Nonlinear Basis Pursuit
Henrik Ohlsson, Allen Y. Yang, Roy Dong and S. Shankar Sastry

Abstract—In compressive sensing, the basis pursuit algorithm aims to find the sparsest solution to an underdetermined linear equation system. In this paper, we generalize basis pursuit to finding the sparsest solution to higher order nonlinear systems of equations, called nonlinear basis pursuit. In contrast to the existing nonlinear compressive sensing methods, the new algorithm that solves the nonlinear basis pursuit problem is convex and not greedy. The novel algorithm enables the compressive sensing approach to be used for a broader range of applications where there are nonlinear relationships between the measurements and the unknowns.

I. INTRODUCTION

Consider the problem of finding the sparsest solution $x$ to a linear system of equations:

$$y_i = b_i^T x \in \mathbb{R}, \quad i = 1, \ldots, N, \quad x \in \mathbb{R}^n,$$  \hspace{1cm} (1)

where $y_i$ and $b_i$ are presumed given. This problem has received considerable attention recently in the area of compressive sensing. The exact solution to (1) is known as $\ell_0$-minimization ($\ell_0$-min):

$$\min_x \|x\|_0$$ \hspace{1cm} (2a)

subject to $y_i = b_i^T x, \quad i = 1, \ldots, N.$ \hspace{1cm} (2b)

It is well known that solving $\ell_0$-min is an NP-hard problem. Hence, it is not practical to directly solve the problem except when the system dimension $n$ is very small.

In the literature, algorithms for finding an approximate solution to (2) can be divided into two categories. The first category includes greedy algorithms [15], [28], [14]. Because these greedy algorithms iteratively update their estimate of $x$ based on local information, their computational complexity in the processing of computing the estimation updates is significantly lower than that of $\ell_0$-min. However, their main disadvantage is a weaker guarantee of convergence, especially comparing to the second category below.

The second category of algorithms solves a convex relaxed problem known as $\ell_1$-minimization or basis pursuit [13]:

$$\min_x \|x\|_1$$ \hspace{1cm} (3a)

subject to $y_i = b_i^T x, \quad i = 1, \ldots, N.$ \hspace{1cm} (3b)

Compared to greedy algorithms, basis pursuit provably recovers the exact solution as $\ell_0$-min under some mild conditions as described in compressive sensing theory [16], [8], [7]. As the nonconvex $\ell_0$-norm function is replaced by a convex $\ell_1$-norm function, basis pursuit can be solved by convex optimization. In sparse optimization literature, there have been extensive discussions about accelerating the implementation of basis pursuit. The interested reader is referred to [24], [39]. For further readings on greedy algorithms, basis pursuit, and compressive sensing, see [17].

More recently, compressive sensing theory has been extended to solving nonlinear problem, which is called nonlinear compressive sensing (NLCS) [4], [1]:

$$\min_x \|x\|_0$$ \hspace{1cm} (4)

subject to $y_i = f_i(x) \in \mathbb{R}, \quad i = 1, \ldots, N,$

where $f_i(x)$ in general can be considered a smooth function.

We are particularly interested in solving (4), as it can extend the compressive sensing approach to a broad range of applications where the relationship between the measurements and the unknowns is nonlinear. For example, the problem of quadratic basis pursuit (QBP) [32], [31], [30] is a special case of the NLCS formulation, which is a fundamental solution to the compressive phase retrieval problem in the applications of diffraction imaging [27] and sub-wavelength imaging [36].

Similar to solving $\ell_0$-min, directly solving (4) in its original form is very expensive and intractable in practice. Therefore, there is a need to seek more efficient numerical algorithms to estimate sparse solutions in (4) while their convergence can be also guaranteed. As the topic of nonlinear compressive sensing is rather new, there are just a few algorithms that have been proposed in the literature. For instance, one of the first papers discussing this problem is [5], which proposed a greedy gradient-based algorithm. Another greedy approach was also proposed in [23]. In [1], the authors proposed several iterative hard-thresholding and sparse simplex pursuit algorithms. As these algorithms are nonconvex greedy solutions, the analysis of their convergence typically only concerns about their local behavior. In [4], the author also considered a generalization of the restricted isometry property (RIP) to support the use of similar iterative hard-thresholding algorithms for solving general NLCS problems.

