Sparse Signal Reconstruction from Limited Data Using FOCUSS: A Re-weighted Minimum Norm Algorithm

Irina F. Gorodnitsky, *Member, IEEE*, and Bhaskar D. Rao

***Abstract*— We present a nonparametric algorithm for finding localized energy solutions from limited data. The problem we address is underdetermined, and no prior knowledge of the shape of the region on which the solution is nonzero is assumed. Termed the FOcal Underdetermined System Solver (FOCUSS), the algorithm has two integral parts: a low-resolution initial estimate of the real signal and the iteration process that refines the initial estimate to the final localized energy solution. The iterations are based on weighted norm minimization of the dependent variable with the weights being a function of the preceding iterative solutions. The algorithm is presented as a general estimation tool usable across different applications. A detailed analysis laying the theoretical foundation for the algorithm is given and includes proofs of global and local convergence and a derivation of the rate of convergence. A view of the algorithm as a novel optimization method which combines desirable characteristics of both classical optimization and learning-based algorithms is provided. Mathematical results on conditions for uniqueness of sparse solutions are also given. Applications of the algorithm are illustrated on problems in direction-of-arrival (DOA) estimation and neuromagnetic imaging.**

I. Introduction

THE PROBLEM OF finding localized energy solutions from limited data arises in many applications including spectral estimation, direction-of-arrival estimation (DOA), signal reconstruction, signal classification, and tomography. Limited data can arise from either limited observation time, nonstationarity of the observed processes, instrument constraints, or the ill-posed nature of the problem and, often, from a combination of these factors. To treat this problem mathematically, we define localized energy or sparse signals as signals that are zero everywhere except on a minimal support of the solution space. We assume that no information is available about this support. A detailed discussion of this definition is given later in the paper. Thus, reconstruction of a sparse signal amounts to finding the "best" basis that represents this signal, where "best" can be measured in terms of a norm. This is different, for example, from basis selection in signal compression where the goal is to find a sparse or perhaps a maximally sparse representation of a signal. In this paper, we address the "best" basis selection and develop a nonparametric algorithm for this problem.

Since estimation from limited data is an underdetermined problem, infinitely many solutions exist, and additional criteria must be used to select a single estimate. The sparsity of the solution is the only *a priori* selection criterion available in our problem. As we show in Section III, the sparsity constraint does not define a unique solution but rather narrows it to a finite subset. Hence, the problem remains underdetermined. The non-uniqueness is worse when data represent a single vector sample, such as a single time series or a single snapshot from a sensor array. Some common techniques used to compute sparse signals include exhaustive searches (e.g., greedy algorithms [1], [2]), evolutionary searches (e.g., genetic algorithms with a sparsity constraint [3]), and Bayesian restoration with Gibbs priors [4]. These algorithms do not utilize any additional information about the solution except its sparsity. Thus, their results are not well constrained, and the bases they select are essentially arbitrary with respect to the real signal. Alternatively, l_1-norm and l_p-norm minimization and Linear Programming (LP) methods which produce a solution by optimizing some cost function are also used for sparse signal estimation [5]. Unfortunately, in most signal processing problems, the relationship of the real signal to the cost functions is not known, and these techniques also return an essentially arbitrary solution with respect to the real signal.

Another approach used to find sparse solutions is to compute maximally sparse solutions [6]. In Section III, we show that in general, maximum sparsity is not a suitable constraint for finding sparse signals and derive the conditions under which its use is appropriate. Some of the techniques listed above can generate maximally sparse solutions, and they can be applied in this case. The problem is N-P complete, however, so the high computational cost and, in some cases, compromised convergence, are serious drawbacks of these methods.

For completeness, we want to mention parametric methods for estimating sparse signals. In [7], we show that sparse solutions can be significantly better constrained by multiple data samples, such as multiple snapshots from a sensor array; therefore, parametric techniques based on such data can provide an advantage here. This holds true when the sparseness of the solution allows the parameter space to be sufficiently small and the signal has favorable statistical properties, e.g.,

Manuscript received February 21, 1995; revised September 20, 1996. The first author was supported in part by NSF Grant MIP-922055 and ONR Grant N00014-94-1-0856. The second author was supported in part by NSF Grant MIP-922055. The associate editor coordinating the review of this paper and approving it for publication was Dr. Farokh Marvasti.

I. F. Gorodnitsky is with Cognitive Sciences Department, University of California, La Jolla, CA 92093 USA (e-mail: igorodni@ece.ucsd.edu).

B. D. Rao is with the Electrical and Computer Engineering Department, University of California, La Jolla, CA 92093, USA.

Publisher Item Identifier S 1053-587X(97)01864-3.

stationarity, in which case, parametric techniques provide good resolution. These are not the problems we address here. What we are interested in are the problems in which parametric methods suffer from poor resolution and/or are very difficult to use either due to unfavorable statistical properties of the signal or because an accurate parametric model is not available. The parametric methods also have three general limitations in our view: the nontrivial requirement that an accurate parametric description of the signal and the dimension of the parametric model be supplied *a priori* and the potential for a rapid rise in the number of model parameters with a small increase in the complexity of the signal. In our experience, these limitations may not be easily overcome in problems such as neuroelectromagnetic imaging (EEG/MEG) [8], which motivated the research presented here.

In what follows, we develop a nonparametric algorithm designed to address the shortcomings of the above techniques. Namely, the algorithm provides a relatively inexpensive way to accurately reconstruct sparse signals. Termed FOcal Underdetermined System Solver (FOCUSS), the algorithm consists of two parts. It starts by finding a low resolution estimate of the sparse signal, and then, this solution is pruned to a sparse signal representation. The pruning process is implemented using a generalized Affine Scaling Transformation (AST), which scales the entries of the current solution by those of the solutions of previous iterations. The solution at each iteration then is found by minimizing the l_2-norm of the transformed variable. The low-resolution initial estimate provides the necessary extra constraint to resolve the non-uniqueness of the problem. Low-resolution estimates are available in most applications, and we describe some particular applications of interest. The AST is a powerful procedure, in general, whose potential has yet to be fully realized. It has also been exploited, but with a different optimization objective, in the design of fast interior point methods in LP, including the Karmarkar algorithm, and in minimizing the l_p-norm $(1 < p < 2)$ of the residual error in overdetermined problems [9].

A posteriori constrained extrapolation and interpolation of bandlimited signals has been vigorously studied in the past but mostly in the context of spectral estimation, and many works pertain to the problem where signal bandwidth is known. Papoulis in [10] and Gerchberg in [11] proposed what is known as the Papoulis–Gerchberg (PG) algorithm which, given a continuous signal of known bandwidth on a finite interval of time, iteratively recovered the entire signal. A one-step extrapolation algorithm for this procedure was later suggested in [12]. Jain [13] unified many of the existing bandlimited extrapolation algorithms under the criterion of minimum norm least squares extrapolation and suggested another recursive least squares algorithm. A similar algorithm, with no restrictions on the shape of the sampled region or the bandwidth, was presented in [14]. In [15], Papoulis and Chamzas modified the PG algorithm by truncating the spectrum of the estimate at each iteration to reduce spectral support of the solution in the subsequent iteration. The first use of what is equivalent to the AST was proposed in a spectral estimation context in [16] and [17]. The authors modified the Papoulis–Chamzas algorithm to use the entire solution from a preceding iteration as the weight for the next iteration. The use of this recursive weighting to enhance resolution in harmonic retrieval was studied in [18], [19], and the references therein. A similar iterative procedure was independently proposed in neuroimaging [8], [20]–[22], although the implementation of the recursive constraints was not explicitly exposed in [20]. In [22], Srebro developed an interesting and slightly different implementation of the recursive weighting. The crucial importance of the correct initialization of these procedures was not recognized in any of these, and the suggested algorithms simply amounted to refinement of a minimum 2-norm type initial estimate. The use of different initializations and generalizations of the basic iterations were suggested in [21] and [24]. The use of a more general, non-AST objective function at each iterative step was suggested in [6].

The contributions of this paper are as follows. We present the development of the re-weighted minimum norm algorithm, which incorporates an initialization and a general form of re-weighted iterations, and we provide a comprehensive theoretical foundation for re-weighted minimum norm algorithms, which has not been previously available. We recognize the generality of the method and convert it from the particular frameworks of spectral estimation and neuroimaging into a general signal processing algorithm. We generalize AST-based iterations by introducing two additional parameters. These parameters are necessary to extend the algorithm to a class of optimization techniques usable for a wide range of applications. The work also provides a formulation of the sparse signal estimation problem in a mathematical framework and develops the theory of uniqueness and non-uniqueness of sparse solutions. The paper is organized as follows. In Section II, we provide background material and definitions. In Section III, we present a theory of uniqueness and non-uniqueness of sparse solutions. Section IV contains a description of the FOCUSS algorithm. In Section V, we present global and local convergence analyses and derive the rate of convergence. In Section VI, we discuss implementation issues revealed by the earlier analysis, including the necessary modifications to the early form of the algorithm to make it applicable to a wider range of problems. In Section VII, we provide a view of the algorithm as a computational strategy partway between classical optimization and learning-based neural networks. Applications of FOCUSS to DOA and neuromagnetic imaging problems are presented in Section VIII. Several other applications of FOCUSS can be found in [23], [25], and [26].

The paper focuses on the theoretical foundation of the *a posteriori* constrained algorithm in which we restrict ourselves to a noise-free environment. Issues pertaining to noisy data, such as performance of the algorithm, are not covered here. These issues must be considered in the context of regularization, which is used to stablize inverse calculations and which could not be addressed in the already lengthy paper here, but we provide some references. In this paper, we provide two ways to regularize FOCUSS that use either of the two common regularization techniques—Tikhonov regularization or truncated singular value decomposition—at each iteration. In [27], we provide the sufficient conditions for convergence

of the regularized FOCUSS algorithms. In [8], we demonstrate the successful regularization of FOCUSS and its performance in a noisy environment for the neuromagnetic imaging problem. We also give an example with noisy data in Section VIII. The computational requirements of inverse algorithms and efficient computational algorithms for large-scale problems are investigated in [28].

II. Nonparametric Formulation and Minimum Norm Optimization

We review the nonparametric formulation of a signal estimation problem and the common minimum norm solutions. We work in complex $\mathcal{C}^n$ space with the usual inner product and norms defined. We carry out the development in the discrete domain because it significantly simplifies the presentation and is relevant to most signal processing applications, as most computations are carried out in discrete form. The results are directly extensible to analog (continuous) signals.

Linear extrapolation (reconstruction, estimation, approximation, interpolation) problems can be expressed in the matrix equation form

$$Ax = b \tag{1}$$

where A is the $m \times n, (m < n)$ matrix operator from an unknown signal $x \in \mathcal{C}^n$ to a limited data set $b \in \mathcal{C}^m$. The conditions for the existence of A are given by the Riesz representation theorem. The problem is to find (reconstruct, estimate, approximate, extrapolate) the signal x from its representation b.

We use an example of harmonic retrieval to facilitate the presentation. In this example, each column of A represents a sampled exponential sinusoid of some frequency ω, i.e., $[1, e^{-j\omega}, e^{-j2\omega}, \ldots, e^{-j(m-1)\omega}]^T$. The columns of A are generated by selecting the values of ω from within the range $(-\pi, \pi)$ to sample the frequency axis with the desired density. The values of ω may be non-uniformly spaced and can be chosen to reflect prior knowledge. The data b is a sample of a process consisting of a few sinusoids. The chosen frequencies $\{\omega_i\}$ may not match exactly the harmonics contained in b. We denote the real solution x to be the solution whose nonzero entries pick out the columns of A that are the closest (in the 1-norm) to the harmonics contained in b. Thus, nonparametric estimation of sparse signals can be thought of as a basis selection process for a signal. It is important to note that the sinusoids represented by each column of A must be sampled at the same sampling density as that used to generate b.

