position). Hence, the new extended Gram matrix denoted by $Z_e$ is given by:

$$Z_e = \begin{bmatrix} H^H & H^H h_n \\ h_n^H & h_n^H h_n \end{bmatrix} = \begin{bmatrix} H^H H & H^H h_n \\ h_n^H H & h_n^H h_n \end{bmatrix}.$$  \hspace{1cm} (6)

In order to find $Z_{e}^{-1}$, we use the general result provided by the inverse of a partitioned matrix \cite{[14]}, which is written as follows (see \cite{[15]} Appendix B):

$$\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}^{-1} = \begin{bmatrix} F_{11}^{-1} & -P_{12} F_{11}^{-1} \\ -P_{21} F_{11}^{-1} & F_{22} \end{bmatrix},$$  \hspace{1cm} (7)

where

$$F_{11} = P_{11} - P_{12} P_{22}^{-1} P_{21};$$  \hspace{1cm} (8)

$$F_{22} = P_{22} - P_{21} P_{11}^{-1} P_{12}. $$  \hspace{1cm} (9)

Using the result provided above, $Z_{e}^{-1}$ is expressed as \cite{[16]} Lemma 11.1:

$$Z_{e}^{-1} = \begin{bmatrix} F_{11}^{-1} & -c Z_{e}^{-1} H H^H h_n \\ -c h_n^H H^H h_n & c \end{bmatrix},$$  \hspace{1cm} (10)

where $c = 1/(h_n^H h_n - h_n^H H H^H h_n)$ and

$$F_{11}^{-1} = Z^{-1} + c Z^{-1} H H^H h_n h_n^H H H^H.$$  \hspace{1cm} (11)

Therefore, it is possible to update the inverse of a matrix $Z = H^H H$ when a column is added to $H$ at position $p$ without explicitly recomputing $Z_{e}^{-1}$ or $Z_{e,k}^{-1}$. The method is summarised in algorithm \cite{[1]} and, for comparison purposes, the corresponding number of required real-valued multiplications and divisions is shown on the right hand side (computation of both $H^H h_p$ and $h_p^H h_p$ were not considered since they are also required in the recomputation of $D$ and $E$ in the Neumann series). If the initial $Z^{-1}$ is the exact inverse, then $Z_{e}^{-1}$ is also exact. On the other hand, if an approximation $Z_{K}^{-1}$ is used, then a propagation of errors is to be expected. Intuitively, the greater the error, the greater the degradation in performance after the update.

Based on the above results, a similar approach can be conducted when a column from $H$ is removed. Decompose the original matrix as $H = [H_r, h_n]$, where $H_r$ is the deflated matrix from which we want to compute the inverse $Z_r^{-1} = (H_r^H H_r)^{-1}$ and $h_n$ the column corresponding to user $n$ and that is to be removed. The initial, already computed, inverse matrix $Z^{-1}$ is given by

$$Z^{-1} = \begin{bmatrix} H_r^H H_r & H_r^H h_n \\ h_n^H H_r & h_n^H h_n \end{bmatrix}^{-1} = \begin{bmatrix} F_{11}^{-1} & -c u \\ u^H H_r & u^H h_n \end{bmatrix},$$  \hspace{1cm} (12)

where $u = Z_r^{-1} H_r h_n$. Noting that $F_{11}^{-1} = Z_r^{-1} + c u u^H$ (from (11)), it is straightforward that

$$Z_r^{-1} = F_{11}^{-1} - c u u^H.$$  \hspace{1cm} (13)

The step-by-step set of operations to compute ([13]) is outlined in algorithm \cite{[2]} (MATLAB indexing notation is used).

B. Updating a User

Formerly, the situations where users are added or removed from the system were considered. Likewise, an update to the inverse when a new channel estimation for user $p$ is obtained can be studied. This operation corresponds to replacing column $p$ of $H$ with the new estimation $h_p$. After $H$ is updated with the new estimation $h_p$, expressed as $H_u$, the new product $Z_u = H_u^H H_u$ corresponds to altering column $p$ and row $p$ in the original $Z$. This rank-2 perturbation can, however, be decomposed into two rank-1 perturbations. Defining a rank-1 perturbation by $\mathbf{ab}^H$, one can use Sherman-Morrison formula to recompute the inverse after the update, as follows \cite{[7]}:

$$(Z + \mathbf{ab}^H)^{-1} = Z^{-1} - (Z^{-1} a)(b^H Z^{-1})^{-1} (1 + b^H Z^{-1} a).$$  \hspace{1cm} (14)

