

# Superposition Coding with Peak-Power Limitation<sup>1</sup>

Jun Tong, Li Ping

Department of Electronic Engineering,  
City University of Hong Kong, Hong Kong,  
email: jun.tong@student.cityu.edu.hk,  
eeliping@cityu.edu.hk

Xiao Ma

Department of Electronics and Communication,  
Sun Yat-sen University, Guangzhou 510275, China,  
email: maxiao@mail.sysu.edu.cn

**Abstract**—This paper presents a peak-power-limited superposition coding scheme based on clipping. A low-complexity soft compensation algorithm (SCA) for combating the clipping effect is investigated. It can be easily combined with soft-input soft-output (SISO) decoding algorithms in an iterative manner. Various numerical results show that the SCA can effectively mitigate the performance loss due to clipping.

**Keywords**—clipping; iterative decoding; superposition coding

## I. INTRODUCTION

Recently, superposition coding [1]-[4] has been studied as an alternative scheme for high throughput transmission [5][6]. One interesting feature of this scheme is that, the transmitted signal exhibits an approximately Gaussian distribution since it is a linear superposition of several independent codewords (each referred to as a layer). This provides a more straightforward approach for achieving the so-called shaping gain [7]-[10], as demonstrated in [3][4] by mutual information analysis and simulation examples. Compared to the shaped schemes in [9][10], where special shaping codes and shaping algorithms are required, superposition coding is conceptually simpler and has lower encoding complexity. Another feature of superposition coding is that it can be treated as a perfectly cooperating multiple-access system by viewing one layer as one user. Hence the low-cost iterative decoding techniques developed in [11] can be employed.

Superposition coding has been used in numerous contexts. One example is capacity analysis, e.g., in the achievability proof of broadcasting channel capacity [1][2]. Another is adaptive modulation, achieved by adjusting the number of layers (and so rate) according to channel conditions [12]. This is more flexible than the traditional approaches, such as switching between, say, trellis coded modulation (TCM) using 8-PSK (8-ary phase shift keying), 16-QAM (16-ary quadrature amplitude modulation), 32-QAM etc, for channel adaptation. The latter has the drawbacks of abrupt rate change and high receiver cost due to the need for many different TCM decoders. With superposition coding, rate changes can be made smoothly by using a low-rate code for each layer. The receiver cost can be kept low by using the same code for all of the layers and time-sharing a common decoder.

However, the approximately Gaussian distribution achieved with superposition coding (and other shaped schemes

[7]-[10]) also implies a high peak-to-average power ratio (PAPR), which can be a serious concern for power amplification [7]. Clipping is a simple and efficient method for tackling this problem, but it may lead to substantial performance degradation [13]-[15] since it introduces non-linear distortion to the transmitted signal.

In this paper, we consider the clipping issue for superposition coding systems. A low-cost soft compensation technique to alleviate the clipping effect is presented. It can be easily combined with iterative decoding algorithms and has the potential to greatly enhance performance.

## II. SYSTEM MODEL

### A. Encoding

Fig. 1 shows the encoder structure for a superposition coding scheme with  $K$  layers. The high-speed binary data sequence  $\mathbf{u}$  is partitioned into  $K$  equal-length subsequences  $\{\mathbf{u}^{(k)}\}$ . The  $k$ th subsequence  $\mathbf{u}^{(k)}$  is encoded by a binary encoder (ENC- $k$ ) at the  $k$ th layer, resulting in a coded bit sequence  $\mathbf{c}^{(k)} = \{c_j^{(k)}\}$  with  $c_j^{(k)} \in \{0, 1\}$ . The randomly interleaved version  $\mathbf{v}^{(k)}$  of  $\mathbf{c}^{(k)}$ , from interleaver- $k$  (INTL- $k$ ), is then mapped to a quadrature phase shifting keying (QPSK) sequence  $x_j^{(k)} = x_{\text{Re},j}^{(k)} + ix_{\text{Im},j}^{(k)}$  according to  $x_{\text{Re},j}^{(k)} = 1 - 2v_{2j}^{(k)}$  and  $x_{\text{Im},j}^{(k)} = 1 - 2v_{2j+1}^{(k)}$ , where  $i = \sqrt{-1}$  and  $j$  is the time index. The subscripts  $\text{Re}$  and  $\text{Im}$  are used to denote the real and imaginary parts of complex numbers, respectively.

The independent QPSK sequences are linearly superimposed to form the output signal:

$$x_j \equiv \sum_{k=0}^{K-1} \rho_k e^{i\theta_k} x_j^{(k)} \quad (1)$$

where  $\rho_k$  ( $\rho_k > 0$ ) is an amplitude factor and  $\theta_k$  ( $0 \leq \theta_k < \pi/2$ ) a rotation angle for layer- $k$ . The overall rate is  $R = 2 \sum_{k=0}^{K-1} R_k$  in bits/symbol, where  $R_k$  is the rate of the  $k$ th binary component code.

This scheme is also referred to as a multilevel coding/sigma-mapping scheme in [3][4]. A notable feature of this scheme is the use of different amplitude factors  $\{\rho_k\}$  for the different layers, which is necessary to facilitate the decoding discussed later. The power allocation strategies developed in [3][4] can be used to optimize  $\{\rho_k\}$ .

<sup>1</sup> This work was fully supported by a grant from the Research Grant Council of the Hong Kong Special Administrative Region, China, under project CityU 117305.