In this paper, we present a novel solution to NLCS, called nonlinear basis pursuit (NLBP). The work extends our previous publications in quadratic compressive sensing and compressive phase retrieval [31], [32], and proposes a convex algorithm to solve NLBP via a high-order Taylor expansion and a lifting technique. The work was inspired by several recent works on CS applied to the phase retrieval problem [27], [26], [12], [36], [11], [9], [33], [38], [21], [34], [35].
The paper is organized as follows. We will first discuss some basic notation for this paper in Section II. Then we will present the NLBP problem and its analysis of theoretical convergence in Section III. In Section V, we will present a numerical efficient convex solver to estimate the sparse solution of NLBP. Finally, in Section VI, we will present numerical evaluation to validate the performance of our algorithm and compare with other existing solutions.

II. NOTATION AND ASSUMPTIONS

In this paper, we use bold face to denote vectors and matrices, and normal font for scalars. We denote the transpose of a real vector by $\mathbf{x}^T$ and the conjugate transpose of a complex vector by $\mathbf{x}^H$. $X(i, j)$ denotes the $(i, j)$th element. Similarly, we let $x(i)$ be the $i$th element of the vector $\mathbf{x}$. Given two matrices $X$ and $Y$, we use the following fact that their product in the trace function commutes, namely, $\text{Tr}(XY) = \text{Tr}(YX)$, under the assumption that the dimensions match. $\| \cdot \|_0$ counts the number of nonzero elements in a vector or matrix; similarly, $\| \cdot \|_1$ denotes the element-wise $\ell_1$-norm of a vector or matrix, i.e., the sum of the magnitudes of the elements; whereas $\| \cdot \|_2$ represents the $\ell_2$-norm for vectors and the spectral norm for matrices. We assume that the functions $f_i(\cdot), i = 1, \ldots, N$ are analytic functions.

III. NONLINEAR BASIS PURSUIT

Since $f_i(\cdot), i = 1, \ldots, N$, are analytic functions, we can express them using their Taylor expansions. Using multi-index notation, we can write the Taylor expansion of $f_i$ around $x_0$ as

$$f_i(x_0) = \sum_{|\alpha| \geq 0} \frac{1}{\alpha!} (x - x_0)^\alpha \partial^\alpha f_i(x_0). \quad (5)$$

If $q$ is an even integer, we can now rewrite a $q$ order Taylor expansion as

$$f_i(x_0) = \sum_{0 \leq |\alpha| \leq q} \frac{1}{\alpha!} (x - x_0)^\alpha \partial^\alpha f_i(x_0) \quad (6)$$

$$= \mathbf{x}^T Q_i \bar{x} \quad (7)$$

where $Q_i$ is a $(n^{q+2} \times n^{q+2})$-symmetric matrix, and $\bar{x}$ contains all the monomials of the elements of $\mathbf{x}$ with degree $\leq q/2$.

Example 1: Let $\mathbf{x} = [x_1 x_2]^T$ and $f(x) = 1 + x_1 + x_2$. We consider a 4th order Taylor expansion around $x_0 = 0$. Then the set of 2-tuple $\alpha$'s is

$$A = \{(0,0), (1,0), (0,1), (1,1), (2,0), (0,2), (2,1), (1,2), (3,0), (0,3), (2,2), (3,1), (1,3), (4,0), (0,4)\}. \quad (8)$$

We can easily verify that $|A| = \binom{n+q}{q} = \binom{3+1}{1} = 15$. Hence, we can rewrite $f(x)$ as

$$f(x) = \begin{bmatrix} 1 & x_1 & x_2 & x_1 x_2 & x_1^2 & x_2^2 & x_1 x_2^2 & x_1^2 x_2 & x_1 x_2^2 \end{bmatrix}^T \begin{bmatrix} 1 & 1/2 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 \ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \ \end{bmatrix} \begin{bmatrix} 1 & x_1 & x_2 \ 0 & 0 & 0 \ 0 & 0 & 0 \ 0 & 0 & 0 \ 0 & 0 & 0 \ \end{bmatrix}.$$

Again, we verify that the dimension of $\bar{x}$ is equal to $\binom{2+2}{2} = 6$.

From the example it is easy to see that elements of $\bar{x}$ are generally dependent. The dependencies between elements of $\bar{x}$ needs to be made explicit.

