
Agenda Item: 7.5

Source: NEC

Title: Multi-resolution Precoding Codebook

Document for: Discussion

1 Introduction

In Multi-resolution precoding, the precoding codebook consists of multiple codebooks that represent different resolution levels and a given channel realization can be quantized to a better resolution by successively choosing from a higher resolution codebook, see [4, 3, 2] for a related discussion. On the other hand, in many systems, the optimal precoders from the codebook for two adjacent transmission blocks are close with respect to a proper distance measure defined on the set of all possible precoders. Here, the adjacent blocks may be considered in time or in frequency, for instance each block can be a set of consecutive tones in an OFDM system. The intuition behind this statement is that in a practical system the channel does not change abruptly from one transmission block to the adjacent one. Thus, the optimal precoders (from the codebook) used for those two blocks should be identical if the channel is slowly varying and the codebook resolution is not relatively high. By increasing the codebook resolution or having a more dynamic channel, the optimal precoders corresponding the adjacent blocks are not equal anymore, nevertheless they will be close. The closeness (or proximity) between two precoders can be measured based by a proper distance metric defined on the space of all such precoders. Consequently, only the differential information needs to be quantized, see [1] for a related discussion. This fact motivates the use of multi-resolution codebook in LTE-A.

2 Precoding Codebook

In this section, we discuss the precoding codebook structure. First, we briefly consider a system employing a conventional codebook, in which only one precoder index is fed back for each resource block (RB) and different RBs are treated separately. Such a codebook can also be used to report a single precoding index for a group of adjacent RBs. Next, we present a system employing a multi-resolution codebook, where for a group of adjacent RBs, the precoder selection algorithm first selects a common precoder index from a low-resolution codebook and then for each RB within the group, a high-resolution codebook is used to fine tune the precoder selected for that RB. Henceforth, the conventional codebook is referred to as the single-resolution codebook.

The concept of multi-resolution codebook may be extended to multiple levels of codebook resolution. For simplicity, in this work, we primarily focus on the multi-resolution codebooks with only two resolution levels: the low-resolution and the high-resolution. It will be shown that by using such multi-resolution codebooks a performance similar to the single-resolution codebook can be achieved with a lower feedback load. Furthermore, both single-resolution and multi-resolution codebooks entail almost the same memory requirement to store the codebooks and the complexity involved in the precoder selection in either case is also comparable.

2.1 Single-resolution codebook

We let \mathbf{H} denote a representative channel matrix corresponding to any RB, which for instance could be the channel matrix corresponding to the center tone of that RB. For a given transmission rank r and B bits of PMI feedback, the codebook design problem is formulated as finding the set $\mathcal{Q} = \{\mathbf{Q}_1, \mathbf{Q}_2, \dots, \mathbf{Q}_{2^B}\}$ of $N_T \times r$ semi-unitary matrices, that is a solution to the optimization problem given by

$$C = \max_{\mathcal{Q}} \mathbb{E} \left[\max_{\mathbf{Q} \in \mathcal{Q}} \log \det \left(\mathbf{I} + \frac{P}{N_T} \mathbf{H} \mathbf{Q} \mathbf{Q}^* \mathbf{H}^* \right) \right]. \quad (1)$$

If a single-resolution codebook is used, the precoder \mathbf{Q} for the channel \mathbf{H} is obtained from the

codebook \mathcal{Q} such that $\log \det \left(\mathbf{I} + \frac{P}{N_T} \mathbf{H} \mathbf{Q} \mathbf{Q}^* \mathbf{H}^* \right)$ is maximized. This capacity measure may be modified accordingly depending on the type of the receiver employed.

2.2 Multi-resolution codebook

In a multi-resolution codebook, the codebook $\mathcal{Q} = \{\mathcal{Q}_1, \mathcal{Q}_2, \dots, \mathcal{Q}_q\}$ consists of q codebooks where the codebook \mathcal{Q}_{i+1} is used to add more resolution to the prior set of codebooks $\mathcal{Q}_1, \dots, \mathcal{Q}_i$. By using a multi-resolution codebook, finding the optimal precoder for a given channel realization \mathbf{H} is performed as follows. First, we find the optimal precoder $\mathbf{Q}_1(\mathbf{H}) \in \mathcal{Q}_1$ for a given channel realization \mathbf{H} . Then, we find the optimal precoder $\mathbf{Q}_2(\mathbf{H}, \mathbf{Q}_1) \in \mathcal{Q}_2$ that depends both on the previous precoder \mathbf{Q}_1 and the channel matrix \mathbf{H} . Depending on the required resolution, this action is successively performed to find, e.g., the precoder $\mathbf{Q}_i(\mathbf{H}, \mathbf{Q}_1, \dots, \mathbf{Q}_{i-1}) \in \mathcal{Q}_i, i \leq q$. It should be pointed out that the multi-resolution codebook is optimized based on the channel statistics. In general, codebooks for different resolution levels may be optimized using different channel statistics. For example, in an OFDM system, the first codebook \mathcal{Q}_1 may be optimized by considering the statistics of the representative matrix of any RB, whereas the second codebook \mathcal{Q}_2 may be optimized by considering the joint statistics including the representative matrices of the adjacent RBs as well.