Non-uniqueness of solutions to (1) is a well-known problem. The infinite set of solutions can be expressed as $x = x_{mn} + v$, where v is any vector in the null space of A, and x_{mn} is the minimum norm solution, which is defined next.

The minimum norm solution is the most widely used estimate for (1) and is found by assuming the minimum Euclidian or l_2-norm criterion $\min \|x\| = (\sum_{i=1}^{n} x^2)^{1/2}$ on the solution.[1] This solution is unique and is computed as

$$x_{mn} = A^+ b \tag{2}$$

[1]Unless explicitly stated, all $\|\cdot\|$ norms in this paper will refer to the 2-norm.

where $A^+ = A^H(AA^H)^{-1}$ denotes the Moore–Penrose inverse [29]. The solution has a number of computational advantages, but it does not provide sparse solutions. Rather, it has the tendency to spread the energy among a large number of entries of x instead of putting all the energy into just a few entries.

A closely related weighted minimum norm solution, on which FOCUSS iterations are based, is defined as the solution minimizing a weighted norm $\|W^{-1}x\|$, where W is a matrix. It is given by

$$x = W(AW)^+ b. \tag{3}$$

To accommodate singular W, we extend the definition of the weighted minimum norm solution to be the solution minimizing $\|W^+x\|$. By changing W, every possible solution to (1) can be generated. When W is diagonal, the cost objective simply becomes $\|W^+x\| = \sum_{i=1, w_i \neq 0}^{n} (\frac{x_i}{w_i})^2$, where w_i are the diagonal entries of W.

For future discussion, it is useful to restate the definition of a weighted minimum norm solution as follows:

$$\begin{aligned} &\text{find} \quad x = Wq, \\ &\text{where } q : \min \|q\|, \quad \text{subject to } AWq = b. \end{aligned} \tag{4}$$

Note that $\|q\| = \|W^+x\|$, i.e., the optimization objective in (4) is preserved. Without further reliance on such terminology, we note that minimum norm-based solutions (2) and (3) constitute Hilbert space optimization, which guarantees their existence and uniqueness.

The common norm minimization methods for finding sparse solutions are the minimum l_1 norm and the related LP problem. The LP problem in the above notation is stated as follows: $\textit{minimize}\, c^T x : Ax \geq b,\ x \geq 0$, where c is an n-vector representing linear cost parameters. If the set of feasible solutions is nonempty, the fundamental theorem of LP guarantees the existence of a solution to (1) that satisfies the LP criterion and in which the number of nonzero elements does not exceed m.

III. Definition and Conditions on Uniqueness of Sparse Solutions

A solution x that has p nonzero terms lies in a p-dimensional subspace of $\mathcal{C}^n$. For convenience, we will refer to such a solution as a p-dimensional solution, where p can take any value from 1 to n.

A. Definition of a Sparse Solution

To study sparse solutions, we suggest a mathematical definition for these solutions. We propose that sparse solutions be defined as the solutions with m or less nonzero terms. Thus, these solutions form the bases, i.e., the minimal representations for the signal. The mathematical properties of these solutions are distinct from the rest, as can be observed from the uniqueness results derived here. In addition, many optimization algorithms, such as LP, naturally return these types of solutions.

The sparse solutions defined above are obviously not unique. Their total number can range from $\binom{n-1}{m}+1$ to $\binom{n}{m}$, as shown

in Section VI. It may appear that we superficially induce non-uniqueness of sparse solutions by including the m-dimensional solutions since an underdetermined system is guaranteed to have at least $m-1$ artifactual m-dimensional solutions. We show, however, that solutions of dimension $p < m$ can also be non-unique. Hence, the definition of sparsity cannot depend on the uniqueness argument. Rather, the m-dimensional solutions must be included in the definition because they provide valid minimum support representations.

Sparse solutions also arise in LP and l_1-norm minimization problems, and we borrow some useful terminology from that area.

Definition [30]: Given a set of m simultaneous linear equations in n unknowns (1), let B be any nonsingular $m \times m$ submatrix made up of columns of A. Then, if all $n-m$ components of x not associated with the columns of B are set equal to zero, the solution to the resulting set of equations is said to be a *basic solution* to (1), with respect to the basis B. The components of x associated with the columns of B are called *basic variables*. If one or more of the basic variables in a *basic solution* has value zero, the solution is said to be a *degenerate basic solution*.

The sparse solutions then are equivalently the basic and degenerate basic solutions. We also refer to the basic and degenerate basic solutions as low-dimensional solutions and to the rest as high dimensional solutions.

B. Uniqueness Conditions for Sparse Solutions

The following uniqueness/non-uniqueness results are derived for systems satisfying the following property.

Unique Representation Property (URP): A system (1) is said to have the URP if any m columns of A are linearly independent.

The URP basically guarantees that every basis component of the real signal is uniquely represented by a column of A. In many problems, the URP can be achieved by using a sufficiently dense sampling rate to create b that unambiguously captures all of the components of the real signal. This density does not need to correspond to the Nyquist frequency used in spectral estimation, as explained below. In other problems, such as physical tomography problems, the URP can never be satisfied. An example of such a problem is the extrapolation of electric currents inside a volume conductor from externally measured electromagnetic fields. Even when the sampling set is completely dense, i.e., the field is completely known everywhere outside the conducting volume, the current inside the volume cannot be uniquely found [31]. Given such intrinsic ill-posedness, sparse solutions, including the maximally sparse solutions, are never unique. However, depending on the physics, the net effect of the intrinsic ill-posedness on the uniqueness of sparse solutions may be limited and must be considered in the context of an individual problem. For example, in the case of the electromagnetic extrapolation problem, its effect is limited to the uncertainty in the neighborhood of each real solution point [7]. How this affects the uniqueness results for sparse solutions is discussed in Section VIII, when we present an example of the neuroimaging problem.

The following theorem gives bounds on dimensions of unique degenerate basic solutions.

Theorem 1: Given a linear system (1) satisfying the URP, which has a $p < m/2$-dimensional solution, there can be no other solution with dimension less than $r = m - p + 1$. A 1-dimensional solution is the unique degenerate basic solution for a given system.

Proof: Suppose two solutions x_1 and x_2 to (1) exist with corresponding dimensions $p < m/2$ and $r < m-p+1$. Then, these solutions satisfy the systems $A_1 x_1 = b$ and $A_2 x_2 = b$, respectively, where A_1 and A_2 consist of p and r columns of A for which the corresponding entries of x_1 and x_2 are nonzero. Hence, $A_1 x_1 = A_2 x_2$, which contradicts the assumption of linear independence of the columns of A. When $p = 1$, we get $r \not< m$; hence, the degenerate basic solution is unique. □

The following two corollaries establish conditions for the uniqueness of maximally sparse solutions.

Corollary 1: A linear system satisfying the URP can have at most one solution of dimension less than $m/2$. This solution is the maximally sparse solution.

Proof: The results follow readily from Theorem 1.

Corollary 2: For systems satisfying the URP, the real signal can always be found as the unique maximally sparse solution when the number of data samples m exceeds the signal dimension by a factor of 2. In this case, if a solution with dimension less than $m/2$ is found, it is guaranteed to represent the real signal. The sampling of the measurement signal does not need to be uniform.

Proof: The result follows readily from Theorem 1 and Corollary 1.

Corollary 2 is a generalization of the Bandpass Filtering Theorem used in spectral estimation that is derived from the Sampling Theorem [32]. The Bandpass Filtering Theorem states that the length of a sampling region twice the bandwidth of a real signal is sufficient to recover this signal. This is different from the condition on the density of the sampling set governed by the Nyquist frequency criterion. The sampling density in our results is specified by the URP and can be significantly lower than the Nyquest frequency. For example, in spectral estimation, the sampling rate equal to the highest frequency contained in the signal is quite sufficient to satisfy the URP.

The preceding results show that the maximum sparsity constraint is not always appropriate for estimating sparse signals. We use the following simple example to reinforce this point.

Example 1: The system

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} x = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

has two equally likely maximally sparse solutions: $x_1 = [1 \;\; 1 \;\; 0 \;\; 0]^t$ and $x_2 = [0 \;\; 0 \;\; 1 \;\; 1]^t$. Both solutions are the degenerate basic solutions of dimension $2 < m$. Obviously, the maximally sparse condition does not define a unique solution in this example, and its relation to the real signal is not defined.

To summarize, general sparse solutions, including ones with less than m nonzero terms, are non-unique. The constraints

that do lead to a unique solution are either the maximum sparsity constraint or the requirement that the solution has "less than $m/2$ nonzero terms." These provide valid optimization criteria for finding sparse signals when the condition of Corollary 2 holds. Note that the "less than $m/2$ nonzero terms" requirement may be cheaper to implement for some search methods than the maximum sparsity constraint. As we will later show, the FOCUSS algorithm can be set up to favor the maximally sparse solution, i.e., to converge to this solution from within a large set of starting points when the dimension of this solution is small relative to the size of b. As the dimension of this solution increases, FOCUSS gradually starts to favor solutions nearest its initialization. Thus, FOCUSS provides a smooth transition between the two desired convergence properties: One is convergence to the maximally sparse solution when the condition of Corollary 2 holds, and the other is convergence to a sparse solution near the initialization when Corollary 2 is not satisfied.

We would like to make a few further comments on the application of the uniqueness results.

Multiple Samples of Data: The above results assume a linear model (1), where the vector b represents a single sample from some data distribution. Such a vector b can be a time series/autocorrelation or a single snapshot from a sensor array. In [7], we have shown stronger uniqueness results when b is composed of multiple data samples, namely, we have shown that sparse solutions of dimensions less than m are unique, provided that the sources are not completely correlated. It is therefore most advantageous to use multiple samples, for example, multiple snapshots from an array of sensors, whenever possible. In addition, the uniqueness result for multiple samples of data is far less sensitive to the presence of noise in the data than the result for the single sample b (see below).

Effects of Noise: Regularized solutions that are used when data is noisy provide only an approximate fit to the data, where the amount of misfit is dependent on signal-to-noise ratio (SNR). In this case, the columns of A that are nearly colinear to the columns that form the basis for the real signal can become equally likely solution candidates. This injects extra uncertainty into the estimator x. Therefore, the effect of noise may weaken the above uniqueness results, but to what extent depends on the SNR and the angular distances between the columns of A, i.e., the condition number of A.

IV. The FOCUSS Algorithm

In this section, we describe the FOCUSS algorithm. We first describe what we call the basic form of the algorithm, which represents a simple implementation of the re-weighted minimum norm idea. The iterative part of this algorithm has appeared in earlier literature for neuroimaging and spectral estimation problems [16], [17], [21]. The basic form captures the main characteristic of the procedure, and we use it here to provide an intuitive explanation as to how the algorithm works. We then discuss more general forms of the algorithm. The initialization of FOCUSS is discussed following the analysis section.

At the basis of the basic FOCUSS algorithm lies the AST

$$q = X_{k-1}^{+} x \tag{5}$$

where $X_{k-1} = \text{diag}(x_{k-1})$ with x_{k-1} being the solution from the previous iteration. Throughout the paper, we use k in a subscript to denote the current iteration step. With this transformation, an optimization problem in x becomes an optimization problem in q. The basic FOCUSS algorithm uses the AST to construct the weighted minimum norm constraint (7) by setting $W_{p_k} = X_{k-1}$, where W_{p_k} denotes the *a posteriori* weight in each iterative step.

