

The Viterbi Algorithm for Subset Selection

Shay Maymon and Yonina C. Eldar, *Fellow, IEEE*

Abstract—We study the problem of sparse recovery in an overcomplete dictionary. This problem has attracted considerable attention in signal processing, statistics, and computer science, and a variety of algorithms have been developed to recover the sparse vector. We propose a new method based on the computationally efficient Viterbi algorithm which is shown to achieve better performance than competing algorithms such as Orthogonal Matching Pursuit (OMP), Orthogonal Least-Squares (OLS), Multi-Branch Matching Pursuit (MBMP), Iterative Hard Thresholding (IHT), and l_1 minimization. We also explore the relationship of the Viterbi-based approach with OLS.

I. INTRODUCTION

IN THE mathematics and signal processing communities, subset selection has been studied extensively in the context of sparse approximation. In this context, the problem consists of selecting a sparse subset of atoms from a large dictionary, whose linear combination best approximates a given signal, in the least-squares sense. Approaches for sparse approximation in linear models can be broadly grouped into two categories: those relying on greedy approximation schemes and those minimizing the l_1 norm of the sparse vector.

Greedy algorithms for sparse signal recovery are iterative in nature and select a sparse subset according to their correlation with the measurements. These methods are rooted in the matching pursuit (MP) algorithm for signal approximation [1]. The Orthogonal Matching Pursuit (OMP) algorithm [2], which draws its popularity from its computational simplicity and guaranteed performance, was developed as an improvement to MP. Both algorithms iteratively update the support of the sparse subset by finding the column most correlated to the signal residual. The difference between MP and OMP is in updating the coefficients over the support set after the support is determined.

Multi-Branch Multi Pursuit (MBMP) [3] belongs to the matching pursuit family and possesses a multibranch structure which allows it to trade-off performance for computational complexity. The MBMP algorithm may be visualized as a tree of nodes where the tree depends on the number of levels and on

the number of branches at each level. Other greedy techniques with a similar flavor include Compressive Sampling Matching Pursuit (CoSaMP) [4] and Subspace Pursuit (SP) [5]. Both CoSaMP and SP add as well as remove elements to the active set in each iteration. Orthogonal Least Squares (OLS) methods were developed in [6] to combine structure determination and parameter estimation for a class of multivariable discrete-time non-linear stochastic systems which are linear in the parameters. The atom selection rule in OLS can be viewed as an extension of the OMP rule: the measurement vector and the dictionary atoms are projected onto the subspace orthogonal to the span of the active atoms, and the normalized projected atom that best correlates with the residual is selected. However, OLS is computationally more expensive than OMP. A simpler iterative technique that is based only on thresholding steps is known as Iterative Hard Thresholding (IHT) [7].

In the second category of approaches are optimization-based methods, in which a convex relaxation replaces the intractable l_0 -norm with an l_1 -norm. This translation is referred to in the literature as Basis Pursuit (BP) [8]. There are many results showing that under appropriate conditions on the measurements, the computationally efficient approaches discussed above yield exact recovery of the sparse vector [9]. Furthermore, these methods are known to be robust in the presence of noise.

In this paper we develop a new approach for subset selection, based on the Viterbi algorithm. The Viterbi algorithm was initially proposed in [10] as an efficient method of decoding convolutional codes, and has since been applied to a number of diverse problems, including intersymbol interference [11], continuous-phase frequency-shift keying (FSK) [12], and text recognition [13]. The relationship of the Viterbi-based approach with OLS is also explored. While OLS is a forward selection algorithm, a backward step is needed in the Viterbi approach to determine the optimal subset. It is demonstrated by simulations that the performance of the Viterbi approach is better than competing methods. It should be noted however that it is computationally more expensive than OLS.

The remainder of the paper uses the following notations: upper (lower) boldface letters will be used for matrices (column vectors), $(\cdot)^H$ denotes conjugate-transposition, \mathbf{I} is the identity matrix, \mathbf{P}_A is an orthogonal projection onto the subspace $\mathcal{R}(A)$, and \mathbf{P}_{A^\perp} is an orthogonal projection onto the space orthogonal to $\mathcal{R}(A)$.

The paper is organized as follows: Section II formulates the subset selection problem and develops the associated optimization using orthogonal decompositions. Section III reviews OLS and its connection with OMP. In Section IV we develop the Viterbi subset selection and discuss its connection with OLS. Simulation results are shown in Section V.

