Insight into Primal Augmented Lagrangian Multiplier Method

Premjith B¹, Sachin Kumar S², Akhil Manikkoth, Bijeesh T V, K P Soman
Centre for Excellence in Computational Engineering and Networking
Amrita Vishwa Vidyapeetham
Coimbatore, India
¹Email: prem.jb@gmail.com
²Email: sachinnme@gmail.com

Abstract—We provide a simplified form of Primal Augmented Lagrange Multiplier algorithm. We intend to fill the gap in the steps involved in the mathematical derivations of the algorithm so that an insight into the algorithm is made. The experiment is focused to show the reconstruction done using this algorithm.

Keywords-compressive sensing; II-minimization; sparsity; coherence

I. INTRODUCTION

Compressive Sensing (CS) is one of the hot topics in the area of signal processing[1,2,3]. The conventional way to sample the signals follows the Shannon’s theorem, i.e., the sampling rate must be at least twice the maximum frequency present in the signal (Nyquist rate) [4]. For practical signals using CS, the sampling or sensing goes against the Nyquist rate. Consider the sensing matrix, \( A \ (A \in \mathbb{R}_{nN}^{nN}) \), as the concatenation of two orthogonal matrices, \( \Psi \) and \( \phi \). A signal, \( b \in \mathbb{R}^n \) (non-zero vector) can be represented as the linear combination of columns of \( \Psi \) or as a linear combination of columns of \( \phi \). That is in \( b = \phi \beta \), \( \alpha \) and \( \beta \) are defined clearly. Taking \( \Psi \) as identity matrix and \( \phi \) as Fourier transform matrix, we can infer that \( \alpha \) is the time domain representation of \( b \) and \( \beta \) is the frequency-domain representation. Here we can see that a signal cannot be sparse in both domains at a time. So this phenomena can be extended as, for any arbitrary pair of bases \( \Psi \) and \( \phi \), either \( \alpha \) can be sparse or \( \beta \) can be sparse. Now what if the bases are same, the vector \( b \) can be constructed using the one of the columns in \( \Psi \) and get the smallest possible cardinality (smallest possible number) in \( \alpha \) and \( \beta \). The proximity between the two bases can be defined through mutual-coherence which can be defined as the maximal inner product between columns from two bases. [5,6]. The CS theory says that the signal can be recovered from very few samples or measurements. This is made true based on two principles: sparsity and incoherence.

Sparsity: a signal is said to be sparse if it is represented using much less number of sample coefficients without loss of information. The advantage is that sparsity gives fast calculation. CS exploits the fact that the natural signals are sparse in nature when expressed using proper basis [5, 6].

Incoherence: we can understand mutual-coherence between the measurement basis and sparsity basis as

\[ \mu = \sqrt{N} \max_{i,j} |\langle \phi_i, \psi_j \rangle| \text{.} \]

For matrices which are having orthonormal basis, \( \mu = 1 \). If the incoherence is higher, the number of measurements required will be smaller [7].

A. Underdetermined Linear System:

Consider a matrix \( A \in \mathbb{R}_{nN}^{nN} \) with \( n < N \), and define the underdetermined linear system of equations \( Ax = b \), \( x \in \mathbb{R}^N \), \( b \in \mathbb{R}^n \). This means that there are many ways of representing a given signal \( b \) (this system has more unknowns than equations, and thus it has either no solution, if \( b \) is not in the span of the columns of the matrix \( A \), or infinitely many solutions). In order to avoid the anomaly of having no solution, we shall hereafter assume that \( A \) is a full-rank matrix, implying that its columns span the entire space \( \mathbb{R}^n \). Consider an example of image reconstruction. Let \( b \) be an image with lower quality, and we need to reconstruct the original image, which is represented as \( x \) using a matrix \( A \). Recovering \( x \) given \( A \) and \( b \) constitutes the linear inversion problem. Compressive sensing is the method to find the sparsest solution to some underdetermined system of linear equations. In...
compressive sensing point of view $b$ is called the measurement vector. $A$ is called the sensing matrix or measurement matrix and $x$ is the unknown signal. A conventional solution is using the linear least square method which increases the computational time.

The above mentioned problem can be solved through optimization method using $l_0$-minimization.

$$\min_{x} \|x\|_0 \quad \text{Subject to} \quad Ax = b$$

The solution to this problem is to find an $x$ that will have few non-zero entries. This is known as $l_0$ norm.

