

# Block Layering Approach in TAST Codes

Zahoor Ahmed, Jean Pierre Cances, Vahid Meghdadi

Université de Limoges - Ecole Nationale Supérieure d'Ingénieurs de Limoges (ENSIL)  
XLIM-Dept. C2S2, UMR CNRS 6172 16, Rue Atlantis Parc ESTER-BP 6804-87068 Limoges cedex, France  
E-mail: [zahoor.ahmed@ensil.unilim.fr](mailto:zahoor.ahmed@ensil.unilim.fr), [cances@ensil.unilim.fr](mailto:cances@ensil.unilim.fr), [meghdadi@ensil.unilim.fr](mailto:meghdadi@ensil.unilim.fr)  
Received July 26, 2010; revised August 20, 2010; accepted September 21, 2010

## Abstract

Threaded Algebraic Space Time (TAST) codes developed by Gamal *et al.* is a powerful class of space time codes in which different layers are combined and separated by appropriate Diophantine number  $\phi$ . In this paper we introduce a technique of block layering in TAST codes, in which a series of layers (we call it Block layers) has more than one transmit antenna at the same time instant. As a result we use fewer layers (Diophantine numbers) for the four transmit antennas scheme, which enhances the coding gain of our proposed scheme. In each block layer we incorporate Alamouti's transmit diversity scheme which decreases the decoding complexity. The proposed code achieves a normalized rate of 2 symbol/s. Simulation result shows that this type of codes outperforms TAST codes in certain scenarios.

**Keywords:** TAST Code, Block Layer, Space Time Coding

## 1. Introduction

It is well known that wireless communications systems over Rayleigh fading channels can benefit from the simultaneous use of multiple antennas at both the transmitter and receiver to convey information either more reliably or at higher rates than would be possible for single antenna system. The remarkable paper of Alamouti [1] which is considered a benchmark in space time coding, is based on orthogonal design for two transmit antennas offer full diversity and simple linear maximum likelihood (ML) detectors that decouple the transmitted symbols.

Unfortunately, the Hurwitz-radon theorem showed that square complex linear processing orthogonal designs cannot achieve full diversity and full rate simultaneously for more than two antennas. Later on such type of proof has also been shown in [2]. In [3] Jaffarkhani *et al.* has generalized the scheme of orthogonal STBC codes construction for more than two transmit antennas by compromising either the diversity or coding gain. Some researchers have also introduced codes with higher rates and better performances by sacrificing the simplicity of ML decoding and thus orthogonality. In [4] a layering concept in STBC, called vertical Bell Lab layered space-time (V-BLAST), was introduced, but the main drawback of this code was its inflexibility with number of antennas.

Extending the work of layering concept of [4], H. Gamal *et al.* [5] introduced a new architecture in STBC codes,

known as Threaded Space Time (TST) codes. In this architecture, independent codes streams are distributed throughout the transmission resource array in different threads. Of course the efficient separation of individual layers from one another was the primary objectives in the design of such codes. The main drawback of this type of code is the complexity of ML decoder which rises exponentially with number of transmit antennas.

Threaded Algebraic codes [5] based on Diophantine approximation theory and number field were further generalized in [6,7] for arbitrary number of transmit and receive antennas, retaining full rate and maximum diversity. Such types of high rate STBC codes have also been constructed using division algebras [8,9].

In this paper we propose a technique of construction TAST codes within the framework of [6]. The proposed codes are flexible both in term of usage of antennas (at both ends) and Diophantine numbers. We use term AF TAST code for being flexible in term of antennas and DNF TAST code for being flexible in term of Diophantine numbers. As a result the DNF TAST code for four transmit antennas scheme provides higher coding gain and higher code rate retaining maximum diversity as that of original layered codes. This framework is based on TAST code with a slight modification in the definition of layer that in this scheme we may use more than one transmit antenna for transmitting same block or a series of layers

The rest of paper is organized as follow:

A brief review of previous work on TAST code is outlined in Section 2. In Section 3 we present the new approach of flexible TAST code construction in term of antennas' flexibility. In Section 4 we discuss flexible TAST codes in term of Diophantine numbers. The decoding is presented in Section 5 and finally Section 6 presents our conclusion.

### 2. Preliminaries

As our proposed framework is based on the threaded space time architecture [6], so for sake of completeness we review some notation from [6].