Taking into account (14) and without recomputing the inverse, calculating $Z_u$ from the previous $Z$ is feasible. For that purpose, define $z_{u,p} = H_u^H h_p$ and compute $a_1 = z_{u,p} - z_p$, where $z_p$ is the $p$th column of $Z$. Further, make $b_2 = a_1^H$, where $a_1^H$ is the same as $a_1$ but with the $p$th entry zeroed. Then, computing $Z_u^{-1}$ is straightforward:

$$Z_u^{-1} = (Z + a_1 e_p^T)^{-1} = Z_1^{-1} - \frac{(Z_1^{-1} a_1) (e_p^T Z_1^{-1})}{(1 + e_p^T Z_1^{-1} a_1)};$$  \hspace{1cm} (15)

$$Z_u^{-1} = (Z + e_p b_2^H)^{-1} = Z_1^{-1} - \frac{(Z_1^{-1} e_p) (b_2^H Z_1^{-1})}{(1 + b_2^H Z_1^{-1} e_p)},$$  \hspace{1cm} (16)

where $e_p$ is the $p$th column of the identity matrix $I$. Equations \cite{[15]} and \cite{[16]} require $24M^2$ scalar multiplications and $4$ divisions; thus, when $M$ is large, updating the inverse via this method might be advantageous over exact inversion in terms of complexity. Note that updating the inverse when only one column in $H$ is changed can also be regarded as removing and adding a column sequentially (algorithms \cite{[2]} and \cite{[1]}). This study will be conducted in the numerical results.

C. Complexity

The involved number of real-valued multiplications and divisions to compute $A_1$ and $A_2$ and perform the former studied updates is expressed in table \cite{[1]}. It is worth emphasising that all of the proposed update algorithms are quadratic in complexity. Moreover, if a recomputation of the term $A_2$ after a small rank perturbation was to be performed, its complexity would still be $O(M^3)$, since all of its entries would have to be recalculated.
Algorithm 2 Update $Z^{-1}$, when a user $h_p$ is removed from $H$ at position $p$

Input: $Z^{-1}$, $p$
Change column $p$ and row $p$ of $Z^{-1}$ to last column and last row
$F^{-1}_{11} \leftarrow Z^{-1}(1 : (M - 1), 1 : (M - 1))$
$c \leftarrow Z^{-1}(M, M)$
$t \leftarrow -Z^{-1}(1 : (M - 1), M)$
$Z^{-1}_{r} \leftarrow F^{-1}_{11} + ttH/c \triangleright 3M^2 + 3M$
Output: $Z^{-1}_{r}$

Table I: Complexity of Neumann series and update operations.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Real-valued multiplications and divisions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compute $A_1$</td>
<td>$3M^2 - 3M$</td>
</tr>
<tr>
<td>Compute $A_2$</td>
<td>$16M^3 - 2M$</td>
</tr>
<tr>
<td>Inflate (alg. 1)</td>
<td>$7M^2 + 9M + 1$</td>
</tr>
<tr>
<td>Deflate (alg. 2)</td>
<td>$3M^2 + 3M$</td>
</tr>
<tr>
<td>(15) and (16)</td>
<td>$24M^2 + 4$</td>
</tr>
</tbody>
</table>

V. Numerical Results - Part I

In this section, the impact in BER performance of the algorithms using the ZF estimates in (2) will be evaluated. The approximation $Z^{-1}_{3}$ will be used as the initial inverse matrix, since it provides a good trade-off between performance and complexity [11].

Denote the number of updates by $U$, corresponding to the number of added, removed or updated users after the last full inverse (in this case $Z^{-1}_{3}$) was computed. The output updated inverse matrix after $U$ sequential inflation or deflation operations is designated by $Z^{-1}_{n=U} \in \mathbb{C}^{(M+U) \times (M+U)}$. For instance, $U = 2$ means that algorithm 1 (or 2) was run twice and its input matrices were $Z^{-1}_{3}$ and $Z^{-1}_{n=1}$ for the first and second repetitions, respectively.

The initial number of antennas was set to $M = 8$ and $N = 80$, resulting in a ratio $\beta = 10$, which guarantees the convergence of (4) with very high probability [12]. Further, the constellation size was set to 64-QAM. For comparison purposes, performance using exact inverse and an entire recomputation of the Neumann series approximation before and after the update will also be exhibited. For the given setup ($K = 3$ and $\beta = 10$), a stalling effect in the BER (error floor) for high SNR using the Neumann series is to be expected due to the error associated with the approximation [11].

Figure 1 depicts the performance after 1 and 2 users are added to the system ($U = 1$ and $U = 2$). The result shows that the degradation in performance when a user is added via matrix inversion lemma is close to negligible, even outperforming the case where the entire recalculation of $Z^{-1}_{n=U}$ using the Neumann series is done. Additionally, only an insignificant increase in BER is noticed from $U = 1$ to $U = 2$. Therefore, the advantages of the studied method are two-fold: not only inflating via algorithm 1 achieves better performance than recomputing $Z^{-1}_{n=K}$, but also a lower number of multiplications are required.

The case where $U$ users are removed from the system is then depicted in figure 2. This time, BER performance using the Neumann series approximation is better if a recalculation from scratch is performed. This is related with the smaller error propagation associated with lower-dimensional matrices (higher ratio $\beta$). Nevertheless, the gain in complexity provided by algorithms 2 might be exploited in applications where BER requirements are not too strict. In addition, this degradation could have been mitigated if a better initial approximation ($K > 3$) of $Z^{-1}$ had been used.