The properties that a norm should satisfy are

1. Zero vectors: $\|v\| = 0$ if and only if $\forall v = 0$
2. Absolute homogeneity: $\forall t \neq 0 \ |t\| = |t| |\|u\||, t \in \mathbb{R}$
3. Triangle inequality: $\|u + v\| \leq \|u\| + \|v\|

This problem is NP-Hard. Candes and Donoho have proved if the signal is sufficiently sparse, the solution can be obtained using $l_1$-minimization. The objective function becomes,

$$\min_{x} \|x\|_1 \quad \text{Subject to} \quad Ax = b$$

Which is a convex optimization problem. This minimization tries to find out a vector $x$ whose absolute sum is small among all the possible solutions. This is known as $l_1$ norm. This problem can be recasted as the linear problem and can be tried to solve using interior-point method. But this becomes computationally complex. Most of the real world applications need to robust to noise, that is, the observation vector, $b$, can be corrupted by noise. To handle the noise a slight change in the constraint is made. An additional threshold parameter is added which is predetermined based on the noise level.

$$\min_{x} \|x\|_1 \quad \text{Subject to} \quad \|b - Ax\|_2 \leq T$$

$T > 0$ is the threshold. For the reconstruction of such signals basis pursuit denoising (BPDN) algorithm is used. [5, 9, 10].

In the light of finding efficient algorithms to solve these problems, several algorithms have been proposed like, Orthogonal Matching Pursuit, Primal-Dual Interior-Point Method, Gradient Projection, Homotopy, Polytope Faces Pursuit, Iterative Thresholding, Proximal Gradient, Primal Augmented Lagrange Multiplier, Dual Augmented Lagrange Multiplier [11]. These algorithms can work better when the signal representation is more sparse. Clearly there are infinitely many solutions out which few will give the good reconstruction. To find out a single solution, a regularization in used. Finally what is trying to achieve is a solution which has the minimum norm.

II. PRIMAL AUGMENTED LAGRANGE MULTIPLIER METHOD

Our aim is to solve the system $Ax = b$. But in practical case there could be some error. So we write $Ax \approx b$. So when it is written in equality form, it will be $Ax + r = b$, where $r$ is the error of residual. So our main objective is to minimize $x$ and $r$, such that $x$ is as sparse as possible. So we take $l_1$ norm of $x$.

The objective function is,

$$\min_{x, r} \|x\| + \frac{1}{2\mu} \|r\|^2$$

Subject to, $Ax + r = b$

Where $X \in C^n$, $r \in C^m$

Take the Lagrangian of the form,

$$L(x, r, y) = \min_{x, r} \|x\| + \frac{1}{2\mu} \|r\|^2 - \text{Re}(y^*(Ax + r - b)) +$$

$$\frac{\beta}{2} \|Ax + r - b\|^2$$

Where, $y \in C^m$ is a multiplier and $\beta > 0$ is a penalty parameter. Since $y \in C^m$ instead of $y^T$ we use $y^*$ (conjugate transpose).

Using iterative methods, for a given $(x, y)$ we obtain $(x^{k+1}, y^{k+1}, r^{k+1})$. We use alternating minimization method to achieve this. i.e., for fixed values of two variables we minimize the other. First, fix $x = x^k$ and $y = y^k$ and find $r^{k+1}$.

Minimize $L(x, r, y)$ with respect to $r$.
Find the derivative of $L(x, r, \mu)$ with respect to $r$

$$\frac{\partial L(x, r, \mu)}{\partial r} = \frac{\partial}{\partial r} \left[ \frac{1}{2} \|x\|_1 + \frac{1}{2} \|y\|_2^2 \right] + \frac{\partial \text{Re}(y^*(Ax + r - b))}{\partial r} + \frac{\beta \partial \|Ax + r - b\|_2^2}{2}$$

(3)

Omit all the terms independent of $r$.

$\|r\|^2$ can be considered as $r^T r$

So, eq. (3) will be,

$$\frac{\partial L(x, r, \mu)}{\partial r} = \frac{1}{2\mu} \frac{\partial}{\partial r} (yr^T) - \frac{\partial \text{Re}(y^*(Ax + r - b))}{\partial r} + \frac{\beta \partial \|Ax + r - b\|_2^2}{2} = 0$$

$$\Rightarrow \frac{2r}{2\mu} - \frac{\partial (yr^T)}{\partial r} + \frac{\beta \partial ((Ax + r) - b)^T ((Ax + r) - b)}{2} = 0$$

$$\Rightarrow \frac{r}{\mu} - \frac{\beta}{2} \frac{\partial ((Ax + r)^T (Ax + r) - 2b(Ax + r)^T)}{\partial r} = 0$$

$$\Rightarrow \frac{r}{\mu} - \frac{\beta}{2} \frac{\partial ((Ax + r)^T (Ax + r) + 2T (Ax + r)^T r - 2b(Ax + r)^T)}{\partial r} = 0$$

$$\Rightarrow \frac{r}{\mu} - \frac{\beta}{2} \frac{\partial ((Ax + r)^T (Ax + r) + 2r^T (Ax + r) + r^T r - 2b(Ax + r)^T)}{\partial r} = 0$$

$$\Rightarrow \frac{r}{\mu} - \frac{\beta}{2} (2Ax + 2r - 2b) = 0$$

$\Rightarrow \frac{r}{\mu} - y + \beta(Ax + r - b) = 0$

Now, fix $r = r^{k+1}$ and $y = y^k$. Do minimization of the Lagrangian with respect to $x$.

