Chapter 4

Large-MIMO Detection via Cuckoo Search

In this chapter, the cuckoo search algorithm by Ramanathan and Jayakumar (2015a) is employed to solve the detection problem in large-MIMO systems. The detection problem in MIMO systems can be thought of solving an optimization problem with finite alphabet constraints. The main motivation behind using cuckoo search is because of its two search capabilities - local search and global search, which is controlled by a discovery probability. This facilitates a better search space exploration and an increased likelihood of finding the global optima. The highlights of the work in this chapter are as follows.

- A discrete space cuckoo search is presented and it is applied to the large-MIMO detection.

- The cuckoo search detector is validated in $8 \times 8$, $16 \times 16$ and $32 \times 32$ large-MIMO systems with various degrees of spatial correlation.

- The BER performance and complexity analysis is presented and compared with
the benchmark lattice reduction aided detector.

Cuckoo search (CS) is a recent addition to the class of nature inspired metaheuristic algorithms to deal with global optimization problems. Cuckoo search idealizes the breeding behavior of species of cuckoo birds, and it can be applied to solve various real time optimization problems. Studies have shown that particle swarm optimization can converge quickly to the current best solution, but not necessarily the global best solutions. Encouragingly, studies by Civicioglu and Besdok (2011) have revealed that cuckoo search can outperform particle swarm optimization in most of the continuous optimization problems. Since its development by Yang and Deb (2009), CS has evolved to be a powerful method and seems to outperform other metaheuristic algorithms in most of the applications across various engineering disciplines. A good description of the cuckoo search is available in Yang (2010); Yang and Deb (2014). An efficient implementation of cuckoo search is provided in Yang and Deb (2010). Modified versions improving the convergence properties are available in Walton et al. (2011) and Zheng and Zhou (2012). Cuckoo search has been applied even to multi-objective optimization by Yang et al. (2011). A brief literature review of cuckoo search is available in Fister Jr. et al. (2014).

Cuckoo search (CS) has found its way in a wide range of applications and we provide a brief survey of them in this section. Cuckoo search has been used to solve reliability optimization problems in Valian et al. (2013) and optimal allocation in distribution networks in Moravej and Akhlaghi (2013). Additionally, the cuckoo search has been
used to solve problems in computer science like selecting the optimal solution in seman-
tic web service composition by Chifu et al. (2012), structural software testing for
code-based criteria in software testing by Srivastava et al. (2012) and Knapsack prob-
lems by Layeb (2011). Cuckoo search has been employed in wireless sensor networks
for data gathering by Dhivya and Sundarambal (2011) and data fusion by Dhivya et al.
(2011). Further, works in Valian et al. (2013) have been carried on applying CS for
feed-forward efficient neural network training. In all the above applications, cuckoo
search has proved to be an effective solution over the existing counterparts in terms
of finding the global optima and computational complexity. This stands a motivation
for employing cuckoo search for the current research problem, detection in large-MIMO
systems.

The chapter is organized as follows. The subsequent section 4.1 presents the cuckoo
search algorithm with its modules for large-MIMO detection. In section 4.2, the perfor-
mance of cuckoo search detector in correlated and uncorrelated large-MIMO systems
are presented. The summary of the results are available in section 4.3.

4.1 Cuckoo Search for Large-MIMO Detection

In this section, the solution to detection in large-MIMO systems via cuckoo search is
presented. In general, this algorithm can be applied to any discrete optimization prob-
lem, more precisely an integer least squares problem. This algorithm of cuckoo search
includes some modifications to the algorithm in Yang (2010) and it has been modified
to accommodate finite alphabet constraints. It is well suited for the application: MIMO
detection. The algorithm with its modules is described in detail as follows.