Example 2: Let, as in the previous example

$$\bar{x} = [1 \ x_1 \ x_2 \ x_1 x_2 \ x_1^2 \ x_2^2]^T. \quad (9)$$

It is clear that e.g., the second element $x_1$ times the third element $x_2$ gives the third element $x_1 x_2$ of $\bar{x}$. This can be expressed as

$$0 = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1 x_2 \\ x_1^2 \\ x_2^2 \end{bmatrix}^T \begin{bmatrix} 0 & 0 & 0 & -1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 0 & 0 \\ -1/2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \ \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1 x_2 \\ x_1^2 \\ x_2^2 \end{bmatrix}.$$

Constructively, the dependencies between elements can be generated as follows. Let $Q_{N+1}$ be a $(n^{q+2} \times n^{q+2})$-symmetric matrix with $Q_{N+1}(1,1) = 1$ and set all other elements to zero. Set $y_{N+1} = 1$, $i = 1$, $k = l = 2$ and $m = N + 2$.

1) If $k \neq l$ and $\bar{x}(i) = \bar{x}(k)\bar{x}(l)$, let $Q_m$ be a $(n^{q+2} \times n^{q+2})$-symmetric matrix with $Q_m(k,l) = Q_m(l,k) = 1/2$, $Q_m(1,i) = Q_m(i,1) = -1/2$, set all other elements to zero, $y_m = 0$ and set $m = m + 1$.

2) If $k = l$ and $\bar{x}(i) = \bar{x}(k)\bar{x}(l)$, let $Q_m$ be a $(n^{q+2} \times n^{q+2})$-symmetric matrix with $Q_m(k,l) = 1$, $Q_m(1,i) = Q_m(i,1) = -1/2$, set all other elements to zero, $y_m = 0$ and set $m = m + 1$.

3) If $k < \binom{n+2}{2}$, set $k = k + 1$ and return to step 1, otherwise continue to the next step.

4) If $l < \binom{n+2}{2}$, set $l = l + 1$, $k = 2$ and return to step 1, otherwise continue to the next step.

5) If $i < \binom{n+2}{2}$, set $i = i + 1$, $k = l = 2$ and return to step 1. Otherwise set $M = m - 1$, return $\{(y_i, Q_i)\}_{i=1}^{M}$ and abort.

The dependencies can now be expressed as

$$y_i = \bar{x} Q_i \bar{x}, \quad i = N + 1, \ldots, M. \quad (10)$$

Note that these constraints are all quadratic.

Assuming that the $q$th order Taylor expansion is a good approximation for $\{f_i(\bar{x})\}_{i=1}^{N}$, then the problem of finding the sparsest solution to (4) can be written as:

$$\min_{\bar{x}} \|\bar{x}\|_0 \quad \text{subject to } y_i = \bar{x}^T Q_i \bar{x}, \quad i = 1, \ldots, M. \quad (11)$$

Next, we employ a lifting technique used extensively in quadratic programming [37], [25], [29], [19] and define a positive semi-definite matrix $\bar{x}^T \bar{x} = \mathbf{X}$, which satisfies, for $i = 1, \ldots, M$, the following equalities

$$y_i = \bar{x}^T Q_i \bar{x} = \text{Tr}(\bar{x}^T Q_i \bar{x}) = \text{Tr}(Q_i \bar{x} \bar{x}^T) = \text{Tr}(Q_i \mathbf{X}). \quad (12)$$
The nonconvex problem in (11) can now be shown equivalent to

\[
\begin{align*}
\min_{X} & \quad \|X\|_0 \\
\text{subj. to} & \quad y_i = \text{Tr}(Q_i X), \quad i = 1, \ldots, M, \quad \text{(13)} \\
& \quad \text{rank}(X) = 1, \quad X \succeq 0.
\end{align*}
\]

Due to the \(\ell_0\)-norm function and the rank condition in (13), the problem is still combinatorial. To relax the \(\ell_0\)-norm function in (13), we can replace the \(\ell_0\)-norm with the \(\ell_1\)-norm. To relax the rank condition, we can remove the rank constraint and instead minimize \(\text{rank}(X)\) in the objective function. Furthermore, since the rank of a matrix is still not a convex function, we choose the rank with the nuclear norm of \(X\), which is known to be a convex heuristic of the rank function \[18\], \[10\]. For a semidefinite matrix \(X\), it is also equal to the trace of \(X\). Note that there are several other heuristics for the rank. For example, the log det heuristic described in \[18\] is an interesting alternative (see also \[36\]). However, we choose to use the nuclear norm heuristic here even though our theoretical results also holds for \(e.g.,\) the log det heuristic. We finally obtain the convex program

\[
\begin{align*}
\min_{X} & \quad \text{Tr}(X) + \lambda \|X\|_1 \\
\text{subj. to} & \quad y_i = \text{Tr}(Q_i X), \quad i = 1, \ldots, M, \quad \text{(14)} \\
& \quad X \succeq 0,
\end{align*}
\]

where \(\lambda \geq 0\) is a design variable. We refer to (14) as nonlinear basis pursuit (NLBP).