Finally, figure 3 depicts the situation after $U$ different columns in $H$ are changed (dimensions of the inverse matrices after the update remain the same). As can be inferred, the sequential operations in (15) and (16) using an approximation as initial inverse led to a propagation of errors, resulting in increased BER values. However, it is interesting to note that performing a deflation followed by an inflation (running algorithm 2 and 1 sequentially) gets better results than computing a 3-term Neumann series from start. This is explained with the vanishing error contribution from “removed” columns in $H$, since algorithms 1 and 2 return the exact inverse if the initial inverse matrix is also exact. Ultimately, if $U = M$ updates were to be performed, then exact inverse would be attained regardless of the initial matrix (though, its cost would be similar to one of an exact inverse).

VI. Symmetric Large MIMO Detection

As was verified in previous sections, in an uplink MIMO channel, when the number of served users $M$ is far less than the number of BS antennas $N$ ($M << N$), linear processing techniques are proven to perform close to optimal. In fact, linear detectors’ theoretical diversity is given by $N - M + 1$ ([18]), which justifies their good performance when the ratio $N/M$ is high. However, when $M \rightarrow N$, their attained diversity approaches one, which is rather undesirable. As a consequence, other detection techniques must be looked at. Despite being optimal, maximum likelihood (ML) and sphere decoding detection (SD) methods are prohibitive in large dimensions due to their exponential complexity. Two other possible alternatives are lattice-reduction methods ([19]) and randomised algorithms, whose performance-complexity trade-offs have been proven to be acceptable.

Advanced statistical methods, such as MCMC, may lead to significant gains in signal processing for wireless communications. In particular, GS based methods can attain ML-like performance, provided the algorithm is run for a sufficient long time [5]. However, the conventional GS technique suffers from the stalling problem, meaning that for high values of SNR, the bit error rate (BER) performance does not improve any further [20].

In this work, capitalising on the results in [5] and [21], we propose a variant of the GS method to perform detection in large MIMO systems. At each iteration, the alternative approach picks one of three possible ”temperature” parameters (TP) according to a given probability distribution. The values are chosen such that the stalling problem is circumvented, whilst keeping convergence rates high.
A. Triple Mixed Gibbs Sampling

For implementation purposes, the complex-valued system model is converted into a real-valued one (as in Eq. (5.3)), therefore, designating \( n = 2M = 2N \), the channel matrix \( \mathbf{H} \in \mathbb{R}^{n \times n} \), while both vectors \( \mathbf{x}, \mathbf{n} \in \mathbb{R}^{n \times 1} \).

Upon receiving \( \mathbf{y} \), the BS starts the detection procedure. Under the assumption that the various transmitted symbols are uncorrelated, retrieving \( \mathbf{x} \) from (1) can be regarded as a closest vector problem (CVP) in a lattice with Gaussian distribution (23). At the initial time instant \( t = 0 \), an \( n \)-dimensional starting candidate \( \mathbf{x}^{(0)} \) is given to the MCMC detector, which then performs an “ingenious” walk over the real alphabet \( \mathcal{A}^n \).

In the GS method herein used, each iteration \( t \) consists of transitioning each coordinate \( j = 1, \ldots, n \) to a possible element \( \omega \in \mathcal{A} \) sequentially, and based on the following sampling rule (5):

\[
p(x^{(t+1)}_j | x^{(t)}_j, j, y, \mathbf{H}) = \frac{\exp \left(-\frac{1}{2 \alpha^2} ||y - \mathbf{H} x^{(t)}_{j | \omega}||^2\right)}{\sum_{x^{(t)}_{j | \omega} \in \mathcal{A}} \exp \left(-\frac{1}{2 \alpha^2} ||y - \mathbf{H} x^{(t)}_{j | \omega}||^2\right)},
\]

where \( x^{(t+1)}_{j | \omega} = [x^{(t+1)}_{1:j-1}, \omega, x^{(t)}_{j+1:n}]^T \) and \( \alpha \) is a positive TP. At the end of each iteration, the ML cost \( f(x^{(t+1)}) = ||y - \mathbf{H} x^{(t+1)}|| \) is computed and the resulting vector is fed into the next iteration. After \( t_{\text{max}} \) iterations, the algorithm is stopped and the vector yielding the lowest ML cost is the output solution. For further implementation details and complexity reduction techniques of GS see, e.g., (5).

Performance of GS based methods are highly dependent on the chosen TP. If \( \alpha^2 \) is set to equal the noise variance (that is \( \alpha^2 = 1 \)), as in the conventional GS, a fast convergence to a minimum is verified but an error floor in the high SNR regime is to be expected. This is related with the existence of several local minima where the algorithm may be stuck for a very long time, a situation that becomes increasingly preponderant when either the \( m \)-arity or \( n \) are large. In order to overcome this problem, the authors in (21) proposed a mixed GS (MGS) where at each iteration there is a small chance that a random walk is performed (which coincides with setting \( \alpha^2 = \infty \) in (17)). However, this method proves to mix slowly to the steady state, hence requiring a considerable amount of restarts and iterations in order to perform reliable detection. On the other hand, (5) demonstrated the existence of an optimal TP \( \alpha^2 \), proportional to SNR, and which maximises the probability of finding the desired solution, a method that will henceforth be denoted as optimised GS (OGS). Notwithstanding the breakthrough developments, the results were only obtained for a \( \{-1,1\} \) constellation set, which is rather limited.

