A heuristic for sparse signal reconstruction

Theofanis Apostolopoulos

1 King’s College, London, Department of Informatics
Strand, London, WC2R 2LS, United Kingdom,
theofanis.apostolopoulos@kcl.ac.uk

Abstract

Compressive Sampling (CS) is a new method of signal acquisition and reconstruction from frequency data which do not follow the basic principle of the Nyquist-Shannon sampling theory. This new method allows reconstruction of the signal from substantially fewer measurements than those required by conventional sampling methods. We present and discuss a new, swarm based, technique for representing and reconstructing signals, with real values, in a noiseless environment. The method consists of finding an approximation of the $l_0$-norm based problem, as a combinatorial optimization problem for signal reconstruction. We also present and discuss some experimental results which compare the accuracy and the running time of our heuristic to the IHT and IRLS methods.

1998 ACM Subject Classification I.5.4 Signal Processing (Applications)

Keywords and phrases Compressive Sampling, sparse signal representation, $l_0$ minimisation, non-linear programming, signal recovery

Digital Object Identifier 10.4230/OASIcs.ICCSW.2012.8

1 Introduction

Over the last few years, a number of different methods for sparse approximation in signal reconstruction have arisen including the Compressive Sampling technique. Compressive Sampling (CS) states that it is possible to reconstruct signals accurately and almost exactly from much fewer number of measurements than those required by the Nyquist-Shannon sampling theory. To achieve this, the method relies on two major principles: sparsity of signal and incoherence of the measurements being taken [2, 7, 8, 9, 10, 11, 13, 15, 20]. Sparsity implies that only a small percentage of the signal entries (less than 40%) in a known transform domain is nonzero or significantly different from zero [8, 11, 15, 20, 21]. Incoherence in measurements states that all the collected samples of a signal are randomly generated and independent to each other [7, 8, 11, 15, 20, 21]. For simplicity, we use signals with real values each of which can be presented as a vector $X = [x_1, x_2, \ldots, x_n]$. In this article we propose a new swarm based method for sparse signal representation and reconstruction based on the key mathematical insights underlying this new theory. We compare the proposed method with two well-known signal reconstruction methods in terms of time and recovery error. The rest of this article is organised as follows: The next section presents the signal reconstruction problem and how the algorithm deals with it. Then, the algorithm is stated in Section 3, while in Section 4, we briefly describe the alternative algorithms used for comparison. Section 5 provides and presents some experimental results of our algorithm and its comparison with the other two methods.
The Signal Reconstruction problem

Obtaining sparse solutions from an under-determined system of linear equations has been of paramount importance in the area of signal processing and analysis. The CS theory aims to obtain the sparsest possible representation of the signal \( X = [x_1, x_2, \ldots, x_N] \), from an under-determined system of linear measurements \( Y \in \mathbb{R}^M \), so as \( Y = CX \), where \( X \in \mathbb{R}^N \) is the signal vector we want to find and \( C \in \mathbb{R}^{M \times N} \) is a Sensing matrix used for under-sampling \( X \) (with \( M \ll N \)). This ill-posed problem can be modelled as an optimisation problem (signal reconstruction problem) as follows \[2, 7, 9, 11, 13, 15, 20, 21\]:

\[
\begin{align*}
\min \ |X|_0 & \quad s.t. \quad Y = CX, \\
\end{align*}
\]  

(1)

where \( |X|_0 \) is the \( l_0 \) norm which is equal to the number of non-zero components in the vector \( X \). Finding the solution to problem (1) is NP-hard due to its nature of non-convex combinational optimization \[2, 7, 9, 11, 13, 15\]. For this reason many researchers suggested replacing the \( l_0 \) norm with the convex approximation of \( l_1 \) norm \[7, 9, 11, 13, 15, 19\]. However, it is still possible to reconstruct sparse signals using the constrained \( l_0 \)-minimisation, which in many situations outperforms even \( l_1 \)-minimisation in the sense that substantially fewer measurements are needed for recovery \[1, 14, 16, 17, 19, 22\]. The main idea is to approximate the \( l_0 \) norm by a smooth continuous function which is easier to handle and does not suffer from the discontinuities of the \( l_0 \) norm. This function can be defined as \[1, 14, 16, 17, 22\]:

\[
\|X\|_0 \approx f_\sigma(X) = N - \sum_{i=1}^{N} f_\sigma(x_i) = N - \sum_{i=1}^{N} \exp\left(-\frac{|x_i|^2}{2\sigma^2}\right),
\]  