For a simpler discussion, we assume $q = 2$ and consider only one low-resolution codebook \mathcal{Q}_1 and one high-resolution codebook \mathcal{Q}_2 . In the sequel, we provide two examples of the multi-resolution codebook design: the multi-resolution codebook design based on partitioning and the multi-resolution codebook design based on unitary rotation.

Multi-resolution codebook based on partitioning

For a simple construction of a multi-resolution codebook, we can use a partitioning concept. Let \mathcal{K} denote a single-resolution codebook of size $|\mathcal{K}|$ which has been designed to meet the highest required resolution. To generate a multi-resolution codebook, we first find a low-resolution codebook \mathcal{Q}_1 by taking $|\mathcal{Q}_1|$ elements from \mathcal{K} which have the maximum minimum distance between any two

elements. For each element \mathbf{Q}_i of the codebook \mathcal{Q}_1 , we then find the set of *closest* $|\mathcal{K}|/|\mathcal{Q}_1|$ elements from \mathcal{K} and denote it by \mathcal{Q}_{2i} . However, the elements are chosen such that each element of \mathcal{K} belongs to one and only one \mathcal{Q}_{2i} . Please note that the set $\mathcal{Q}_2 = \{\mathcal{Q}_{21}, \mathcal{Q}_{22}, \dots, \mathcal{Q}_{2,|\mathcal{Q}_1|}\}$ eventually includes all the elements of \mathcal{K} but each element is doubly indexed, i.e., an index corresponding to a unique element of \mathcal{Q}_1 and an index which enumerates the possible choices of the precoders in \mathcal{Q}_2 for a given precoder index in \mathcal{Q}_1 . Thus, the set \mathcal{Q}_2 is divided into $|\mathcal{Q}_1|$ partitions such that each partition \mathcal{Q}_{2i} has equal number of elements. It should also be noted that such partitioning is not unique, because the set of *closest* elements depends on the order that we assign the elements of \mathcal{K} to each partition. Thus, an optimization may be performed to maximize the overall codebook performance by taking the best partitioning.

An alternative approach to design a multi-resolution codebook is to first design a low-resolution codebook \mathcal{Q}_1 directly. Next, for each entry of the codebook, say \mathbf{Q}_i , we find the set of channel conditions (realizations) for which this entry is chosen and denote it as $\mathcal{H}(\mathbf{Q}_i)$. Then, we design an optimal codebook \mathcal{Q}_{2i} using only the channel conditions in $\mathcal{H}(\mathbf{Q}_i)$, i.e., solving the optimization problem (1) where the expectation is taken over $\mathcal{H}(\mathbf{Q}_i)$. The size of \mathcal{Q}_{2i} is chosen to meet the desired resolution level. The codebook $\mathcal{Q}_2 = \{\mathcal{Q}_{21}, \mathcal{Q}_{22}, \dots, \mathcal{Q}_{2,|\mathcal{Q}_1|}\}$ is then constructed as the set of all such designed codebooks.

In some cases, it is beneficial to design a nested multi-resolution codebook. The codebook $\mathcal{Q} = (\mathcal{Q}_1, \mathcal{Q}_2)$ is called *nested multi-resolution codebook* if and only if

$$\forall i, 1 \leq i \leq |\mathcal{Q}_1| : \mathcal{Q}_{2i} \subset \mathcal{Q}_1 \quad (2)$$

One possible design can be obtained by taking the low-resolution codebook $\mathcal{Q}_1 = \mathcal{K}$ and then finding the set $\mathcal{Q}_{2i} \subset \mathcal{K}$ as the $|\mathcal{Q}_{2i}|$ closest neighbors of each element $\mathbf{Q}_i \in \mathcal{Q}_1$ to construct the high-resolution codebook $\mathcal{Q}_2 = \{\mathcal{Q}_{2i}\}_{i=1}^{|\mathcal{Q}_1|}$.

Please note that the representation of the latter codebook constructions requires storing $|\mathcal{Q}_1|$ and $|\mathcal{Q}_2|$ precoders and a set of indices to denote the partitioning in \mathcal{Q}_2 . Also, the former codebook construction needs storing $|\mathcal{K}|$ precoders plus two sets of indices to represent \mathcal{Q}_1 and \mathcal{Q}_2

codebooks.

