Comparison of Basis Selection Methods*

J. Adler, B. D. Rao, and K. Kreutz-Delgado
Electrical and Computer Engineering Department
Univ. of California, San Diego, R-007
La Jolla, California 92093-0407

Abstract

In this paper, we describe and evaluate three forward sequential basis selection methods: Basic Matching Pursuit (BMP), Order Recursive Matching Pursuit (ORMP) and Modified Matching Pursuit (MMP), and a parallel basis selection method: the **FOC***al* **U***nderdetermined* **S***ystem* **S***olver (FOCUSS) algorithm. Computer simulations show that the ORMP method is superior to the BMP method in terms of its ability to select a compact basis set. However, it is computationally more complex. The MMP algorithm is developed which is of intermediate computational complexity and has performance comparable to the ORMP method. All the sequential selection methods are shown to have difficulty in environments where the basis set contains highly correlated vectors. The drawback can be traced to the sequential nature of these methods suggesting the need for a parallel basis selection method like FOCUSS. Simulations demonstrate that the FOCUSS algorithm does indeed perform well in such correlated environments. However, a drawback of FOCUSS is that it is computationally more intense then the sequential selection methods.*

1 INTRODUCTION

In this paper, we evaluate several basis selection methods. The problem of best basis selection consists of determining a small, often smallest, subset of vectors chosen from a large redundant set of vectors to match the given data. The importance of this problem is evident from the numerous applications in which it arises, e.g. time/frequency representations [1, 2], biomagnetic inverse problems [3], speech coding [4], bandlimited extrapolation and spectral estimation [5, 6], direction of arrival estimation [7], functional approximation [8, 9, 10], and failure diagnosis [11].

*This work was supported by the National Science Foundation under Grants No. MIP-922055 and IRI-9202581.

Despite the wide application potential for basis selection methods, they have been generally solved or treated with a specific application in mind. In contrast, we attempt to look at these algorithms in a more general context. Towards this end we present some recently proposed basis selection methods, evaluate their effectiveness and complexity, and provide some insight into their pros and cons. In particular we examine four algorithms: Basic Matching Pursuit (BMP) [1, 4], Order Recursive Matching Pursuit (ORMP) [8, 9], a Modified Matching Pursuit (MMP) developed herein, and the **FOC**al **U**nderdetermined **S**ystem **S**olver (FOCUSS) algorithm of [7]. BMP, ORMP, and MMP are forward sequential selection methods, while FOCUSS is a parallel selection method.

2 Problem Formulation

The basis selection problem is as follows. Let $D = \{a_k\}_{k=1}^n$ be a set/dictionary of vectors which is highly redundant, i.e. $a_k \in R^m$ and $m << n$ with $R^m = \text{Span}(D)$. Without loss of generality, it is assumed that the vectors a_k are of unit norm. Given a signal vector $b \in R^m$, and a preset error tolerance, ϵ, the problem is to find the most compact representation of b to within the given tolerance using the basis vectors in the dictionary D. Therefore it involves determining the number r (the *sparsity index*) and the set of vectors $\{a_{k_i}\}_{i=1}^r$ that best model b. Because we are *pursuing* the goal of determining a small subset of vectors in the dictionary D that best *match* the vector b, algorithms that accomplish this goal are often referred to as *matching pursuit* algorithms.

The best basis selection (matching pursuit) problem can be viewed as solving for a sparse solution to a linear system of equations. More precisely, forming a matrix A with columns taken from the dictionary, $A = [a_1, a_2, ..., a_n]$, the problem can be stated as finding an x with a small number of non-zero entries such that $\|Ax - b\| \leq \epsilon$. Formulating the problem in this way

gives new additional insights. Notice that for $\epsilon = 0$, the problem reduces to solving the system $Ax = b$.

Since A has a null space of dimension greater than zero, there are infinitely many solutions to the problem of minimizing $||Ax - b||$. A common approach to obtaining a unique solution is to seek a minimum 2-norm solution [7]. The goal of minimizing the 2-norm favors solutions with many nonzero entries, which is contrary to the goals of basis selection methods [7, 2]. Finding an optimal (smallest) basis set is NP hard and requires an combinatorial search [2, 9]. For example if we were interested in selecting p vectors that best represented the data, this would require searching over the $\binom{n}{p}$ possible ways in which the basis sets can be chosen to find the best solution. The cost of such searches is prohibitive, making finding an optimal solution using an exhaustive search infeasible. Suboptimal methods, such as the ones described in this paper, have been developed to deal with this problem.