Manuscript received June 09, 2014; revised September 10, 2014; accepted September 10, 2014. Date of publication October 01, 2014; date of current version October 24, 2014. This work was supported in part by an Aly Kaufman Fellowship, in part by the Israel Science Foundation under Grant 170/10, in part by the Ollendorf Foundation, in part by the SRC, and in part by the Intel Collaborative Research Institute for Computational Intelligence (ICRI-CI). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jie Liang.

The authors are with the Department of Electrical Engineering, The Technion, Haifa, Israel (e-mail: maymon@technion.ac.il; yoninag@ee.technion.ac.il).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/LSP.2014.2360881

II. SUBSET SELECTION

A. Problem Formulation

We consider the following sparse linear model

$$\mathbf{y} = \mathbf{A}\mathbf{s} + \boldsymbol{\eta} \quad (1)$$

where \mathbf{y} is the measurement vector of length $M \times 1$, \mathbf{s} is a sparse vector of length $N \times 1$ whose support Λ has cardinality $|\Lambda| = K$ with $K < M \ll N$. The random vector $\boldsymbol{\eta}$ is assumed to be Gaussian distributed with zero mean and covariance matrix $\sigma_\eta^2 \mathbf{I}$.

Given the measurement vector \mathbf{y} , we aim to recover the sparse vector \mathbf{s} . Denote by \mathbf{s}_Λ the nonzero components of \mathbf{s} and by \mathbf{A}_Λ a submatrix of \mathbf{A} whose columns correspond to the nonzero entries in \mathbf{s} . We can then formulate the following least-squares optimization for finding Λ and the nonzero elements of \mathbf{s} :

$$\min_{\Lambda, \mathbf{s}_\Lambda} \|\mathbf{y} - \mathbf{A}_\Lambda \mathbf{s}_\Lambda\|^2. \quad (2)$$

Assuming that \mathbf{A}_Λ has full-column rank, minimizing (2) with respect to \mathbf{s}_Λ yields:

$$\mathbf{s}_\Lambda = (\mathbf{A}_\Lambda^H \mathbf{A}_\Lambda)^{-1} \mathbf{A}_\Lambda^H \mathbf{y}. \quad (3)$$

Substituting (3) into (2) leads to the following optimization:

$$\min_{\Lambda} \mathbf{y}^H \mathbf{P}_{\mathbf{A}_\Lambda^\perp} \mathbf{y} \quad (4)$$

where $\mathbf{P}_{\mathbf{A}_\Lambda^\perp} = \mathbf{I} - \mathbf{P}_{\mathbf{A}_\Lambda} = \mathbf{I} - \mathbf{A}_\Lambda (\mathbf{A}_\Lambda^H \mathbf{A}_\Lambda)^{-1} \mathbf{A}_\Lambda^H$ is an orthogonal projection onto $\mathcal{R}^\perp(\mathbf{A}_\Lambda)$. The optimization problem in (4) is challenging because of its combinatorial complexity. In principle, to optimize the objective function, we need to consider $\binom{N}{K}$ different possibilities of sparse subsets of size K from the $M \times N$ dictionary \mathbf{A} .

B. Orthogonal Decomposition

The objective function in (4) depends on the support Λ through the projection matrix $\mathbf{P}_{\mathbf{A}_\Lambda^\perp}$. Since the orthogonal projection is unique, using the columns of \mathbf{A}_Λ to form $\mathbf{P}_{\mathbf{A}_\Lambda}$ is equivalent to using any other set of vectors that spans the same subspace. We next show that by applying an orthogonal decomposition on \mathbf{A}_Λ , the objective in (4) is reduced into a sum of components where each introduces a new additional index to those previously introduced. This fact will be exploited in the following sections in developing sequential algorithms for subset selection.