Multi-resolution codebook based on unitary rotation

To simplify the codebook storage, we may use functions, $f_i, 1 \leq i \leq Q$ which convert each element $\mathbf{Q}_1 \in \mathcal{Q}_1$ to a new precoder of the same size. An example of such function is a unitary rotation, i.e., $\mathbf{Q}_2 = f_i(\mathbf{Q}_1) = \mathbf{F}_i \mathbf{Q}_1$. For the precoder of size $N_T \times r$, the unitary rotations \mathbf{F}_i are of size $N_T \times N_T$. Using such a transformation, one needs to store only Q unitary matrices to generate the entire high-resolution codebook \mathcal{Q}_2 . Thus, the average performance of such a generated codebook over all entries of the codebook \mathcal{Q}_1 should be considered in the design of the functions f_i . More specifically, we divide the set of all the channel realizations into $|\mathcal{Q}_1|$ partitions $\mathcal{H}(\mathbf{Q}_i), 1 \leq i \leq |\mathcal{Q}_1|$ where the precoder $\mathbf{Q}_i \in \mathcal{Q}_1$ is the optimal precoder.

Thus the codebook design problem is given by

$$\max_{\mathcal{F}} \sum_{i=1}^{|\mathcal{Q}_1|} \mathbb{E}_{\mathcal{H}(\mathbf{Q}_i)} \left[\max_{\mathbf{F} \in \mathcal{F}} \log \det \left(\mathbf{I} + \frac{P}{N_T} \mathbf{H} \mathbf{F} \mathbf{Q}_i \mathbf{Q}_i^* \mathbf{F}^* \mathbf{H}^* \right) \right]. \quad (3)$$

where $\mathcal{F} = \{\mathbf{F}_1, \mathbf{F}_2, \dots\}$ is the set of unitary matrices that equivalently represent the codebook \mathcal{Q}_2 .

3 Simulation Results

First, we define the system setup and the codebooks used for the simulations. We assume a NodeB with 4 transmit antennas and a user with 2 receive antennas. The precoding schemes use two codebooks corresponding to rank 1 and rank 2, each of the same size. An extra feedback bit is also used to feedback the rank of the selected precoder. The PMI feedback of size 3,4,5, or 6 bits represents 8,16,32, or 64 choices per rank. The single-resolution codebook of size 2^B is denoted by $\mathcal{C}^{(B)}$. The other important system parameters are summarized in Table 1.

For the multi-resolution codebook, we use the following designs. For each codebook $\mathcal{C}^{(B)}$:

- The multi-resolution codebook $\mathcal{N}^{(B)}, B \geq 5$ is designed by using the partitioning approach. We first find $\mathcal{N}_1^{(B)}$ of size $2^{(B-3)}$ by taking the elements from $\mathcal{C}^{(B)}$ which have the maximum

minimum pairwise distance. Then, we divide the set $\mathcal{C}^{(B)}$ into $2^{(B-3)}$ partitions, where each partition contains the set of 8 closest neighbors for each codebook entry $\mathbf{Q} \in \mathcal{N}_1^{(B)}$. Thus, $\mathcal{N}_2^{(B)}$ is the set $\mathcal{C}^{(B)}$ along with the acquired partitioning.

- The multi-resolution codebook $\mathcal{O}^{(B)}$ is also designed by using the partitioning approach. Here, we take $\mathcal{O}_1^{(B)} = \mathcal{C}^{(B)}$ and then we find the set of 8 closest neighbors of each codebook entry $\mathbf{Q}_i \in \mathcal{O}_1^{(B)}$ to obtain $\mathcal{O}_{2i}^{(B)}$, and we have $\mathcal{O}_2^{(B)} = \{\mathcal{O}_{21}^{(B)}, \mathcal{O}_{22}^{(B)}, \dots, \mathcal{O}_{2,|\mathcal{C}^{(B)}|}^{(B)}\}$.
- The multi-resolution codebook $\mathcal{M}^{(B)}$ is obtained by taking its first (low) resolution codebook as $\mathcal{M}_1^{(B)} = \mathcal{C}^{(B)}$ and then designing a 3-bit unitary rotation codebook $\mathcal{F}^{(B)}$ which equivalently represents $\mathcal{M}_2^{(B)}$.

Please note that, as a point of comparison, the multi-resolution codebook is designed such that only 3 bits is required to represent the high-resolution information about the precoder in all mentioned cases. Thus, the value B in the codebooks $\mathcal{C}^{(B)}$, $\mathcal{M}^{(B)}$, and $\mathcal{O}^{(B)}$ only refers to the size of low-resolution codebook. However, the size of low-resolution codebook in $\mathcal{N}^{(B)}$ is $B - 3$ bits.

We compared the throughput performances of the different codebooks through a link level simulation. We consider precoding over M resource blocks. When a single-resolution codebook $\mathcal{C}^{(B)}$ is used, the precoder index for each resource block is found individually and total of BM bits is fed back to the NodeB. However, when any one of the codebooks $\mathcal{M}^{(B)}$, $\mathcal{N}^{(B)}$, and $\mathcal{O}^{(B)}$ is used, the precoding requires B bits for the center resource block and only 3 bits for the rest of the $M - 1$ resource blocks, thus, the total number of feedback bits can be calculated as $B + 3(M - 1)$.