In order to gain insight into the comparative behavior of the algorithms discussed in this paper, we will perform a simulation study of their behavior on two test cases, T1 and T2, described as follows:

T1: In this test case, a random $m \times n$ matrix A is created whose entries are each Gaussian random variables with mean zero and variance 1. A sparse solution, x_s, with a specified number of nonzero entries r is then created; the indices of these r entries is random, and their amplitudes are random. The vector b is then computed as $b = Ax_s$. With a known sparse solution, x_s, now at hand to provide a benchmark, the various best basis selection methods are run to select vectors (columns of A) which yield an error tolerance, ϵ, set to zero. The number of vectors chosen are compared with the actual number, r, used to generate the data. The experiment is repeated 100 times, and a histogram of a *redundancy index*, defined as the ratio of the number of distinct columns chosen by the method to the number of columns actually used to generate the data, is plotted. Algorithms with a redundancy index histogram concentrated around 1 indicate a good procedure. The same experiment is repeated with the ideal b being contaminated with noise and a suitable nonzero error tolerance, ϵ, having been chosen. The results of these experiments give insight into the robustness of the various methods.

T2: In this test case, the A matrix is more structured and is chosen to be the rows of the Discrete Fourier Transform (DFT) matrix. The number of columns indicates the resolution in the frequency domain and the number of rows the length of the time series data. This matrix provides the opportunity to evaluate an algorithm when the columns are correlated and there is structure in the dictionary vectors. b is again generated as before by selecting a few columns of A. Special attention is paid to the high resolution case where the frequencies selected are closely spaced, i.e. the indices of the columns of A chosen are near.

3 Sequential Basis Selection Methods

The methods described in this section select the basis vectors sequentially, i.e. the basis set is built up one vector at a time. To facilitate the presentation, we develop some notation. The residual vector after the pth iteration is denoted by b_p, with $b_0 = b$. The indices of the p vectors selected are stored in the index set denoted by I_p, where $I_p = \{k_1, k_2, .., k_p\}$, $I_0 = \emptyset$, and the vectors themselves are stored as columns in the matrix S_p, i.e. $S_p = [a_{k_1}, a_{k_2}, ..., a_{k_p}]$, $S_0 = \emptyset$. The orthogonal projection matrix onto the range space of S_p is denoted by P_{S_p} and its orthogonal complement $P_{S_p}^{\perp} = (I - P_{S_p})$, $P_{S_0} = 0$, $P_{S_0}^{\perp} = I$. The projection matrix onto the space spanned by a vector a_k, $||a_k|| = 1$, is denoted by P_{a_k} where $P_{a_k} = a_k a_k^H$.

3.1 Basic Matching Pursuit (BMP)

This method was suggested in [1] and independently for speech coding [4]. In this basis selection method, in the pth iteration the vector most closely aligned with the residual b_{p-1} is chosen. The computation involved for this selection is

$$k_p = \arg\max_l |a_l^H b_{p-1}|. \quad (1)$$

If $k_p \notin I_{p-1}$, then the index and basis sets are updated, i.e. $I_p = I_{p-1} \cup k_p$, and $S_p = [S_{p-1}, a_{k_p}]$. Otherwise $I_p = I_{p-1}$ and $S_p = S_{p-1}$. The new residual vector is then computed as

$$b_p = P_{a_{k_p}}^{\perp} b_{p-1} = b_{p-1} - (a_{k_p}^H b_{p-1}) a_{k_p}. \quad (2)$$

Equations (1) and (2) give the Basic Matching Pursuit (BMP) algorithm (with $b_0 = b$). The procedure terminates when either $p = r$ (for specified sparsity index r) or $||b_p|| \leq \epsilon$ (for specified ϵ). Note that at each iteration the optimization (1) is over *all* vectors in the dictionary and therefore it is possible to re-select an already selected vector, slowing down convergence [1].