4.1.1 Neighborhood Matrix

A discrete search space is defined by the constellation set $S^{N_r}$. The constellation set consists of complex entries and the transmitted symbol takes a point from this set. For example, with 4-QAM modulation, the set $S^{N_r}$ is defined as follows.

$$S^{N_r} = \{1 + i, -1 + i, -1 - i, 1 - i\}$$

The neighborhood of any element in the set is defined based on the Euclidean distance. For example, the neighborhood vector of the alphabet $(1 + i)$ is defined by $\{-1 + i, -1 - i, 1 - i\}$, where the neighborhood vector is of size 3. It can be defined accordingly for size of two or one. However the neighborhood vector size is limited by $M - 1$, where $M$ is the cardinality of the set $S^{N_r}$. In this way, the neighborhood matrix can be formed, in which each row corresponds to a neighborhood vector of an alphabet in the constellation set $S^{N_r}$. There will be totally $M$ rows in this neighborhood matrix.

For ease of implementation, the neighborhood matrix can be extended to accommodate the alphabet itself as an index. This is accomplished by placing the alphabet as the first entry in each row, followed by the corresponding neighborhood vector. In other words, the first column of the neighborhood matrix contains the alphabet itself. In the example, for neighborhood vector size of 3, the dimension of the neighborhood matrix will be a $4 \times 4$ matrix.
4.1.2 Get a Cuckoo

This role of this function in the algorithm is to get a candidate solution. A candidate solution is termed as cuckoo and the vector containing the solution is called a nest. The main difference between the algorithm in this section and the conventional cuckoo search is in the process of getting a cuckoo (solution) and subsequent evaluation. Since the search space is discrete in this case, the Lévy flight or random walk is not used. The random walk or Lévy flight is suitable for continuous optimization or integer optimization. In this case, the solution space is a finite set and it is better to perform a neighborhood search. But this may increase the complexity as a $1 - opt$ search is performed every time to get a solution. Instead, a simple technique is presented, in which the random walk is used for getting an index of a solution and not the solution. The search space is defined by the neighborhood matrix and the following procedure is adopted to get a cuckoo.

1. The current solution in a nest say $C$ is obtained.

2. An integer random number say $R$ in the interval two to maximum size of neighborhood vector is generated. In the example, $R \in \{2, 3, 4\}$.

3. The row in the neighborhood matrix with $C$ as its first entry is traced and the $R^{th}$ element in this row is picked as the required solution.

4. The current solution is replaced with this new solution.
It is to be noted that the random number generated in step 2 above is not the solution. It is only an index which is used to pick a solution from the matrix. The row value is given by the current solution and column value is given by this random number.

4.1.3 Objective Function

The objective function used to evaluate the fitness of the solution vectors can be directly obtained from the problem defined in (2.8). By expanding the terms and ignoring the terms that does not contribute to the optimization, the simplified objective function is derived.

\[ \phi(x) = y^H y + x^H H^H H x - 2 \Re\{y^H H x\} \]  
\[ \text{(4.1)} \]

ignoring the first term, which is independent of \( x \) results in,

\[ \phi(x) = x^H H^H H x - 2 \Re\{y^H H x\} \]  
\[ \text{(4.2)} \]

The solution that minimizes the expression above is a solution to the problem in (2.8).

\[ \hat{x} = \arg \min_{\tilde{x} \in S^N_T} \tilde{x}^H H^H H \tilde{x} - 2 \Re\{y^H H \tilde{x}\} \]  
\[ \text{(4.3)} \]

The cuckoo search is employed with the objective function given by (4.2). This will be the fitness function using which the solutions are to be evaluated. The output of cuckoo search is the solution vector \( \hat{x} \).

4.1.4 Cuckoo search algorithm

There are several variants of the cuckoo search algorithm and the simplest form of cuckoo search with the modifications discussed above was implemented. A detailed
description of conventional cuckoo search and its implementation is given by [Yang and Deb (2009, 2010)], and the crux of the algorithm used in this work is provided here. The description of the CS algorithm are presented below.

*Input:* The received vector $y$, the channel matrix $H$, Neighborhood matrix.

*Output:* Detected transmitted vector $\hat{x}$.

1. The stopping criteria, nest size and the discovery probability (probability of abandoning the worst nests) are fixed.

2. An initial population of cuckoos are generated by using the ‘get a cuckoo’ algorithm with zero forcing solution or a random vector as initial solution.

3. A new solution is generated using ‘get a cuckoo’ algorithm and its fitness is evaluated using the objective function.

4. For a randomly chosen nest, the existing solution is compared and replaced with the new solution if it is a minimizer compared to the existing solution.