Let us denote by $\{\mathbf{a}_{i_j}\}_{j=1}^K$ the columns of \mathbf{A}_Λ where $\{\mathbf{a}_i\}_{i=1}^N$ are the columns of \mathbf{A} and $\{i_j\}_{j=1}^K$ represent the support of \mathbf{s} . Gram-Schmidt (GS) orthogonalization can then be used to obtain an orthogonal set $\{\mathbf{u}_{i_j}\}_{j=1}^K$ from the columns of \mathbf{A}_Λ . The GS procedure computes U one column at a time and orthogonalizes \mathbf{A}_Λ as follows: at the j th stage it makes the j th column orthogonal to each of the $j-1$ previously orthogonalized columns by subtracting from the current vector \mathbf{a}_{i_j} the projections in the direction of previous vectors $\{\mathbf{a}_{i_m}\}_{m=1}^{j-1}$. The computational procedure is represented as

$$\mathbf{u}_{i_j} = \mathbf{P}^{(j-1)} \cdot \mathbf{a}_{i_j}, \quad (5)$$

where $\mathbf{P}^{(j-1)} = (\mathbf{I} - \sum_{m=1}^{j-1} \frac{\mathbf{u}_{i_m} \mathbf{u}_{i_m}^H}{\mathbf{u}_{i_m}^H \mathbf{u}_{i_m}})$ and $\mathbf{P}^{(0)} = \mathbf{I}$. The projection $\mathbf{P}_{\mathbf{A}_\Lambda^\perp}$ can be expressed as

$$\mathbf{P}_{\mathbf{A}_\Lambda^\perp} = \mathbf{I} - \sum_{j=1}^K \frac{\mathbf{u}_{i_j} \mathbf{u}_{i_j}^H}{\mathbf{u}_{i_j}^H \mathbf{u}_{i_j}} = \mathbf{I} - \sum_{j=1}^K \frac{\mathbf{P}^{(j-1)} \mathbf{a}_{i_j} \mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)}}{\mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)} \mathbf{a}_{i_j}}. \quad (6)$$

Substituting (6) into (4), the optimization becomes

$$\begin{aligned} & \max_{i_1, \dots, i_K} \sum_{j=1}^K \frac{|\mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)} \mathbf{y}|^2}{\mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)} \mathbf{a}_{i_j}} \\ &= \max_{i_1, \dots, i_K} \left\{ \frac{|\mathbf{a}_{i_1}^H \mathbf{y}|^2}{\mathbf{a}_{i_1}^H \mathbf{a}_{i_1}} + \frac{|\mathbf{a}_{i_2}^H \mathbf{P}^{(1)} \mathbf{y}|^2}{\mathbf{a}_{i_2}^H \mathbf{P}^{(1)} \mathbf{a}_{i_2}} + \dots + \frac{|\mathbf{a}_{i_K}^H \mathbf{P}^{(K-1)} \mathbf{y}|^2}{\mathbf{a}_{i_K}^H \mathbf{P}^{(K-1)} \mathbf{a}_{i_K}} \right\}, \end{aligned} \quad (7)$$

where $\mathbf{P}^{(j)}$ can also be expressed as

$$\begin{aligned} \mathbf{P}^{(j)} &= \mathbf{I} - \sum_{m=1}^j \frac{\mathbf{u}_{i_m} \mathbf{u}_{i_m}^H}{\mathbf{u}_{i_m}^H \mathbf{u}_{i_m}} = \left(\frac{\mathbf{I} - \mathbf{u}_{i_1} \mathbf{u}_{i_1}^H}{\mathbf{u}_{i_1}^H \mathbf{u}_{i_1}} \right) \cdot \dots \cdot \left(\frac{\mathbf{I} - \mathbf{u}_{i_j} \mathbf{u}_{i_j}^H}{\mathbf{u}_{i_j}^H \mathbf{u}_{i_j}} \right) \\ &= \mathbf{P}_{i_1^\perp} \cdot \mathbf{P}_{i_{j-1}^\perp} \cdot \dots \cdot \mathbf{P}_{i_1^\perp}. \end{aligned} \quad (8)$$

Due to the combinatorial complexity of the optimization in (7), it is intractable, in general, to obtain an exact solution. We next discuss sub-optimal approaches whose computational complexity are significantly lower. In Section III we treat OLS and its relation to OMP. Section IV introduces the Viterbi approach for solving the subset selection problem.