3.2 Order Recursive Matching Pursuit (ORMP)

This method was developed in [8, 9]. In this method, the pursuit of the matching pth basis vector conceptu-

ally involves solving $(n-p+1)$ order recursive least squares problems of the type $\min_y \|[S_{p-1}, a_l]y - b\|$ ([12],page 232), and selecting the vector $a_l \notin S_{p-1}$ that reduces the residual the most. With the notation $S_{p,l} = [S_{p-1}, a_l]$, we have

$$k_p = \arg\min_l \|P^{\perp}_{S_{p,l}} b\|, \; l \notin I_{p-1}, \tag{3}$$

in which case $S_p = [S_{p-1}, a_{k_p}]$ and $b_k = P^{\perp}_{S_p} b$. Note that the projection operator $P_{S_{p,l}}$ can be recursively updated via

$$\begin{aligned} P_{S_{p,l}} &= P_{S_{p-1}} + \frac{1}{\|P^{\perp}_{S_{p-1}} a_l\|^2} P^{\perp}_{S_{p-1}} a_l a_l^H P^{\perp}_{S_{p-1}} \\ &= P_{S_{p-1}} + \frac{a_l^{(p-1)} (a_l^{(p-1)})^H}{\|a_l^{(p-1)}\|^2} \end{aligned}$$

where

$$a_l^{(p)} \equiv P^{\perp}_{S_p} a_l = P^{\perp}_{S_p} a_l^{(p-1)}, \tag{4}$$

using the fact that $P^{\perp}_{S_p} = P^{\perp}_{S_p} P^{\perp}_{S_{p-1}}$ (which also shows that $b_p = P^{\perp}_{S_p} b = P^{\perp}_{S_p} b_{p-1}$). The index selection criteria (3) therefore simplifies to

$$k_p = \arg\max_l \frac{|(a_l^{(p-1)})^H b_{p-1}|}{\|a_l^{(p-1)}\|}, \; l \notin I_{p-1}, \tag{5}$$

resulting in $I_p = I_{p-1} \cup k_p$, $S_p = [S_{p-1}, a_{k_p}]$, and $P_{S_p} = P_{S_p,k_p} = P_{S_{p-1}} + q_p q_p^H$ where

$$q_p \equiv \frac{a_{k_p}^{(p-1)}}{\|a_{k_p}^{(p-1)}\|}. \tag{6}$$

This allows (4) to be expanded as

$$a_l^{(p)} = P^{\perp}_{S_p} a_l^{(p-1)} = a_l^{(p-1)} - (q_p^H a_l^{(p-1)}) q_p. \tag{7}$$

Similarly, the residual vector b_p can be recursively computed as

$$b_p = P^{\perp}_{S_p} b_{p-1} = b_{p-1} - (q_p^H b_{p-1}) q_p. \tag{8}$$

Equations (5)–(8) constitute the ORMP algorithm (with $b_0 = b$, $a_l^{(0)} = a_l$, $l = 1, \ldots, n$). The procedure terminates when either $p = r$ (for specified sparsity index r) or $\|b_p\| \leq \epsilon$ (for specified ϵ). Note that the residual $b_p = P^{\perp}_{S_p} b$ is the orthogonal projection of b onto the orthogonal complement of the range space of S_p, and therefore is the smallest possible error (in the 2-norm sense) when b is to be represented in the span of the columns of S_p. Also note that in (5) the optimization is only over previously unselected dictionary vectors, thereby avoiding the re-selection problem of the BMP.

3.3 Comparison of BMP and ORMP

The results of applying BMP and ORMP to test case T1 in the no noise case are shown in figs. 1 and 2. The parameters are $m = 20$, $n = 30$, and sparsity index $r = 4$ and 8. The ORMP method is clearly superior in performance to the BMP method. However, the computational complexity of ORMP is much higher. In particular, the cost of computing (7) is dependent on the size of the dictionary and can be quite large. On the other hand, the computations involved in BMP method (given by (1) and (2)) are much simpler, providing the incentive to try and improve it.