A layer in an  $N_T \times T$  (where  $N_T$  denote the number of transmit antennas) transmission resource array is identified by an indexing set  $l \subset I_{N_T} \times I_T$  where  $I_{N_T} = \{1, 2, \dots, N_T\}$  and the  $t$ -th symbol interval on antenna  $a$  belongs to the layer if and only  $(a, t) \in l$ . This indexing set must satisfy the requirement that if  $(a, t) \in l$ , then either  $t \neq t'$  or  $a = a'$  (i.e., that  $a$  is a function of  $t$ ). The definition will be clearer from **Table 1** given below, which depicts a view for four transmit antennas having four layers.

Where for layer  $l$ ,  $l = 1, \dots, n$  of the codeword, the set of matrix entries in positions are given by

$$(t, (l+t-1) \bmod(n)+1), \quad \text{for } k = 1, \dots, n$$

With an arbitrary number of threads, the TAST codes are constructed by transmitting a scaled DAST code [10] in each thread, i.e.,

$$x_l = \phi_l x_l = \phi_l M_l s_l \tag{1}$$

is transmitted over thread  $l_l$ . Where  $x_l$  are encoded symbols,  $M_l$  is an  $N_T \times N_T$  real or complex rotation matrix,  $x_l = M_l s_l$  are rotated complex information symbol vectors and  $\phi_l, l = 1, \dots, L$  are the Diophantine numbers chosen to ensure full diversity and maximize the coding gain of the component codes. In [6]  $\phi_l$  is given by

$$\phi_l = \phi^{(l-1)/N_T} \tag{2}$$

where  $\phi = e^{i\lambda}$  ( $\lambda \neq 0$ ) is an algebraic number

### 3. AF TAST Codes

In some communication systems (for example UMTS), the number of antennas varies among base stations and

**Table 1. Thread distribution.**

|   |   |   |   |
|---|---|---|---|
| 1 | 4 | 3 | 2 |
| 2 | 1 | 4 | 3 |
| 3 | 2 | 1 | 4 |
| 4 | 3 | 2 | 1 |

mobile devices, so it is vital to design a flexible MIMO transmission scheme supporting various multi-element antennas. As a minimum requirement, the mobile station might only be informed about the number of transmit antennas at the base station. Based on its own number of receive antennas, it can then decide which decoding algorithm to apply. Conventional STBC codes offer great complexity in varying the number of receive/transmit antennas. The TAST codes [6] are flexible with respect to number of transmit/receive antennas. In this section we introduce a different and simple technique of flexible ST codes construction which are also flexible with respect to number of transmit/receive antennas and reducing decoding complexity.

We start with basic simple Alamouti code.

$$A_1 = \phi_l \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1 \end{bmatrix} \tag{3}$$

$$A_2 = \phi_l \begin{bmatrix} s_3 & -s_4^* \\ s_4 & s_3 \end{bmatrix} \tag{4}$$

$$A_3 = \phi_l \begin{bmatrix} s_5 & -s_6^* \\ s_6 & s_5 \end{bmatrix} \tag{5}$$

$$A_4 = \phi_l \begin{bmatrix} s_7 & -s_8^* \\ s_8 & s_7 \end{bmatrix} \tag{6}$$

For  $1 \leq l \leq L$  ( $L$  being the numbers of layers)

where  $\phi_l$  is Diophantine number and it is not difficult to verify that taking any one matrix from (3) to (6) results a simple Alamouti codes as we know from (2) that  $\phi_l = 1$ .

As our proposed scheme is flexible with respect to number of transmit and receive antennas, so by simple reshuffle of (3) to (6) we get different structure of TAST codes for different set up of transmit/ receive antennas. Below is a body of a simple program that might be used for this purpose.

Let  $N_T, N_R, L, A$ , denote number of transmit antennas, number of receive antennas, number of layers, and number of Alamouti matrices (given in (3) to (6)), respectively.

*Initialization,  $N_T, N_R$*

*Condition (No. of transmit & receive antenna)*

*Select (value for  $L$  and  $A$ )*

*Process (build TAST codeword matrix with given no. of  $L$ , and  $N_T$ )*

*end*

Note that for all the following structure of codes we

consider Diophantine number  $\phi_l$  same as in (2).

For case of  $N_T = 2$ , (Alamouti code) we simply take any one matrix from ((3) to (6)). For  $N_T = 2$  and  $N_R \geq 2$ , we shall add any two matrices from ((3) to (6)) with a minor manipulation. To save space we avoid going in detail. Likewise for  $N_T = 3$  and  $N_R \geq 2$ , we add any three matrices from ((3) to (6)) with a slight modification. In same way we can develop a code for  $N_T = 4$ . In next section we discuss one of such type of code for  $N_T = 4$  and  $L = 2$ .