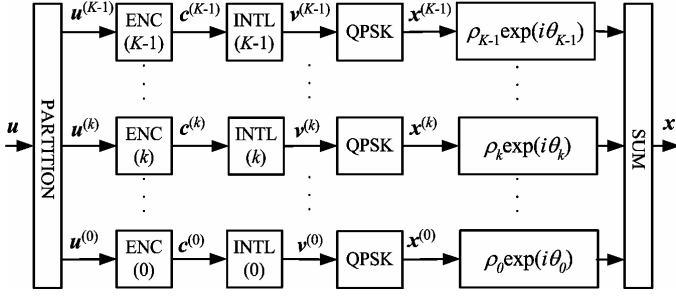


Fig. 1. Encoder of a superposition coding system.

### B. Peak-to-Average Power Ratio

The PAPR of  $x_j$  is defined as

$$\text{PAPR} = \frac{\max_j (|x_j|^2)}{E(|x_j|^2)}. \quad (2)$$

where  $E(\cdot)$  is the expectation and  $|\cdot|$  the absolute value. We assume that all coded bits  $c_j^{(k)}$  are independent, identically distributed (i.i.d.) random variables. It can be verified that, given  $\{\rho_k\}$ , the PAPR is maximized when all  $\theta_k$ 's are equal. As an example, for a 5-layer system with  $\{\rho_k\} = \{0.1634, 0.2380, 0.3467, 0.5051, 0.7358\}$ , the maximum PAPR = 5.97 dB is reached at  $\theta_k = 0, \forall k$ . On the other hand, if we set  $\theta_k = k\pi/10, \forall k$ , then the PAPR is reduced to 5.39 dB.

In order to further reduce the PAPR, we can clip  $x_j$  to  $\bar{x}_j$  before transmission according to the following rule:

$$\bar{x}_j \equiv \begin{cases} x_j, & |x_j| \leq A \\ Ax_j/|x_j|, & |x_j| > A \end{cases} \quad (3)$$

where  $A > 0$  is the clipping threshold (a real value). Following [13], we define the clipping ratio (CR) as  $\gamma = A^2/E(|x_j|^2)$ . The clipped signal  $\bar{x}_j$  is then transmitted over an AWGN channel. The PAPR of the transmitted signal is given by  $A^2/E(|\bar{x}_j|^2)$ . The received signal can be written as

$$y_j = \bar{x}_j + w_j, \quad (4)$$

where  $\{w_j\}$  are samples of a circularly symmetric complex Gaussian random process with zero mean and variance  $\sigma^2$  per dimension. The ratio of energy per bit ( $E_b$ ) to the noise power spectral density ( $N_0$ ) is given by  $E_b/N_0 = E(|\bar{x}_j|^2)/(2R\sigma^2)$ .

In this paper,  $\{\rho_k\}$  are obtained using the simulation-based power allocation method in [4] under the assumption that no clipping is applied. It is of interest to develop new power allocation strategies taking the clipping effect into account for best performance. However, this is still an open problem due to the non-linear characteristic of clipping and is beyond the scope of this paper.

### C. Performance Limit of Clipped Superposition Coding

In general, the linear superposition in (1) induces a signal constellation consisting of  $4^K$  nonequispaced signal points which are used with equal probabilities. The achieved mutual

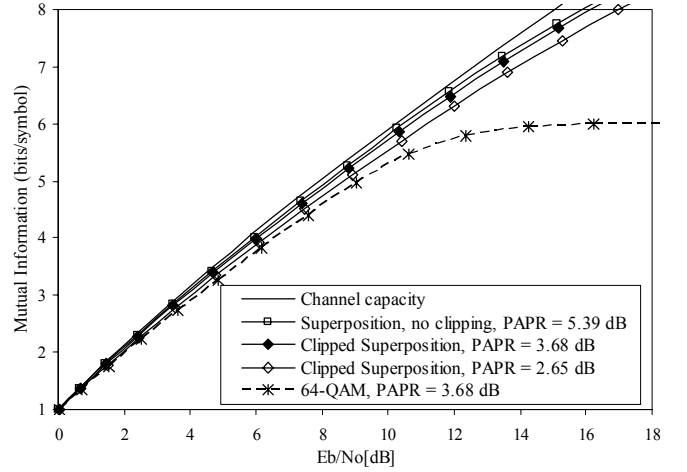


Fig. 2. Mutual information versus  $E_b/N_0$  for a 5-layer superposition coding scheme parameterized by  $\{\theta_k = k\pi/10, \forall k\}$  and  $\{\rho_k\} = \{0.1634, 0.2380, 0.3467, 0.5051, 0.7358\}$ .

information between the clipped transmitted signal and the received signal for a 5-layer example is shown in Fig. 2. The performance limit of the equiprobable 64-QAM constellation and the channel capacity are also included for comparison. We can see that without clipping, the superposition signaling can provide noticeable shaping gains over the equiprobable 64-QAM signaling, and hence can closely approach the channel capacity. Although clipping degrades the achievable rate, the effect is not serious. For example, at the rate of 5 bits/symbol, only 0.16 dB loss in  $E_b/N_0$  is introduced by clipping with PAPR = 3.68 dB.

### III. ITERATIVE DECODING

We now develop an iterative decoding algorithm for clipped superposition coding systems.