Potential dificiencies of the performance of BMP relative to ORMP can be traced to the way the dictionary vectors are selected and the residuals are computed in the BMP method,

$$b_p^{BMP} = P^{\perp}_{a_{k_p}} b_{p-1} = \Pi_{l=1}^{p} P^{\perp}_{a_{k_l}} b \neq P^{\perp}_{S_p^{BMP}} b, \; a_{k_l} \in S_p^{BMP}.$$

Note that the sequence of one-dimensional projections defining the BMP residual b_p^{BMP} is *not*, in general, equal to a orthogonal projection onto the orthogonal complement of the range space of S_p^{BMP}. This generally produces a residual at each step larger than $P^{\perp}_{S_p^{BMP}} b$, resulting in the BMP procedure selecting additional columns to reduce it below a specified tolerance level. We have also noted that the BMP algorithm repeatedly optimizes over all dictionary vectors, which can result in re-selecting a column already selected in the past iterations. These aspects of the BMP algorithm are noted and discussed in [1, 4], where a procedure for monitoring and improving performance *a posteriori* is given. Fortunately, a closer examination of the BMP algorithm indicates that the cost of performing the superior $P^{\perp}_{S_p} b$–residual computation on-line is quite minimal, and an efficient way to do this is developed in the next subsection.

3.4 Modified Matching Pursuit (MMP)

Modifying the BMP procedure, we propose that in the pth iteration the index k_p should be selected by finding the vector best aligned with the residual obtained by projecting b onto the orthogonal complement of the range space of S_{p-1},

$$\begin{aligned} k_p &= \arg\max_l |a_l^H P^{\perp}_{S_{p-1}} b| \\ &= \arg\max_l |a_l^H b_{p-1}|, \; l \notin I_{p-1}. \end{aligned} \tag{9}$$

Then $I_p = I_{p-1} \cup k_p$, $S_p = [S_{p-1}, a_{k_p}]$. In contrast to the ORMP basis selection step (5), note that in (9)

there is no need to compute $a_l^{(p-1)} = P_{S_{p-1}}^{\perp} a_l, \forall l \notin I_{p-1}$. As in the ORMP method, quantities in (9) can be computed efficiently using a Modified Gram-Schmidt type of procedure. More precisely, with the initialization $\hat{a}_{k_p}^{(0)} = a_{k_p}, q_0 = 0$, we have $P_{S_p} = P_{S_p,k_p} = P_{S_{p-1}} + q_p q_p^H$ where

$$\begin{aligned} \hat{a}_{k_p}^{(\ell)} &= \hat{a}_{k_p}^{(\ell-1)} - (q_{\ell-1}^H \hat{a}_{k_p}^{(\ell-1)}) q_{\ell-1}, \ \ell = 1, .., p \quad (10) \\ q_p &= \frac{\hat{a}_{k_p}^{(p)}}{\|\hat{a}_{k_p}^{(p)}\|} \end{aligned}$$

The residual b_p is updated via

$$b_p = P_{S_p}^{\perp} b_{p-1} = b_{p-1} - (q_p^H b_{p-1}) q_p. \quad (11)$$

Equations (9)–(11) define the Modified Matching Pursuit (MMP) algorithm. The stopping rules are the same as for ORMP. Note that the burdensome step (7) required by ORMP, of having to project *all* the vectors remaining in the dictionary at each iteration onto $S_p^{\perp}$, has been replaced by the need to project only the optimal vector in (10). Also note that in (9) the optimization is only over previously unselected dictionary vectors, thereby avoiding the re-selection problem of the BMP. Comparison of (1) and (2) with (9)–(11) shows that MMP retains much of the computational simplicity of BMP; the two algorithms differing essentially only by the addition of the projection step (10). We have constructed an algorithm that is intermediate in cost between the BMP and the ORMP, and which should exhibit the benefits of working with the optimal $P_{S_p}^{\perp} b$–residuals and avoiding the vector re-selection problem.

3.5 Comparison of ORMP and MMP

The MMP algorithm is run on test case T1 with no noise and sparsity $r = 4$ and the result shown in fig. 1(c). It shows a significant improvement over BMP, and is comparable to ORMP. Similar results are shown for a sparsity of $r = 7$ (fig. 2) and in the presence of noise (fig. 3). The results suggest that ORMP and MMP are comparable, with ORMP being slightly better.

BMP, MMP, and ORMP are now run on test case T2 with $m = 32$, and $n = 128$. Two closely spaced columns (columns 5 and 9) are chosen. Again ORMP and MMP are comparable in performance. However, a problem with both the methods is that even though they can identify a small set, they do not choose the correct columns. Instead, as shown in fig. 4 for ORMP, they choose columns close to the original ones. The drawback can be traced to the sequential nature of the basis selection process. Such situations indicate the possible utility of nonsequential methods as discussed in the next section.