(2)

where \( x_i \) is the \( i \)-th element of the signal (vector) \( X \) of \( N \) terms (length) and \( f_\sigma(X) \) is a continuous function, which belongs to the Gaussian family of functions. The \( \sigma \) is actually a decreasing sequence of constants \([\sigma_1, \sigma_2, \ldots, \sigma_j]\) for every iteration of the method so as to maximise the smoothed \( l_0 \) norm of the problem. Then, the problem can be defined as:

\[
\begin{align*}
\max \ f_\sigma(X) &= (N - \sum_{i=1}^{N} \exp\left(-\frac{|x_i|^2}{2\sigma^2}\right)) \quad s.t. \quad Y = CX, \\
\end{align*}
\]  

(3)

Now we have a smooth objective function, though non-linear, which is much easier for calculations. The purpose is to maximise the objective function in (3) together with the minimisation of the real parameter \( \sigma \). The value of this parameter represents the tradeoff between accuracy and smoothness of the approximation. The smaller the \( \sigma \), the better the approximation, while the larger the \( \sigma \), the smoother the approximation. Also note that the minimisation of \( l_0 \) norm is actually equivalent with the maximisation of the \( f_\sigma \) for sufficiently small \( \sigma \). For small values of \( \sigma \), \( f_\sigma \) contains a lot of local maxima and thus it is difficult to maximise it. Therefore, we need to set this parameter initially very large so as to make the objective function convex and then gradually decrease it according to the value of the objective function so as to enter the region close to its global maximiser.

3 The Proposed Algorithm

In this section we present the pseudocode of the proposed method (Pseudocode (1)) together with the parameter settings used. The method is an iterative process which is based on the swarm optimisation. The computation is conducted by a group of agents, where every agent carries a solution which is slightly different from the other agents. At each iteration \( t \) the
A heuristic for sparse signal reconstruction

Pseudocode 1

| Problem: Determine a vector $X$ s.t. $CX = Y$. |
| Inputs: $\sigma$, $C$, $Y$, Iterations, Agents, Sparsity level $S$, $f_\sigma(X)$. |
| Outputs: best value $f_\sigma(X)$, best sparse vector $X_*$. |

Proposed swarm based method:

1. Generate Initial $X^{(0)}_i$ using (4) for every swarm $i$.
2. Set $\sigma^{(0)}_i = 2 \times \max X$ for every swarm $i$.
3. While ($t < \text{Iterations}$) (for all iterations)
   - For all Agents (for all swarms)
     - Evaluate $f_\sigma(X)$ for every $X^{(t)}_i$.
     - Find current best $X^{(t)}_i$ so as max $f_\sigma(X)$ and min $\sigma$.
     - Set $X^{(t)}_i = X^{(t)}_i$ (keep the best $i$'th solution).
     - Check $X^{(t)}_i$ entries for non-feasible values (Pseudocode (2)).
     - Consider the constraints $CX = Y$ (project back to feasibility set): $X^{(t)}_i = X^{(t)}_i - C^T (C C^T)^{-1} (C X^{(t)}_i - Y)$.
     - Set all but $S$ largest entries of $X^{(t)}_i$ to zero.
     - Generate new solutions for all the other agents based on (5).
   - End For all Agents (for all swarms).
4. Set $\sigma^{(t+1)} = \sigma^{(t)} \times 0.5$.
5. End While ($t < \text{Iterations}$).
6. Display the signal reconstruction error using equation (9).

Current best solution $X^{(t)}_*$ that maximises $f_\sigma$ and minimises $\sigma$ is chosen. It is then corrected in terms of feasibility and bounds of its values. All the other agents are destroyed and a new solution is generated for each of them based on the previously created one. Again all the solutions are evaluated against the current best solution, which is updated, till the method completes all the number of iterations given. Also, note that the $\sigma$ value is initially assigned to twice the maximum value of the vector $X$ and then it is gradually decreased by half at each iteration. This particular assignment was chosen based on the nature of the given test vector (signal). Finally, it is notable that every solution vector generated is projected back to the feasibility set based on the constraints equation $CX = Y$ and then only the $S$ largest entries are kept, setting all the others equal to zero. This step of the method is very important as it achieves the necessary feasibility of the new solution and also follows the sparsity level of the original vector (signal).