III. ORTHOGONAL LEAST-SQUARES

In the first term of the objective function in (7), the index i_1 is the only unknown variable. It can thus be selected based on optimizing this term only. This selection however is not optimal since other terms of the objective function also depend on i_1 through the projection operators. The second term of the objective function depends both on i_1 and on i_2 , but since i_1 has already been selected, its estimate can be substituted in $\mathbf{P}^{(1)}$ and the second term may be optimized with respect to i_2 alone. Similarly, in the j th iteration of the algorithm, an estimate of i_j is obtained by solving

$$\max_{i_j} \frac{|\mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)} \mathbf{y}|^2}{\mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)} \mathbf{a}_{i_j}} \quad (9)$$

where $\mathbf{P}^{(j-1)}$ can be computed according to (8) using estimates of i_1, \dots, i_{j-1} from previous iterations. The algorithm described above is referred to in the literature as OLS [6]. We present it here in a way that highlights its relationship with the Viterbi approach developed in Section IV.

OLS is similar in some ways to OMP [2]. In the j th iteration of the latter i_j is selected to maximize

$$\max_{i_j} \frac{|\mathbf{a}_{i_j}^H \mathbf{r}^{j-1}|^2}{\mathbf{a}_{i_j}^H \mathbf{a}_{i_j}} = \max_{i_j} \frac{|\mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)} \mathbf{y}|^2}{\mathbf{a}_{i_j}^H \mathbf{a}_{i_j}} \quad (10)$$

where $\mathbf{r}^{j-1} = \mathbf{P}^{(j-1)} \mathbf{y}$ is the residual vector after the $(j-1)$ th iteration. According to (9), we first project all columns of \mathbf{A} onto the subspace orthogonal to the subspace spanned by $\{\mathbf{a}_{i_m}\}_{m=1}^{j-1}$, normalize the projected columns, and finally select the one which correlates best with \mathbf{y} . In OMP, on the other hand, the next index is selected by first normalizing the columns of \mathbf{A} ,

and then finding the column which has the best correlation with the residual vector.

To better understand the relation between OLS and OMP, we write (10) as

$$\max_{i_j} \frac{|\mathbf{a}_{i_j}^H / b f P^{(j-1)} \mathbf{y}|^2 / \mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)} \mathbf{a}_{i_j}}{1 + \mathbf{a}_{i_j}^H (\mathbf{I} - \mathbf{P}^{(j-1)}) \mathbf{a}_{i_j} / \mathbf{a}_{i_j}^H \mathbf{P}^{(j-1)} \mathbf{a}_{i_j}}. \quad (11)$$

While the numerator coincides with the objective function of OLS, the denominator depends on the norms of the columns projected onto the subspace spanned by previously selected vectors, i.e., $\{\mathbf{a}_{i_m}\}_{m=1}^{j-1}$. These norms that affect the selection in OMP are irrelevant in OLS. Numerical experiments show that neither of the algorithms is uniformly better than the other but for correlated dictionaries, guaranteed exact recovery may be obtained after fewer iterations for OLS than for OMP [14].

IV. VITERBI SUBSET SELECTION

We now propose a Viterbi-based approach for solving the optimization in (7), which can be viewed as an extension of OLS. While the latter is a forward selection algorithm which optimizes each component of the objective function separately, the former considers the coupling between the various components of the objective function. Therefore, the optimal subset is determined only at the end of the forward path, resulting in a computationally more expensive method. We first review the Viterbi algorithm [10], [15] and then discuss how to apply it for solving the subset selection problem. In Section V we demonstrate by simulations that the Viterbi approach yields better results than OMP, OLS, MBMP, IHT, and l_1 minimization in various cases.

The Viterbi algorithm is a recursive optimal solution to the problem of maximum a-posteriori probability (MAP) estimation of the state sequence of a finite-state discrete-time Markov process observed in memoryless noise. A trellis diagram is often used to represent the Markov process, in which each node corresponds to a distinct state at a given time, and each branch represents a transition to some new state at the next time instant. In general, there are several paths that reach a node in the trellis. The path segment with the shortest length is called the survivor for that node. Each node has one survivor and consequently, the total number of survivors for any time instant is equal to the number of states. The Viterbi algorithm is based on the observation that the shortest complete path must begin with one of these survivors. Therefore, at any given time, we only need to remember the survivors and their lengths. To transition to the next time instant, we extend all survivors by one time unit, compute the lengths of the extended paths, and select survivors among the extended paths, one per node.