4 Parallel Basis Selection

We only discuss one such method, the algorithm FOCUSS [7, 5, 6, 13]. Others can be found in [2, 11], and in the pruning literature of neural networks [14]. In this method *all* the vectors of the dictionary are initially selected, and processed and vectors are asymptotically eliminated until a requisite number remain.

4.1 FOCUSS

We consider only the no noise case so that b can be exactly represented by a few columns from the dictionary. Given an initial solution, x_0, the iterations of the basic FOCUSS algorithm are

$$x_{k+1} = W_k (A W_k)^{+} b, \ \text{where} \ W_k = \text{diag}(x_k). \quad (12)$$

"+" denotes the Moore Penrose pseudoinverse. Other variations can be found in [7]. The main intuitive idea behind the method can be grasped by noting that (12) is the solution to the following weighted norm minimization problem

$$\min_x \|W_k^{-1} x\|^2 \ \text{ subject to } \ Ax = b,$$

or, equivalently,

$$\min_q \|q\|^2 \ \text{ subject to } \ AW_k q = b,$$

with $x_{k+1} = W_k q_{k+1}$. The cost function being minimized at iteration $k + 1$is

$$\|q\|^2 = \sum_{l=1}^{n} \left(\frac{x[l]}{x_k[l]} \right)^2.$$

Keeping in mind that this minimization is being carried out under an equality constraint, one can make the following inference. Entries $x_k[l]$ with large magnitude and corresponding to columns that are aligned with b will be enhanced or retained, while entries that correspond to columns that do little to explain b will diminish with every iteration. The potential advantages of this procedure are:

1. The solution is initial condition dependent. This allows prior knowledge to be incorporated into the iterations by a proper choice of initial conditions.

In the absence of prior knowledge, the minimum-norm method is a suitable solution. This can be justified by noting that all solutions are of the form $x = x_{mn} + v$, where x_{mn} is the minimum norm solution, v is a vector orthogonal to x_{mn} and in the null space of A. Experimental studies suggest that choosing $v = 0$ appears to be non-preferential towards any particular solution and a desirable initial choice. The dependence on initial condition is advantageous in that should the algorithm fail to yield a satisfactory sparse solution, one can retry with a new initialization [15]. Such flexibility is beneficial and currently does not exist with the other sequential methods.

2. Problems with the forward sequential selection algorithms BMP, ORMP, and MMP arise when a poor/bad choice is made in the initial stages of the basis vector selection process. There is no mechanism to recover from this problem. FOCUSS does not suffer from this since no final decision is made until the end. Therefore, the method is able to often finding more suitable representations. This is particularly true when the desired basis set involves correlated vectors like in the high resolution frequency estimation problem in test case T2.

Of course there are potential drawbacks, too. A difficulty with the FOCUSS method is that it is computationally more expensive. The computation in (12) can be made efficient [15], but still it is more computationally involved as it involves the entire dictionary D. Further, the mathematical underpinnings of the algorithm are more involved and less familiar. However, we feel that real progress can be made to address all these potential difficulties [15].

4.2 Evaluation of FOCUSS

FOCUSS, initialized by the minimum-norm solution, is tested on test case T1, with sparsity 4 and 7. Initial results are inferior to the MMP and ORMP. However, recent analysis has demonstrated that one can test for and identify the failures and, when a failure is detected, rerun the method with a random initialization until the method succeeds [15]. The performance of FOCUSS with multiple initializations, shown in fig. 2(e), is seen to be clearly superior to the sequential methods. A histogram of the number of initializations is also shown in fig. 2(f). The results of FOCUSS applied to test case T2 are also superior. Fig. 4 shows that unlike ORMP, FOCUSS successfully identifies and isolates the two highly correlated columns. These studies appear to show that FOCUSS is better able to deal with the coupling among basis set vectors.

Acknowledgements

The authors would like to thank Mr. Shane Cotter for his assistance with the simulations.

References

[1] S. G. Mallat and Z. Zhang. "Matching Pursuits with Time-Frequency Dictionaries". *IEEE Trans. ASSP*, 41(12):3397–3415, Dec. 1993.

[2] S. Chen and D. Donoho. "Basis Pursuit". In *Twenty-Eight Asilomar Conference on Signals, Systems and Computers, Vol. I*, pages 41–44, CA, Nov. 1994.