The main drawback of clipping is that it introduces additional noise at the transmitter [13]-[15]. If the clipping noise is not carefully treated at the receiver, significant degradation in the bit-error-rate (BER) performance may be observed, even when the clipping threshold  $A$  is moderate. On the other hand, in contrast to channel noise, the clipping noise is introduced by a process characterized by the clipping threshold. Exploiting this fact, we propose the soft compensation algorithm (SCA) to combat the clipping effect.

#### A. Receiver Structure

We first outline the overall receiver structure. The decoding/detection principle discussed below is derived based on the similarity between the superposition coding scheme and the interleave-division multiple-access (IDMA) scheme [11]. As illustrated in Fig. 3, the receiver consists of one elementary signal estimator (ESE) and  $K$  soft-input soft-output (SISO) decoders (DECs). These are connected through the INTLs and DEINTLs (de-interleavers), operating in a turbo-type iterative manner. The messages passing between the ESE and the DECs are the so-called *extrinsic* information. The DECs perform

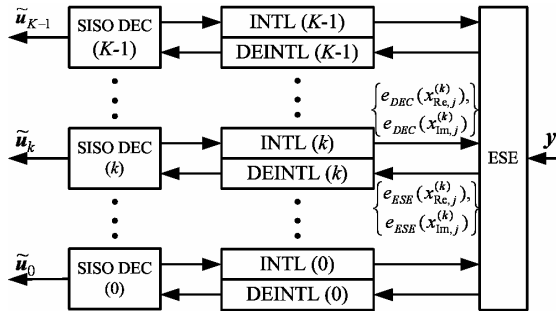


Fig. 3. Block diagram of the iterative decoding/detection algorithm.

standard *a posteriori* probability (APP) decoding, so we will only focus on the ESE which undertakes SISO detection to generate the *a posteriori* log-likelihood ratios (LLRs) for all coded bits [11].

In the following, we concentrate on the detection for layer- $k$ . The signal model (4) is rewritten as

$$y_j = \rho_k e^{i\theta_k} x_j^{(k)} + \xi_j^{(k)} \quad (5)$$

where  $\xi_j^{(k)}$  denotes the distortion contained in  $y_j$  with respect to  $x_j^{(k)}$ . There are two parts in  $\xi_j^{(k)}$ :

$$\xi_j^{(k)} = \zeta_j^{(k)} + z_j \quad (6)$$

where  $z_j$  is the clipping noise and

$$\zeta_j^{(k)} = \sum_{k' \neq k} \rho_{k'} e^{i\theta_{k'}} x_j^{(k')} + w_j \quad (7)$$

is the superposition of the inter-layer-interference and the channel noise. From (3),  $z_j$  can be represented by

$$z_j = \begin{cases} 0, & |x_j| \leq A \\ Ax_j/|x_j| - x_j, & |x_j| > A \end{cases} \quad (8)$$

Consider the detection of  $x_j^{(k)}$  based on (5) and (6). If we simply ignore the clipping noise  $z_j$  in (6), i.e., assuming that  $z_j = 0$ , then the ESE detection algorithm (with linear complexity with respect to  $K$ ) developed for IDMA systems [11] can be directly applied to deal with  $\zeta_j^{(k)}$ , as in the un-clipped cases [3][4]. Unfortunately, with this approach, the performance can degrade significantly compared to unclipped cases if deep clipping (with small  $\gamma$ ) is used.

### B. Soft Compensation Algorithm (SCA)

We now derive the SCA which uses a joint process to handle  $\zeta_j^{(k)}$  and  $z_j$ . The intuition is that, with *extrinsic* information generated by the DECs, an estimate of the clipping noise can be obtained and then subtracted from the received signals, which in turn facilitate the decoding. As will be shown later, the performance can be significantly improved by the SCA with complexity only slightly higher than the detection algorithm for unclipped schemes.

First, some definitions. Let  $x$  be a complex random variable and  $E(x)$  be its mean. Define its covariance matrix as

$$\mathbf{Cov}(x) = \begin{pmatrix} \text{Var}(x_{\text{Re}}) & E(x_{\text{Re}}x_{\text{Im}}) - E(x_{\text{Re}})E(x_{\text{Im}}) \\ E(x_{\text{Re}}x_{\text{Im}}) - E(x_{\text{Re}})E(x_{\text{Im}}) & \text{Var}(x_{\text{Im}}) \end{pmatrix}$$

where  $\text{Var}(\cdot)$  denotes the variance function.

Following the strategy developed in [11], we approximate  $\xi_j^{(k)}$  by an additive complex Gaussian variable. Moreover, we treat  $\zeta_j^{(k)}$  and  $z_j$  in (6) as independent variables. Provided that the statistics of  $\zeta_j^{(k)}$  and  $z_j$  have been computed (the details will be discussed later), the mean and covariance matrix of  $\xi_j^{(k)}$  can be estimated as

$$E(\xi_j^{(k)}) = E(\zeta_j^{(k)}) + E(z_j), \quad (9a)$$

$$\mathbf{Cov}(\xi_j^{(k)}) = \mathbf{Cov}(\zeta_j^{(k)}) + \mathbf{Cov}(z_j). \quad (9b)$$