$$\min_x \|x\|_1 + \frac{1}{2\mu} \|r\|^2 - \text{Re}(y^*(Ax + r - b)) + \frac{\beta}{2} \|Ax + r - b\|^2$$

(5)

Omit all the terms independent of $x$.

$$\Rightarrow \min_x \|x\|_1 - \text{Re}(y^*(Ax)) + \frac{\beta}{2} (Ax + r - b)^T (Ax + r - b)$$

$$= \min_x \|x\|_1 - \text{Re}(y^*(Ax)) + \frac{\beta}{2} ((Ax + r)^T (Ax + r) - 2b^T (Ax + r))$$
Here we assume \( A^T A = I \)

\[
\min_x \|x\|_1 + \beta \left( \frac{1}{2} \left( T x + 2 r A x - 2 b A x - 2 y A x \right) / \beta \right)
\]

This can be written as,

\[
\min_x \|x\|_1 + \beta \left( \frac{1}{2} \left( A x + r x + k + b - y \right) / \beta \right)^2
\]

Since (7) is in quadratic form, we can approximate it by first two terms of Taylor series.

\[
\min_x \|x\|_1 + \beta \left( \Re(g^k) * (x - k) \right) + \frac{1}{2} \left( x - k \right)^2
\]

Where \( \tau > 0 \) is a proximal parameter and,

\[
k^+ = \text{Shrink} \left( k - \tau g^k, \frac{\tau}{\beta} \right)
\]

Finally update the multiplier \( y \) by fixing, \( r = r^+ + k + 1 \) and \( x = x^+ + k + 1 \).

Since the Lagrangian is quadratic in \( y \) and for any fixed \( x^+ \) and \( r^+ \),

\[
y^+ = y - \gamma \beta \left( A x^+ + r^+ + k + 1 - b \right)
\]

Where \( \gamma > 0 \) is a constant.

So the PALM algorithm in 3 steps is,

1. \[
k^+ = \left( \frac{\mu \beta}{1 + \mu \beta} \right) \left( y - (A^k x - b) \right)
\]

2. \[
x^+ = \text{Shrink} \left( k - \tau g^k, \frac{\tau}{\beta} \right)
\]

3. \[
y^+ = y - \gamma \beta \left( A x^+ + r + k + 1 - b \right)
\]

III. RESULTS

Primal Augmented Lagrangian Multiplier algorithm is tested on three different images of size 256x256 pixels. Then Gaussian, Salt & pepper and Speckle noises with varying percentage of intensities are added onto these images and reconstruct the image using PALM algorithm. PSNR, RMSE and execution time are calculated in each experiment and reported in the tables Table1 and Table2.

In the case of first two images, PSNR is less and RMSE is more for original image. But in the case of
third image, when we add 20% Gaussian noise, it shows good recovery.

IV. CONCLUSION

We have presented a simple mathematical derivation for Primal Augmented Lagrangian Multiplier method which is considered as one of the best compressive sensing algorithms. PALM can be used for reconstructing various signals with varying noise intensities. Experiments show that this algorithm gives very good reconstruction result.

REFERENCES

[1] D. L. Donoho, Compressed Sensing, Manuscript, September 2004.
[2] Compressive Sensing Resources, http://dsp.rice.edu/cs
[3] Rauhut, H. (2010). Compressive sensing and structured random matrices, in Theoretical Foundations and Numerical Methods for Sparse Recovery, edited by Fornasier, M. Radon Series Comp. Appl. Math., Volume 9, pp. 1–92. deGruyter.
[4] Abdul J.Jerri,”The Shannon Sampling Theorem –Its various Extensions and Applications: A Tutorial Review”, IEEE Proc. Vol.65, November 1977
[5] Michael Elad, “From Theory to Applications in Signal and Image Processing”, Springer Text Book, 2010
[6] Candès, Emmanuel J. and Wakin, Michael B. (2008) An Introduction To Compressive Sampling [A sensing/sampling paradigm that goes against the common knowledge in data acquisition]. IEEE Signal Processing Magazine, 25 (2). pp. 21-30. ISSN 1053-5888
[7] E. Candès, “Compressive Sampling,” Proc. Int’l Congress of Mathematicians, 2006.
[8] A. Bruckstein, D. Donoho, and M. Elad. “From sparse solutions of systems of equations to sparse modeling of signals and images” (in press) SIAM Review, 2007
[9] E. Candès, J. Romberg, and T. Tao “Stable signal recovery from incomplete and inaccurate measurements” Communications on Pure and Applied Math, 59(8):1207–1223, 2006
[10] S. Chen, D. Donoho, and M. Saunders “Atomic decomposition by basis pursuit” SIAM Review, 43(1):129–159, 2001
[11] 11.http://www.eecs.berkeley.edu/~yang/software/l1benchmark.