3.1 Initial Solution

The initial solution generated in vector format, for each swarm $i$, is given as:

$$X^{(0)}_i = ((C^T C)^{-1} C^T Y) + k,$$

where, $(C^T C)^{-1} C^T Y$ is the pseudo-inverse of matrix $C$, $X^{(0)}_i$ is the initial solution vector for agent $i$ and $k$ is a vector of small random numbers based on the lowest value of the original signal $X$. This $k$ value is slightly different for every agent that carries a solution.

3.2 Solution Generation

The generation of a new solution in vector format for each swarm $i$ is generated as:

$$X^{(t)}_i = 2 \times k^t \times X^{(t-1)}_i \times \sigma^{4L} + (1 - k^t) \times 1/M \times L,$$

(5)
where, \( M \) is the number of samples, \( k \) is a vector of small random numbers between 0 and 1, different for every swarm \( i \), \( t \) is the current iteration, while \( X_i^{(t)} \) and \( X_i^{(t-1)} \) is the current and the previously generated solution vector of the \( i \)-th swarm. \( L \) is the norm \( \| Y - CX_i^{(t)} \|_2 \) which stands for the Euclidean distance between the samples vector \( Y \) and the product between the Sampling matrix \( C \) and the current best solution at iteration \( t \), \( X_i^{(t)} \).

### 3.3 Solution Correction

Every solution vector \( X_i^{(t)} \) created in Equation (5) is tested and corrected so as to be within the given ranges of the original vector (signal). \( X_{\text{min}} \) and \( X_{\text{max}} \) are the minimum and maximum value of the given original signal, which remain the same for all iterations. The whole procedure is presented in Pseudocode (2).

### 4 Alternative Algorithms

Several methods have been proposed to find the sparsest solution of the under-determined system of linear equations in (1), including many methods for obtaining signal representations in over-complete dictionaries. These methods range from general approaches, like the Basis Pursuit (BP), Orthogonal Matching Pursuit (OMP) and the method of Matching Pursuit (MP) \[6, 18\] to more sophisticated ones such as a Steepest Descent/Ascent methods \[1, 16\] together with the IHT \[3, 4, 5\] and IRLS \[12\] methods, which will be briefly described in this Section. In our point of view, all these methods have both advantages and shortcomings; some are very slow in convergence, such as BP and OMP methods, while others have low estimation quality especially for large systems of equations, such as IRLS. Furthermore, to the best of our knowledge, we are not aware of any swarm based techniques used in the Compressive Sampling framework so far.

#### 4.1 Iterative Hard Thresholding (IHT)

Iterative Hard Thresholding Algorithm (IHT) is a simple, yet efficient, iterations based method for signal reconstruction, which uses a non-linear operator (\( P_k \)) to reduce the value of the \( l_0 \) norm at every iteration. The new solution is generated as follows \[3, 4, 5\]:

\[
X^{(t)} = P_k(X^{(t-1)} + C^T(Y - CX^{(t-1)})),
\]

where, \( Y \) is the samples vector, \( C \) the Sensing matrix, and \( X^{(t-1)}, X^{(t)} \) are the current and the new generated solution. \( P_k \) is a hard thresholding operator that sets all but \( K \) largest elements to zero. The algorithm can be summarised in Pseudocode (3) \[3, 4, 5\].