In order to detect  $x_j^{(k)}$ , we generate

$$\hat{y}_j^{(k)} = e^{-i\theta_k} y_j = \rho_k x_j^{(k)} + \hat{\xi}_j^{(k)}, \quad (10)$$

where  $\hat{\xi}_j^{(k)} = e^{-i\theta_k} \xi_j^{(k)}$ . The statistics of  $\hat{\xi}_j^{(k)}$  are given by

$$E(\hat{\xi}_j^{(k)}) = e^{-i\theta_k} E(\xi_j^{(k)}) \quad (11a)$$

$$\mathbf{Cov}(\hat{\xi}_j^{(k)}) = (\mathbf{R}^{(k)})^T \mathbf{Cov}(\xi_j^{(k)}) \mathbf{R}^{(k)}, \quad (11b)$$

where

$$\mathbf{R}^{(k)} = \begin{pmatrix} \cos(\theta_k) & -\sin(\theta_k) \\ \sin(\theta_k) & \cos(\theta_k) \end{pmatrix},$$

and the superscript  $T$  denotes transpose of matrixes. Then, the output LLR ( $e_{ESE}(x_{\text{Re},j}^{(k)})$ ) for  $x_{\text{Re},j}^{(k)} \in \{+1, -1\}$  can be estimated as follows. (We can handle  $x_{\text{Im},j}^{(k)}$  in a similar way.)

$$\begin{aligned} e_{ESE}(x_{\text{Re},j}^{(k)}) &= \ln \left( \frac{\Pr(\hat{y}_{\text{Re},j}^{(k)} | x_{\text{Re},j}^{(k)} = +1)}{\Pr(\hat{y}_{\text{Re},j}^{(k)} | x_{\text{Re},j}^{(k)} = -1)} \right) \\ &= \ln \left( \frac{(\text{Var}^+(\hat{\xi}_{\text{Re},j}^{(k)}))^{-1/2} \exp(-(\hat{y}_{\text{Re},j}^{(k)} - \rho_k - E^+(\hat{\xi}_{\text{Re},j}^{(k)}))^2 / (2\text{Var}^+(\hat{\xi}_{\text{Re},j}^{(k)})))}{(\text{Var}^-(\hat{\xi}_{\text{Re},j}^{(k)}))^{-1/2} \exp(-(\hat{y}_{\text{Re},j}^{(k)} + \rho_k - E^-(\hat{\xi}_{\text{Re},j}^{(k)}))^2 / (2\text{Var}^-(\hat{\xi}_{\text{Re},j}^{(k)})))} \right) \\ &= \frac{1}{2} \ln \left( \frac{\text{Var}^-(\hat{\xi}_{\text{Re},j}^{(k)})}{\text{Var}^+(\hat{\xi}_{\text{Re},j}^{(k)})} \right) + \frac{(\hat{y}_{\text{Re},j}^{(k)} + \rho_k - E^-(\hat{\xi}_{\text{Re},j}^{(k)}))^2}{2\text{Var}^-(\hat{\xi}_{\text{Re},j}^{(k)})} \\ &\quad - \frac{(\hat{y}_{\text{Re},j}^{(k)} - \rho_k - E^+(\hat{\xi}_{\text{Re},j}^{(k)}))^2}{2\text{Var}^+(\hat{\xi}_{\text{Re},j}^{(k)})} \end{aligned} \quad (12)$$

where  $E^+(\hat{\xi}_{\text{Re},j}^{(k)})$  and  $E^-(\hat{\xi}_{\text{Re},j}^{(k)})$  denote the means of  $\hat{\xi}_{\text{Re},j}^{(k)}$  under the hypothesis that  $x_{\text{Re},j}^{(k)}$  is either  $+1$  or  $-1$ , respectively.  $\text{Var}^+(\hat{\xi}_{\text{Re},j}^{(k)})$  and  $\text{Var}^-(\hat{\xi}_{\text{Re},j}^{(k)})$  are defined similarly. From (9) and (12), we can see that the SCA is essentially a turbo-type clipping noise cancellation technique.

For comparison, if there is no clipping, then (12) becomes

$$e_{ESE}(x_{\text{Re},j}^{(k)}) = 2\rho_k \cdot \frac{\hat{y}_{\text{Re},j}^{(k)} - E(\hat{\xi}_{\text{Re},j}^{(k)})}{\text{Var}(\hat{\xi}_{\text{Re},j}^{(k)})} \quad (13)$$

where  $\hat{\zeta}_j^{(k)} = e^{-i\theta_k} \zeta_j^{(k)}$ .

It can be seen that (12) is slightly more complicated than (13). This is because that the statistics of  $\hat{\zeta}_j^{(k)}$  are different for different hypothesis on  $x_{\text{Re},j}^{(k)}$  since the clipping effect depends on hypothesis. However, the overall complexity is not significantly increased compared to unclipped cases, since it is usually dominated by the APP decoding of the component codes, especially when turbo-like code is used.

The computational details of the generation of the statistics for  $\zeta_j^{(k)}$  and  $z_j$  are discussed separately in Subsections III-C and -D. An SNR evolution technique, to analyze the above detection algorithm for unclipped superposition coding systems, is also discussed later.