Figure 1: (a) Original Image (b) Reconstructed Image
Figure 2: Added with 2% Gaussian noise
Figure 3: Added with 5% Gaussian noise
Figure 4: Added with 10% Gaussian noise
Figure 6: Added with 20% Gaussian noise

Figure 7: Added with 2% Salt & pepper noise

Figure 8: Added with 5% Salt & pepper noise

Figure 9: Added with 10% Salt & pepper noise

Figure 10: Added with 20% Salt & pepper noise

Figure 11: Added with 2% Speckle noise

Figure 12: Added with 5% Speckle noise

Figure 13: Added with 10% Speckle noise

Figure 14: Added with 20% Speckle noise

Figure 15: Added with 2% Gaussian noise
Figure 16: Added with 5% Gaussian noise

Figure 17: Added with 10% Gaussian noise

Figure 18: Added with 20% Gaussian noise

Figure 19: Added with 2% Salt & pepper noise

Figure 20: Added with 5% Salt & pepper noise

Figure 21: Added with 10% Salt & pepper noise

Figure 22: Added with 20% Salt & pepper noise

Figure 23: Added with 2% Speckle noise

Figure 24: Added with 5% Speckle noise

Figure 25: Added with 10% Speckle noise
Figure 26: Added with 20% Speckle noise

Figure 27: Added with 2% Gaussian noise

Figure 28: Added with 5% Gaussian noise

Figure 29: Added with 10% Gaussian noise

Figure 30: Added with 20% Gaussian noise

Figure 31: Added with 2% Salt & pepper noise

Figure 32: Added with 5% Salt & pepper noise

Figure 33: Added with 10% Salt & pepper noise

Figure 34: Added with 20% Salt & pepper noise

Figure 35: Added with 2% Speckle noise
TABLE 1: PSNR, RMSE AND EXECUTION TIME OF ORIGINAL IMAGES

| Image name | PSNR (dB) | RMSE   | Time (Sec) |
|------------|-----------|--------|------------|
| Image_1    | 17.8557   | 32.6403| 20.940833  |
| Image_2    | 14.0948   | 50.3272| 21.732722  |
| Image_3    | 10.5039   | 76.0933| 26.502142  |

TABLE 2: PSNR, RMSE AND EXECUTION TIME OF ORIGINAL IMAGES WHEN ADDED WITH NOISES OF VARYING INTENSITIES

| Image name | Noise     | % of error | PSNR (dB) | RMSE   | Time (Sec) |
|------------|-----------|------------|-----------|--------|------------|
| Image_1    | Gaussian  | 2          | 15.721    | 41.734 | 23.74473   |
|            |           | 5          | 15.77     | 41.4995| 19.613224  |
|            |           | 10         | 15.7262   | 41.7089| 25.044838  |
|            |           | 20         | 15.7814   | 41.445 | 26.502142  |
|            | Salt & pepper | 2       | 16.8149   | 36.7956| 28.512797  |
|            |           | 5          | 15.5122   | 42.7494| 22.795465  |
|            |           | 10         | 14.009    | 50.8263| 27.19995   |
|            |           | 20         | 11.9891   | 64.1333| 27.10357   |
| Image_2    | Speckle   | 2          | 16.7394   | 37.1169| 23.400962  |
|            |           | 5          | 15.5125   | 42.7481| 22.950852  |
|            |           | 10         | 13.987    | 50.9556| 22.495303  |
|            |           | 20         | 12.0597   | 63.6144| 23.600862  |
| Image_3    | Gaussian  | 2          | 13.0878   | 56.5134| 28.182993  |
|            |           | 5          | 13.1057   | 56.3966| 24.715946  |
|            |           | 10         | 13.0987   | 56.4423| 24.711972  |
|            |           | 20         | 13.0964   | 56.4571| 26.883013  |
|            | Salt & pepper | 2       | 13.6543   | 52.9451| 26.492682  |
|            |           | 5          | 13.0383   | 56.8364| 21.902348  |
|            |           | 10         | 12.2250   | 62.4152| 23.998090  |
|            |           | 20         | 10.8429   | 73.1803| 26.126452  |
|            | Speckle   | 2          | 10.4489   | 76.5762| 21.228414  |
|            |           | 5          | 10.3024   | 77.8793| 22.401311  |
|            |           | 10         | 10.0795   | 79.9031| 24.353466  |
|            |           | 20         | 9.6208    | 84.2366| 24.639351  |