[3] I.F. Gorodnitsky, J.S. George, and B.D. Rao. "Neuromagnetic source imaging with FOCUSS: a recursive weighted minimum norm algorithm". *Journal of Electroencephalography and Clinical Neurophysiology*, 95(4):231–251, Oct. 1995.

[4] A. M. Kondoz. *Digital Speech*. Wiley, 1996.

[5] H. Lee, D. P. Sullivan, and T. H. Huang. "Improvement of discrete band-limited signal extrapolation by iterative subspace modification". In *Proc. ICASSP*, volume 3, pages 1569–1572, Dallas, Texas, April 1987.

[6] S. D. Cabrera and T. W. Parks. "Extrapolation and Spectral Estimation with Iterative weighted norm modification". *IEEE Trans. on ASSP*, April 1991.

[7] I.F. Gorodnitsky and B.D. Rao. "Energy Localization in Reconstructions using FOCUSS: A Recursive weighted norm minimization algorithm". *IEEE Trans. on Signal Processing*, accepted and to appear in 1997.

[8] S. Chen and J. Wigger. "Fast Orthogonal Least Squares Algorithm for Efficent Subset Model Selection". *IEEE Trans. on SP*, July 1995.

[9] B. K. Natarajan. "Sparse Approximate Solutions to Linear Systems". *SIAM Journal on Computing*, 24(2):227–234, April 1995.

[10] R. E. Carlson and B. K. Natarajan. "Sparse Approximate Multiquadric Interpolation". *Computer Math and Applications*, 27(6):99–108, 1994.

[11] P. Duhamel and J. C. Rault. "Automatic Test Generation Techniques for Analog Circuits and Systems: A Review". *IEEE Trans. on CAS*, July 1979.

[12] S. M. Kay. *Fundamentals of Statistical Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1993.

[13] A. A. Ioannides, J. P. R. Bolton, and C. J. S. Clarke. "Continuous Probabilistic Solutions to the Biomagnetic Inverse problem". *Inverse Problems*, 1990.

[14] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Macmillan, New York, 1994.

[15] B. D. Rao. "Analysis and Extensions of the FOCUSS Algorithm". In *Proc. of the 30th Asilomar Conference on Signals, Systems and Computers*, Nov. 1996.

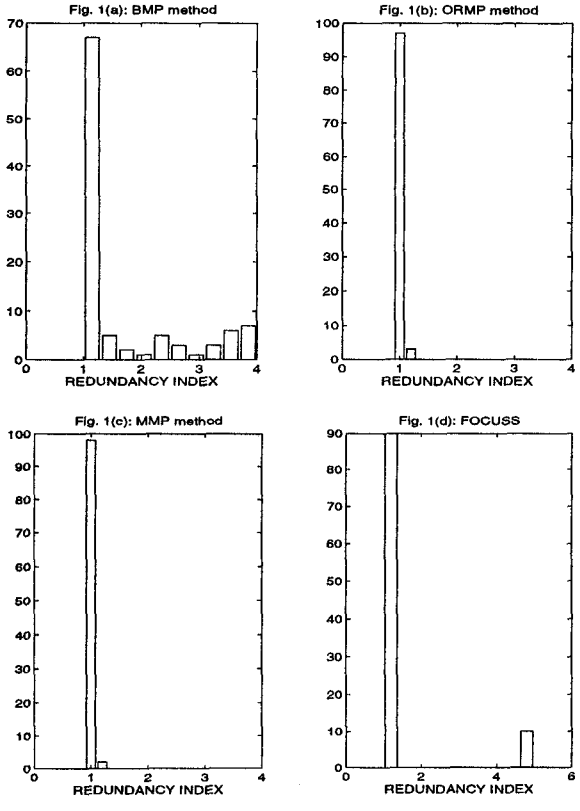
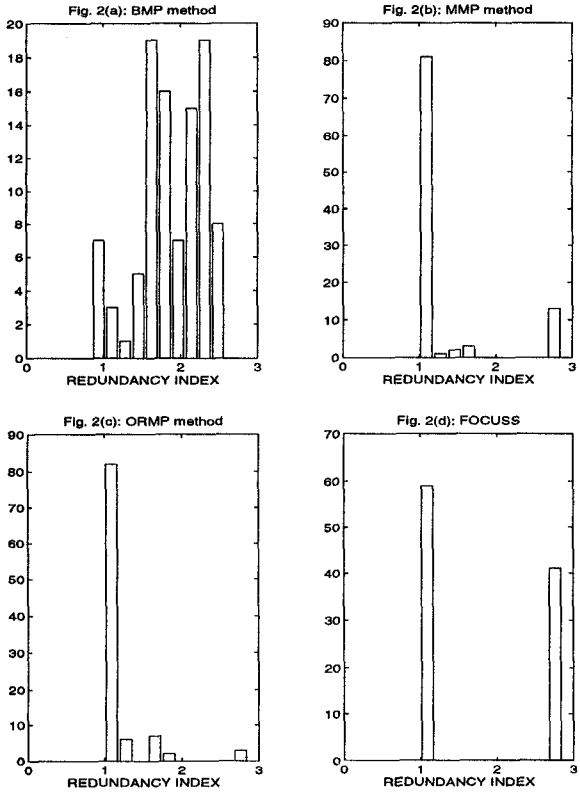