### C. Estimation of $E(\zeta_j^{(k)})$ and $\mathbf{Cov}(\zeta_j^{(k)})$

Let  $e_{\text{DEC}}(x) = \ln(\Pr(x=+1)/\Pr(x=-1))$  be the extrinsic LLR for coded bit  $x$  from the DEC's (see Fig. 3). Following [11],  $E(x_j^{(k)})$  and  $\mathbf{Cov}(x_j^{(k)})$  can be estimated by

$$E(x_j^{(k)}) = \tanh(e_{\text{DEC}}(x_{\text{Re},j}^{(k)})/2) + i \tanh(e_{\text{DEC}}(x_{\text{Im},j}^{(k)})/2), \quad (14a)$$

$$\mathbf{Cov}(x_j^{(k)}) = \begin{pmatrix} 1 - (E(x_{\text{Re},j}^{(k)}))^2 & 0 \\ 0 & 1 - (E(x_{\text{Im},j}^{(k)}))^2 \end{pmatrix}. \quad (14b)$$

(Initially, we set  $E(x_j^{(k)}) = 0$  and  $\mathbf{Cov}(x_j^{(k)}) = \mathbf{I}$ , implying no feedback from the DEC's.) Then, from (7), the statistics of  $\zeta_j^{(k)}$  which are required to compute (9) can be generated by

$$E(\zeta_j^{(k)}) = \sum_{k' \neq k} \rho_{k'} e^{i\theta_{k'}} E(x_j^{(k')}), \quad (15a)$$

$$\mathbf{Cov}(\zeta_j^{(k)}) = \sum_{k' \neq k} \rho_{k'}^2 \mathbf{R}^{(k')} \mathbf{Cov}(x_j^{(k')}) (\mathbf{R}^{(k')})^T + \sigma^2 \mathbf{I}, \quad (15b)$$

where  $\mathbf{I}$  is a  $2 \times 2$  identity matrix.

### D. Estimation of $E(z_j)$ and $\mathbf{Cov}(z_j)$

Now the key to evaluate  $E(\zeta_j^{(k)})$  and  $\mathbf{Cov}(\zeta_j^{(k)})$  is to find  $E(z_j)$  and  $\mathbf{Cov}(z_j)$ . We assume that the clipping threshold is known at the receiver. From (8), if the probability density function  $p(x_j)$  of  $x_j$  is available, then  $E(z_j)$  and  $\mathbf{Cov}(z_j)$  can be evaluated using numerical integration. However, it is impractical to find  $p(x_j)$  in realistic systems. We propose the following suboptimal strategy based on Gaussian approximation of  $x_j$ .

From (5) and (7), we can express  $x_j$  as

$$x_j = \rho_k e^{i\theta_k} x_j^{(k)} + \zeta_j^{(k)} - w_j. \quad (16)$$

Then  $\mu_j \equiv E(x_j)$  and  $\mathbf{V}_j \equiv \mathbf{Cov}(x_j)$  can be obtained from (16) if  $E(x_j^{(k)})$  and  $\mathbf{Cov}(x_j^{(k)})$  are given and  $E(\zeta_j^{(k)})$  and  $\mathbf{Cov}(\zeta_j^{(k)})$  are available (see Subsection III-C). We now model  $x_j$  as a complex Gaussian random variable. Consequently,  $E(z_j)$  and  $\mathbf{Cov}(z_j)$  are fully determined by  $(\mu_j, \mathbf{V}_j)$  from (8). We denote these two relationships using two functions below:

$$E(z_j) = \phi(\mu_j, \mathbf{V}_j), \quad (17a)$$

$$\mathbf{Cov}(z_j) = \gamma(\mu_j, \mathbf{V}_j). \quad (17b)$$

In general, the two functions in (17) can be generated numerically using the Monte Carlo method. We can create two look-up tables to characterize them. Assuming that these two tables are available, then the computational load of the SCA is only slightly higher than the ESE algorithm in [11]. We examine the required memory usage of the look-up tables below.

Since  $\mu_j$  and  $\mathbf{V}_j$  involve five parameters, we need two 5-dimensional (5D) tables. We now consider an approximate technique to reduce memory cost using

$$\mathbf{V}_j \approx v_j \mathbf{I} \quad (18)$$

where

$$v_j = (\text{Var}(x_{\text{Re},j}) + \text{Var}(x_{\text{Im},j})) / 2.$$

This is to approximately characterize  $x_j$  using a complex Gaussian distribution  $\mathcal{CN}(\mu_j, v_j \mathbf{I})$ . Now there are only three parameters involved. Furthermore, it is easily shown that

$$\phi(\mu_j, v_j \mathbf{I}) = \mu_j / |\mu_j| \phi(|\mu_j|, v_j \mathbf{I}) \quad (19a)$$

$$\gamma(\mu_j, v_j \mathbf{I}) = \Psi_j \gamma(|\mu_j|, v_j \mathbf{I}) \Psi_j^T \quad (19b)$$

where

$$\Psi_j = \frac{1}{|\mu_j|} \begin{pmatrix} \mu_{\text{Re},j} & -\mu_{\text{Im},j} \\ \mu_{\text{Im},j} & \mu_{\text{Re},j} \end{pmatrix}.$$

Therefore we only need two 2-dimensional (2D) tables to characterize  $\phi(|\mu_j|, v_j \mathbf{I})$  and  $\gamma(|\mu_j|, v_j \mathbf{I})$  and then use (19) to obtain  $\phi(\mu_j, v_j \mathbf{I})$  and  $\gamma(\mu_j, v_j \mathbf{I})$  during SCA. This requires slightly more operations but can greatly reduce the memory cost.