Following the aforementioned results, we propose an algorithm that at each iteration picks a different value for the TP from the set \( \{\infty, \text{SNR}/\log(n), 1\} \) with probabilities \( q_{\infty}, (1 - q_{\infty})q_{\text{SNR}}, \) and \( (1 - q_{\infty})q_1 \), respectively. In this manner, at each iteration there is a chance \( q_{\infty} \) that a random walk is performed; otherwise, the algorithm chooses \( \alpha^2 = 1 \) or \( \alpha^2 = \text{SNR}/\log(n) \) with probabilities \( q_1 \) or \( q_{\text{SNR}} \) and samples accordingly. For each sampling process, choosing \( \alpha^2 = 1 \)}, the algorithm chooses \( \alpha^2 = 1 \) or \( \alpha^2 = \text{SNR}/\log(n) \) with probabilities \( q_1 \) or \( q_{\text{SNR}} \) and samples accordingly. For each sampling process, choosing \( \alpha^2 = 1 \)
Algorithm 3 Triple Mixed Gibbs Sampling

Input: \( y, g\mathbf{H}, q_1, q_\infty, \mathbf{x}^{(0)} \)

Define \( c_0 \) as the cost of best candidate so far, set \( t = 0 \) and compute:

while \( t < t_{\text{max}} \)

for \( i = 1 \) to \( n \) do

\begin{align*}
\text{if } r_1 < q_\infty & \text{ then } \quad \text{generate } r_1, r_2 \sim U[0, 1] \quad \text{generate pmf } p(x_{i}^{(t+1)} = \omega) \sim U[0, 1], \forall \omega \in \mathcal{A} \quad \text{sample } x_{i}^{(t+1)} \text{ from this pmf} \\
\text{else} & \quad \text{if } r_2 < q_1 \text{ then } \quad \alpha^2 = 1 \\
& \quad \text{else} \quad x_{i}^{(t+1)} \propto \text{SNR} \quad x_{i}^{(t+1)} \sim p(x_{1}^{(t)}, ..., x_{i-1}^{(t)}, x_{i+1}^{(t)}, ..., x_{n}^{(t)}) \\
& \quad \text{if } c \leq c_0 \text{ then } \quad \hat{x} = x^{(t+1)}; \quad c_b = c \\
& \quad t = t + 1
\end{align*}

Output: \( \hat{x} \)

ensures fast convergence to a local solution, \( \alpha^2 \propto \text{SNR} \) guarantees a quick mix to steady state and \( \alpha^2 = \infty \) is necessary to avoid “deep” local minima. Whilst yielding the same complexity, we find that with a suitable choice for \( q_\infty \) and \( q_1 \) (\( q_\text{SNR} \) is redundant), it is possible to effectively mitigate the stalling effect and achieve near-ML performance. The pseudocode related with the proposed variant, denoted as triple MGS (T-MGS), is presented in Algorithm 3 where \( U[0, 1] \) denotes the uniform distribution between 0 and 1, and pmf stands for probability mass function.

B. Mixing Ratio Choice

Finding the optimal choice for the mixing ratios \( q \) is a non-trivial task, since, as it will be seen in the numerical results, their impact in performance is very dependent on the SNR. Nevertheless, one is to find parameters that on average perform well for any SNR. The authors in [21] proposed the MGS method with \( q_\infty = 1/n, q_1 = 1, \) whereas [5] suggested the OGS with \( q_\infty = 0, q_\text{SNR} = 1. \) With the T-MGS algorithm, one expects to experience fewer stalling occurrences than in MGS, as long as \( q_\text{SNR} > 0. \) This fact was verified through intensive simulations and a choice of \( q_\infty = 1/(10n), q_1 = q_\text{SNR} = 1/2 \) proved to attain very satisfactory results. Note that with this choice of parameters, one random walk is performed each 10 iterations on average and whenever it is not carried out, there is a similar chance that \( \alpha^2 = 1 \) or \( \alpha^2 \propto \text{SNR} \) are chosen.

VII. NUMERICAL RESULTS - PART 2

The results provided here consider an \( N \times M = 16 \times 16 \) large MIMO system and a 16-QAM constellation. In order to effectively verify the superiority of T-MGS over MGS and OGS, a comparison of the average BER performance against the number of iterations is first provided in figure 4. Two distinct values for the SNR were chosen (15 and 20 dB), \( t_{\text{max}} \) was set to 1000 iterations and, for comparison purposes, SD performance (ML-like) is also depicted. Additionally, the conventional GS method (\( \alpha^2 = 1 \)) is shown and the “optimal” TP parameter in the OGS method is computed according to [5, Eq. (23)]. In all instances, the MMSE filter output was chosen as the initial candidate vector. As can be inferred, for low values of SNR, all of the methods (apart from MGS) attain optimal performance after 500 iterations. However, when SNR increases to 20 dB, it is verified that different mixing ratios indeed exhibit different behaviours. Firstly, it is confirmed that conventional GS has a rapid decay in BER in the first few iterations but then becomes stalled in a local minima. Secondly, it is observed that the proposed method T-MGS outperforms both MGS and OGS by a non-negligible margin, thus showing the importance of choosing adequate mixing ratios. Nonetheless, the performance is still far from optimal, which proves the usefulness of executing multiple restarts (MR) as suggested in [21].

