Performance Analysis of Adaptive Beamforming Algorithms for Smart Antennas

Prerna Saxena*, A.G. Kothari

Department of Electronics Engineering, VNIT, Nagpur-440010, India

Abstract

In this paper, adaptive beamforming techniques for smart antennas based upon Least Mean Squares (LMS), Sample Matrix Inversion (SMI), Recursive Least Squares (RLS) and Conjugate Gradient Method (CGM) are discussed and analyzed. The beamforming performance is studied by varying the element spacing and the number of antenna array elements for each algorithm. These four algorithms are compared for their rate of convergence, beamforming and null steering performance (beamwidth, null depths and maximum side lobe level).

© 2014 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/3.0/).
Selection and peer review under responsibility of Information Engineering Research Institute

Keywords: Smart antenna; adaptive beamforming algorithms; null steering

1. Introduction

A smart antenna system is a multi-element antenna where the signals received at each antenna element are intelligently combined to improve the performance of the wireless system. Smart antennas can increase signal range, reduce signal fading, suppress interfering signals, and increase the capacity of wireless systems. The block diagram of a smart antenna system is shown in Fig.1 where signals received by the antenna array are multiplied by adjustable weights and then combined to produce the system output. The processor receives...
array signals, system output, and direction of the desired signal as additional information. These are used by the processor to calculate the weights to be used for each channel [2][9].

![Basic Adaptive Beamformer](image)

**Fig. 1. Basic Adaptive Beamformer**

Beamforming is the process of forming the radiation pattern of the antenna array by nulling out the interference and pointing the beam in the direction of the user. Fixed beamforming is applied to fixed arrival angle sources. However, if the angles of arrival of the sources change with time, fixed beamforming cannot be used. The optimum array weights need to be continuously adapted to the ever-changing environment. This process is known as adaptive beamforming. An adaptive array system consists of antenna array elements terminated in an adaptive processor which is designed to update and compensate the array weights as the source moves. There are two basic adaptive approaches [1]: 1. Block Adaptation, where a temporal block of data is used to estimate the optimum array weights and 2. Continuous Adaptation, in which the weights are adjusted as the data is sampled such that the weight vector converges to the optimum solution. Beamforming algorithms used to control the smart antenna patterns are based upon certain criteria like minimizing the variance, maximizing the signal to interference ratio, minimizing the mean square error, etc. In this paper, adaptive beamforming techniques based upon Least Mean Squares (LMS), Sample Matrix Inversion (SMI), Recursive Least Squares (RLS) and Conjugate Gradient Method (CGM) are discussed and analyzed in section 2. Section 3 presents MATLAB simulations of these algorithms and discusses the advantages and disadvantages of each algorithm. Section 4 concludes the paper.

**2. Adaptive beamforming algorithms**

**2.1. LMS Algorithm**

The LMS algorithm [2-8] was introduced by Widrow. In this algorithm, the weights are updated at every iteration by estimating the gradient of the quadratic Mean Square Error (MSE) surface, and then moving the weights in the negative direction of the gradient by a small amount, known as the step size. The convergence of this algorithm is directly proportional to the step-size parameter $\mu$. When the step size is within a range that ensures convergence, the process leads the estimated weights to the optimal weights. Stability is ensured provided that the following condition is met [3].

$$0 \leq \mu \leq \frac{1}{2\lambda_{\text{max}}}$$

(1)
where $\lambda_{\text{max}}$ is the largest eigen value of the array correlation matrix $R_{xx}(k)$ which is given by equation 2.

\[ R_{xx}(k) = x(k)x^H(k) \]  

(2)

$x(k)$ denotes the received signal vector. The array weights are updated according to the following equation:

\[ w(k+1) = w(k) + \mu e^*(k)x(k) \]  

(3)

where the error signal is given by equation 4.

\[ e(k) = d(k) - w^H(k)x(k) \]  

(4)

2.2. SMI Algorithm

The SMI algorithm [2][9-10] is based on block adaptation. The weights for the $k^{th}$ block of length $K$ are calculated by using equation 5.

\[ w(k) = R_{xx}^{-1}(k)r(k) \]  

(5)

where $R_{xx}(k)$ is the array correlation matrix given by equation 6.

\[ R_{xx}(k) = \frac{1}{K} \sum_{k=1}^{K} x(k)x^H(k) \]  

(6)

where $K$ is the observation interval.

This algorithm is suitable for a rapidly changing environment as it converges much faster than the LMS algorithm, thereby allowing the tracking of the desired signal. However, computational complexity and matrix singularities can cause problems.