### E. SNR Evolution

We now outline an SNR evolution technique [11] to evaluate the performance of unclipped superposition coding systems, which also provides insight into the convergence property of the iterative decoding. Recall that in unclipped cases,  $z_j = 0$  and the detection is based on (13). With simple manipulations, we can rewrite (13) as

$$e_{\text{ESE}}(x_{\text{Re},j}^{(k)}) = \frac{2\rho_k}{\text{Var}(\hat{\zeta}_{\text{Re},j}^{(k)})} (\rho_k x_{\text{Re},j}^{(k)} + \hat{\zeta}_{\text{Re},j}^{(k)} - E(\hat{\zeta}_{\text{Re},j}^{(k)})). \quad (20)$$

The SNR in  $e_{\text{ESE}}(x_{\text{Re},j}^{(k)})$  with respect to  $x_{\text{Re},j}^{(k)}$  after observing  $y_j$  is calculated as

$$\text{snr}_{\text{Re},j}^{(k)} = \frac{E((\rho_k x_{\text{Re},j}^{(k)})^2)}{E((\hat{\zeta}_{\text{Re},j}^{(k)} - E(\hat{\zeta}_{\text{Re},j}^{(k)}))^2)} = \frac{\rho_k^2}{\text{Var}(\hat{\zeta}_{\text{Re},j}^{(k)})}.$$

Following [11], we define the average SNR as

$$\text{snr}_{\text{Re}}^{(k)} = E(\text{snr}_{\text{Re},j}^{(k)}) = E\left(\frac{\rho_k^2}{\text{Var}(\hat{\zeta}_{\text{Re},j}^{(k)})}\right) \geq \frac{\rho_k^2}{E(\text{Var}(\hat{\zeta}_{\text{Re},j}^{(k)}))}, \quad (21)$$

where Jensen's inequality [1] is used. Define  $v_{\text{Re}}^{(k)} \equiv E(\text{Var}(x_{\text{Re},j}^{(k)}))$ . Similarly, we can define  $snr_{\text{Im}}^{(k)}$  and  $v_{\text{Im}}^{(k)}$ . Since  $x_{\text{Re},j}^{(k)}$  and  $x_{\text{Im},j}^{(k)}$  are the coded bits of the same component code, we have  $v^{(k)} \equiv v_{\text{Re}}^{(k)} = v_{\text{Im}}^{(k)}$ . Then it can be easily verified from (14) and (15) that the input SNR to DEC- $k$  is bounded by

$$snr^{(k)} \equiv snr_{\text{Re}}^{(k)} = snr_{\text{Im}}^{(k)} \geq \gamma^{(k)} \equiv \frac{\rho_k^2}{\sum_{k' \neq k} \rho_{k'}^2 v^{(k')} + \sigma^2}. \quad (22)$$

Notice that  $v^{(k)}$  is a function of  $snr^{(k)}$ . We express this function as

$$v^{(k)} = f(sn r^{(k)}).$$

We also define the BER performance of DEC- $k$  as a function of  $snr^{(k)}$  as

$$\text{BER} = g(sn r^{(k)}).$$

Both  $f(\bullet)$  and  $g(\bullet)$  can be obtained by applying Monte-Carlo method to the underlying forward error correction code. From (22), we can obtain a lower bound of  $snr^{(k)}$  during the iterations

$$\gamma_{\text{new}}^{(k)} = \frac{\rho_k^2}{\sum_{k' \neq k} \rho_{k'}^2 f(\gamma_{\text{old}}^{(k')}) + \sigma^2}, \quad \forall k, \quad (23)$$

where  $\gamma_{\text{old}}^{(k)}$  and  $\gamma_{\text{new}}^{(k)}$  are  $\gamma^{(k)}$  values before and after one iteration. Initially, we set  $f(\gamma_{\text{old}}^{(k)}) = 1, \forall k$ . Repeating (23), the SNR evolution for the iterative detection process can be tracked. During the final iteration, the BER performance can be estimated by substituting  $\{\gamma^{(k)}\}$  into  $g(\bullet)$ .

#### IV. NUMERICAL RESULTS

##### A. Simulation Model

In this section, we show numerical results that demonstrate the performance of the clipped superposition coding scheme. The following examples are considered. A rate-1/2 doped code with data length  $10^5$  [16] is chosen as the component code for each layer. We set  $K = 4$  and 5. The corresponding total rates  $R = 4$  and 5 bits/symbol, and the PAPRs without clipping are 4.74 and 5.39 dB, respectively. For both systems, we set the clipping ratio  $\gamma$  to 4.0, 3.5, and 3.0 dB, and the corresponding PAPR is reduced to about 4.1, 3.7 and 3.3 dB, respectively. The number of iterations in the component decoders is set to 200, and the number of iterations between detection and decoding is set to 6. The entropy based stop criterion introduced in [3] is used.

##### B. Results

The simulated BER performance (averaged over all layers) is shown in Fig. 4 and Fig. 5. For comparison, the performance limit of the conventional equiprobable 64-QAM signaling is also included. The channel capacities for  $R = 4$  and 5 bits/symbol are  $E_b/N_0 = 5.74$  and 7.92 dB, respectively.