### 4. DNF TAST Code

In case of Diophantine numbers flexibility, the case is interesting for  $N_T = N_R = 4$  and  $L = 2$ . Therefore in what follows, we discuss a case for  $N_T = 4$  and  $L = 2$ , and at the end of this section we give the numeral representations for others set up as well.

The necessary condition of layering concept in [5] that the more than one antenna cannot transmit symbols from a given layer at a given time instant has been relaxed. A group of transmit antennas may now belong to a series of layers (for simplicity we call a series of layers as block) for a given symbol period.

A block layer is indexed by a set  $b$ ,  $b \subset b_{N_T} \times b_T$ ,  $b = (w, t) \in \{1, 2, \dots, N_T\}$ . Like TAST [6] and DAST [10] schemes, the idea is to map each block layer to a different subspace so that they are as far away from each other as possible. With the concept of block layers, the total number of layers becomes less and consequently a less number of Diophantine numbers are required which increases the coding gain. Also, real or complex rotated symbols are used to further increase the coding gain. In each block we use Alamouti's transmit diversity scheme that ensures simple decoding at the receiver.

Combining (3) to (6)

$$\begin{bmatrix} \phi_1(A_1) & \phi_2(A_2) \\ \phi_2(A_4) & \phi_1(A_3) \end{bmatrix} \tag{7}$$

Or more precisely

$$\begin{bmatrix} \phi_1 s_1 & -\phi_1 s_2^* & \phi_2 s_3 & -\phi_2 s_4^* \\ \phi_1 s_2 & \phi_1 s_1^* & \phi_2 s_4 & \phi_2 s_3^* \\ \phi_2 s_7 & -\phi_2 s_8^* & \phi_1 s_5 & -\phi_1 s_6^* \\ \phi_2 s_8 & \phi_2 s_7^* & \phi_1 s_6 & \phi_1 s_5^* \end{bmatrix} \tag{8}$$

It is straightforward to verify that the modified representation in (8) has the same property as the original Alamouti code. However, this modified representation clearly falls within the scope of the threaded coding framework.

In (8)  $\phi_1$  and  $\phi_2$  are two Diophantine numbers

and  $[s_1, s_2, \dots, s_8]$  is the rotated information vector to be transmitted.

In matrix form the DNF-TAST code for  $N_T = N_R = 4$  and  $L = 2$  is given in (9) which uses  $2^q$  PSK or QAM signal constellation, and has a rate of  $R = 2q$ . For TAST code we use the notation  $T_{N_T, L, R}$  while for flexible TAST code we use  $\bar{T}_{N_T, L, R}$ , where the subscripts in both cases show the numbers of transmit antennas, number of layers, and symbols per channel use, respectively.

$$\bar{T}_{4,2,4} = \begin{bmatrix} 1 & 1 & 2 & 2 \\ 1 & 1 & 2 & 2 \\ 2 & 2 & 1 & 1 \\ 2 & 2 & 1 & 1 \end{bmatrix} \tag{9}$$

The transmitted symbol  $x_l$  corresponding to source information symbol  $s_l$  over  $l^{th}$  block layer is

$$x_l(s_l) = \phi_l x_l = \phi_l M_l s_l, \quad l = 1, \dots, L$$

where  $L$  represents the total number of block layers and  $x_l = M_l s_l$  are the rotated information symbol vectors. Here  $M_l$  is an  $N_T \times N_T$  real or complex rotation matrix built on an algebraic number field  $Q(\theta)$  with  $\theta$  an algebraic number of degree  $n$ , and the numbers  $\phi_l$ ,  $l = 1, \dots, L$  are the Diophantine numbers. Both for real and complex rotation matrices we use the matrices same as given in [6].

In general, one can use different rotation matrices in different blocks. A general and simple MATLAB program which generate rotation matrix  $M_d$  of any dimension  $d = 2^q$  on a number field  $Q(\cos 2\pi/8d)$  is given in [10].

$$M = \text{sqrt}(2/d) * \cos(\pi / (4*d)) * (4*[1:d]-1) * (2*[1:d]-1); \tag{10}$$

To construct a rotation matrix  $M_d$  of higher dimensions in  $d$  the following recursive approach can be used [11].

$$M_d = \begin{bmatrix} M_{d/2}^1 & -M_{d/2}^2 \\ M_{d/2}^2 & M_{d/2}^1 \end{bmatrix} \tag{11}$$

where  $M_{d/2}^1$  is the optimal real rotation in dimension  $d/2$  and  $M_{d/2}^2$  is an orthogonal transformation in dimension  $d/2$ . The Diophantine approximation intends to achieve full diversity and maximize the coding gain [6]. For a DNF-TAST code with  $L$  layers the Diophantine numbers are chosen same as (2) with  $L$  denoting number of block layers.