Figure 1: Comparison of BMP, ORMP, MMP, and FOCUSS on data set T1. $m = 20, n = 30, r = 4$, no noise.



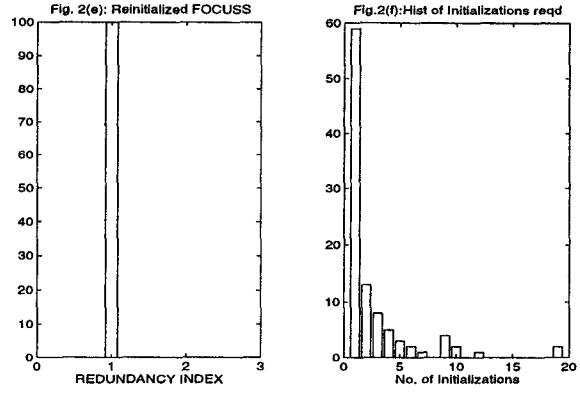


Figure 2: Comparison of MMP, ORMP, FOCUSS and Reinitialized FOCUSS on test data T1 with sparsity $r = 7$ and no noise. $m = 20, n = 30$.

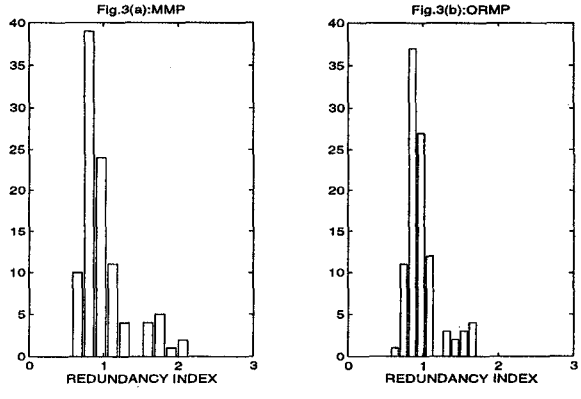


Figure 3: Comparison of MMP and ORMP on data set T1 with sparsity $r = 7$ and SNR of 27db. $m = 20, n = 30$.

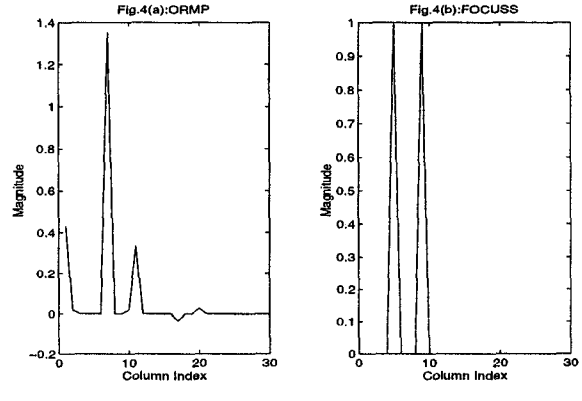


Figure 4: Comparison of ORMP and FOCUSS on sinewave data (T2). Column 5 and 9 with amplitude 1 were used to generate the data, as correctly identified by FOCUSS. $m = 32, n = 128$. Only the first 30 entries are shown as the rest are zero.