A. The Basic FOCUSS Algorithm

The basic form of the FOCUSS algorithm is

$$\begin{aligned} &\text{Step 1:} \quad W_{p_k} = (\text{diag}(x_{k-1})) \\ &\text{Step 2:} \quad q_k = (AW_{p_k})^{+} b \\ &\text{Step 3:} \quad x_k = W_{p_k} q_k. \end{aligned} \tag{6}$$

Since entries that are zero at initialization remain zero for all iterations, we assume without loss of generality that the number of nonzero components in an initial vector, which defines the dimension of the problem, is always n. The final solutions produced by the algorithm will be denoted by $\bar{x}$ to differentiate them from the all feasible solutions x to the linear system (1).

Steps 2 and 3 of the algorithm together represent the weighted minimum norm computation (3). The algorithm is written in three steps solely for exposition purposes. In the implementation, all the steps can be combined into one.

To understand how the AST constraint leads to pruning of the solution space, we consider the objective minimized at each step

$$\|W^{+}x\|^2 = \|q\|^2 = \sum_{i=1, w_i \neq 0}^{n} \left(\frac{x_i}{w_i}\right)^2. \tag{7}$$

The relatively large entries in W reduce the contribution of the corresponding elements of x to the cost (7), and vice versa. Thus, larger entries in x_{k-1} result in larger corresponding entries in x_k if the respective columns in A are significant in fitting b as compared to the rest of the columns of A. By starting with some feasible approximate solution to (1), minimization of (7) gradually reinforces some of the already prominent entries in x while suppressing the rest until they reach machine precision and become zeros.[2] The algorithm stops when a minimal set of the columns of A that describe b is obtained. Note that the algorithm does not simply increase the largest entries in the initial x_0. In fact, the largest entries in x_0 can become zeros in the final $\bar{x}$. Note also that (7) is never explicitly evaluated in (6). The weights $w_i = 0$ and the corresponding subspaces are eliminated from the computation through the product $A \cdot W_{p_k}$.

While the entries of x_k converge to zero and nonzero values, the corresponding entries in q_k converge to zeros or ones,

[2] Theoretically, the elements of a solution asymptotically converge to zeros but never reach zeros. In finite precision, the asymptotically diminishing elements become zeros.

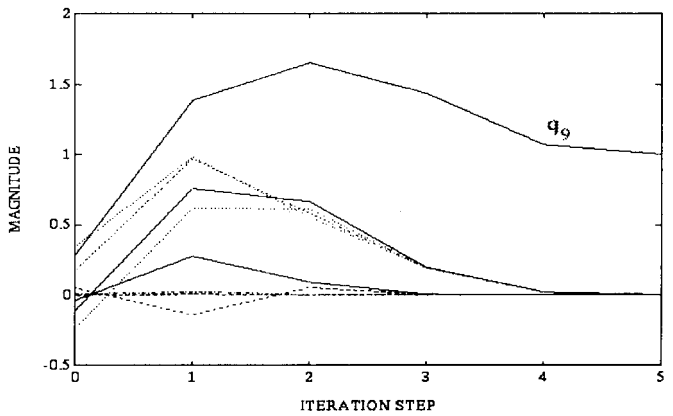


Fig. 1. Elements of q at each iteration for the 10×4 example described in the text.

i.e., $q_k(i) \rightarrow 0$ as $x_k(i) \rightarrow 0$, and $q_k(j) \rightarrow 1$ as $x_k(j)$ approach nonzero values. Fig. 1 illustrates the convergence of the elements of q_k. The example uses a 4×10 matrix A and the vector b equal to the ninth column of A. The correct sparse solution then is a vector of zeros with a one in the ninth entry. Each line in Fig. 1 shows an element of the vector q_k as a function of the iteration index k from initialization to the *fifth* iteration. The ninth element $q_k(9)$ converges to *one*, whereas the rest become *zero*. This indicates that the desired sparse solution x^* with only one nonzero element in the ninth position was found. The minimum norm solution (2) was used for the initialization. Note that the ninth element was not the largest in the initialization.

From our experience, the pattern of change in q emerges after few iterations, from which it is possible to identify the entries converging to ones and zeros. Significant savings in computation, as well as better convergence and performance properties, are gained by eliminating the diminishing entries of x that are indicated by q at each iteration. Further savings can be achieved by implementing a hard thresholding operation to obtain the final result once the convergence pattern becomes clear. Although for the purposes of the analysis we do not explicitly include these truncation operations in the algorithm, they should always be an integral part of FOCUSS implementation.

B. General FOCUSS

We extend basic FOCUSS into a class of recursively constrained optimization algorithms by introducing two parameters. In the first extension, we allow the entries of x_{k-1} to be raised to some power l, as shown in (8). The second extension is the use of an additional weight matrix—denoted W_{a_k}—which is independent of the *a posteriori* constraints. This extension makes the algorithm flexible enough to be used in many different applications. It also provides a way to input *a priori* information. The general form of the algorithm then is

$$\begin{aligned} W_{p_k} &= \text{diag}(x_{k-1}^l), \quad l \in \mathcal{I}_+ \\ q_k &= (AW_{a_k}W_{p_k})^+ b \\ x_k &= W_{a_k}W_{p_k}q_k \end{aligned} \tag{8}$$

where $\mathcal{I}_+$ denotes the set of all positive integers. For the uses of the algorithm considered here, it is sufficient to assume W_{a_k} to be constant for all iterations.

In the applications where the positivity constraint $x_i > 0$ is imposed, we can expand the range of l to all real $l > 0.5$. The lower bound on l is explained in Section V-A. The positivity constraint on x can be enforced by incorporating the step size $p_k = x_k - x_{k-1}$ into the algorithm, as is done in many LP methods. The iterative solution then becomes $\hat{x}_k = x_{k-1} + \alpha p_k$, where the step size α is chosen to keep all entries of $\hat{x}_k$ positive.

More generally, other nondecreasing functions of x_{k-1} can be used to define the weights in (8), although the need for more complicated weight functions is not evident for the applications we have considered.

A *cumulative* form of the FOCUSS algorithm can be derived by using *cumulative a posteriori* weights in (8) that are a function of more than one iteration, e.g., $W_{p_k} = \text{diag}(\prod_{i=1}^{k-1} x_i^{l_i})$. This form may prove to be more robust in terms of convergence to solutions near the initialization, as was found to be the case for the neuromagnetic imaging problem. The convergence analysis of general FOCUSS (8), which is presented next, is extensible to the cumulative form of the algorithm.

V. Analysis

We concentrate our analysis on the form (8) of FOCUSS, unless indicated otherwise. The results are extensible to the other forms. Since W_{a_k} is constant for all iterations, we assume that $W_{a_k} = I$ without affecting the results of the analysis.

The steps of the FOCUSS algorithm always exist and are unique since the transformation (8) is a one-to-one mapping. We next consider the global behavior of the algorithm. For an algorithm to be a useful estimation tool, it must converge to point solutions from all or at least a significant number of initialization states and not exhibit other nonlinear system behaviors, such as divergence or oscillation. *Global convergence analysis* is used to investigate this behavior. The term *global convergence*, however, is sometimes used to imply convergence to a global minimum, which is not the appropriate meaning here. To avoid confusion, we use the term *fixed point convergence* or *absolute convergence* to describe the convergence properties of the algorithm. These terms mean that an algorithm converges to a point solution from any starting condition. The term *absolute stability* has also been used for this property.

Global convergence analysis is not sufficient to understand the complete behavior of even an absolutely convergent nonlinear algorithm. Typically, not all the convergence points form a valid solution set, but this cannot be revealed by the global convergence analysis alone. This point is sometimes overlooked. Here, we add local convergence to our analysis to characterize the different convergence points. We first provide some background in nonlinear systems to motivate our analysis steps. This material is a compilation from several sources. For references, see, for example, [33] and the references therein.

A phase space is a collection of trajectories that trace the temporal evolution of a nonlinear algorithm from different initial points. The points at which a nonlinear algorithm is stationary are called *fixed points*. These can be *stable fixed points (s-f-ps)*, to which the algorithm converges from anywhere within some closed neighborhood around such a point, or *saddle fixed points*, to which the algorithm converges only along some special trajectories. The third type, known as *unstable fixed points*, are stationary points from which an algorithm moves away given any perturbation. The largest neighborhood of points from which an algorithm converges to a given s-f-p is called the basin of attraction of that s-f-p. For a fixed-point convergent algorithm, its entire solution space is divided up by the basins of attraction containing s-f-ps. The borders separating individual basins do not belong to any of the basins. These borders can be made up of trajectories leading to saddle points or to infinity, or they can be a dense set of unstable fixed points, or they can be a combination of the two. Thus, it is important to recognize that an absolutely convergent algorithm does not converge to an s-f-p from any initialization point. It can converge to a saddle point or it may get stuck at an unstable fixed point. Because the saddle points are reached only along special trajectories whose total number has measure zero, an algorithm converges to these solutions with probability 0 (w.p. 0). The unstable fixed points also have measure zero; therefore, an algorithm returns these points w.p. 0. It follows then that an absolutely stable algorithm converges to s-f-ps w.p. 1. Ideally, the s-f-ps of such an algorithm would form the set of valid solutions, such as the sparse solutions in our case, and the other fixed points would be outside of the solution set. The unlikely case of an algorithm becoming fixed in a saddle or unstable fixed point can be resolved by choosing a different initialization state.

Equivalent global convergence theorems exist in two independent fields. In Nonlinear Programming (NP), the theory is based on the analysis of the general theory of algorithms, which was developed mainly by Zangwill. In nonlinear dynamical systems, the Lyapunov stability theory was developed by others based on Lyapunov's first and second theorems. We use elements of both theories in our analysis, as follows. We first define the solution set to contain all the fixed points of FOCUSS and use the global convergence theorem from NP to show that FOCUSS is absolutely convergent. We then use local convergence analysis to determine the nature of the individual fixed points. We show that the sparse solutions are the s-f-ps of FOCUSS and the non-sparse solutions are the saddle points. The rate of local convergence is shown to be at least $2l$. Local analysis of saddle points is difficult and we use nonlinear dynamical system theory concepts for this part of the work.

A. Global Convergence

Theorem 2: The FOCUSS algorithm (8) is absolutely convergent, i.e., for any starting point x_0, it converges asymptotically to a fixed point. The descent function associated with the algorithm is $L(x) = \prod_{i=1}^{n} |x_k(i)|$. The set of fixed points Γ of the algorithm are solutions $\bar{x}$ to $Ax = b$ that have one or more zero entries.

Proof: See Appendix.

Convergence of FOCUSS for $l < 1$ is discussed in Section V-C. The absolute convergence of FOCUSS means that it produces a point solution from any initial condition, but this point can be either a stable, a saddle, or an unstable fixed point. We next determine which solutions of FOCUSS correspond to which fixed states.

B. Analysis of Fixed Points

1) Sparse Solutions: The following theorem shows that the sparse FOCUSS solutions are the s-f-ps of the algorithm.

Theorem 3: Let x^* denote a sparse solution to (1). For any x^*, there exists a neighborhood Ω around it such that for any $x_0 \in \Omega$, the FOCUSS generated sequence $\{x_k\}_{k=0}^{\infty}$ converges to x^*. The local rate of convergence is at least quadratic for the basic algorithm and at least $2l$ for the general class of algorithms (8).

Proof: See Appendix.

Note that the number of sparse FOCUSS solutions is limited to, at most, one solution per each $\mathcal{C}^r$ $(r \leq m)$ subspace of $\mathcal{C}^n$. What is left is to determine the nature of the non-sparse solutions, which we show correspond to saddle and unstable fixed points.

2) Non-sparse Solutions:

Corollary 3: Non-sparse FOCUSS solutions in $\mathcal{C}^r$, $m < r < n$, are its saddle points. Convergence to these points is along special trajectories on which groupings of two or more elements of x do not change relative to each other. A fixed point in $\mathcal{C}^n$ is the unstable fixed point of FOCUSS.