IV. THEORETICAL ANALYSIS

NLBP is a convex relaxation of the combinatorial problem given in (11). It is of interest to know when the relaxation is tight. As a special case, when the degree of the analytic functions \(f_i(x)\) are no greater than two, the problem is known as quadratic basis pursuit (QBP) \[32\], \[31\]. In fact, the underlying optimization problem of NLBP is the same as that of QBP. Therefore, the theoretical analysis about the performance guarantees of QBP also applies to NLBP in this paper. In this section, we only highlight several key results that extend from the proofs given in \[32\], \[31\].

First, it is convenient to introduce a linear operator \(B:\)

\[
B: X \in \mathbb{R}^{(n+\frac{q}{2})\times(n+\frac{q}{2})} \mapsto \{\text{Tr}(Q_i X)\}_{1 \leq i \leq M} \in \mathbb{R}^M. \quad \text{(15)}
\]

We consider a generalization of the restricted isometry property (RIP) of the linear operator \(B\):

Definition 1 (RIP): A linear operator \(B(\cdot)\) as defined in (15) is \((\epsilon, k)\)-RIP if

\[
\left| \frac{\|B(X)\|^2}{\|X\|^2} - 1 \right| < \epsilon
\]

for all \(\|X\|_0 \leq k\) and \(X \neq 0\).

We can now state the following theorem:

Theorem 2 (Guaranteed recovery using RIP): Let \(\bar{x}\) be the solution to (11). The solution of NLBP \(\bar{X}\) is equal to \(\bar{x}\bar{x}^T\) if it has rank 1 and \(B(\cdot)\) is \((\epsilon, 2)\)-RIP with \(\epsilon < 1\).

The RIP-type argument may be difficult to check for a given matrix and are more useful for claiming results for classes of matrices/linear operators. For instance, it has been shown that random Gaussian matrices satisfy the RIP with high probability. However, given realization of a random Gaussian matrix, it is indeed difficult to check if it actually satisfies the RIP.

In the literature, there exist two alternative arguments, namely, the spark condition \[13\] and the mutual coherence \[16\]. The spark condition usually gives tighter bounds but is known to be difficult to compute as well. On the other hand, mutual coherence may give less tight bounds, but is more tractable. Next, we focus our discussion on mutual coherence, which is defined as:

Definition 3 (Mutual coherence): For a matrix \(A\), define the mutual coherence as

\[
\mu(A) = \max_{1 \leq i, j \leq n, i \neq j} \frac{|A^T_{i,j}|}{\|A_i\| \|A_j\|}. \quad \text{(17)}
\]

Let \(B\) be the matrix satisfying \(y = BX^* = B(X)\) with \(X^*\) being the vectorized version of \(X\). We are now ready to state the following theorem:

Theorem 4 (Recovery using mutual coherence): Let \(\bar{x}\) be the solution to (11). The solution of NLBP, \(\bar{X}\), is equal to \(\bar{x}\bar{x}^T\) if it has rank 1 and \(\|X\|_0 < 0.5(1 + 1/\mu(B))\).

V. NUMERICAL SOLVERS

Naturally, efficient numerical solvers that implement nonlinear basis pursuit are desirable. There are many classes of methods, commonly used in compressed sensing, which can be used to solve non-smooth SDPs. Among these include interior point methods, which is used in the popular software package CVX \[20\], gradient projection methods \[2\], and augmented Lagrangian methods (ALM) \[2\], outer approximation methods \[12\], and the alternating directions method of multipliers (ADMM), see for instance \[3\], \[6\].

Interior point methods are known generally to scale poorly to moderate- or large-scale problems. Gradient projection methods require a projection onto the feasible set; for our problem, this feasible set is the intersection of a subspace with the positive semidefinite cone. The complexity of this projection operator limits the benefits of using a gradient projection method. For ALM, the augmented primal and dual objection functions are still SDPs, which are equally difficult to solve in each iteration as the original problem. Also, we found that outer approximation methods converge very slowly.