Let us now introduce a Viterbi approach for solving the optimization in (7). In the first step of the algorithm, we solve

$$\max_{i_1 \neq i_2} \left(\frac{|\mathbf{a}_{i_1}^H \mathbf{y}|^2}{\mathbf{a}_{i_1}^H \mathbf{a}_{i_1}} + \frac{|\mathbf{a}_{i_2}^H \mathbf{P}^{(1)} \mathbf{y}|^2}{\mathbf{a}_{i_2}^H \mathbf{P}^{(1)} \mathbf{a}_{i_2}} \right), \quad (12)$$

where the maximization is performed for each choice of i_2 . In contrast to OLS, in which only the first term of the objective function in (7) is considered in the optimization with respect to i_1 and the dependence of all other terms on i_1 is ignored, the Viterbi method considers also the second term. Then, denoting by \hat{i}_1 the value of i_1 selected in the first step, which is a function

of i_2 , and substituting it both in $\mathbf{P}^{(1)}$ and $\mathbf{P}^{(2)}$ we obtain the following optimization for i_2 :

$$\max_{i_2 \neq \hat{i}_1, i_3} \left[\frac{|\mathbf{a}_{i_1}^H \mathbf{y}|^2}{\mathbf{a}_{i_1}^H \mathbf{a}_{i_1}} + \frac{|\mathbf{a}_{i_2}^H \mathbf{P}^{(1)} \mathbf{y}|^2}{\mathbf{a}_{i_2}^H \mathbf{P}^{(1)} \mathbf{a}_{i_2}} + \frac{|\mathbf{a}_{i_3}^H \mathbf{P}^{(2)} \mathbf{y}|^2}{\mathbf{a}_{i_3}^H \mathbf{P}^{(2)} \mathbf{a}_{i_3}} \right].$$

Here i_2 is selected to maximize the objective function for each choice of i_3 . We continue similarly where in each iteration the component added to the objective function introduces only one new index. The algorithm terminates when we reach the K th stage in which the optimal value of i_K is obtained and the complete path segment is backtraced. After estimating the support Λ , we finally use it in (3) to recover the sparse vector.

In the objective function of (7), associated with the subset selection problem, the j th component in the sum depends in general on all values i_1 through i_j . This is in contrast to the state sequence estimation problem, in which the Markov property holds. Therefore, the solution to the subset selection problem obtained via Viterbi is not guaranteed to be optimal unless $K \leq 2$. Furthermore, as opposed to the state sequence estimation problem, in which each sequence is in general valid and permutations are considered as different solutions, column repetition is not allowed in the subset selection problem and different permutations are considered equivalent. The effect this has on the algorithm is that it does not consider states which have already been selected.

The computational complexity of the subset selection problem is reduced with the Viterbi algorithm and the problem becomes feasible even for large values of N . In the Viterbi approach, the first stage considers $N \times (N - 1)$ path segments, the second stage examines $N \times (N - 2)$ paths, and the $(K - 1)$ th stage checks $N \times (N - K + 1)$ segments, adding up to a total of $N \times (N - K/2) \times (K - 1)$ path segments. This is in contrast to the optimal solution for which $\binom{N}{K}$ paths need to be treated. Comparing the complexity of the Viterbi approach to OLS, the former is roughly N times more expensive. At each iteration of the Viterbi method, the computations involved in the transition from one node to all candidate nodes at the next stage are roughly equivalent to the computation involved in the corresponding iteration of OLS. The factor N appears since we have to perform these computations for each of the N nodes.

V. ILLUSTRATIVE EXAMPLES & SIMULATIONS

For illustration, we first consider a dictionary consisting of rows $\{1, 4, 6, 8, 9, 11\}$ from a Fourier matrix of size $N = 12$. Given $\mathbf{y} = [3, -1, \sqrt{3} - 1, \sqrt{3} - 1, 0, 2]^T$ and assuming a sparsity level $K = 3$, we aim to recover the sparse vector \mathbf{s} .