Proof: See Appendix.

From the proof of Corollary 3, it follows that the set of saddle fixed points of (8) is not dense. Since a nonlinear system converges to a saddle or unstable fixed point w.p. 0, the probability of FOCUSS converging to a non-sparse solution is also 0.

C. Relationship to Newton's Method and Cost Functions Associated with FOCUSS

In a broad sense, quadratic minimization of the AST generated cost functions is a *Newton's method* because it replaces a global optimization problem by a series of local quadratic optimization steps. In fact, as shown below, FOCUSS is equivalent to a modified Newton's method minimizing a concave cost function.

Theorem 4: An iterative step from the current state x to the new state x_k, $s_k = x_k - x$, of the FOCUSS algorithm (8) is equal to a step λp_k, with $\lambda = -\frac{1}{2l-1}$ of the modified Newton's method minimizing the function

$$C(x,l) = \begin{cases} \sum_{i=1, x_i \neq 0}^{n} \ln |x_i|, & l = 1 \\ -\frac{2l-1}{2l-2} \sum_{i=1, x_i \neq 0}^{n} \frac{1}{x_i^{2l-2}}, & \begin{cases} l > .5,\ l \neq 1,\ x_i > 0 \\ l = \text{integer}, \quad x_i \in \mathcal{C}^n \end{cases} \end{cases} \quad (9)$$

subject to $Ax = b$. The modification can be viewed equivalently as using a modified Hessian of $C(x,l)$, in which the signs of its negative eigenvalues are reversed, and the positive scaling $\lambda = \frac{1}{2l-1}$. Further, the modified Newton search criteria for constrained minimization of $C(y,l) : y = \lambda x$ is

equivalent to the constrained weighted minimum norm criteria $\min \|W_{p_k}^{+} x_k\|^2$ of the algorithm. For the basic FOCUSS algorithm, $\lambda = 1$.

Proof: See [26].

FOCUSS finds a local minimum of $C(x,l) \cap Ax = b$. The initialization determines the valley of $C(x,l)$ minimized by FOCUSS. The valleys of $C(x,l)$ then define the basins of attraction of the algorithm. The parameter l and *a priori* weights shape these valleys and influence the outcome of the algorithm.

The cost function is useful in understanding the behavior of FOCUSS. It can be used to show that basic FOCUSS always converges to the minimum of the valley of $C(x,l)$ in which it starts, whereas general FOCUSS can move away and converge to the minimum of another valley [26]. We can also show that if we constrain the entries of x to not change their signs throughout all iterations, we have $C(x_k,l) < C(x_{k-1},l)$, i.e., FOCUSS is convergent to the local minimum, for any $l > 0.5$ [26].

The breakdown of convergence for $l \leq 0.5$ can also be observed from $C(x,l)$. When $l = 0.5$, $C(x,l)$ is the 1-norm of x. Since quadratic approximation to a linear function is not defined, FOCUSS steps are also not defined, and $l = 0.5$ in (8) produces no change in x_k for $k > 1$. For $l < 0.5$, $C(x,l)$ is piecewise convex; therefore, FOCUSS steps maximize the local cost, which leads first to a sparse solution followed by an oscillation cycle between two sparse points.

Although we do not emphasize the following use of the algorithm here, FOCUSS also offers simple and relatively inexpensive means of global costs optimization when a good approximate solution is already known. Examples of such use are LP problems, in which solutions may change only slightly in day-to-day operations. If initialized sufficiently near the solution, only one or two iterations may be needed to identify the convergence pattern and, thus, the solution. Using $l \geq 2$ in (8) and efficient implementations of the inverse operation [28] can further speed up the convergence.

VI. Implementational Issues

Here, we discuss factors pertaining to implementation of the re-weighted minimum norm algorithms. We first discuss the regularization, the computational requirements of FOCUSS, and the use of the parameter l. We then discuss how to achieve the desired convergence properties.

Each iteration of FOCUSS requires the evaluation of $(A_W)^{+}$. $A_W = AW_k$ (with $W_k = W_{a_k} W_{p_k}$) is the weighted A matrix at step k. When $(A_W)^{+}$ is ill conditioned, the inverse operation must be regularized to prevent arbitrarily large changes in x in response to even small noise in the data. Here, we suggest two regularized versions of FOCUSS based on the two most common regularization techniques [8]. One is Tikhonov regularization [34] used at each iteration. The second is truncated singular value decomposition (TSVD), which is also used at each iteration.

Tikhonov Regularization: In this method, the optimization objective is modified to include a misfit parameter

$$\min_x [\|Ax - b\|^2 + \lambda^2 \|W_k x\|^2]. \tag{10}$$

λ is the *regularization parameter* that must be chosen before (10) can be solved. When the condition number of A_W is not very large, the solution to (10) can be found by solving the normal equations

$$\left(A_W^T A_W + \lambda^2 I\right) x_{k+1} = A_W^T b. \tag{11}$$

Otherwise, solving

$$(A_W \quad \lambda I)\begin{pmatrix} x_{k+1} \\ s \end{pmatrix} = b \tag{12}$$

in the minimum norm sense is recommended instead. Standard implementations for solving (12) that include finding the optimal λ are not very computationally efficient. A novel algorithm for solving this problem efficiently is given in [28], and we omit it here in the interest of space.

TSVD: Here, A_W is replaced with a well-conditioned approximation A_{W_T}, given by SVD expansion of A truncated to the first t components

$$A_{W_T} = U_t S_t V_t^T. \tag{13}$$

The matrices U_t and V_t^T are composed of the first t left and right singular vectors of A_W. S_t is the diagonal matrix containing t corresponding singular values. The TSVD FOCUSS iteration is then

$$x_{k+1} = W_k V_t S_k^{-1} U_t^T b. \tag{14}$$

The parameter t can be found using the L-curve criteria [35], for example. The performance of both regularized versions of FOCUSS was studied in the context of the neuromagnetic imaging problem in [8].

The cost of inverse operations and efficient algorithms for computing regularized inverse solutions for large-scale problems are presented in detail in [28]. The Tikhonov regularization implementation proposed in [28] is approximately three times more efficient than the TSVD regularization that utilizes the R-SVD algorithm. In either case, the cost of both regularized inversions is only a linear function in n, i.e., $O(m^2 n + m^3)$ floating-point operations.

The truncation of entries of x at each iteration and the hard thresholding operation to terminate iterations were already discussed in Section III. These provide a very significant saving in computational cost and improve the performance. They should be used in all FOCUSS implementations.

The parameter l can be used to increase the rate of convergence and so further reduce the cost of computation. Although convergence to the minimum of the basin where the algorithm starts is not guaranteed for $l \neq 1$, convergence to this minimum can be shown from any point in its neighborhood for which $\|W_k^{l-1} q_k\| < n$ holds in the subsequent iteration. Thus, the inequality $\|W_k^{l-1} q_k\| < n$ defines a neighborhood of local convergence for a given realization of (8). To utilize $l \geq 2$, we can begin the calculations using $l = 1$ and switch to $l \geq 2$ once an x_k is reached for which the above inequality holds.

In principle, the parameter l can also be used to shape the basins of attraction and thus control the convergence outcome, but we do not advise this because it is difficult to predict the effects of a change in l on the outcome. Instead, we concentrate

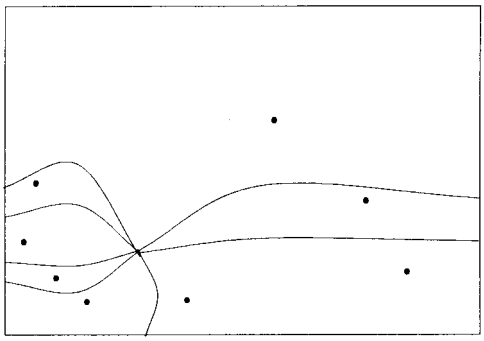


Fig. 2. Schematic representation of basins of attraction of FOCUSS. The dots indicate stable fixed points of the algorithm, and the lines mark the boundaries of the basins.

on the two factors that can be used to control convergence properties.

We assume here that the desired FOCUSS behavior is convergence to a sparse solution in the neighborhood of the initialization. Additionally, we would like the algorithm to favor the maximally sparse solution when its dimension is relatively small, for the reasons discussed in Section III.

Fig. 2 presents a schematic picture of the solution space tessellation via FOCUSS into basins of attraction around each s-f-p. In order for the algorithm to converge to the real solution, the initial estimate must fall into the correct basin of attraction. Thus, the shapes of the basins and the quality of the initialization are two interrelated factors that control the FOCUSS outcome.

To avoid having the algorithm favor any one solution, all its basins of attraction should be equally sized. The exception may be the maximally sparse solution, which we may want to favor, in which case, it should be quite large. Such basin sizes occur naturally in problems that include spectral estimation and far-field DOA estimation, which explains the noted success of the basic FOCUSS algorithm in these applications [16], [17], [19]. Physical inverse problems, such as biomedical or geophysical tomography, have a distinct bias to particular solutions, and the basins must be adjusted for proper convergence to occur. We discuss this issue next. Initialization options are discussed at the end of the section.

A. Basins of Attraction

The factors that control the shape of the basins are the relative sizes of the entries in the columns of A and the total number of sparse solutions in a given problem, as shown next.

1) Effect of A on the Basins: In any minimum norm based solution the magnitude differences in the entries of different columns of A act analogously to the weights W of a weighted minimum norm solution. This can be seen as follows. Suppose we can express matrix A as a product of two matrices A_n and N so that (1) becomes

$$A_n N x = b \tag{15}$$

where A_n is such that the entries in each of its columns span exactly the same range of values. N is then a diagonal matrix that reflects the "size" differences between the columns of the original A. The minimum norm solution to (15) is

$$x = (A_n N)^{+} b = N A_n^H \left(A_n N^2 A_n^H\right)^{-1} b \tag{16}$$

where A_n affects the solution only through the degree of correlation of individual columns with the vector b, whereas the entries of N act as weights on the corresponding elements of x, i.e., small/large entries of N reduce/increase the penalty assigned by the minimum norm criterion to the corresponding x_i. This means that the amplitudes of the entries in the columns of A can modify the effect of the weights w_{p_i} in a re-weighted minimum norm algorithm, producing a bias toward the entries x_i corresponding to the columns of A containing larger terms. The reason why no bias occurs in standard signal processing problems, such as spectral estimation, should be clear now. It is because the values in each column of A span the same $[-1 \text{ to } 1]$ range.

The intrinsic bias produced by A toward particular solutions translates into larger basins of attraction around these solutions in the re-weighted minimum norm algorithms. To eliminate the bias, the basin sizes must be equalized. Ideally, we would like to use a weight in (8), such as N^{-1} from (15), to cancel the penalties contributed to the weighted minimum norm cost by the magnitude differences in entries of the columns of A. This weight would be used at each iterative step. Unfortunately, the size of a column is not a well-defined quantity and cannot be completely adjusted via a scalar multiple. We found, however, that an approximate adjustment through such a scaling that makes the range of values in each column of A as similar as possible works well for such problems as electromagnetic tomography. We use this particular scaling in the example presented in Section VIII.

2) Effect of the Number and Dimension of the Solutions on the Basins: The larger the number of sparse solutions to a given problem, the greater the fragmentation of the solution space of the FOCUSS algorithm into correspondingly smaller basins. As the sizes of individual basins diminish, the algorithm must start progressively closer to the real solution in order to converge to it.