Figure 1 shows the link level throughput results comparing the single-resolution codebook $\mathcal{C}^{(B)}$ with the multi-resolution codebook $\mathcal{M}^{(B)}$ for different B . Here, we have used $M = 3$ resource blocks. It is observed that the performance of the multi-resolution codebook is better than that of the single-resolution codebook for $B = 3, 4, 5$ while the feedback load has reduced to 9, 10, 11 from 9, 12, 15 bits. The performance of the multi-resolution codebook $\mathcal{M}^{(6)}$ is, however, slightly lower than that of the single-resolution codebook $\mathcal{C}^{(6)}$. Nonetheless, there is a considerable saving

of 6 bits in the feedback load in this case.

Figures 2-4 show the link level throughput results comparing the single-resolution codebook $\mathcal{C}^{(B)}$ with the multi-resolution codebook $\mathcal{M}^{(B)}$ for different B by increasing the number of resource blocks from $M = 3$ to $M = 7$ and $M = 11$. It is observed that the performance of the multi-resolution codebook starts degrading with respect to that of the single-resolution codebook as M increases. This is due to the fact that we assume that the low-resolution index for all M RBs is the same and only calculate the high-resolution indices for the $M - 1$ blocks adjacent to the center RB. It is observed that for $M = 7$ and $B = 3$ the performance of the multi-resolution codebook is better than that of the single-resolution codebook. However, for the $M = 11$ the multi-resolution codebook $\mathcal{M}^{(B)}$ shows considerable degradation in performance as shown in Figures 4 and 5. However, one should note that the feedback loads in this case are 36, 35, 34, 33 instead of 66, 55, 44, 33 for $B = 6, 5, 4, 3$, respectively.

Figure 6 shows the link level throughput results comparing the single-resolution codebook $\mathcal{C}^{(B)}$ with three different designs of multi-resolution codebooks, i.e., $\mathcal{M}^{(B)}$, $\mathcal{N}^{(B)}$, and $\mathcal{O}^{(B)}$. In this case we consider $M = 11$ resource blocks and only present the result for the codebook size $B = 6$. It is seen that although the performance of $\mathcal{M}^{(B)}$ is not so appealing, the performance of the multi-resolution codebook $\mathcal{O}^{(B)}$ is comparable with that of the original single-resolution codebook $\mathcal{C}^{(B)}$. The performance of the other multi-resolution codebook $\mathcal{N}^{(B)}$ is somewhere in between that of $\mathcal{M}^{(B)}$ and $\mathcal{O}^{(B)}$. Thus, by proper design of the multi-resolution codebook, it is possible to achieve a performance close to a single-resolution codebook but with much lower feedback load. In this case, the feedback load has been reduced to 36 from 66 that is a huge saving of 30 bits.

4 Conclusion

When the channel is correlated in time (or in frequency, e.g., across adjacent tones in OFDM systems), the correlation can be used to considerably reduce the feedback requirement without loss in performance. The idea of multi-resolution codebook is to quantize and feedback only

the variation in the channel, since typically quantizing the channel variation requires a smaller codebook than quantizing the channel itself. Moreover, the complexity and memory requirement of such multi-resolution codebook precoding schemes are comparable with those of the conventional single-resolution codebook. Based on the promising initial results presented here, we propose to investigate the use of multi-resolution codebooks in LTE-Advanced.

References

- [1] R1-090743 Nortel, “Differential codebook feedback scheme for LTE-A”, *3GPP TSG RAN WG1 Meeting 56*, Athens, Greece, Feb 9 - 13, 2009.
- [2] R1-091288 Philips, “CSI feedback improvements for LTE-A based on multiple codebooks”, *3GPP TSG RAN WG1 Meeting 56bis*, Seoul, Mar. 2009.
- [3] R1-091287 Philips and Qualcomm, “High-Level Principles for CSI feedback for DL MIMO in LTE-A”, *3GPP TSG RAN WG1 Meeting 56bis*, Seoul, Mar. 2009.
- [4] R1-090866 Qualcomm, “Multiple Description Coding for Spatial Feedback Payload Reduction”, *3GPP TSG RAN WG1 Meeting 56*, Athens, Greece, Feb 9 - 13, 2009.

Parameter	Assumption
Access	OFDM
RF carrier frequency	2.0 GHz
Bandwidth	10.0 MHz
Number of occupied sub-carriers	720
FFT point	1024
Number of antennas at BS	4
Number of antennas at MS	2
Codebook size (in bits)	3,4,5,6 bits for low- resolution, 3 bits for high-resolution
Number of the resource blocks (M)	3,7,11
Channel models	SCM channel model case 1A: 3kmph, L=6
Channel estimation	Ideal channel estimation
MCS	4, 16 and 64 QAM, code rates 1/2, 2/3, 3/4 and 5/6
Quantization codebook	Single-resolution and Multi-resolution

Table 1: Simulation parameters and channel model.