Let us begin with the Viterbi approach, which is illustrated in Fig. 1. The dictionary size N and the sparsity level K are equivalent in the state sequence estimation problem to the number of states and the length of the sequence. Thus, the trellis diagram consists of N nodes, one for each column \mathbf{a}_i of the sensing matrix \mathbf{A} , and K stages. We first consider all appropriate transitions from nodes at the first stage to nodes at the second stage, where their costs are computed according to the objective function in (12). Among all transitions reaching the same node at stage 2, only the path whose cost is maximal, i.e., the survivor for this node, is depicted in the figure. In the second step, we

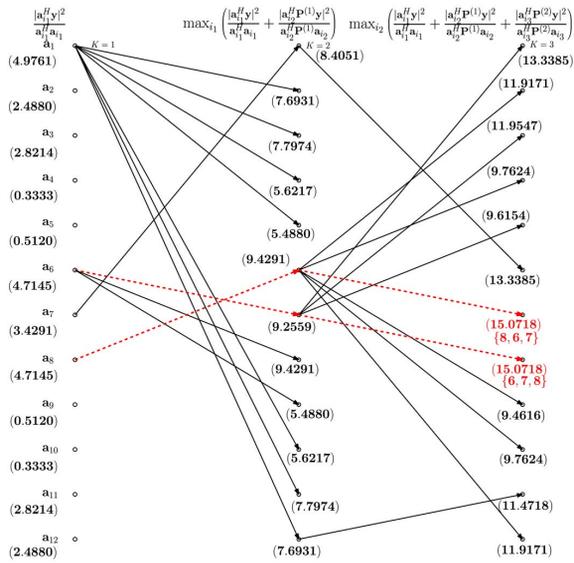


Fig. 1. Illustration of the Viterbi algorithm for a partial Fourier matrix.

consider all appropriate transitions from nodes at the second stage to nodes at the third stage. When computing the cost of the extended path, we take the cost of each of the survivors from the previous stage and add to it the contribution due to the transition to each node at the third stage. Then, we select as the survivor for each node the path from stage 2 whose overall cost is maximal, and represent it with a line. We finally select the node at stage 3 whose overall cost is maximal and determine the optimal complete path, which is highlighted with a dashed line.

Fig. 2. illustrates both OLS and OMP. The first iteration in both approaches is the same, i.e., finding the column that best correlates with the given measurement, which results in a_1 . In the second iteration we compute the objective functions associated with OLS and OMP as given in (9) and (10) for all paths originating in a_1 to all appropriate nodes at the second stage. These values are shown in parentheses near the paths where the left number corresponds to OLS and the right one to OMP. As illustrated, the objective function of both approaches is optimized for a_7 , whose path is highlighted with a dashed line. In the third iteration, we substitute a_1 and a_7 in $\mathbf{P}^{(2)}$ and compute (9) and (10) for all appropriate nodes at the third stage. Columns a_6 and a_8 are equally likely to be selected for OLS whereas a_3 and a_{11} are equally likely to be selected for OMP.

While the Viterbi approach correctly recovers the complete support $\Lambda = \{6, 7, 8\}$ of s and achieves zero MSE, only partial recovery of the support is obtained in OLS (2/3) and in OMP (1/3), whose corresponding MSEs are given by $\|s_{OLS} - s\|^2 = 2.6711$ and $\|s_{OMP} - s\|^2 = 3.3591$.

We now compare the performance of the Viterbi method, OLS, OMP, MBMP, IHT, and l_1 minimization as a function of the sparsity level K . With a total number of branches equal to $N = 64$, two different settings are considered for MBMP: one in which all branches are assigned at the first level and another which equally allocates the branches between all levels. To solve the l_1 optimization we use CVX [16]. We consider two different dictionaries: a real-valued random matrix \mathbf{A} whose entries are drawn independently from a Gaussian distribution with