For an $m \times n$ system, the maximum number of sparse solutions occurs when all the solutions are basic, i.e., there are no degenerate basic solutions. That number is given by

$$\binom{n}{m} = \frac{n!}{(n-m)!m!}. \tag{17}$$

The number of basins is reduced when degenerate basic solutions are present. Each p-dimensional solution $(p < m)$ reduces the number of s-f-ps by

$$\binom{n-p}{m-p} - 1. \tag{18}$$

When the degenerate solution is 1-dimensional, there can be no other degenerate basic solutions, and the total number of sparse solutions is minimal: $\binom{n-1}{m} + 1$.

To summarize, the number of basins decreases with an increase in the number of data points m and a decrease in the dimensions of the degenerate basic solutions and increases with an increase in the dimension of the solution space n.

Basin sizes also depend on the dimensions of s-f-ps they contain. When there is no intrinsic bias due to A (either naturally or because we cancel it with a weight matrix), the basins around degenerate basic solutions are larger than the basins around the basic solutions, and the smaller the dimension of the degenerate basic solution, the larger its basin of attraction. This is because when the sparse solutions are all basic, each m-dimensional subspace has an equal probability of being the solution, and all basins are equal. When a degenerate basic solution exists, a 1-dimensional solution for example, the m-dimensional solutions in the subspaces containing this dimension are no longer present, and an initialization that would have lead to one of those basic solutions now leads to the 1-dimensional solution. For a 2-dimensional solution, all the basic solutions containing its two dimensions would be eliminated, but the basic solutions containing only one of its dimensions would still exist. The basin of this solution would be large, but not as large as the one for the 1-dimensional solution.

It follows then that when the algorithm is adjusted so that there is no bias due to A, the maximally sparse solution has the largest basin. The algorithm then favors the maximally sparse solution in its convergence, as desired, and the smaller the dimension of the maximally sparse solution, the greater the likelihood of convergence to it. At the other end of the spectrum of convergence behavior, when the dimension of the real solution approaches m, the algorithm must start progressively closer to this solution in order to end up in the right basin. Therefore, convergence to a solution neighboring the real solution becomes more common, but the distances between the real solution and the neighboring ones also become significantly smaller. In this case, the error is manifested in poorer resolution, rather than in gross discrepancy between the real signal and the solution obtained.

B. Initialization

It is clear that an initialization as close to the true solution as possible should be used. Earlier versions of the re-weighted minimum norm algorithm [16], [17], [20], [21] were initialized with the minimum norm solution (2). It is a very popular low-resolution estimate that is used when no *a priori* information is available because it is often thought to contain no bias toward any particular solution. Instead, this solution should be viewed as one that minimizes the maximum possible error for a bounded solution set. In [26], we show that depending on A, this estimate can strongly bias particular solutions, as described in the above subsection, and is not necessarily best, even in the absence of *a priori* information. Even when bias compensation is used, minimum norm-type estimates cannot be used universally to initialize FOCUSS. They are derived from only single samples of data and, hence, cannot resolve general non-unique sparse signals (see Section III) as they select only one out of several possible basins.

Instead, the best available low resolution estimate of the sparse solution should be used for the initialization. Any *a priori* information should be incorporated into it as well. The final choice of the algorithm clearly depends on the particular application. When multiple samples of data are available, however, the sparse signal of dimension less than m that can generate this data is unique [7]. In many applications, the sparse signal of interest is expected to be of dimension less than m; therefore, it can be estimated uniquely from multiple samples of data. Standard algorithms, however, suffer from decreased resolution under unfavorable conditions, such as nonstationarity of sources. In this case, they provide good initialization for FOCUSS, which can then refine the solution to a higher degree of accuracy. From our experience, beamforming is a good choice for FOCUSS initialization when no special constraints are present. This suggests array processing as one class of applications for FOCUSS, and we present an example of this application in Section VIII.

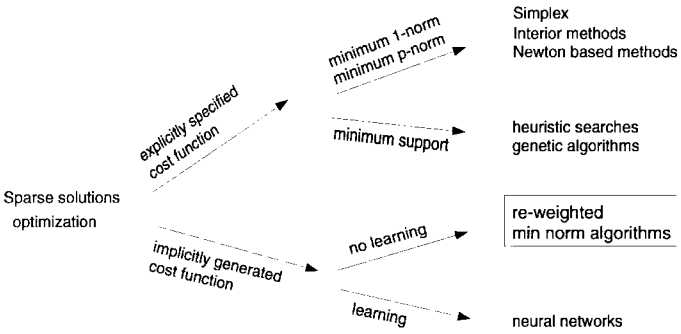


Fig. 3. Diagram of optimization methods for finding sparse solutions. The position of the FOCUSS algorithm is highlighted by the boxed area.

Note that when sparse initial estimates are used, they should be "blurred," and all the entries should be made nonzero so that potentially important components are not lost. In general, the initialization does not have to satisfy a given linear system exactly; therefore, any estimate, including guesses, can be used. In the neuroimaging application, for example, an estimate of brain activity from other modalities may be used to initialize FOCUSS.

VII. Relationship of FOCUSS to Other Optimization Strategies

The re-weighted minimum norm algorithms can be viewed as a novel class of computational strategies that combines elements of both the direct cost optimization and neural network methods, as depicted in Fig. 3. Like classical direct cost optimization methods, FOCUSS descends a well-defined cost function, but the function is generated in the process of computation rather than being explicitly supplied. Like an associative network, FOCUSS retrieves a stored fixed state in response to an input, but no learning is involved. Learning can be added, however, if desired, to fine tune the cost function. What sets FOCUSS apart is its utilization of the initial state, which defines the cost function being optimized. We next discuss how FOCUSS relates computationally to these optimization strategies.

The computational aspects of FOCUSS differ fundamentally from those of classical optimization methods for finding sparse solutions, the most popular of which are the Simplex algorithm and the interior methods, which include the Karmakar algo-

rithm. FOCUSS can be considered to be a boundary method, operating on the boundary of the simplex region $Ax = b$. An initialization near the final solution directly benefits FOCUSS convergence. The Simplex algorithm operates on the vertices of the simplex region defined by $Ax \leq b$, and the interior methods operate in the interior of this region, i.e., $Ax < b$. Interior methods do not benefit from initialization near the final solution because in the course of their computation, the intermediate iterative solutions move away from the boundary.

The connection of FOCUSS to pseudoinverse-based neural networks and its application to a pattern classification problem was presented in [25]. The input/output function in FOCUSS networks is well defined, and the stability of these networks is guaranteed for any input by the convergence analysis presented here. If desired, learning can be incorporated into FOCUSS to produce the desired associative recall. This can be done, for example, by introducing an additional weight matrix that is learned and modifies the shape of the basins of attraction. A crucial advantage of this type of network is that a process of regularization can be built into the algorithm to deal with noise [8]; therefore, a model of noise is not required. This is important for applications where noise models are hard to obtain, such as biomedical tomography. Another possible modification of FOCUSS that borrows from neural networks involves using past solutions as the memory states and cross-referencing them with the current solution to try to anticipate the final convergence state. This can speed up the convergence and provide a form of regularization.

VIII. Applications

The FOCUSS algorithm is suitable for application to linear underdetermined problems for which sparse solutions are required. The use of the basic form of the algorithm in spectral estimation and harmonic retrieval has been extensively investigated, e.g., [16]–[19], [26]. Several other utilizations have been studied, e.g., [8], [22], [25], [26]. Here, we present two examples. The first example is the narrowband farfield direction-of-arrival (DOA) estimation problem, for which we give the results of a detailed study of FOCUSS performance. We use scenarios with moving sources, a very short non-uniformly spaced linear array and short record lengths (single snapshots). The aim is to illustrate the implementation and the advantages of the algorithm on a familiar signal processing application. FOCUSS can also be applied to much broader *Sensor Array Processing* problems, given the appropriate forward model. One such example is the neuroimaging problem, which is the second application we present. This is a nearfield problem, where the sources are dynamic and multidimensional, the array is nonlinear and nonuniform, and no assumption is made on the bandwidth of the signals. Sources of brain magnetic fields that are not resolvable with more conventional methods are shown to be resolved by FOCUSS in this example.

A. DOA

DOA estimation deals with the estimation of incoming directions of waves impinging on an array of sensors. This problem is a special case of general sensor array processing. Our example addresses the narrowband far-field estimation problem, where the sources can be considered as point sources and the incoming waves as plane waves.

The nonparametric DOA model is constructed as follows. The data vector $b(t)$ denotes the output of the sensors at a time t, which is known as a *snapshot*. The noise-free output of the lth sensor at time t, i.e., $b_l(t)$, is the result of a superposition of m plane waves. This can be expressed as

$$b_l(t) = \sum_{i=1}^{m} x_i(t) e^{jw_0 \tau_l(\theta_i)} \tag{19}$$

where $x_i(t)$ is the response of the lth sensor to the ith source at time t, and $e^{jw_0\tau_l(\theta_i)}$ is the complex exponential representing the ith incoming wave with DOA θ_i, $(-\pi/2 \leq \theta_i < \pi/2)$, center temporal frequency w_0, and a time delay $\tau_l(\theta_i)$ of the wavefront between the reference sensor and the lth sensor. The parameters θ_i represent the *DOA* we want to estimate. The ith column of the matrix A is the output of the array due to a unit strength source at angular location θ_i. The columns are constructed by varying θ_i through the range of possible DOA and computing array outputs. The nonzero entries in the solution x select the angular directions of the sources that compose the signal.

We demonstrate the high-resolution performance of FOCUSS under challenging conditions. We use three moving sources whose location and intensity change from one snapshot to the next, a very short non-uniformly spaced linear array (eight sensors), and short record lengths (we use single snapshots for all cases). We run tests with varying noise level and DOA, relative angular separations, and amplitudes of the waves. The spatial frequencies of the waves do not match the frequencies represented in the columns of A. The sensors are spaced sufficiently close to avoid aliasing, i.e., the sampling density of the array is such that the URP of Section III is satisfied. FOCUSS with $l = 1$ and a hard thresholding that eliminates all entries below $10^{-5} \cdot x_{\max}$, where $x_{\max} = \max\{x_k(i)\}$, is used for all iterations. The use of such thresholding significantly improves the performance and the convergence rate. The initialization is done using a regularized MVDR estimate computed as

$$x_i = \frac{1}{a_i^H R_{\text{reg}}^{-1} a_i} \tag{20}$$

where a_i is the ith column of A, and $R_{\text{reg}} = b \cdot b^H + \delta I$ is a regularized covariance matrix of the data. The I here is the identity matrix, and δ is the regularization parameter. The regularization of the standard covariance matrix $R = b \cdot b^H$ is required because using so few snapshots results in a singular R. R^+ can be used in place of R_{reg}^{-1} in (20) to obtain comparable results, but only when the number of snapshots is equal to or exceeds the number of incoming waves.