5. Based on the discovery probability, some of the solutions are ignored and these nests are replaced with newly generated solutions by using the `get a cuckoo` algorithm.

6. Finally, the solutions are ranked in terms of its ML cost and the solution with the least ML cost is chosen as the current best.

7. The above steps are repeated until stopping criteria - either iterations or tolerance. The final best solution is output as the detected symbol vector.
In the above algorithm, the stopping criteria can be either number of iterations or the tolerance value. If the number of iterations is chosen, then the algorithm terminates at a fixed point irrespective of whether the global solution is reached or not. If the tolerance value is used as a stopping criteria, then in certain cases, the iterations fails to converge. This happens in MIMO detection when the system is ill conditioned i.e. the channel matrix is poor. This is likely to happen due to spatial correlation in the channels. Hence, it is recommended to use the number of iterations. In the simulations to follow, the number of iterations is fixed to be 20% of the total number of solutions in the solution space.

The nest size in the algorithm will determine the length of the solution vector and number of solution vectors. It should be chosen as $N_T \times M$, where $N_T$ and $M$ refer to number of transmitting antennas and cardinality of constellation set respectively. This is required to make an initial assessment and further carry on with the optimization. Since, a local neighborhood search is performed, the choice of the initial solution affects the convergence of the algorithm. It is observed that the zero forcing solution provides a good start yielding better convergence.

The discovery probability is a critical factor affecting the convergence of the algorithm. For a discovery probability of 0.25, the local search is very intensive with about 25% of the search time and the global search takes about 75% of the total search time. Fixing the probability is found be crucial and is fixed empirically in the range of 0.15 - 0.45.
4.2 Performance in Large-MIMO systems

In this section, the simulation results of the cuckoo search detector in large-MIMO systems are analyzed. The CS detector is employed in various MIMO configurations like $8 \times 8$, $16 \times 16$ and $32 \times 32$ MIMO systems. The neighborhood size is set to be 3 in all the simulations. The performance is evaluated in uncorrelated and correlated channels as well. The correlated channel is characterized with the correlation coefficient $p$ taking the values 0, 0.3, 0.5 and 0.7 for channels with no, low, medium and high correlation. The benchmark detector in this case is LR aided MMSE-SIC in [Zhou and Ma (2013a)].

4.2.1 Performance comparison in uncorrelated channels

Fig. 4.1 shows the performance in $8 \times 8$ MIMO. The CS detector offers a gain of around 1 dB in low SNR regime and around 2 dB in medium SNR. The performance of CS and LR-MMSE-SIC detectors tends to close-in as SNR increases. Despite this, the CS detector ensures a minimum gain of 1 dB - 2 dB in $8 \times 8$ MIMO system.

Fig. 4.2 shows the performance comparison in a $16 \times 16$ MIMO system. The trend is encouraging with the gain of CS detector increasing with increasing number of antennas, i.e. higher order MIMO system. For instance, a BER of $10^{-2}$ is achieved at an SNR of 9.25 dB by CS detector, whereas the LR-MMSE-SIC detector requires and SNR of 13 dB. This means that the gain of CS over LR detector is around 3.75 dB. We observe that the gain in low SNR is minimum 1 dB and the gain increases with increase in SNR.

From Fig. 4.3 it is observed that in a $32 \times 32$ MIMO system, the CS detector shows a consistent superior performance over LR-MMSE-SIC detector. The gain is around 4
Figure 4.1: BER performance comparison of CS & LR-MMSE-SIC detectors in 8 × 8 MIMO with 4-QAM in uncorrelated channels.

Figure 4.2: BER performance comparison of CS & LR-MMSE-SIC detectors in 16 × 16 MIMO with 4-QAM in uncorrelated channels.
dB for SNRs greater than 7 dB. For instance, to achieve a BER of $10^{-2}$ is 9 dB and 13 dB for CS and LR-MMSE-SIC detectors respectively. Similarly, to achieve a BER of $10^{-3}$, LR-MMSE-SIC detector requires SNR of 16 dB, whereas the CS detector requires only 12 dB. This implies that the gain of CS detector for achieving a BER of $10^{-2}$ and $10^{-3}$ is around 4 dB demonstrating the consistency in performance gain. The large system behavior of metaheuristic search has been illustrated by this trend.