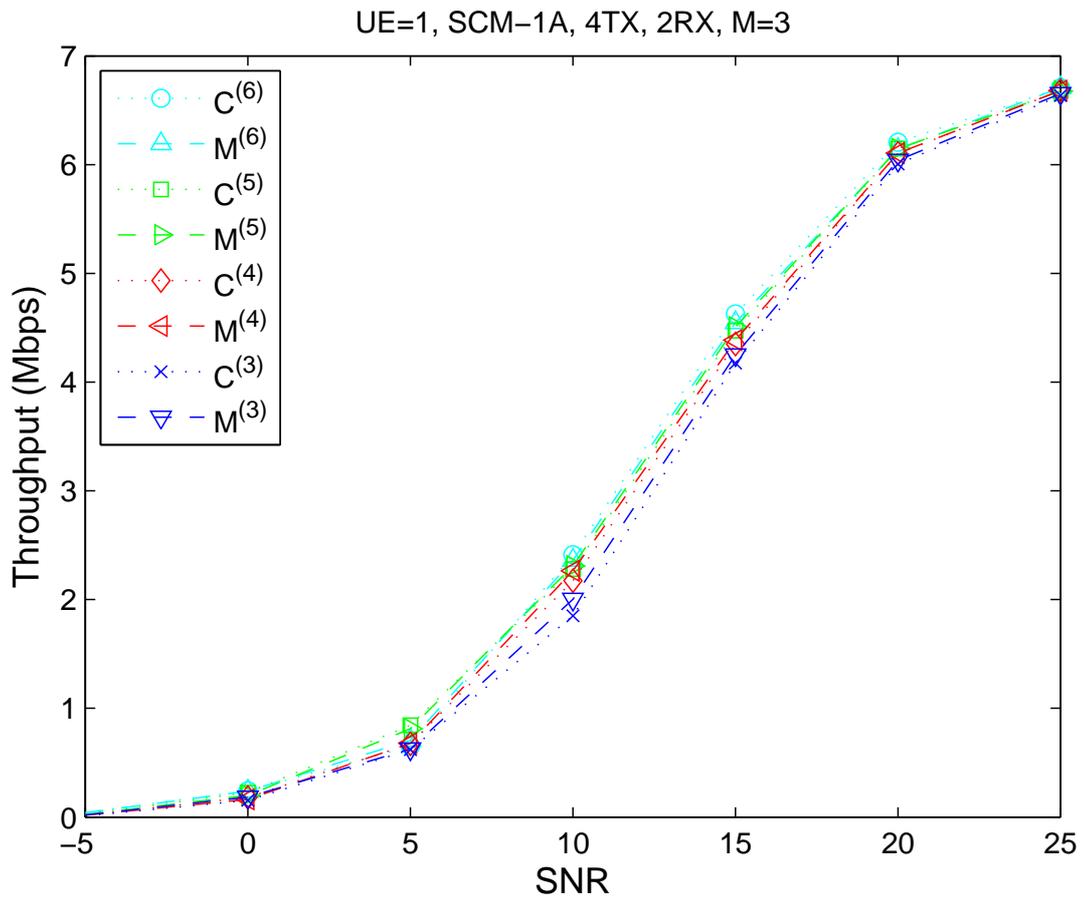


Figure 1: Link level throughput performance over M resource blocks.