The simulation results are as follows. We find FOCUSS performance to be consistent across the tested range of amplitudes and angular separations. In the case of zero noise, the error in the model is only due to the mismatch between the frequencies of the signal and those contained in the columns of A. In this case, the algorithm successfully recovers the DOA

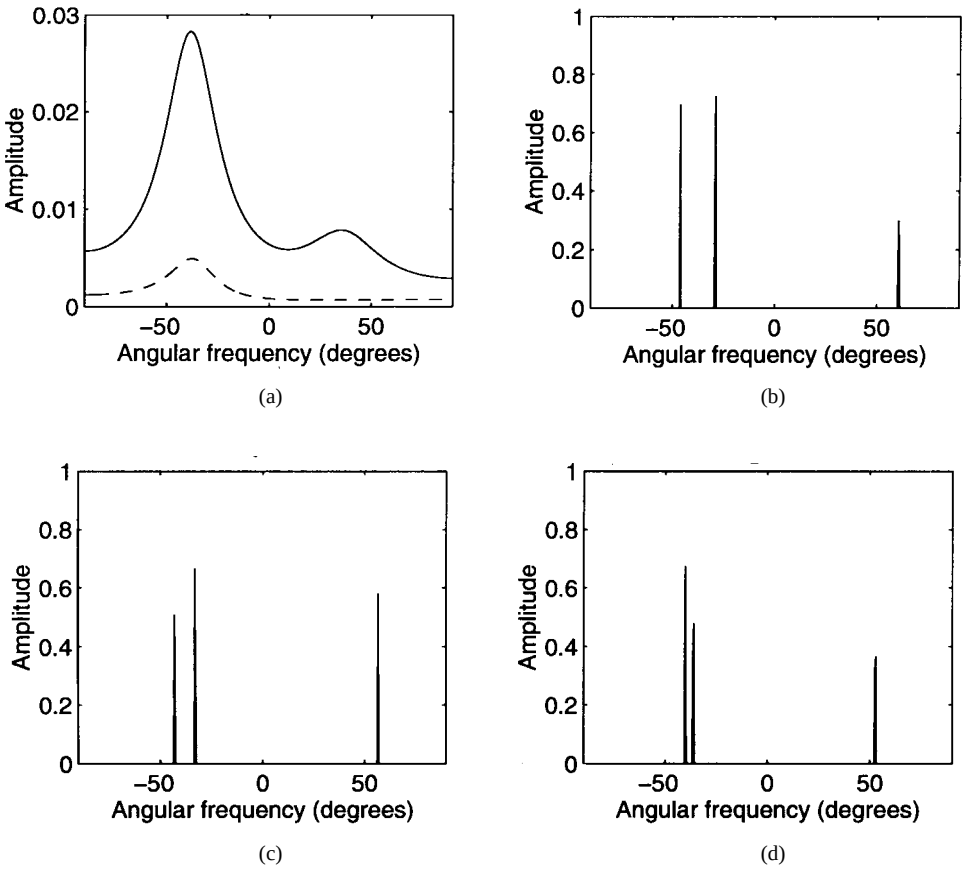


Fig. 4. DOA estimation from single snapshots, with an array of eight unevenly spaced sensors, of three nonstationary sources with DOA $[-46^\circ\ -29^\circ\ 60^\circ]$ for the first snapshot, $[-43^\circ\ -33^\circ\ 56^\circ]$ for the second snapshot, and $[-40^\circ\ -36^\circ\ 52^\circ]$ for the third. (a) MVDR estimates from the first snapshot (dashed line) and from the three snapshots combined (solid line). (b) FOCUSS solution for the first snapshot. (c) FOCUSS solution for the second snapshot. (d) FOCUSS solution for the third snapshot.

of each wave, typically by representing it by the two columns of A that span it. The exception is when the DOA is very close to the angular direction of a particular column. In that case, it is represented by this column alone. For this reason, the true amplitudes of the sources are not readily resolved. More precise DOA solutions that give accurate amplitude estimates can be found by hierarchically refining the solution grid in the areas of nonzero energy in $\bar{x}$ and executing a few additional FOCUSS iterations, as demonstrated in [24]. We do not demonstrate this step here.

In simulations with varying levels of noise, white Gaussian noise is used with the variance given as a percentage of the power of the weakest wave in the signal. The unregularized algorithm reliably assigns the highest power to the entries of x surrounding the true DOA's for noise power of up to 50%, i.e., signal to noise (SNR) of 3 dB with respect to the weakest source, but the power in the solution for each DOA tends to be spread among a number of neighboring columns of A. This number can be as high as 8 for the highest noise levels (3 dB). The unregularized solutions also have smaller spurious nonzero values in other entries of x. We use TSVD with the truncation level determined by the L-curve criteria [35] for the regularization of inverse operations. The regularization allows FOCUSS to handle slightly higher levels of noise, and it eliminates spurious energy in the solutions that are due to noise. It also concentrates the energy in the solutions for each DOA into a smaller (2–3) number of columns of A. As can be expected, due to noise, FOCUSS solutions may contain small errors. The columns of A that are found may no longer be the absolute closest ones to the DOA of the real signal, but they still provide a good estimate of the solution. In addition, the very closely spaced DOA's can, at times, get represented as a single DOA by the intervening columns of A.

The results are demonstrated with the following example. Three snapshots of three sources moving toward each other are used. Two sources start with a moderate angular separation in the first snapshot and are closely spaced by the third snapshot. The third source remains well separated from the other two at all times. The directions of arrival are $[-46^\circ\ -29^\circ\ 60^\circ]$ for the first snapshot, $[-43^\circ\ -33^\circ\ 56^\circ]$ for the second snapshot, and $[-40^\circ\ -36^\circ\ 52^\circ]$ for the third. The amplitudes of three sources are $[0.7\ 1\ 0.5]$, $[0.72\ 0.95\ 0.6]$, and $[0.75\ 0.8\ 0.7]$ for the respective snapshots. FOCUSS solutions in the noise-free case for each snapshot are shown in Fig. 4(b)–(d). The figures show the successful recovery of the three sources, including resolution of the two very closely spaced sources. In each case, the algorithm converges to the solution in four iterations. Fig. 4(a) shows the regularized MVDR estimates found using the first snapshot of data (dashed line) and all three snapshots combined (solid line). The FOCUSS reconstructions

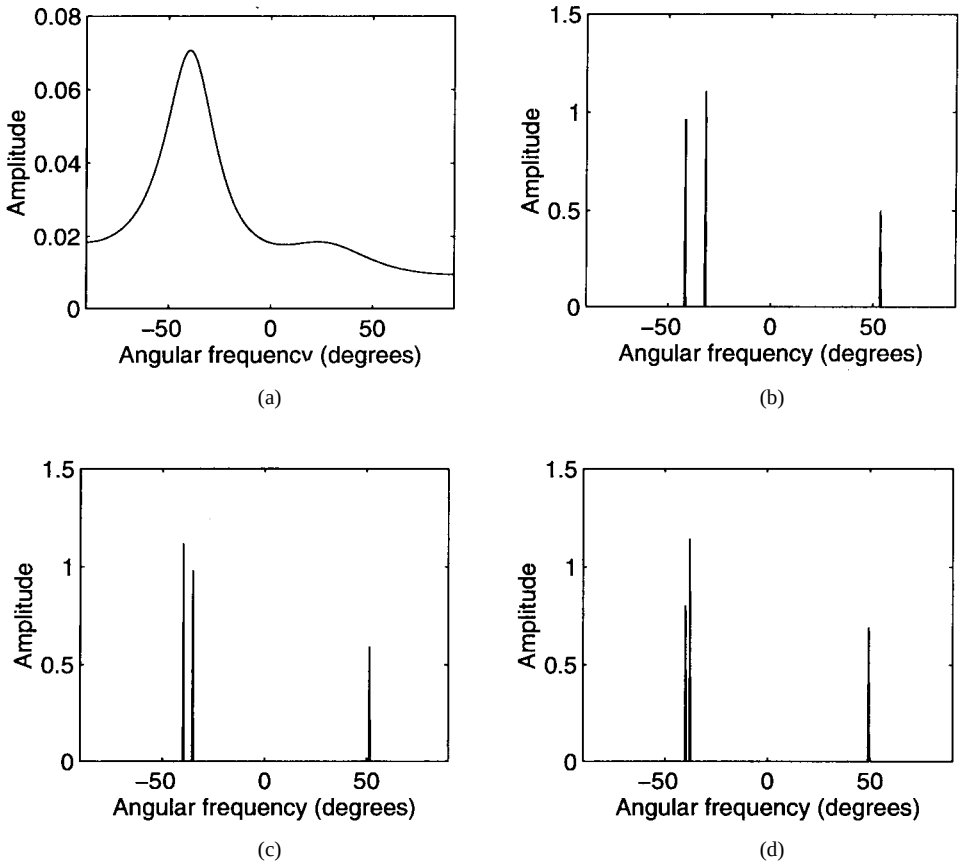


Fig. 5. DOA estimation using the same example as in Fig. 4 with random Gaussian noise added to the data. The SNR is 10 dB with respect to the weakest source. (a) MVDR estimates using all three snapshots. (b) FOCUSS solution for the first snapshot. (c) FOCUSS solution for the second snapshot. (d) FOCUSS solution for the third snapshot.

were independent of the MVDR solution that was used for the initialization. FOCUSS solutions for this example with a SNR of 10 db are shown in Fig. 5.

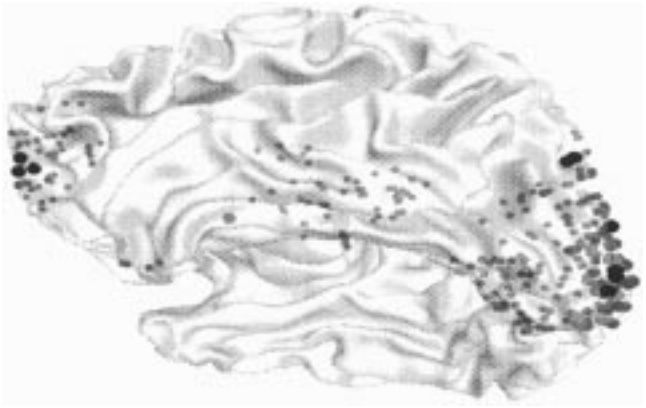
B. Neuroimaging

Functional imaging of the brain using the scalp electric potentials (EEG) or the brain magnetic fields measured outside the head (MEG) is an extrapolation problem, where the objective is to find the current inside the head that generates the measured fields. The problem is physically ill posed as well as underdetermined. Physical ill posedness means that the current cannot be determined uniquely even with the absolute knowledge of the fields outside the head, i.e., when the number of data is infinite. The only *a priori* constraint available is the knowledge that current distributions imaged by EEG/MEG are spatially compact, i.e., these currents are produced by synchronous firing of neurons clustered in 1 to 4 cm^2 areas. Because neuronal activity is highly dynamic, i.e., the intensity and location of the current flow can change fairly rapidly and because simple models, such as electric or magnetic dipoles, do not accurately describe extended and irregularly shaped areas of activity, EEG/MEG solutions are difficult to model either parametrically or through cost functions. The problem is further compounded by complex statistical properties of the noise and by the physical ill posedness. As a result, the success of conventional algorithms under realistic conditions has not been demonstrated for this problem. For a more complete description of the physics of the EEG/MEG imaging problem and its approaches, see references in [8].

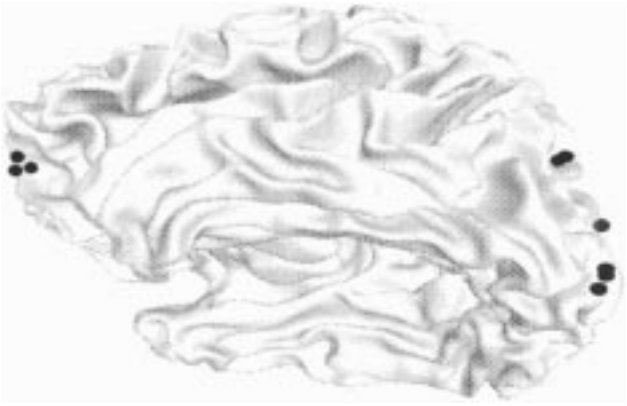
In [8], we show that the sparseness constraint is well suited for recovery of EEG/MEG signals if the problem is made physically well posed. This can be achieved by constraining the solutions to lie in a 2-D plane. In [7], we showed that the net effect of physical ill posedness is limited in any case to the existence of a small uncertainty envelope around each active site. Thus, by using the sparseness constraint, we can identify the neighborhoods where the activity occurs but not the exact shape of the current distributions. This is the best estimate obtainable for the EEG/MEG imaging problem. Our experimental results suggest that the maximally sparse solutions may not always work well to recover the neighborhoods of activity, even when Corollary 2 holds in this situation. The sparsity constraint can still be used here, however. AST constrained algorithms such as FOCUSS are a good choice of an estimator in this case, but they provide only one solution from an infinite set of possible current distributions within each neighborhood of activity. In particular, FOCUSS finds the maximally sparse representation for the current within each active site. We demonstrate finding an MEG solution with FOCUSS here.