In summary, the BER performance results confirms and demonstrates the effectiveness of the CS detector compared to LR aided detectors in uncorrelated large-MIMO systems.
4.2.2 Performance comparison in correlated channels

In this section, the performance of cuckoo search detector is analyzed in spatially correlated large-MIMO systems.

![Graph showing BER performance comparison](image)

Figure 4.4: BER performance comparison of CS & LR-MMSE-SIC detectors in 8 × 8 MIMO with 4-QAM in low correlated channels.

The figures 4.4 to 4.6 depicts the BER performance in 8 × 8 MIMO under correlated channels. In low correlated channels \((p = 0.3)\), the CS detector has a gain of around 1.75 dB in moderate SNR and 2.5 dB in high SNR. However, the low SNR performance is similar for both CS and LR-MMSE-SIC detectors. In medium correlated channels the SNR gain for CS detectors is around 2.75 dB to 3 dB. For instance, the CS detector achieves a BER of \(10^{-3}\) at 15 dB and LR-MMSE-SIC requires 18 dB to achieve the same performance. The CS detector exhibits a consistency in performance, offering a gain of 3 dB in high correlated channels as well. The discussion suggest that an average
Figure 4.5: BER performance comparison of CS & LR-MMSE-SIC detectors in 8 × 8 MIMO with 4-QAM in medium correlated channels.

Figure 4.6: BER performance comparison of CS & LR-MMSE-SIC detectors in 8 × 8 MIMO with 4-QAM in high correlated channels.
gain of around 3 dB can be expected in CS detector in correlated channels for an 8 × 8 MIMO system.

In Fig. 4.7, the performance in 16 × 16 MIMO systems with low correlation is analyzed. The results are encouraging in the sense that the performance gain of CS detector has increased to around 1 dB in low SNR regime, 4 dB for medium SNRs and 6 dB for high SNRs. The gain of the CS detector increases as the size of MIMO increases exhibiting the large system behavior. A similar trend is noticed for medium and high correlated channels in a 16 × 16 MIMO system depicted in Fig. 4.8 and Fig. 4.9 respectively. The CS detector consistently outperforms the LR-MMSE-SIC detector with approximately 4 dB gain for SNR in the range of 12 - 18 dB and 6 dB gain in the region of above 18 dB SNR. The degradation in the performance from medium to
Figure 4.8: BER performance comparison of CS & LR-MMSE-SIC detectors in $16 \times 16$ MIMO with 4-QAM in medium correlated channels.

Figure 4.9: BER performance comparison of CS & LR-MMSE-SIC detectors in $16 \times 16$ MIMO with 4-QAM in high correlated channels.
high correlation is around 3.5 dB for LR and 2.5 dB for CS detector. This suggests that the degradation rate of CS detector is low for increased correlation compared to LR detector.

![BER performance comparison of CS & LR-MMSE-SIC detectors in 32x32 MIMO with 4-QAM in low correlated channels.](image)

**Figure 4.10:** BER performance comparison of CS & LR-MMSE-SIC detectors in 32x32 MIMO with 4-QAM in low correlated channels.