For the assessment of BER as a function of the SNR, and taking into account the previous result, \( R = 10 \) restarts are performed, fulfilling \( t_{\text{max}} = 600 \) iterations each, and the output coincides with the candidate vector yielding the lowest ML cost. For the first restart, the MMSE solution is used as the initial vector, and for the remaining ones a randomly chosen candidate is picked from the lattice instead. In addition to the SD, a state-of-the-art element-based lattice reduction (ELR) method was implemented (D-ELR-SLB-SV-SIC-MMSE as in [19]). Performance is depicted in figure 5. The first thing to be noticed is the existence of a stalling effect in OGS, which proves that [5, Eq. (23)] does not necessary scale for larger constellations than \( \{-1, 1\} \). Further, it can be verified that, with the inclusion of the MR approach, T-MGS performs very closely to the SD, whilst achieving near-optimal diversity for the whole studied SNR range. Also, the difference in performance between the proposed method and the ELR is around 6 dB, which is significant. It is worth emphasising that the MR instances are parallel-architecture friendly, therefore such hardware design model, increasingly available at the BS, can be exploited to increase speed.

VIII. IN-BAND FULL-DUPLEX

Thus far, we have seen that high performance, low complexity detection and precoding in large MIMO systems is indeed possible. Taking that into account, we shall now evaluate the situation where both massive MIMO and IBFD technologies are available.

A. IBFD System Model

Let us consider \( M \) user pairs establish a wireless connection through a relay station, which operates in IBFD. For this purpose, each user is equipped with a single antenna, while the relay has \( N_{rx} \) receiving antennas and \( N_{tx} \) transmitting antennas. The number of antennas at the relay may go up to a few hundred and the number of served users is in general considered to be much lower (\( M << N_{rx}, N_{tx} \)). The nonexistence of a direct link between the sources and destinations is assumed, hence all links are established via the
Figure 4: BER performance as a function of the number of iterations $t$. Two distinct values for the SNR are considered.

Figure 5: BER performance as a function of the SNR of the proposed variant, including a comparison with other methods.

B. Detection and Precoding

Let us consider for the moment an equivalent received signal $\tilde{r}$, whose LI component has been minimised. Hence, we are in the presence of an uplink channel where $M << N_{rx}$. Under this assumption, linear processing techniques are proven to perform close to optimal [25], therefore we consider the usage of ZF filtering for both detection and beamforming [26]. Using the estimation of $G_{SR}$, denoted by $\hat{G}_{SR} = H_{SR}D_{SR}^{1/2}$, the estimated symbols after a ZF filter $W_{zf}$ are given by

$$\hat{x} = Q(W_{zf}\hat{F}) = Q((\hat{G}_{SR}^H\hat{G}_{SR})^{-1}\hat{G}_{SR}^HH_{SR}F).$$  \hfill (20)

Upon detection based on $\hat{x}$, the estimated symbols are forwarded to the destinations $y_d$, which we assume to have very limited processing capabilities. Thus, a ZF precoder $A_{zf}$ is used. The corresponding expression is

$$\hat{t} = A_{zf}\hat{x} = \alpha_{zf}\hat{G}_{RD}^H(\hat{G}_{RD}^H\hat{G}_{RD})^{-1}\hat{x},$$  \hfill (21)

where $\hat{G}_{RD} = D_{RD}^{1/2}H_{RD}$ is the estimation of the true $G_{RD}$ and $\alpha_{zf} = (E(tr((\hat{G}_{RD}^H\hat{G}_{RD})^{-1})))^{-\frac{1}{2}}$ is a scalar chosen to

whereas the symbols from the relay to the destinations pass through $G_{RD} \in \mathbb{C}^{M \times N_{dz}}$. Both matrices are expressed as $G_{SR} = H_{SR}D_{SR}^{1/2}$ and $G_{RD} = D_{RD}^{1/2}H_{RD}$, where $D_{SR}$ and $D_{RD}$ are diagonal matrices with entries $\beta_{SR,k}$ and $\beta_{RD,k}$ taken from a log-normal distribution. The fast-fading channel components are present in $H_{SR}$ and $H_{RD}$, both taken from a $CN(0,1)$ distribution. The LI channel is represented by $H_{LI}$ with distribution $CN(0,\sigma_{LI}^2)$, where $\sigma_{LI}^2$ accounts for the residual LI power. The vectors $n_r \sim CN(0,\sigma_{n_r}^2)$ and $n_d \sim CN(0,\sigma_{n_d}^2)$ account for additive white Gaussian noise at the relay and destinations, respectively. Finally, the diagonal matrix $D_{ps}$ with entries $p_{s,k}$ regulates each source transmit power, while $p_r$ denotes the relay’s average transmit power.