For a neat comparison for  $N_T = 4$  and  $L = 2$ , we reproduce the code as given in [8] in (12). It is crystal clear that the performance of the code in (8) is much better

than in (12), as the former contain no zeros in transmission matrix.

$$\mathcal{T}_{4,2,2} = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 2 & 1 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 2 & 1 \end{bmatrix} \quad (12)$$

For  $N_T = 3$  and  $L = 2$ , we can get flexible TAST code by deleting last row and adding last and second last columns in (12).

$$\bar{\mathcal{T}}_{3,2,2} = \begin{bmatrix} 1 & 0 & 2 \\ 2 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix} \quad (13)$$

For  $N_T = 3$  and  $L = 3$ , we can get flexible TAST code by adding third thread on empty layer in (13).

$$\bar{\mathcal{T}}_{3,3,3} = \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 3 \\ 3 & 2 & 1 \end{bmatrix} \quad (14)$$

For  $N_T = 2$  and  $L = 2$ , we get a code by deleting last row and adding last and second last columns in (13).

$$\bar{\mathcal{T}}_{2,2,2} = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \quad (15)$$

### 5. Decoding

For the set up with one and three Diophantine numbers, we can use simple decoding schemes given in [1] and [6] respectively. Here we elaborate decoding scheme for our proposed code in (9).

The received signal can be written as

$$Y = H\bar{\mathcal{T}}_{N_T,L,R} + N \quad (16)$$

where  $H$  is the  $N_R \times N_T$  complex Gaussian random channel matrix with element  $h_{i,j}$ ,  $i = 1, 2, \dots, N_R$  and  $j = 1, 2, \dots, N_T$ , and  $N$  is a complex Gaussian random noise vector.

Let

$$y = \text{vec}(Y^T) \quad (17)$$

arranges the matrix  $Y^T$  in one column vector by stacking its columns one after other, and let

$$y = [y_1, y_2, \dots, y_{N_R N_T}] \quad (18)$$

Simplifying equation (1) and (16), we get

$$y' = \mathbf{H}'\phi\mathbf{M}'u + n \quad (19)$$

where

$$\mathbf{M}' = \begin{bmatrix} \mathbf{M} & \mathbf{A} \\ \mathbf{A} & \mathbf{M} \end{bmatrix} \quad (20)$$

where  $\mathbf{A}$  is a  $0_{4 \times 4}$  matrix,  $\mathbf{M}'$ ,  $\phi'$  and  $\mathbf{H}'$  are respectively rotation matrix, Diophantine matrix and the channel matrix given in (20), (21) and (22), and  $n$  is obtained by converting  $\text{vec}(N^T)$  into column vector by stacking its columns one after other, and  $u$  is a vector carrying source information symbols.

$$\hat{\phi} = \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} \quad (21)$$

where

$$\phi_1 = \begin{bmatrix} \phi_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \phi_1^* & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \phi_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \phi_1^* & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \phi_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \phi_1^* & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \phi_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \phi_1^* & 0 & 0 & 0 & 0 \end{bmatrix}$$

and

$$\phi_2 = \begin{bmatrix} \phi_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \phi_2^* & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \phi_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \phi_2^* & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \phi_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \phi_2^* & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \phi_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \phi_2^* & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{H}' = [h_1 \ h_2] \quad (22)$$

where

$$h_1 = \left( \begin{bmatrix} h_{ij} & 0 & h_{ij+1} & 0 & 0 & 0 & 0 & 0 \\ 0 & h_{ij+1}^* & 0 & -h_{ij}^* & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_{ij+2} & 0 & h_{ij+3} & 0 \\ 0 & 0 & 0 & 0 & 0 & h_{ij+3}^* & 0 & -h_{ij+3}^* \end{bmatrix} \right)_{\substack{i=1,2,3,4 \\ j=1}}$$