The nonparametric EEG/MEG imaging model is constructed as follows. The elements of a solution x represent point

current sources at each location node in the cortex (the cortex is discretized using a 3-D lattice of nodes). Thus, the current distribution at an active site is represented in this model by a uniform distribution of point sources within the site. The data is generated by assigning current values to the selected locations in the cortex, shown in Fig. 6(a), and computing the resultant electromagnetic field at the sensor locations as determined by Maxwell equations. We use a boundary element model with realistic head geometry in this example. We assume 150 sensors and three neuronal ensembles containing, respectively 2, 4, and 3 active nodes. The distribution of current within each ensemble is maximally sparse. The coefficients of the matrix A map a unit current at each location in the cortex to the field values at the sensors. A weighted minimum norm solution (see Fig. 6(a)) that includes a compensation for the bias contained in A, as described in Section VI, is used for the initial estimate. Note the low resolution and the false active sites returned by the minimum norm-based estimate. The FOCUSS solution from (8) using $l = 1$ and the same bias compensation matrix W_a as used in the initialization is shown in Fig. 6(b). FOCUSS recovers the correct maximally sparse current at each active site.



(a)



(b)

Fig. 6. MEG reconstructions of three neuronal ensembles represented by 3, 2, and 4 active nodes, respectively, with a 150 sensor array: (a) Solid black circles mark the nodes of the three active sites. The weighted minimum norm solution that includes a compensation for the bias as described in the text is marked by gray transparent circles. (b) FOCUSS reconstruction of the active nodes shown in solid black circles.

APPENDIX A
PROOF OF THEOREM 2

To show fixed-point convergence, we use the solution set Γ as defined in Theorem 2 and show that the conditions of the global convergence theorem [30] hold:

i) The set $\{x_k\} \subset X$ is compact.

For the finite-dimensional case, this condition is equivalent to showing that a sequence of points $\{x_k : k = 1 : t\}$ generated by the mapping (8) of the algorithm is bounded.

At each iterative step, the solution is finite; therefore, we need only to examine the convergence limits of the algorithm. From ii), the convergence limits of $\{x_k\}$ are the minima of the descent function $L(x)$, which occur only when at least one of the entries of x_k becomes zero. The limit points of $\{x_k\}$ that are sparse solutions are clearly bounded. We later show that the non-sparse limit points are only reachable through special trajectories for which the limit of convergence has the same properties as a sparse solution. Therefore, these solutions are also bounded.

ii) There is a continuous descent function $L(x)$ such that

$$\begin{cases} L(x_k) < L(x_{k-1}) & \text{outside } \Gamma \\ L(\bar{x}) \le L(x_k) & \text{when } \bar{x} \in \Gamma. \end{cases}$$

We show that the descent function is

$$L(x) = \prod_{i=1}^{n} |x_k(i)|. \tag{A.1}$$

From $x_k(i) = x_{k-1}^l(i) q_k(i)$, we have $|x_k(i)| = |x_{k-1}^l(i)|$ $|q_k(i)|$ and

$$\prod_{i=1}^{n} |x_k(i)| = \prod_{i=1}^{n} |x_{k-1}(i)| \cdot \prod_{i=1}^{n} \left|x_{k-1}^{l-1}(i) q_k(i)\right|. \tag{A.2}$$

Taking the logarithm of both sides, we have

$$\sum_{i,x_i \neq 0} \ln|x_k(i)| = \sum_{i,x_i \neq 0} \ln|x_{k-1}(i)| + \sum_{i,x_i \neq 0} \ln\left|x_{k-1}^{l-1}(i) q_k(i)\right|. \tag{A.3}$$

Thus, $L(x_k) < L(x_{k-1})$ iff $\sum_{i,x_i \neq 0} \ln |x_{k-1}^{l-1}(i) q_k(i)| < 0$. To show that $\sum_{i,x_i \neq 0} \ln |x_{k-1}^{l-1}(i) q_k(i)| < 0$, we note that $\ln y^2$ is a purely concave function, that is

$$\sum_{i=1}^{n} \frac{1}{n} \ln\left(y_i^2\right) \le \ln\left(\frac{1}{n} \sum_{i=1}^{n} y_i^2\right).$$

We can rewrite this as

$$\frac{2}{n} \sum_{i=1}^{n} \ln|y_i| \le \ln \frac{1}{n} + \ln \sum_{i=1}^{n} y_i^2$$

or

$$\sum_{i=1}^{n} \ln|y_i| \le -\frac{n}{2} \ln n + \frac{n}{2} \ln \sum_{i=1}^{n} y_i^2. \tag{A.4}$$

We next observe that the norm minimized at each step of the algorithm is bounded by the the value of the norm when no

change in x occurs, that is

$$\left\|W_{p_k}^{-1}x_k\right\|^2 < \sum_{i=1}^{n}\left(\frac{1}{x^{l-1}(i)}\right)^2.$$

Since $\sum_{i=1}^{n} q_k(i)^2 = \|W_{p_k}^{-1}x_k\|^2$, we have

$$\sum_{i=1}^{n}\left(x_{k-1}^{l-1}(i)q_k(i)\right)^2 < n.$$

Substituting this into (A.3), we get

$$\sum_{i,x_i\neq 0}\ln\left|x_{k-1}^{l-1}(i)q_k(i)\right| \le -\frac{n}{2}\ln n + \frac{n}{2}\ln\sum_{i=1}^{n}\left(x_{k-1}^{l-1}(i)q_k(i)\right)^2 < -\frac{n}{2}\ln n + \frac{n}{2}\ln n = 0. \tag{A.5}$$

Substituting (A.4) into (A.2), we obtain the desired result

$$L(x_k) < L(x_{k-1}).$$

For $\bar{x}\in\Gamma$, we have $L(\bar{x}) = -\infty \le L(x_k)$.

iii) The mapping (8) is closed at points outside Γ.

This is true since (8) is a continuously differentiable mapping outside Γ. □

Appendix B
Proof of Theorem 3

Consider the system (1) partitioned as follows:

$$[A_1 \quad A_2]\begin{bmatrix}x_1\\ x_2\end{bmatrix} = b$$

where A_1 has any p ($p\le m$) columns such that $A_1x_1 = b$ is compatible, and A_2 contains the remaining columns. Similarly, let A_{1_k} and A_{2_k} be the two partitions of the weighted matrix AW_k at an iteration k. Then

$$A_{1_k} = A_1W_{1_k}, \quad A_{2_k} = A_2W_{2_k}$$

where $W_{1_k} = \text{diag}(x_{1_{k-1}})$ and $W_{2_k} = \text{diag}(x_{2_{k-1}})$. Define

$$B_k = \left(I - A_{1_k}A_{1_k}^{+}\right)A_{2_k} \quad \text{and} \quad C_k = A_{1_k}^{+}A_{2_k}\left(I - B_k^{+}B_k\right).$$

In the proof, we will also make use of the following results:

$$A_{1_k}^{+} = W_{1_k}^{-1}A_1^{+} \tag{B.1}$$
$$A_{1_k}^{+}A_{2_k} = W_{1_k}^{-1}A_1^{+}A_2W_{2_k} \tag{B.2}$$
$$C_k = W_{1_k}^{-1}A_1^{+}A_2W_{2_k}\left(I - B_k^{+}B_k\right) \tag{B.3}$$
$$\left\|\left(I - B_k^{+}B_k\right)\right\| \le 1 \tag{B.4}$$
$$\left\|\left(I + C_kC_k^H\right)^{-1}\right\| = \left\|\left(I + C_k^HC_k\right)^{-1}\right\| \le 1. \tag{B.5}$$

The inequality (B.4) holds because $(I - B_k^{+}B_k)$ is a projection operator.

From the general form of the pseudoinverse of partitioned matrices [29], together with the fact that $b\in\mathcal{R}(A_1)$, the kth iterate of the basic algorithm can be written as

$$x_{1_k} = W_{1_k}\left(I + C_kC_k^H\right)^{-1}(A_{1_k})^{+}b;$$

and

$$x_{2_k} = W_{2_k}C_k^H\left(I + C_kC_k^H\right)^{-1}(A_{1_k})^{+}b. \tag{B.6}$$

Consider an arbitrary point x_k in a neighborhood of x^*

$$x_{k-1} = \begin{pmatrix}x_1^*(1+\delta_1)\\ \vdots\\ x_p^*(1+\delta_p)\\ \epsilon_{p+1}\\ \vdots\\ \epsilon_n\end{pmatrix}$$

where we assume, without loss of generality, that $x_1^*(1+\delta_1)$ is the largest element of x_{k-1}. Note that

$$A_{1_k}^{+}b = \begin{pmatrix}\frac{1}{(1+\delta_1)}\\ \vdots\\ \frac{1}{(1+\delta_p)}\end{pmatrix}.$$

Using the Matrix Inversion Lemma $P^H(I+PP^H)^{-1} = I - P(I+P^HP)^{-1}P^H$ with (B.6) and (B.1)–(B.5), we have

$$\begin{aligned}&\left\|x_{1_k} - x_1^*\right\|\\ &= \left\|W_{1_k}C_k\left(I + C_k^HC_k\right)^{-1}C_k^HA_{1_k}^{+}b\right\|\\ &\le \|W_{1_k}C_k\|\left\|\left(I + C_k^HC_k\right)^{-1}\right\|\left\|C_k^H\right\|\left\|A_{1_k}^{+}b\right\| \qquad \text{(B.7)}\\ &\le \left\|A_1^{+}A_2\right\|\|W_{2_k}\|\left\|W_{1_k}^{-1}\right\|\left\|A_1^{+}A_2\right\|\|W_{2_k}\|\left\|W_{1_k}^{-1}\right\|\left\|A_1^{+}b\right\|\\ &\le \frac{\epsilon^2}{x_1^{*^2}(1+\delta_1)^2|1+\delta|}\sqrt{p}\left\|A_1^{+}A_2\right\|^2 \le \epsilon^2\sqrt{p}\left\|A_1^{+}A_2\right\|^2. \qquad \text{(B.8)}\end{aligned}$$

The ϵ^2 denotes the largest of the ϵ_i terms, and $(1+\delta)$ is the smallest of all the $(1+\delta_i)$ terms.

From (B.8), for sufficiently small ϵ, we get

$$x_{1_k} = x_1^* + O(\epsilon^2) = \begin{pmatrix}x_1^*(1+O(\epsilon^2))\\ \vdots\\ x_p^*(1+O(\epsilon^2))\end{pmatrix}.$$

Similarly

$$\begin{aligned}\|x_{2_k}\| &= \left\|W_{2_k}C_k^H\left(I + C_kC_k^H\right)^{-1}A_{1_k}^{+}b\right\|\\ &\le \|W_{2_k}\|^2\left\|W_{1_k}^{-1}\right\|^2\left\|A_1^{+}A_2\right\|\left\|A_1^{+}b\right\|\\ &\le \frac{\epsilon^2}{x_1^{*^2}(1+\delta_1)^2|1+\delta|}\sqrt{p}\left\|A_1^{+}A_2\right\| \le \epsilon^2\sqrt{p}\left\|A_1^{+}A_2\right\| \qquad \text{(B.9)}\end{aligned}$$

from which we get

$$x_{2_k} = \begin{pmatrix}O(\epsilon^2)\\ \vdots\end{pmatrix}.$$

In the next iteration, we repeat the same calculations, starting with x_k, to obtain $x_{1_{k+1}} = x_1^* + O(\epsilon^4)$ and $x_{2_{k+1}} = O(\epsilon^4)$.