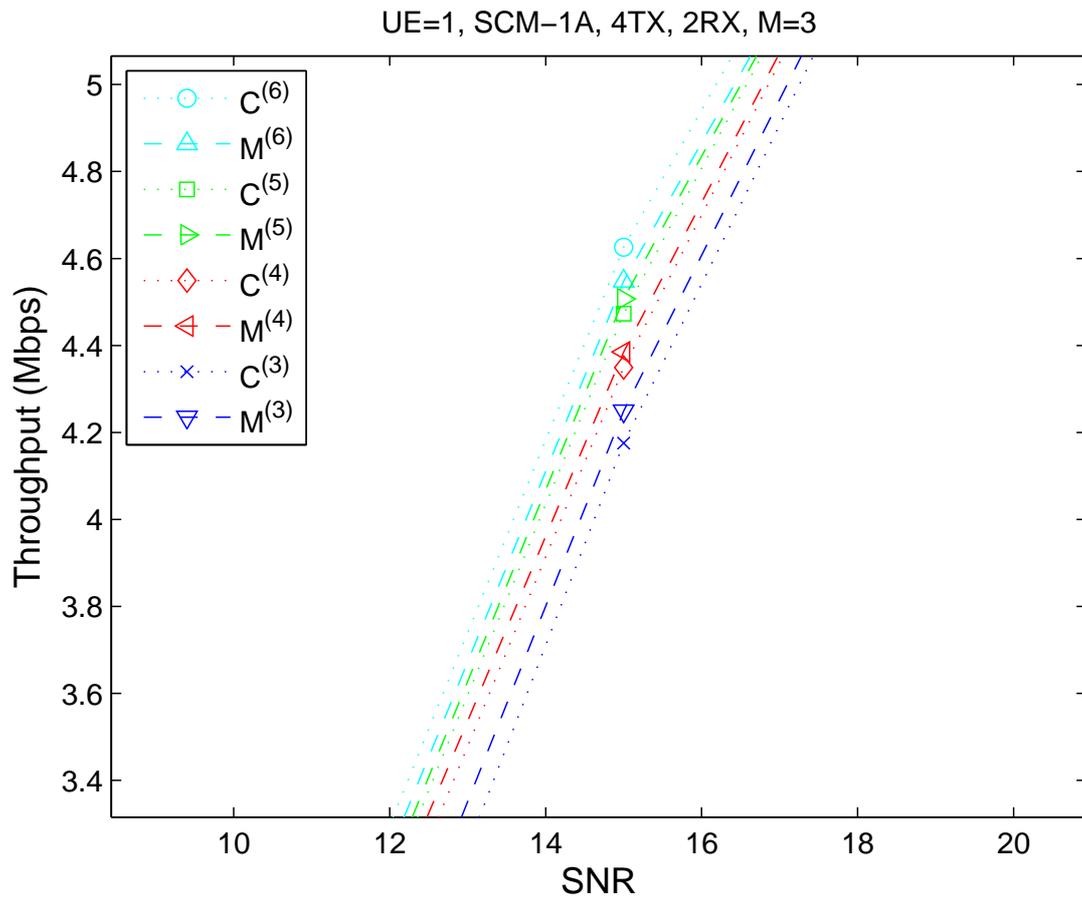


Figure 2: Link level throughput performance over M resource blocks.

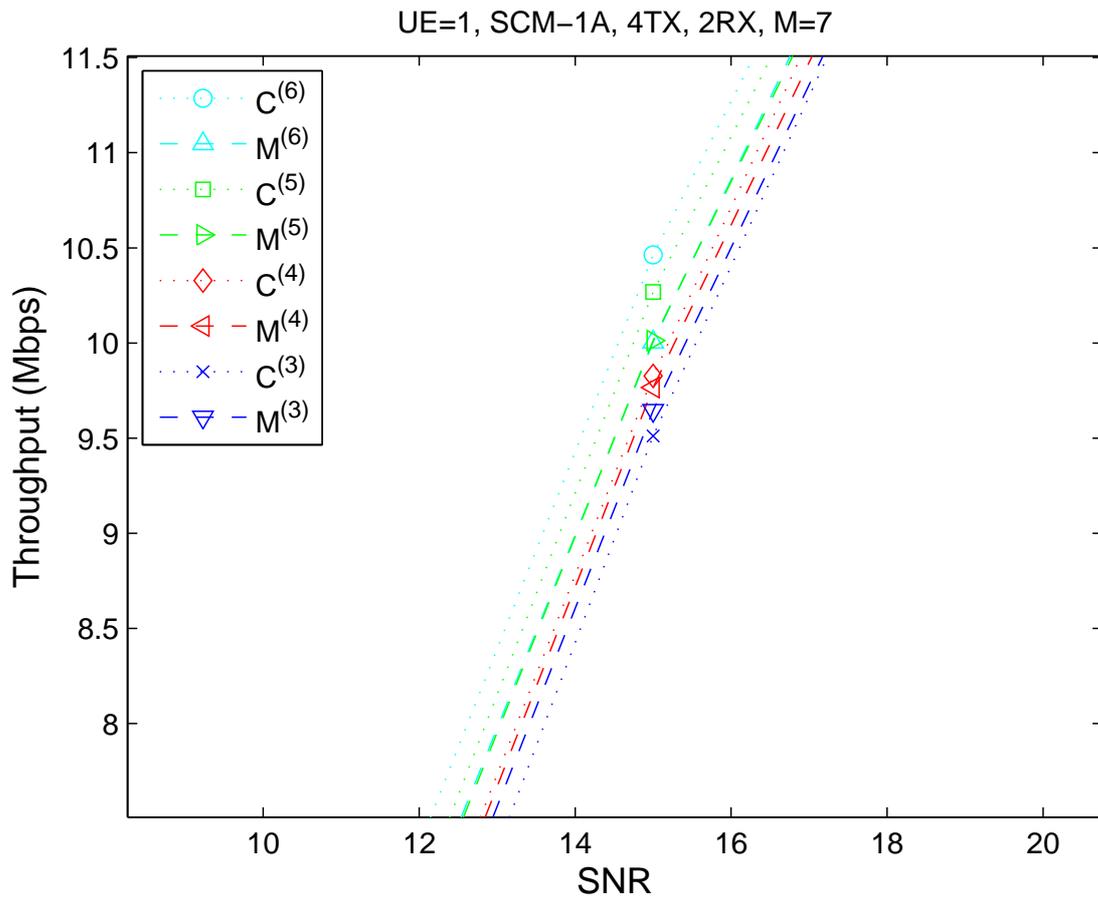


Figure 3: Link level throughput performance over M resource blocks.