2.3. RLS Algorithm

The convergence speed of LMS algorithm depends on the eigen values of the array correlation matrix. In a fast changing mobile environment which yields an array correlation matrix with large eigen value spread, the LMS algorithm converges with a slow speed. This problem is solved with the RLS algorithm [2] by replacing the gradient step size $\mu$ with a gain matrix $R^{-1}(k)$ at the $k^{th}$ iteration, producing the weight update equation given by equation 7.

\[ w(k) = w(k-1) - R^{-1}(k)x(k)e^*(w(k-1)) \]  

(7)

2.4. CGM Algorithm

CGM algorithm[2][9][3][11][12] increases the rate of convergence by iteratively searching for the optimum solution by choosing conjugate (perpendicular) paths for each new iteration. Thus, the path taken for the $(n+1)^{th}$ iteration is perpendicular to that for the $n^{th}$ iteration. The weights are updated according to the following equation

\[ w(n+1) = w(n) - \mu(n)D(n) \]  

(8)

$\mu(n)$ is the step size and $D(n)$ is the direction vector.

3. Simulation Results and Analysis

In this section, the beamforming algorithms discussed in section 2 are simulated and analyzed. A uniform linear array with operating frequency of 2.4 GHz is considered for simulations. The desired signal is taken as a cosine signal at an angle of 15°. Two interfering signals are considered at angles of -15° and 30°. The effect
of variation of the distance between two antenna elements (d) is analyzed as shown in Fig. 2. When the antenna element spacing is less than 0.5 of the wavelength (\( \lambda \)), the side lobe level (SLL) is very high and null depth is very low as shown in Fig. 2 for \( d=0.3\lambda \). If the antenna element spacing (d) is increased above 0.5\( \lambda \), the beamwidth decreases resulting in higher directivity. However, the null depth decreases as shown in Fig. 2 for \( d=0.7\lambda \). When \( d=0.5\lambda \), the SLL is low and the null depth is highest resulting in greatest interference suppression capability. Thus, \( d=0.5\lambda \) is the optimum separation distance between two antenna elements. The result shown in Fig. 2 is for the LMS algorithm, however, it is found true for all algorithms and thus d is considered as 0.5\( \lambda \) in all further simulations. Further, the effect of number of antenna elements and convergence speed of each algorithm are discussed.

![Fig. 2. Effect of variation of antenna array spacing on beamforming](image)

### 3.1 Analysis of Beamforming

The following section shows the effect of variation of the number of antenna elements (M) on the beamforming of LMS, SMI, RLS and CGM algorithms. The beam is steered at 15° and nulls are formed along the directions of the interferers at -15° and 30°.

![Fig. 3. (a) Normalized Array Factor Plot for LMS algorithm; (b) Normalized Array Factor Plot for SMI algorithm](image)
Table 1. Beamforming Analysis for LMS algorithm

<table>
<thead>
<tr>
<th>M</th>
<th>Beamwidth (degree)</th>
<th>Max. SLL (dB)</th>
<th>Null Depth at -15°(dB)</th>
<th>Null Depth at 30°(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>13.181</td>
<td>-12.55</td>
<td>-35.66</td>
<td>-34.44</td>
</tr>
<tr>
<td>16</td>
<td>6.88</td>
<td>-13.54</td>
<td>-34.05</td>
<td>-37</td>
</tr>
<tr>
<td>20</td>
<td>5.16</td>
<td>-13.09</td>
<td>-30.22</td>
<td>-39.29</td>
</tr>
</tbody>
</table>

Table 2. Beamforming Analysis for SMI algorithm

<table>
<thead>
<tr>
<th>M</th>
<th>Beamwidth (degree)</th>
<th>Max. SLL (dB)</th>
<th>Null Depth at -15°(dB)</th>
<th>Null Depth at 30°(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>12.605</td>
<td>-9.435</td>
<td>-31.8</td>
<td>-35.33</td>
</tr>
<tr>
<td>16</td>
<td>5.73</td>
<td>-9.553</td>
<td>-27.04</td>
<td>-31.69</td>
</tr>
<tr>
<td>20</td>
<td>5.16</td>
<td>-12.571</td>
<td>-28.91</td>
<td>-26.05</td>
</tr>
</tbody>
</table>

Fig. 4. (a) Normalized Array Factor Plot for RLS algorithm; (b) Normalized Array Factor Plot for CGM algorithm