All of the aforementioned channel matrices $H$ are assumed to be known at the receiver apart from an error, denoted by $E \sim CN(0,\sigma_E^2)$; that is $H = H^* + E$, where $H$ corresponds to the estimated matrix. The impact of any hardware imperfection at the relay is modelled by means of an additive error [24] in the transmitted vector given by $t = \hat{t} + \epsilon_t$, where $\hat{t}$ is the vector to be transmitted after baseband filtering and where all the elements of $E_t \sim CN(0,\sigma_E^2)$ are assumed to be uncorrelated with $\hat{t}$.
normalise the power of  \( \hat{t} \), i.e.,  \( \mathbb{E}(t^H \hat{t}) = 1 \). From the definition of  \( \mathbf{H}_{RD} \sim \mathcal{CN}(0, 1 + \epsilon_r^2 \mathbf{I}) \),  \( \alpha_{ef} \) comes as  \( \left[27 \right] \)

\[
\alpha_{ef} = \sqrt{\frac{(\mathbf{N}_r - \mathbf{M})}{\sum_{k=1}^{M} (\beta_{RD,k}(1 + \epsilon_r^2 \mathbf{I}))^{-1}}}. \tag{22}
\]

IX. LOOPBACK INTERFERENCE MITIGATION

A. Linear Filtering

We now evaluate the problem of suppressing LI, using linear processing. Under the model given by  \( \left[18 \right] \), we are interested in minimising the LI term  \( \mathbf{\sqrt{p}_R \mathbf{H}_{lt} \hat{t}} \), while preserving the signal  \( \mathbf{G}_{SR} \mathbf{D}_{p_s}^{1/2} \mathbf{x} \) and taking into account the covariance of the noise vector  \( \mathbf{R}_{n_r} \) at the relay. Hence, the mean square error between the desired signal and the one received by the relay station is  \( \mathbb{E}((\mathbf{G}_{SR} \mathbf{D}_{p_s}^{1/2} \mathbf{x} - \hat{r})(\mathbf{G}_{SR} \mathbf{D}_{p_s}^{1/2} \mathbf{x} - \hat{r})^H) \). Using the method proposed in  \( \left[24 \right] \), we consider a linear prefilter  \( \mathbf{F}_t \) and postfilter  \( \mathbf{F}_{rx} \), such that  \( \hat{t} = \mathbf{F}_t \mathbf{t} \) and  \( \hat{r} = \mathbf{F}_{rx} r \). The filter that finds the MMSE when  \( \mathbf{F}_{rx} \) is fixed satisfies  \( \left[27 \right] \)

\[
\mathbf{F}_{rx} = \mathbf{G}_{SR} \mathbf{D}_{p_s} \mathbf{G}_{SR}^H + p_R \mathbf{H}_t \mathbf{R}_t \mathbf{H}_t^H + \mathbf{R}_{n_r} \mathbf{I}^{-1}, \tag{23}
\]

where  \( \mathbf{R}_t = \mathbf{F}_t \mathbf{A}_x \mathbf{R}_x \mathbf{A}_x^H + \epsilon_t^2 \mathbf{I} \). Equation  \( (23) \) is a closed-form expression containing the channel estimates and covariance matrices of transmitted vectors and noise, that are assumed to be known. Under the considered model and for a given set of parameters, a trade-off in the end-to-end (e2e) BER is to be expected: on the one hand, a lower transmitted power at the relay  \( p_t \) reduces the LI power and diminishes any impact of the estimation errors and radio-frequency impairments; on the other hand, a higher  \( p_t \) leads to improved signal-to-noise ratios (SNR) levels at the destinations and hence a lower BER in the forward link channel. Thus, for a fixed source transmit power, an optimal choice for  \( p_R \) that minimises the e2e BER can be anticipated.

B. Optimal Power Allocation

The level of interference suffered by the relay station depends critically on its transmission power and directly perturbs the e2e performance. Thus, finding the optimal power that meets the requirements of the system is desired. Let us consider the achievable rate for each individual link. The procedure done in  \( \left[28 \right] \) allows us to obtain an expression for the transmission link rate between each source and destination

\[
R_k = \min\{R_{SR,k}, R_{RD,k}\}, \tag{24}
\]

where  \( R_{SR,k} \) and  \( R_{RD,k} \) denote the achievable rates between the sources and the relay and between the relay and the destinations, respectively. To obtain  \( R_k \), we start by considering the received signal at the relay before detection given by

\[
y_{k} = \sqrt{p_{S,k}} (\mathbf{W}_d \mathbf{F}_{rx}^T)_{k} g_{SR,k} x_k + \sum_{j \neq k} \sqrt{p_{S,j}} (\mathbf{W}_d \mathbf{F}_{rx}^T)_{j} g_{SR,j} x_j + \sqrt{p_R} (\mathbf{W}_d \mathbf{F}_{rx}^T)_{k} \mathbf{H}_t t + (\mathbf{W}_d \mathbf{F}_{rx}^T)_{k} \mathbf{n}_r, \tag{25}\]

where  \( g_{SR,k} \) denotes the  \( k \text{th} \) column of  \( \mathbf{G}_{SR} \). The received signal at each destination link before detection is given by

\[
y_{d,k} = \sqrt{p_{R} g_{RD,k}^T (\mathbf{F}_t \mathbf{A}_d)} x_k + \sqrt{p_R} \sum_{j \neq k} g_{RD,k}^T (\mathbf{F}_t \mathbf{A}_d) x_j + n_{d,k}. \tag{26}\]