#### 4.2 Iteratively Re-weighted Least Squares (IRLS)

This algorithm tries to reconstruct sparse signals, using a re-weighted least squares method for computing local minima of the non-convex problem. It replaces the \( l_0 \) norm with a
Pseudocode 3

Input: Matrix $C$, vector $Y$, sparsity level $k$, number of iterations $T$
Output: Approximation vector $X$

The IHT Method:
Set $X^{(0)} = 0$
while ($t < T$) (number of iterations)

$$X^{(t)} = P_k(X^{(t-1)} + C^T(Y - CX^{(t-1)}))$$
end while ($t < T$)

weighted $l_2$ norm, as follows [12]:

$$\min \sum_{i=1}^{N} w_i x_i^2, \quad s.t. \quad CX = Y,$$

where, the weights $w_i$ are calculated based on the previous solution so as the objective function is a first order approximation of the $l_p$ objective function ($0 \leq p \leq 1$). The new solution at $k$-th iteration is generated as follows [12]:

$$x^{(k)} = Q_n C^T (CQ_n C^T)^{-1} Y,$$

where, $Q_n$ is a diagonal matrix with entries $1/w_i = 1/((x_i^{(k-1)})^2 + \epsilon)^{p/2-1}$ and $\epsilon > 0$ is a small constant used to regularise the optimisation problem. The whole procedure is repeated a number of iterations based on the nature of the problem.

5 Experimental Results

In this Section we conduct numerical experiments to test the performance and the efficiency of the proposed heuristic. Table (1) shows the average time and the recovery error of the methods for the test run. It can be seen that the proposed heuristic performed faster than the others with better results. Notice that all the algorithms are based on non-linear problems and that all of them performed well in the under-sampled case of 70 samples. In experiments conducted, the Revised Simplex method (used for solving the $l_1$ equivalent convex problem) performed better than all the previous methods ($10^{-16}$ error) for more than 200 samples and failed in smaller sample sizes (70, 100, 150 samples), where the three methods discussed achieved very good results. However, all the three methods failed to recover a signal using less than 70 samples, which appears to be the limit for efficient recovery. All the computations were performed on an Intel Core2 Duo CPU (2 GHz) with 2 GB RAM, using Matlab R2010a under MS Windows 7 Ultimate. The whole experiment took less than 2 mins. A discrete time randomly generated signal (in vector format) of 500 entries with 10% sparsity (non-zero entries) has been used for 100 test runs with 70 samples. This simple signal was constructed using the Real Gaussian model (i.e. using Standard Normal distribution) to generate real
values between 0 and 10, which constitutes a realistic model for testing the efficiency of the methods. The signal reconstruction error is defined as \([7, 9, 11, 15, 20]\):

\[
\text{Recovery Error} = \frac{\|X - \hat{X}\|_2}{\|X\|_2},
\]

where \(X\) and \(\hat{X}\) is the original and the recovered signal, while \(\|X - \hat{X}\|_2\) stands for the Euclidean distance between these two vectors. Note that the Euclidean distance of the vector \(\|X\|_2\) is simply the square root of the sum of the squares of its elements. The CPU time was used as a rough estimation of time in secs, while 12 agents have been used by the proposed method, during this simulation.

6 Conclusions – Future work

In this article, an efficient heuristic for finding a sparse approximation of a signal, by solving an under-determined system of linear equations with non-linear objective function, has been proposed. It is based on maximising a smooth approximation of the \(l_0\) norm. Although the presented heuristic has no guarantee of achieving a global minimum as does its convex \(l_1\) analogue, the local minimum found by solving the non-convex problem in (1) typically allows for accurate and successful signal reconstruction even at much higher under-sampling rates where linear optimisation fails. Overall, the method has shown to be better in accuracy for a small number of samples and a bit faster than other alternative algorithms, without adding complexity, for the same randomly generated signal in a noiseless environment. A potential improvement of this heuristic is to re-weight the smooth \(l_0\) norm using coefficients at every iteration; a technique that has been applied successfully to similar \(l_0\) and \(l_1\) norm based CS problems \([10, 12, 17]\). The algorithm’s adaption in noisy environments constitutes another realistic improvement with much higher applicability since it is already known that the IHT and IRLS have not been extensively tested in noisy environments. Finally, potential applications of this method include the areas of signal separation, de-noising in images and signals, image sparse representation and inpainting (i.e. the process of reconstructing lost parts of images) \([15, 20]\).

Acknowledgements The author would like to particularly thank his primary supervisor, Dr. Tomasz Radzik, for his insight and constructive comments at an earlier version of this article, and the anonymous reviewers for their valuable suggestions.

References

A heuristic for sparse signal reconstruction