For a sufficiently small ϵ, this shows asymptotic convergence from an arbitrary point $x\in\Omega$ to x^* with at least a quadratic rate of convergence.

The rate of convergence can be similarly derived for the general algorithm (8) with $l > 0.5$ using (B.7) and (B.9). The powers of $x_1^*(1+\delta_1)$ and ϵ in W_k in the general case will be

equal to l. Correspondingly, the rate of convergence will be $2l$. Thus, using $l = 2$ in (8), for example, generates a local rate of convergence order of 4. □

Proof of Corollary 3: Analytical investigation of saddle points is difficult because the behavior of the algorithm is different at different points in a neighborhood of a saddle point. Instead, we use the interpretation of the geometry of solution curves from nonlinear dynamical system theory for this part of the analysis.

We first establish the existence of FOCUSS generated sequences $\{x_k : k = 1 : t\}$ along which ratios between two or more entries of x_k do not change. Consider two "neighboring" sparse solutions $\bar{x}_1$ and $\bar{x}_2$ that differ in two entries as follows. For the first solution, we have $\bar{x}_1(l) = 0$ and $\bar{x}_1(r) \neq 0$, whereas for the second, we have $\bar{x}_2(l) \neq 0$ and $\bar{x}_2(r) = 0$. Along the trajectories $\{x_{1_k}\}$ within the basin of attraction of $\bar{x}_1$, the ratio $x_{1_k}(l)/x_{1_k}(r)$ is decreasing, whereas the reverse is true for the trajectories in the basin of $\bar{x}_2$. Then, by continuity of the parameters in the phase space, there must exist a trajectory along which the ratio between lth and rth entree of x_k is constant. This trajectory forms the boundary between the basins of attraction of $\bar{x}_1$ and $\bar{x}_2$ since it does not lead to either of these solutions. Similarly, we can extend the argument to show the existence of other boundary trajectories along which several sets of two or more entries of x do not change relative to each other. Each of these sets of entries acts as a single element of x and can be replaced by a linear combination of its entries without affecting the overall behavior of the algorithm along the special trajectories.

We now consider the convergence of FOCUSS along such a trajectory, which we denote by $\{\tilde{x}\}$. Let us substitute a single element for each set of entries of $\tilde{x}$ that are constant relative to each other. We similarly substitute a single column for each corresponding set of columns of A to produce a reduced system $A_r\tilde{x}_r = b$. Convergence of FOCUSS to sparse solutions of this system is guaranteed by Theorem 3. Some of these sparse solutions are the same as the sparse solutions of the original system (1). The others, namely, the solutions that contain the reciprocally constant terms of $\tilde{x}$, correspond to the non-sparse solutions of (1). For example, the trajectories along which any two terms of $\tilde{x}$ are reciprocally constant lead to $m + 1$ dimensional solutions. The trajectories along which any three entries of $\tilde{x}$ are reciprocally constant lead to $m + 2$ dimensional solutions, etc. Since FOCUSS converges to these solutions along these special trajectories only, these solutions are saddle points. Using the argument from the above paragraph, we can show that the total number of the trajectories leading to such non-sparse solutions is limited, meaning that the number of these solutions is also limited. In fact, the trajectories along which the ratio between any two terms of $\tilde{x}$ is constant are unique.

Thus far, we have established the existence of non-sparse solutions that are saddle fixed points and that the sparse solutions must be s-f-ps. Because the sparsity assumption is required in the proof of stability in Theorem 3, we propose that the non-sparse solutions in $\mathcal{C}^r$, $m < r < n$ must be the saddle points described above. Simulations verify this conclusion as well.

By the continuity argument used above, there also must exist a point in $\mathcal{C}^n$ where the recursive algorithm remains completely stationary. This point is the unstable fixed point of the algorithm. □

ACKNOWLEDGMENT

The original FOCUSS algorithm was developed while the first author was a visiting researcher at Los Alamos National Laboratories, in collaboration with Dr. J. George. The functional brain visualization project from which the image of the brain in Fig. 6 was produced is being carried out at the Visualization Laboratory at the San Diego Supercomputer Center with the assistance of D. Beransky and P. Vollrath. The authors are indebted to R. Beucker for bringing to their attention additional works on the use of the AST in overdetermined problems and to E. Baker for numerous readings and comments on the manuscript. The authors also would like to thank two unknown referees for their thorough review of the manuscript and useful comments, many of which they incorporated.

REFERENCES

[1] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.
[2] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, pp. 227–234, Apr. 1995.
[3] J. Holland, "Genetic algorithms," *Sci. Amer.*, pp. 66–72, July 1992.
[4] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, no. 6, pp. 721–741, 1984.
[5] M. S. O'Brien, A. N. Sinclair, and S. M. Kramer, "Recovery of a sparse spike time series by l_1 norm deconvolution," *IEEE Trans. Signal Processing*, vol. 42, pp. 3353–3365, Dec. 1994.
[6] G. Harikumar and Y. Bresler, "A new algorithm for computing sparse solutions to linear inverse problems," in *Proc. IEEE ICASSP*, May 1996.
[7] I. F. Gorodnitsky, "Can compact neural currents be uniquely determined?," in *Proc. 10th Int. Conf. Biomagn.*, Santa Fe, NM, Feb. 1996.
[8] I. F. Gorodnitsky, J. S. George, and B. D. Rao, "Neuromagnetic source imaging with focuss: A recursive weighted minimum norm algorithm," *J. Electroenceph. Clinical Neurophysiol.*, vol. 95, no. 4, pp. 231–251, Oct. 1995.
[9] G. Merle and H. Spath, "Computational experience with discrete l_p-approximation," *Comput.*, vol. 12, pp. 315–321, 1974.
[10] A. Papoulis, "A new algorithm in spectral analysis and bandlimited extrapolation," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 735–742, Sept. 1975.
[11] R. W. Gerchberg, "Super-resolution through error energy reduction," *Optica Acta*, vol. 21, no. 9, pp. 709–720, 1974.
[12] J. A. Cadzow, "An extrapolation procedure for band-limited signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 4–12, Feb. 1979.
[13] A. K. Jain, "Extrapolation algorithm for discrete signals with application in spectral estimation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 830–845, Aug. 1981.
[14] J. Sanz and T. Huang, "Discrete and continuous band-limited signal extrapolation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 1276–1285, Oct. 1983.
[15] A. Papoulis and C. Chamzas, "Detection of hidden periodicities by adaptive extrapolation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 492–500, Oct. 1979.
[16] S. D. Cabrera and T. W. Parks, "Extrapolation and spectral estimation with iterative weighted norm modification," *IEEE Trans. Signal Processing*, vol. 39, pp. 842–851, Apr. 1991.
[17] H. Lee, D. P. Sullivan, and T. H. Huang, "Improvement of discrete band-limited signal extrapolation by iterative subspace modification," in *Proc. ICASSP*, Dallas, TX, Apr. 1987, vol. 3, pp. 1569–1572.
[18] S. D. Cabrera, J. T. Yang, and C. H. Chi, "Estimation of sinusoids by adaptive minimum norm extrapolation," in *Proc. Fifth ASSP Workshop Spectrum Estimation, Modeling*, Oct. 1990, pp. 35–39.

[19] G. Thomas and S. D. Cabrera, "Resolution enhancement in time-frequency distributions based on adaptive time extrapolations," in *Proc. IEEE-SP Int. Symp. Time-Frequency Time-Scale Anal.*, Oct. 1994, pp. 104–107.
[20] A. A. Ioannides, J. P. R. Bolton, and C. J. S. Clarke, "Continuous probabilistic solutions to the biomagnetic inverse problem," *Inverse Problems*, vol. 6, pp. 523–542, 1990.
[21] I. F. Gorodnitsky, B. D. Rao, and J. S. George, "Source localization in magnetoencephalography using an iterative weighted minimum norm algorithm," in *Proc. 26th Asilomar Conf. Signals, Syst. Comput.*, Oct. 1992, vol. 1, pp. 167–171.
[22] R. Srebro, "Iterative refinement of the minimum norm solution of the bioelectric inverse problem," *IEEE Trans. Biomed. Eng.*, vol. 43, pp. 547–552, May 1996.
[23] I. F. Gorodnitsky and B. D. Rao, "A new iterative weighted norm minimization algorithm and its applications," in *Proc. 6th SP Workshop Stat. Signal Array Processing*, Oct. 1992, pp. 412–415.
[24] ——, "A recursive weighted minimum norm algorithm: Analysis and applications," in *Proc. Int. Conf. Acoustic, Speech Signal Processing*, Apr. 1993, vol. III, pp. 456–459.
[25] B. D. Rao and I. F. Gorodnitsky, "A novel recurrent network for signal processing," in *Proc. 1993 IEEE Workshop: Neural Net. Signal Processing*, Sept. 1993, pp. 108–117.
[26] I. F. Gorodnitsky, "A novel class of recursively constrained algorithms for localized energy solutions: Theory and application to magnetoencephalography and signal processing," Ph.D. dissertation, Univ. California, San Diego, La Jolla, 1995.
[27] B. D. Rao and I. F. Gorodnitsky, "Affine scaling transformation based methods for computing low complexity sparse solutions," in *Proc. Int. Conf. Acoustic, Speech Signal Processing*, May 1996.
[28] I. F. Gorodnitsky and D. Beransky, "Fast algorithms for biomedical tomography problems," in *Proc. 30th Asilomar Conf. Signals, Syst. Comput.*, Nov. 1996.
[29] S. L. Campbell and C. D. Meyer, Jr., *Generalized Inverses of Linear Transformations*. London, U.K.: Pitman, 1979.
[30] D. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 1989.
[31] H. Helmholtz, "Über einige gesetze der vertheilung elektrischer ströme in körperlichen leitern, mit anhwendung auf die thierisch-elektrischen versuche," *Pogg. Ann. Phys. Chemie*, vol. 89, pp. 211–233; 353–377, 1853.
[32] D. A. Linden, "A discussion of sampling theorem," in *Proc. IRE*, vol. 47, pp. 1219–1226, July 1959.
[33] S. Wiggins, *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. New York: Springer-Verlag, 1990.
[34] M. Foster, "An application of the Wiener-Kolmogorov smoothing theory to matrix inversion," *J. SIAM*, vol. 9, pp. 387–392, 1961.
[35] P. C. Hansen, "Analysis of discrete ill-posed problems by means of l-curve," *SIAM Rev.*, vol. 34, pp. 561–580, 1992.

Irina F. Gorodnitsky (S'90–M'95) received the B.S. degree in mathematics and biology and the M.S. degree in applied mathematics from the University of Washington, Seattle, in 1982 and 1984, respectively. In 1995, she received the Ph.D. degree in electrical engineering from the University of California, San Diego.

She worked as a Research Scientist at Honeywell Systems and Research Center from 1984 to 1986. She is currently a Postdoctoral Researcher at the Department of Cognitive Sciences at the University of California, San Diego. Her research interests are the mathematical foundation of signal analysis, digital signal and image processing, functional brain imaging, signal-based medical diagnostic algorithms, and human-computer interfaces.

Bhaskar D. Rao received the B. Tech. degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India, in 1979 and the M.S. and Ph.D. degrees from the University of Southern California, Los Angeles, in 1981 and 1983, respectively.

Since 1983, he has been with the University of California, San Diego, where he is currently a Professor in the Electrical and Computer Engineering Department. His interests are in the areas of digital signal processing, estimation theory, and optimization theory, with applications to communications, biomedical imaging, and speech.