Both expressions may be seen as a known mean gain times the desired signal plus an uncorrelated effective noise term that includes channel impairment effects, interpair and loopback interference, and Gaussian noise. A commonly used technique in large MIMO systems  \( \left[29 \right] \) is to approximate the effective noise term by a Gaussian noise term under the central limit theorem, and which gives a good approximation for  \( R_{SR,k} \) as in  \( \left[28 \right] \) and for  \( R_{RD,k} \) as in  \( \left[29 \right] \).

The goal is to find the system power allocation, i.e., compute the required power transmitted by both sources and relay, such that the desired rate for each communication pair  \( k \) is guaranteed. Moreover, peak power constraints should be satisfied and overall power consumption taken into consideration; in other words, the system energy efficiency, defined as  \( EE = \sum_{k=1}^{M} R_k / (p_R + \sum_{k=1}^{M} p_{S,k}) \), should be maximised. This can be described as the following optimisation problem

\[
\begin{align*}
\min_{p_{S,0}, \ldots, p_{S,M}, p_R} & \sum_{k=1}^{M} R_k + p_R, \\
\text{s.t.} & R_k \geq R_{0,k}, \quad k = 1, \ldots, M; \\
& 0 \leq p_{S,k} \leq p_{S,0}, \quad k = 1, \ldots, M; \\
& 0 \leq p_R \leq p_{R_0},
\end{align*} \tag{27}
\]

where  \( R_{0,k} \) and  \( p_{S,0} \) are the required rate and peak power for pair  \( k \) respectively, and  \( p_{R_0} \) is the relay station peak power. The solution to problem  \( (27) \) involves the computation of the channel statistics which are the output of involved expressions due to the non-linear dependence of the optimisation variables in  \( \mathbf{F}_{rx} \). For that reason, we propose a simple and computationally efficient iterative algorithm, based on linear programming, capable of obtaining the referred OPA. The algorithm computes the channel statistics coefficients in  \( \left[28 \right] \) and  \( \left[29 \right] \), the loopback mitigation filters and the power vectors iteratively until the optimal power vector is reached. The procedure is summarised in algorithm  \( \left[4 \right] \).

X. NUMERICAL RESULTS - PART 3

In this section, we compare the performance of the proposed filters with special emphasis on the e2e link reliability and on the efficiency of the power allocation scheme.

We assume a symmetric system with the same number of users  \( M \) on both sides with one antenna each and  \( N = N_{tx} = N_{rx} \) antennas at the relay. Transmission and reception at the relay are done in the same time-slot and frequency band, and an arbitrary processing delay  \( d \geq 1 \) is assumed such that at time instant  \( i \),  \( x[i] = \hat{f}(x[i-d]) \). We consider  \( \mathbf{F}_{rx} \) as in  \( \left[23 \right] \) and  \( \mathbf{F}_t = \mathbf{I} \). Without loss of generality, we also consider  \( \sigma_{fL}^2 = 1 \). The SNR at the relay is defined as  \( \mathbb{SNR}_R = \frac{\sum_{k=1}^{M} p_{S,k} p_{R} + p_R}{\sigma_n^2} \).

A. Transmitted Power vs. BER

We start by evaluating the performance of the system in terms of BER at both the relay and destinations, considering...
\[ R_{SR,k} = \log_2 \left( 1 + \frac{p_{SR,k}M_{VS,k}}{p_{SR,k} + \sum_j^K p_{SR,j}M_{PR,(k,j)} + p_{R}L_{SR,k} + A_{NSR,k}} \right), \] 

where \( M_{VS,k} = |\mathbb{E}\{ (W_{SR}F_{SR})_k^T g_{SR,k}^2 \}|^2, \) \( V_{SR,k} = \text{Var} \{ (W_{SR}F_{SR})_k^T g_{SR,k}^2 \}, \) \( M_{PR,(k,j)} = \mathbb{E}\{ |(W_{SR}F_{SR})_k^T g_{SR,j}^2| \}, \) \( L_{SR,k} = \mathbb{E}\{ ||(W_{SR}F_{SR})_k^T||^2 \} \) and \( A_{NSR,k} = \sigma_{n,k}^2 \mathbb{E}\{ ||(W_{SR}F_{SR})_k^2|| \}. \)

\[ R_{RD,k} = \log_2 \left( 1 + \frac{p_{RD,k}M_{VRD,k}}{p_{RD,k} + p_{PR}M_{PRD,k} + A_{NRD,k}} \right), \]

where \( M_{VRD,k} = |\mathbb{E}\{ g_{VRD,k}^T (F_{SR}A_{SR})_k^2 \}|^2, \) \( V_{RD,k} = \text{Var} \{ g_{VRD,k}^T (F_{SR}A_{SR})_k^2 \}, \) \( M_{PRD,k} = \sum_{j \neq k} K p_{PR}M_{PRD,k} + A_{NRD,k} \) and \( A_{NRD,k} = \sigma_{n,k}^2. \)