In Fig. 4, we present simulation results without clipping effect compensation. First, we can see that the performance of

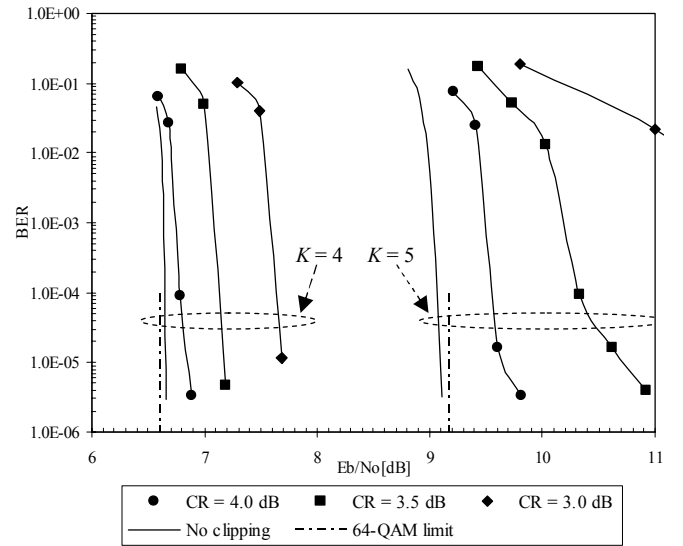


Fig. 4. Performance of clipped superposition coding systems without clipping effect compensation. For  $K = 4$ ,  $\{\rho_k\} = \{0.2430, 0.3529, 0.5124, 0.7442\}$  and  $\{\theta_k = k\pi/8, \forall k\}$ . For  $K = 5$ ,  $\{\rho_k\} = \{0.1634, 0.2380, 0.3467, 0.5051, 0.7358\}$  and  $\{\theta_k = k\pi/10, \forall k\}$ .

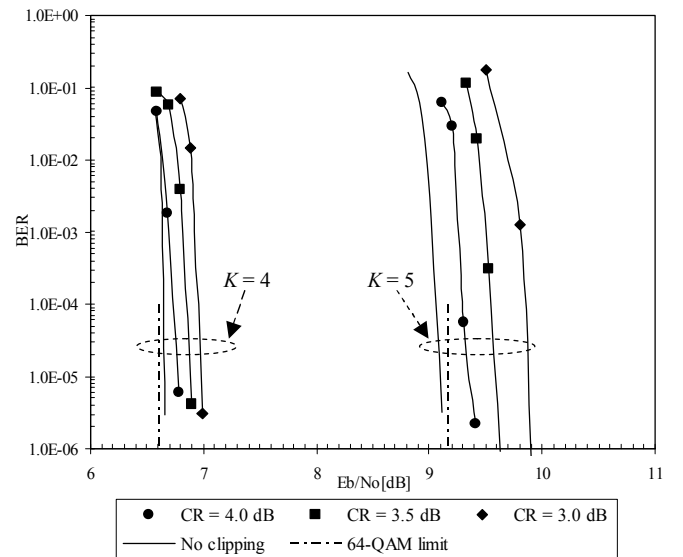


Fig. 5. Performance of clipped superposition coding systems with SCA. The parameters are the same as those in Fig. 4.

the unclipped superposition coding schemes is quite close to the channel capacity. At BER of  $10^{-5}$ , the gaps between the required  $E_b/N_0$  and the channel capacity are only 0.9 and 1.2 dB for  $K = 4$  and 5, respectively. Note that the performance of the 5-layer scheme surpasses the theoretical limit of the equiprobable 64-QAM signaling.

It is clearly evident in Fig. 4 that the performance of the clipped systems deteriorates significantly when the clipping effect is not compensated. The performance loss increases as  $\gamma$  decreases, as expected. It can also be seen that the system with more layers is more sensitive to the clipping effect. For the 5-layer example, the performance degrades by about 0.7 and 1.8 dB with  $\gamma = 4.0$  and 3.5 dB, respectively, and the performance with  $\gamma = 3.0$  dB is even worse.

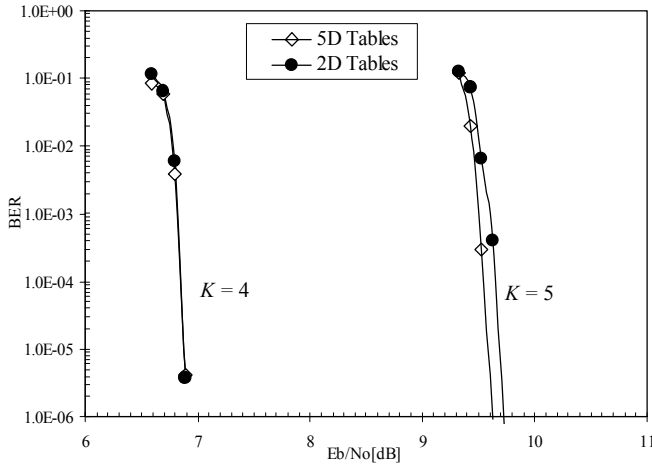


Fig. 6. Performance comparison between 5-D and 2-D look-up tables for clipped superposition coding systems with  $\gamma=3.5$  dB. The parameters are the same as those in Fig. 4.