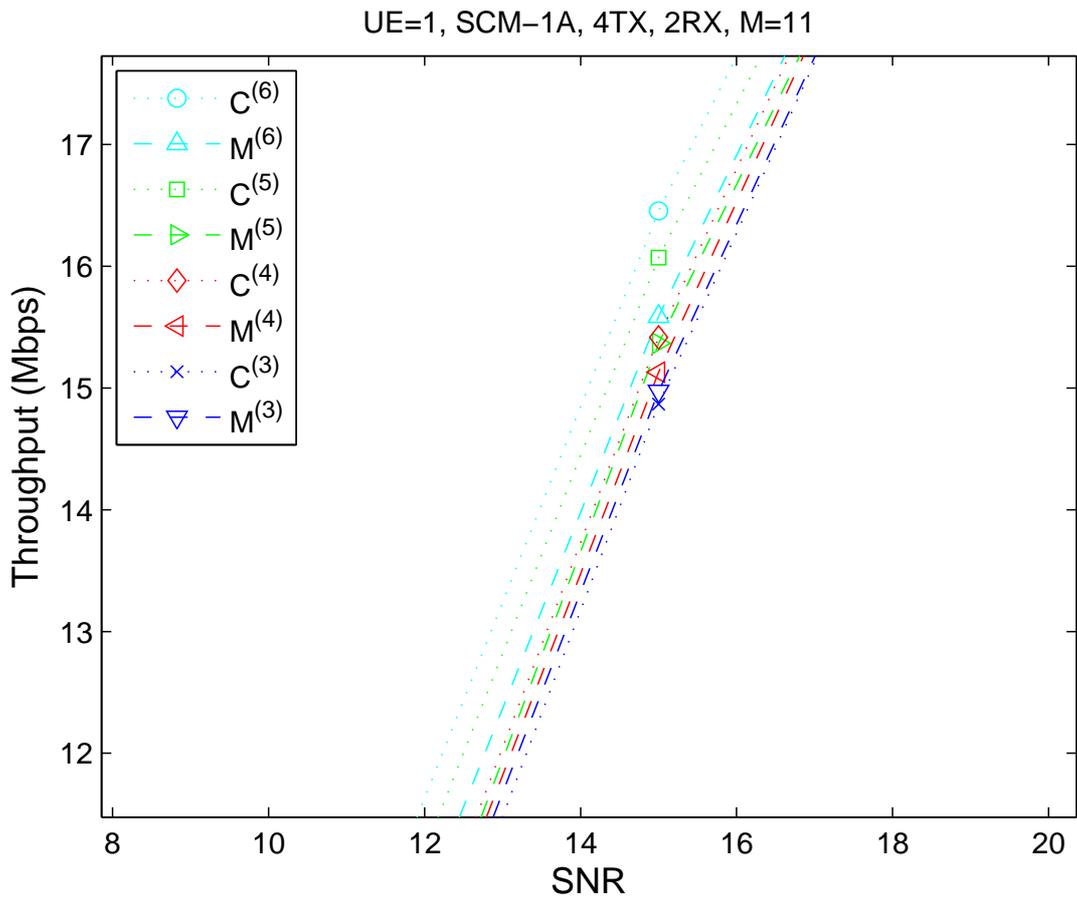


Figure 4: Link level throughput performance over M resource blocks.