Algorithm 4 - Input: Peak powers \( p_{S0,k} \) and \( p_{R0} \)
1. Initialisation: Set \( i = 1; \) initialise powers \( p_{SR,k,i} = p_{S0,k} \) and \( p_{PR,i} = p_{R0}; \) define \( L \) as the total number of iterations and set \( N_i \) as the number of channel realisations per iteration.
2. Iteration i:
   1) Compute channel statistics:
      for \( n = 1 \) to \( N_i \)
      i) Generate \( G_{SR}, G_{RD}, H_{LL} \) and compute \( W_{SR} \) and \( A_{SR} \)
      ii) Compute filter \( F_{SR} \) with \( p_{SR,k,i} \) and \( p_{PR,i} \)
      iii) Compute instantaneous rate coefficients for all \( k \) pairs (as in (28) and (29))
   2) Average the coefficients over the realisations to obtain channel statistics.
   3) Solve linear program (27) with the coefficients found in step 2) to obtain the new \( p_{SR,k,i} \) and \( p_{PR,i}. \)
   4) Set \( p_{SR,k,i+1} = p_{SR,k,i} \) and \( p_{PR,i+1} = p_{PR,i}. \)
3. Check:
   if \( i = L \) then Output: \( p_{SR,L} \) and \( p_{PR,L} \)
   else set \( i \leftarrow i + 1. \)

only small-scale fading, that is \( D_{SR}^{1/2} = D_{RD}^{1/2} = I. \) Firstly, for a given SNR and fixed uniform transmitted power \( p_{SR,k} = 1 \) for all \( k, \) we are interested in studying the allocated powers at the relay \( p_{PR} \) and for an increasing number of antennas \( N. \) For comparison purposes, we implemented natural isolation (NI) (when \( F_{SR} = F_{SR} = I, \) which corresponds to ignoring the LI component) and the HD counterpart (when \( t = 0 \) and \( F_{SR} = F_{SR} = I. \) Setting the variance in the errors to be \( \epsilon^2 = \epsilon^2_H = 10^{-3}, \) curves of e2e BER for different numbers of antennas and relay power \( p_{PR} \) are depicted in figure 7. One can confirm that for a given configuration there is an optimal choice for the power at the relay that minimises the e2e BER. Moreover, it is concluded that a larger number of antennas attains the minimum BER with less power.

B. Optimal Power Allocation Algorithm

Finally, we evaluate the results of the proposed algorithm in section IX-B by finding the OPA that meets the rate constraints of each link, while minimising the overall power of the system. Large-scale fading effects are now taken into consideration, more precisely, \( \beta_{SR,k} \) and \( \beta_{RD,k} \) are independent variables generated from a log-normal distribution with mean value

\[ m = 1 \text{ and standard deviation } \sigma = 6 \text{ dB. Additionally, we set the normalised peak power of the sources and relay as } p_{S0,k} = 3 \text{ dB and } p_{R0} = 10 \text{ dB, respectively. In addition, parameters } N_i = 10^3 \text{ and } L = 5 \text{ are empirically defined, such that both channel statistics and output powers in algorithm 4 are good approximations of their real value. The performance of algorithm 4 is determined in terms of EE for the case where interference effects are disregarded (OPA-NI) and when the MMSE filter is used (OPA-MMSE). This method is also compared with the situation where an optimal uniform power allocation (OUPA) is used, which corresponds to considering } p_{SR,k} = p_{SR}, \text{ for all } k, \text{ in (27). Figure 8 shows the curves of average EE for different values of desired e2e sum-rate, defined as } \sum_{k=1}^M R_{0,k}, \text{ where the individual required rates } R_{0,k} \text{ are taken from a discrete uniform distribution. As one may see, for the same desired sum-rate, the EE of OPA-MMSE is improved significantly when compared to OUPA, while guaranteeing that no link is in outage } (R_k \geq R_{0,k}, \forall k). \text{ Furthermore, since MMSE filter effectively reduces the LI, a lower amount of energy is spent to achieve the same sum-rate when compared with OPA-NI, an effect that becomes more preponderant in higher rates regimes.}

XI. Conclusions

On the whole, it was demonstrated in this paper that the problem arising in massive MIMO detection can be efficiently surpassed. Firstly, the applicability of both Neumann
Figure 8: Energy efficiency for different power allocation schemes, filters and number of antennas. The remaining parameters are $\sigma^2_{n_k} = 1$, $M = 10$ pairs and $\text{SNR}_R = 16 \text{ dB}$.

series and matrix inversion lemma in massive MIMO systems were exploited to construct high-diversity, low-complexity detectors. Secondly, the effectiveness of randomised algorithms based on Gibbs sampling using distinct values for the “temperature” parameter was shown. Lastly, linear filters to jointly perform detection, precoding and loopback interference mitigation in full-duplex systems were designed, and the corresponding impact in performance was assessed. All of the herein provided algorithms are suitable for hardware implementation and, hence, satisfy the high throughputs requirements of future wireless networks.

REFERENCES