Table 3. Beamforming Analysis for RLS algorithm

<table>
<thead>
<tr>
<th>M</th>
<th>Beamwidth (degree)</th>
<th>Max. SLL (dB)</th>
<th>Null Depth at -15°(dB)</th>
<th>Null Depth at 30°(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>6.88</td>
<td>-9.977</td>
<td>-27.94</td>
<td>-33.79</td>
</tr>
<tr>
<td>20</td>
<td>5.16</td>
<td>-8.374</td>
<td>-17.68</td>
<td>-21.28</td>
</tr>
</tbody>
</table>

Table 4. Beamforming Analysis for CGM algorithm

<table>
<thead>
<tr>
<th>M</th>
<th>Beamwidth (degree)</th>
<th>Max. SLL (dB)</th>
<th>Null Depth at -15°(dB)</th>
<th>Null Depth at 30°(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>6.31</td>
<td>-10.05</td>
<td>-48.5</td>
<td>-36.89</td>
</tr>
<tr>
<td>20</td>
<td>5.16</td>
<td>-10.07</td>
<td>-47.91</td>
<td>-34.23</td>
</tr>
</tbody>
</table>

As shown in Fig.3 and Fig.4, as the number of antenna elements increases, the beamwidth decreases, thus increasing the directivity of the antenna array. The SLL decreases with increase in the number of antenna
array elements as shown in Table 1 and Table 2. The SMI algorithm is found to have a narrower beamwidth and a higher directivity as compared to the LMS algorithm. Also, the maximum SLL is higher for the SMI algorithm as compared to the LMS algorithm. These results are shown in Fig.3, Table 1 and Table 2. The beamwidth and SLL obtained for the RLS algorithm is slightly higher than that for the SMI algorithm as shown in Fig.3 and Fig.4. However, the RLS algorithm overcomes the computational complexity and singularity problems associated with the correlation matrix in the SMI algorithm. Fig. 4(b) and Table 4 show the beamwidth, maximum SLL and the null depths obtained for the CGM algorithm. It is seen that the CGM algorithm gives the maximum null depths at the two interferences. Beamwidth obtained for CGM is similar to that for LMS. However, the SLL is slightly higher than that for LMS.

3.2 Analysis of Convergence Speed

Next, the convergence speed of the algorithms is discussed. For this analysis, \( M=8 \) and \( d=0.5\lambda \) are chosen.

3.2.1 LMS Algorithm

Fig.5(a) shows that the weights of LMS algorithm converge to their optimum solution after 60 iterations. If the signal characteristics are rapidly changing due to high mobility rate of the user, then the LMS algorithm may not be able to track the signal due to its slow convergence rate. This is the main drawback of this algorithm.

3.2.2. SMI Algorithm

The SMI algorithm is used to overcome the slow convergence of the LMS algorithm. For \( M=8 \) and \( d=0.5\lambda \), the SMI pattern is similar to the LMS pattern as shown in Fig.2. The SMI pattern was generated with no iterations. The block size \( K \) for the SMI algorithm was chosen to be 30 which is less than the number of iterations (60) taken by the LMS algorithm to achieve convergence.

3.2.3. RLS Algorithm

Fig.5(b) shows that the weights of the RLS algorithm converge to their optimum values in just 15 iterations. This convergence is much faster than LMS (60 iterations) and SMI (30 snapshots).

3.2.4. CGM Algorithm

The plot of the norm of the residual is shown in Fig.5(c). It can be seen that the residual drops to almost zero in 5 iterations only. This shows that the convergence of CGM algorithm is fastest as compared to all other algorithms. This is because CGM is an accelerated gradient based approach as discussed in section 2. The convergence results are summarized in Table 5.
Table 5. Rate of convergence for each algorithm

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>No. of iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS</td>
<td>60</td>
</tr>
<tr>
<td>SMI</td>
<td>No iteration, block size 30</td>
</tr>
<tr>
<td>RLS</td>
<td>15</td>
</tr>
<tr>
<td>CGM</td>
<td>5</td>
</tr>
</tbody>
</table>

4. Conclusion

In this paper, four adaptive beamforming algorithms have been simulated, analyzed and compared on the basis of beamforming (Beamwidth, max SLL and Null Depth) and rate of convergence. It is shown that d=0.5λ is the optimum spacing between the antenna array elements. The effect of number of antenna elements (M) on beamforming is analyzed and it is seen that with increase in M, the beamwidth reduces; making the array more directional and the SLL reduces, thus improving beamforming. It is seen that the LMS algorithm has good performance except that it has a very slow rate of convergence. SMI improves the convergence speed at the cost of more computational complexity and singularity problem of correlation matrix. It is shown that RLS algorithm overcomes the problems of SMI and improves the rate of convergence of LMS at the cost of higher SLL and lower null depths. It is observed that the CGM algorithm has the fastest convergence and greatest null depths ensuring good performance.

References