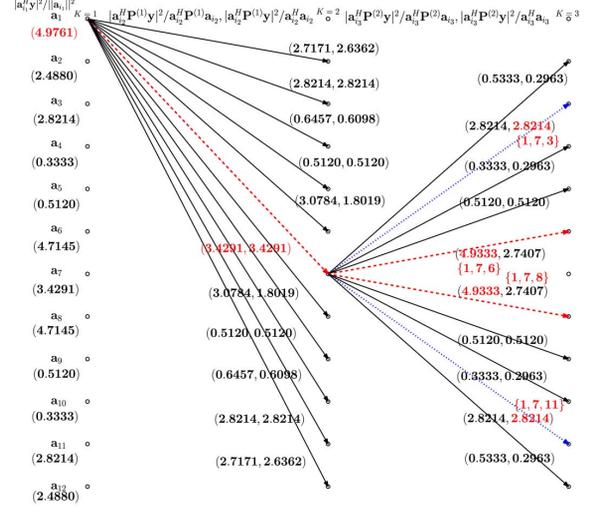
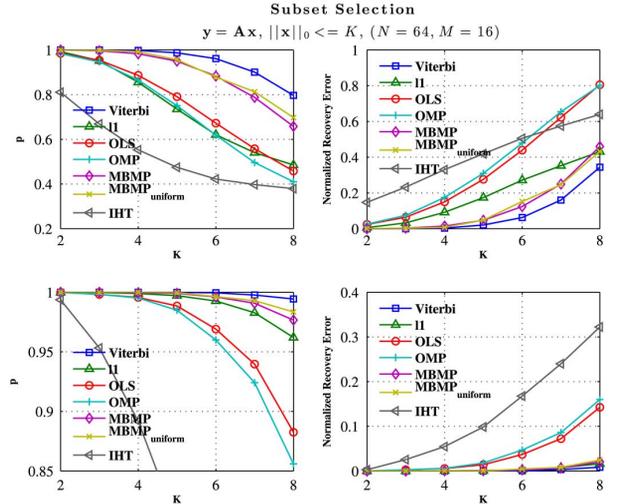


Fig. 2. Illustration of OLS and OMP for a partial Fourier matrix.

Fig. 3. Performance comparison of Viterbi, OLS, OMP, MBMP, IHT, and l_1 minimization as a function of the sparsity level for two choices of \mathbf{A} : iid Gaussian matrix (top), random partial Fourier matrix (bottom).

zero mean and variance one and a random partial Fourier matrix. For each value of $2 \leq K \leq M/2$, we construct a K -sparse vector s of size $N \times 1$. The nonzero locations are drawn uniformly at random and the nonzero values are drawn from a Gaussian distribution with zero mean and unit variance. The experiment is repeated 10,000 times for each sparsity value where we choose $M = 16$ and $N = 64$.

Fig. 3 shows the performance of the reconstruction algorithms. The top corresponds to \mathbf{A} being a Gaussian matrix while \mathbf{A} is a random partial Fourier matrix in the bottom. The figures on the left were obtained by averaging the fraction of correct indices detected at each experiment, whereas the figures on the right represent the normalized recovery error, namely the norm of the error between the true vector and the recovered vector divided by the norm of the true vector. The simulations show that for both dictionaries the Viterbi algorithm outperforms all other methods with respect to both criteria: support detection as well as recovery error.

REFERENCES

- [1] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, pp. 3397–3415, 1993.
- [2] R. Rezaifar, Y. Pati, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, USA, Nov. 1993.
- [3] M. Rossi, A. M. Haimovich, and Y. C. Eldar, "Multi-branch matching pursuit with applications to MIMO radar," *IEEE Trans. Signal Process.*, submitted for publication.
- [4] D. Needell and J. A. Tropp, "Cosamp: Iterative signal recovery from incomplete and cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, 2008.
- [5] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.
- [6] S. A. Billings, S. Chen, and W. Luo, "Orthogonal least squares methods and their application to non-linear system identification," *Int. J. Control*, vol. 50, no. 5, pp. 1873–1896, 1989.
- [7] T. Blumensath and M. Davies, "Gradient pursuits," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2370–2382, 2008.
- [8] D. Donoho, S. Chen, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, pp. 33–61, 1998.
- [9] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
- [10] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theory*, vol. IT-13, pp. 260–269, Apr. 1967.
- [11] G. D. Forney, Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inf. Theory*, vol. IT-18, pp. 363–378, May 1972.
- [12] R. C. Davis, M. G. Pelchat, and M. B. Luntz, "Coherent demodulation of continuous-phase binary fsk signals," in *Proc. Int. Telemetry Conf.*, Washington, DC, USA, 1971.
- [13] D. L. Neuhoff, *The Viterbi Algorithm as an Aid in Text Recognition*. Stanford, CA, USA: Stanford Electronic Labs.
- [14] J. Idier, C. Soussen, R. Gribonval, and C. Herzet, Joint k-step analysis of orthogonal matching pursuit and orthogonal least squares Centre Recherche en Autom. Nancy, Nancy, France, Tech. Rep., 2012.
- [15] G. David Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, Mar. 1973.
- [16] M. Grant and S. Boyd, "Cvx: Matlab software for disciplined convex programming cvx: Matlab software for disciplined convex programming (web page and software)," Mar. 2008.