On the other hand, (14) decomposes nicely into the ADMM framework. That is, we can define the following functions:

\[
\begin{align*}
\text{h}_1(X) & = \left\{ \begin{array}{ll} 
\text{Tr}(X) & \text{if } y_i = \text{Tr}(Q_i X), \quad i = 1, \ldots, M \\
\infty & \text{otherwise} \end{array} \right. \quad \text{(18a)} \\
\text{h}_2(X) & = \left\{ \begin{array}{ll} 
0 & \text{if } X \succeq 0 \\
\infty & \text{otherwise} \end{array} \right. \quad \text{(18b)} \\
g(Z) & = \lambda \|Z\|_1 \quad \text{(18c)}
\end{align*}
\]

Then, (14) becomes:

\[
\begin{align*}
\min_{X_1, X_2, Z} & \quad \text{h}_1(X_1) + \text{h}_2(X_2) + g(Z) \\
\text{subj. to} & \quad X_1 = X_2 = Z \quad \text{(19)}
\end{align*}
\]
which has the form that ADMM applies to. ADMM has strong convergence results, converges quickly in practice, and scales well to large data sets [6]. For more details of this implementation, we refer the reader to [32], [31].

VI. NUMERICAL EVALUATION

A. A Simple Example

Let \( n = 5, N = 50 \) and consider

\[
y_i = \sum_{4 \geq |\alpha| \geq 0} q_\alpha x^\alpha, \quad i = 1, \ldots, N, \quad x \in \mathbb{R}^n, \quad (20)
\]

\( \{q_\alpha\}_{4 \geq |\alpha| \geq 0} \) was generated by sampling from a Gaussian distribution and \( x \) was zero except for two elements that both were one. A Monte Carlo simulation consisting of 100 trials was performed. We compared NLBP (fourth order Taylor expansion), QBP (second order Taylor expansion), and LASSO (first order Taylor expansion). \( x_0 = 0 \) was used. The results are given in Table I.

![Box plot comparing NLBP and QBP for finding the dense solution of polynomial equation systems. The box plot summarizes the result from 100 trials.](image)

**TABLE I**

<table>
<thead>
<tr>
<th>Method</th>
<th>NLBP</th>
<th>QBP</th>
<th>LASSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success rates</td>
<td>100%</td>
<td>74%</td>
<td>0%</td>
</tr>
</tbody>
</table>

B. Polynomial Equation System

In this example we consider the problem of finding the dense \( x \) that solves a system of 4th order polynomials

\[
y_i = \sum_{4 \geq |\alpha| \geq 0} q_\alpha x^\alpha, \quad i = 1, \ldots, N, \quad x \in \mathbb{R}^n, \quad (21)
\]

given the coefficients \( \{q_\alpha\}_{4 \geq |\alpha| \geq 0} \). We will let \( N = 60, n = 5 \) and generate \( x \) by sampling from a Gaussian distribution with a standard deviation of 10. Figure 1 shows the box plots of the squared residuals from a Monte Carlo simulation consisting of 100 trials. New polynomial coefficients were generated at random for each trial and a new \( x \). NLBP found the true solution (within machine precision) in 99 out of the 100 trials. QBP never found the correct solution. Both used \( \lambda = 0 \). LASSO with \( \lambda = 0 \) (the least squares estimate) did not give a satisfactory estimate.

VII. CONCLUSION AND FUTURE WORK

The main contribution of this paper is a nonlinear compressive sensing algorithms based on convex relaxations. The algorithm, referred to as nonlinear basis pursuit, is rather general in that it applies to any analytic nonlinearity by approximate it by a Taylor expansion of desired order. Nonlinear basis pursuit inherits theoretical guarantees, such as guaranteed recovery etc from its linear relative (basis pursuit) and therefore comes with theoretical guarantees that greedy algorithms often lack. Nonlinear basis pursuit takes the form of a convex non-smooth SDP which can be solved using conventional software for problems of interest.

It should be noticed that solving a nonlinear equation system is by itself a difficult problem. It is therefore quiet remarkable that we with rather high success rate manage to find the sparest solution to the nonlinear equation system. In addition, solving an overdetermined nonlinear equation system is also difficult. As shown in the example section, nonlinear basis pursuit not only find sparse solutions but also can also provide dense solutions to nonlinear equation system when no sparse solutions is available.

Convexifying nonlinear constraints using SDPs via its Taylor expansion is up to our knowledge novel and should find applications in many areas. This is seen as future research. Last, we have not discussed noise in this paper. However, this extension is trivial and a nonlinear extension of basis pursuit denoising follows.

VIII. ACKNOWLEDGMENTS

The authors would like to acknowledge useful discussions and inputs from Yonina C. Eldar, Laura Waller, Filipe Maia, Stefano Marchesini and Michael Lustig.

REFERENCES