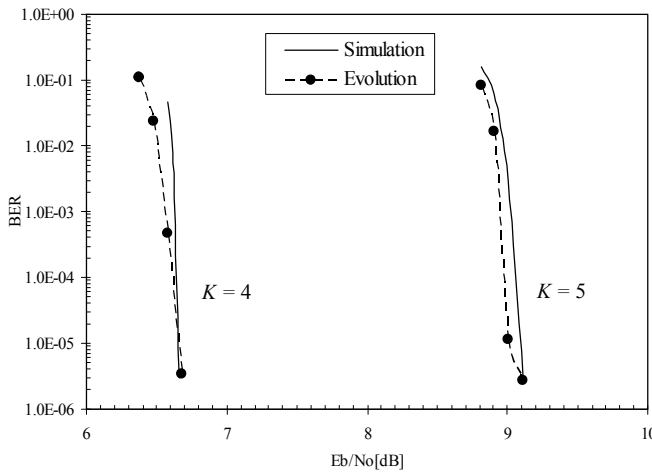


Fig. 7. Performance comparison between evolution and simulation for unclipped superposition coding systems. The parameters are the same as those in Fig. 4.

Fig. 5 illustrates the performance with SCA for different clipping ratios. Comparing Fig. 4 and Fig. 5, we can see that the performance loss due to clipping can be largely removed by SCA. For  $K = 5$ , the performances loss relative to the un-clipped scheme is reduced to within 0.3, 0.5 and 0.8 dB for  $\gamma = 4.0, 3.5$  and  $3.0$  dB, respectively.

In the simulations, we observed that the poor performance without compensation is mainly due to the high error rates of the layers with lower powers (smaller  $\rho_k$ ), which is quite reasonable. With high probability, the signals from high-power layers, which contribute more to the unclipped signal  $x_j$ , can be perfectly recovered. So it is not surprising that the SCA, which is based on the estimation of  $x_j$ , can offer significant performance improvement.

The performance loss due to the approximation in (18) is shown in Fig. 6, where each dimension is quantized to 20 levels. We can see that the difference between the 5D and 2D methods is marginal (within 0.1 dB). For the latter, two small tables of size  $20 \times 20$  are sufficient for  $\phi(\bullet)$  and  $\gamma(\bullet)$ .

Fig. 7 compares the simulated and predicted (using SNR evolution) performance for the unclipped 4- and 5-layer superposition coding schemes. We can see that the predicted results are in good agreement with the simulated results. This indicates that the SNR evolution technique provides a fast and reasonably accurate way to predict the performance of the detection algorithm based on (13). Furthermore, given  $f(\bullet)$ ,  $g(\bullet)$  and target BER, this technique can also be used to optimize the amplitude factors  $\{\rho_k\}$ . (See [11] for related discussions.)

## V. CONCLUSIONS

We have studied a peak-power-limited superposition coding system based on clipping. A turbo-type soft compensation algorithm to alleviate the clipping effect is proposed. Simulation results show that the proposed technique can efficiently enhance the system performance. Possible subjects for future work include performance analysis and power allocations for the clipped schemes considered in this paper.

## REFERENCES

- [1] T. M. Cover and J. M. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.
- [2] D. N. C. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge: Cambridge University Press, 2005.
- [3] X. Ma and Li Ping, "Coded modulation using superimposed binary codes," *IEEE Trans. Inform. Theory*, vol. 50, no. 12, pp. 3331–3343, Dec. 2004.
- [4] X. Ma and Li Ping, "Power allocations for multilevel coding with sigma mapping," *Electron. Lett.*, vol. 40, no. 10, pp. 609–611, May 2004.
- [5] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT–28, pp. 55–67, Jan. 1982.
- [6] H. Imai and S. Hirakawa, "A new multilevel coding method using error-correcting codes," *IEEE Trans. Inform. Theory*, vol. IT–23, pp. 371–377, May 1977.
- [7] Fischer, R. *Precoding and Signal Shaping for Digital Transmission*. New York: Wiley, 2002.
- [8] G. D. Forney, Jr., "Trellis shaping," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 281–300, Mar. 1992.
- [9] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel codes: theoretical concepts and practical design rules," *IEEE Trans. Inform. Theory*, vol. 45, no.5, pp. 1361–1391, July 1999.
- [10] N. Varnica, X. Ma, and A. Kavčić, "Iteratively decodable codes for bridging the shaping gap in communication channels," in *Proc. Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Nov. 2002.
- [11] L. H. Liu, J. Tong, and Li Ping, "Analysis and optimization of CDMA systems with chip-level interleavers," *IEEE J. Select. Areas Commun.*, vol. 24, no. 1, pp. 141–150, Jan. 2006.
- [12] H. Schoeneich and P. A. Hoeher, "Adaptive interleave-division multiple access-A potential air interference for 4G bearer services and wireless LANs," in *Proc. WOCN 2004*, Muscat, Oman, June 2004, pp. 179–182.
- [13] D. Kim and G. L. Stüber, "Clipping noise mitigation for OFDM by decision-aided reconstruction," *IEEE Commun. Lett.*, vol. 3, no. 1, pp. 4–6, Jan. 1999.
- [14] H. Chen and A. M. Haimovich, "Iterative estimation and cancellation of clipping noise for OFDM signals," *IEEE Commun. Lett.*, vol. 7, no. 7, pp. 305–307, July 2003.
- [15] G. Gelle, M. Colas, and D. Declercq, "Turbo decision aided reconstruction of clipping noise in coded OFDM," in *Proc. IEEE SPAWC 2004*, 2004.
- [16] S. ten Brink, "Rate one-half code for approaching the Shannon limit by 0.1 dB," *Electron. Lett.*, vol. 36, no. 15, pp. 1293–1294, July 2000.