and

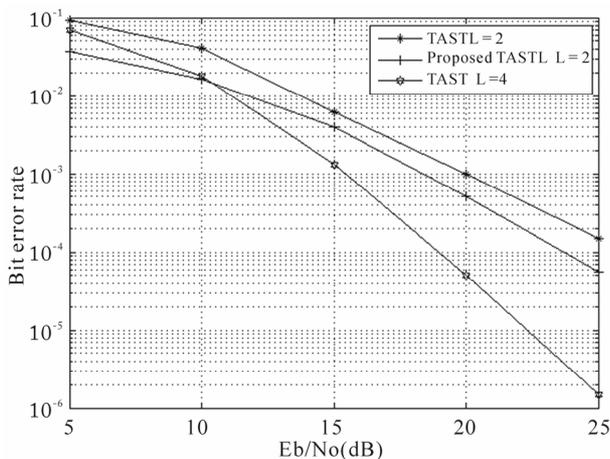
$$h_2 = \left( \begin{bmatrix} h_{ij-1} & 0 & h_{ij} & 0 & 0 & 0 & 0 & 0 \\ 0 & h_{ij}^* & 0 & -h_{ij-1}^* & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_{ij-3} & 0 & h_{ij-2} & 0 \\ 0 & 0 & 0 & 0 & 0 & h_{ij-2}^* & 0 & -h_{ij-3}^* \end{bmatrix} \right)_{\substack{i=1,2,3,4 \\ j=4}}$$

Note that  $h_1$  and  $h_2$  are stacked into column for different values of  $i$ .

The simulation results given in **Figure 1** confirm our mathematical analysis for obtaining better performances of our proposed code in (9) over his brethren codes with  $L = 2$  and 4. When comparing with the code when  $L = 2$  as given in (12), our proposed codes absolutely, however in case when  $L = 4$ , our code outperform at low SNR. Due to the hardware constraint we could not carried out simulation for large no of antennas but we intelligently guess that our code gains better performance over the both codes for large no of antennas.

## 6. Conclusions

TAST codes with different number of transmit/ receive antennas and Diophantine numbers have been proposed. A code having four transmit antennas and two layers is discussed which attains a better performance as compared to same class of code having four layers in certain scenarios. For four receive antennas our proposed code outperform TAST code at low SNR, but for higher SNR TAST code works better. Due to limitation of hardware we could not simulate for higher number of receive antennas but we guess intelligently that increasing the number of receive antennas may enhance the performance of our proposed code. In addition ML decoding is



**Figure 1.** Comparison of different class of TAST codes.

another positive point of our scheme.

## 7. References

- [1] S. M. Alamouti, "A Simple Transmit Diversity Technique for Wireless Communication," *IEEE Journal on Selected Areas in Communications*, Vol. 16, No. 8, 1998, pp. 1451-1458.
- [2] X. -B. Liang and X. -G. Xia, "On the Nonexistence of Rate-One Generalized Complex Orthogonal Designs," *IEEE Transactions on Information Theory*, Vol. 49, No. 11, 2003, pp. 2984-2988.
- [3] V. Tarokh, H. Jafarkhani and A. R. Calderbank, "Space Time Block Codes from Orthogonal Designs," *IEEE Transactions on Information Theory*, Vol. 45 No. 5, 1999, pp. 1456-1467.
- [4] G. J. Foschini, "Layered Space Time Architecture or Wireless Communication in a Fading Environment when Using Multiple Antennas," *Bell Labs Technical Journal*, Vol. 1, No. 2, 1996, pp. 41-59.
- [5] H. Gamal and A. R. Hammon, "A New Approach to Layered Space Time Coding and Signal Processing," *IEEE Transactions on Information Theory*, Vol. 47, No. 6, 2001, pp 2321-2334.
- [6] H. E. Gamal and M. O. Deman, "Universal Space Time Coding," *IEEE Transactions on Information Theory*, Vol. 49, No. 5, 2003, pp. 1097-1119.
- [7] M. O. Damen, A. Tewfik and J.-C. Belfiore, "A Construction of a Space-Time Code Based on Number Theory," *IEEE Transactions on Information Theory*, Vol. 48, No. 3, 2002, pp. 753-760.
- [8] F. Oggier, J. Belfiore and E. Viterbo, "Cyclic Division Algebras: A Tool for Space Time Coding," Boston, Delft, 2007.
- [9] B. A. Sethuraman, B. S. Rajan and V. Shashidhar, "Full-Diversity, High-Rate Space Time Block Codes from Division Algebras," *IEEE Transactions on Information Theory*, Vol. 49, No. 10, 2003, pp. 2596-2616.
- [10] M. O. Damen, K. A. Meraim and J.-C. Belfiore, "Diagonal Algebraic Space-Time Block Codes," *IEEE Transactions on Information Theory*, Vol. 48, No. 3, 2002, pp. 628-636.
- [11] J. Boutros and E. Viterbo, "Signal Space Diversity: A Power and Bandwidth Efficient Diversity Technique for the Rayleigh Fading Channel," *IEEE Transactions on Information Theory*, Vol. 44, No. 4, 1998, pp. 1453-1467.