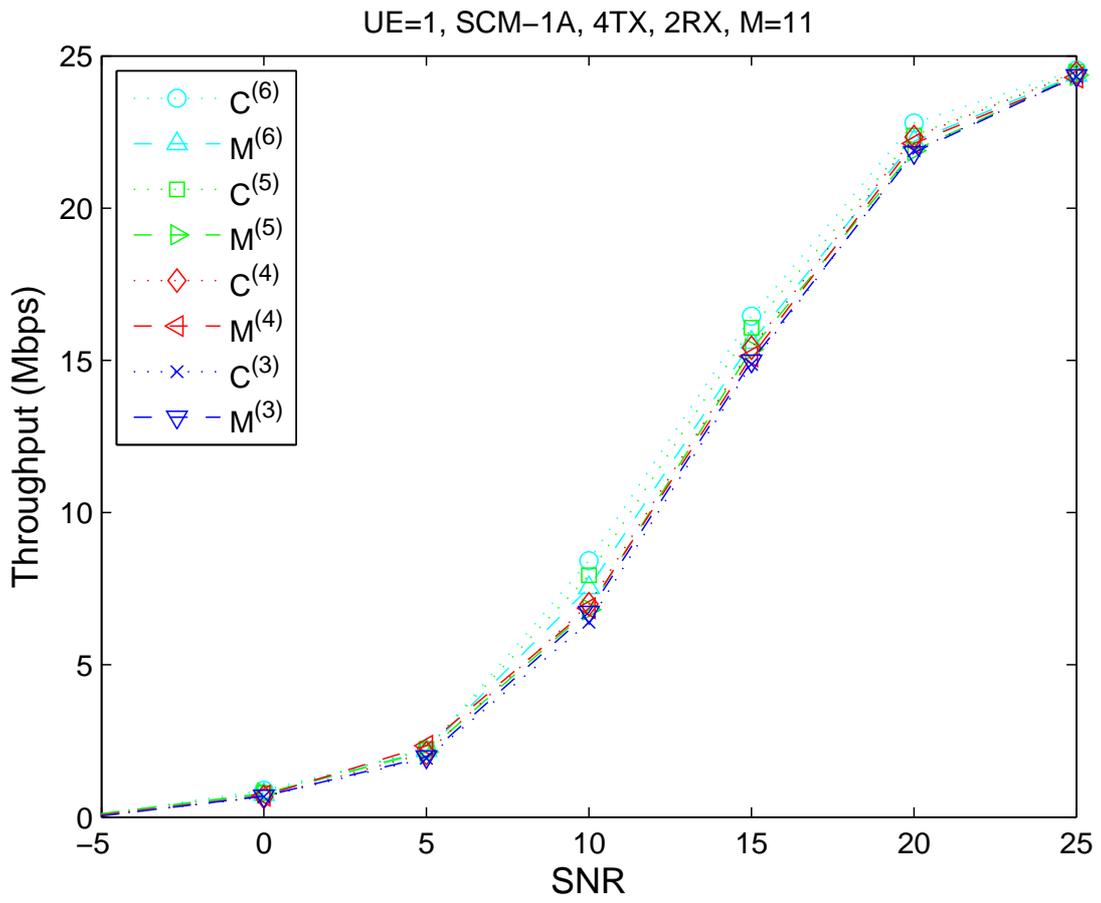


Figure 5: Link level throughput performance over M resource blocks.

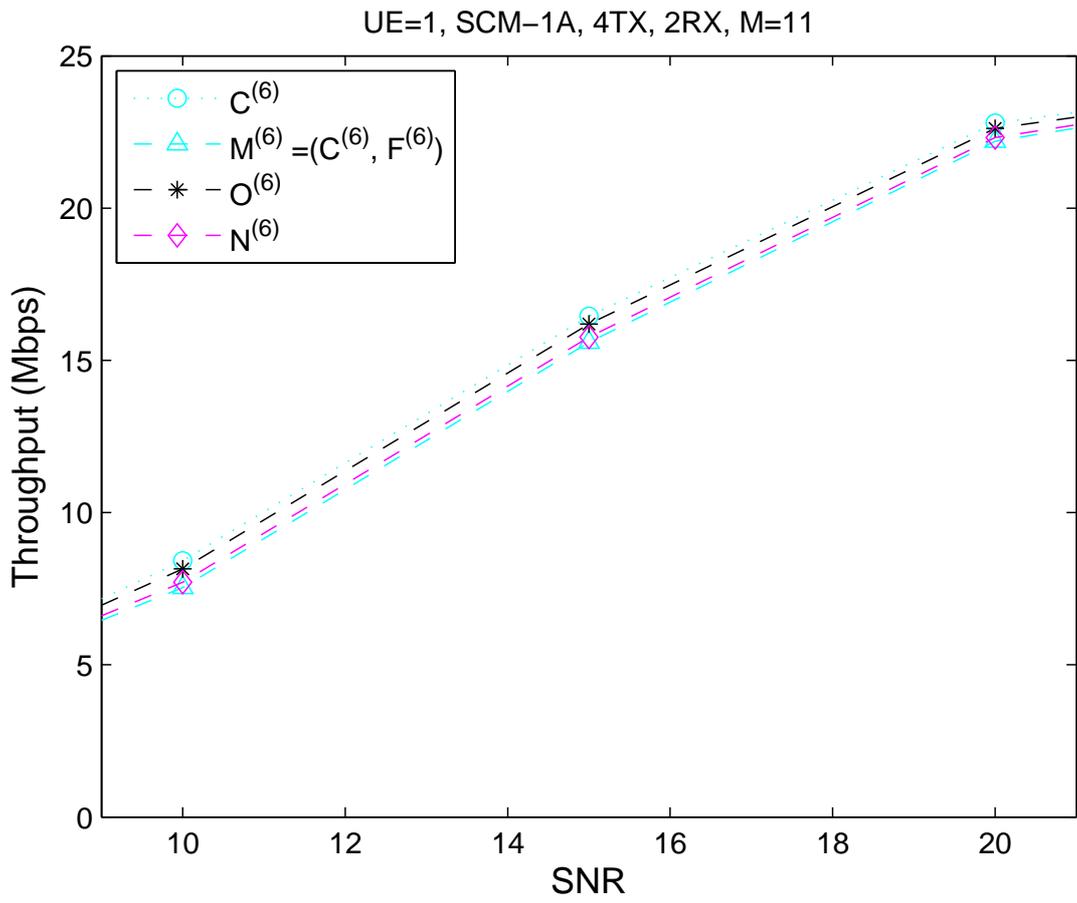


Figure 6: Link level throughput performance over M resource blocks.