**Table 4.1:** Performance and Average running Time (ART) comparison of the CS & LR-MMSE-SIC detectors in 8x8, 16x16 and 32x32 large-MIMO systems.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>SNR</th>
<th>ART</th>
<th>SNR</th>
<th>ART</th>
<th>SNR</th>
<th>ART</th>
<th>SNR</th>
<th>ART</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>8 x 8 MIMO</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LR-MMSE-SIC</td>
<td>12.4</td>
<td>4.4E-6</td>
<td>13</td>
<td>4.4E-6</td>
<td>16</td>
<td>4.4E-6</td>
<td>17.5</td>
<td>4.4E-6</td>
</tr>
<tr>
<td>CS</td>
<td>10</td>
<td>1.9E-5</td>
<td>10.5</td>
<td>1.9E-5</td>
<td>13</td>
<td>1.9E-5</td>
<td>14.5</td>
<td>1.9E-5</td>
</tr>
<tr>
<td><strong>16 x 16 MIMO</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LR-MMSE-SIC</td>
<td>13</td>
<td>6.3E-6</td>
<td>14.2</td>
<td>6.3E-6</td>
<td>18</td>
<td>6.3E-6</td>
<td>20.7</td>
<td>6.3E-6</td>
</tr>
<tr>
<td>CS</td>
<td>9.2</td>
<td>3.5E-5</td>
<td>9.6</td>
<td>3.5E-5</td>
<td>13.25</td>
<td>3.5E-5</td>
<td>15.7</td>
<td>3.5E-5</td>
</tr>
<tr>
<td><strong>32 x 32 MIMO</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LR-MMSE-SIC</td>
<td>13.25</td>
<td>9.8E-6</td>
<td>15</td>
<td>1.3E-5</td>
<td>20</td>
<td>1.6E-5</td>
<td>24</td>
<td>2.8E-5</td>
</tr>
<tr>
<td>CS</td>
<td>8.8</td>
<td>5.3E-5</td>
<td>9.5</td>
<td>5.3E-5</td>
<td>13.25</td>
<td>5.3E-5</td>
<td>16.25</td>
<td>5.3E-5</td>
</tr>
</tbody>
</table>
Figure 4.11: BER performance comparison of CS & LR-MMSE-SIC detectors in $32 \times 32$ MIMO with 4-QAM in medium correlated channels.

Figure 4.12: BER performance comparison of CS & LR-MMSE-SIC detectors in $32 \times 32$ MIMO with 4-QAM in high correlated channels.
The BER performance in a 32 × 32 MIMO system are presented in the Figs. 4.10 to 4.12. Unlike the uncorrelated channels, where the performance in 16 and 32 antenna systems were similar, the performance in correlated channels show a different pattern. The performance gap between CS and LR-MMSE-SIC detectors increases in the case of correlated channels and increase in SNR. The degradation in performance is slower in CS compared to LR-MMSE-SIC detector. In addition, the gap in performance increases with increase in correlation and SNR. We corroborate that the CS detector outperforms LR-MMSE-SIC detector in all correlated cases with significant SNR gain.

The performance and complexity comparison of the CS and LR-MMSE-SIC detectors for various MIMO configurations and correlation conditions are presented in Table 4.1. The major observations are as follows.

1. It is observed that the CS detector exhibits large system behavior, i.e. the BER performance improves as the MIMO size increases.

2. The SNR gap between the LR-MMSE-SIC and CS detector increases with MIMO size and correlation. The instance of \( p = 0.7 \) in 32 × 32 MIMO system is a good example for this aspect.

3. The degradation in performance form uncorrelated to correlated scenarios is also at a slower rate for CS detector compared to the LR-MMSE-SIC detector.

4. The computational complexity of CS detector is slightly higher compared to LR-MMSE-SIC detector. However, this is tolerable for the performance gain it offers.
The CS detector is proposed to overcome the shortcoming of the SDR detector in high correlated channels. For a BER of $10^{-2}$, the CS detector offers a gain of around 1.3 to 1.75 dB compared to SDR detector and at the same time having an ART which is reduced by a factor of 100. In short, the performance in highly correlated channels of CS detector is slightly higher than the LD-LR detector with slightly lower complexity. This significant performance improvement of CS detector is due to the characteristics of metaheuristics algorithms: intensification and diversification. Intensification intends to search around the current best solutions and select the best candidates or solutions, while diversification makes sure that the algorithm can explore the search space efficiently. In addition, substantial fractions of the new solutions are generated by far field randomization, which is neighborhood search in our discrete version of CS and whose locations are far enough from the current best solution. This will make sure the algorithm will not be trapped in a local optimum.

4.3 Summary

A novel cuckoo search approach for detection in large-MIMO systems was presented by modifying the existing algorithm suitably for discrete space optimization problems. In general, CS can be applied to any integer least squares problem. The algorithm was applied to the detection problem in MIMO wireless systems and studied. The performance comparison for large-MIMO systems in uncorrelated channels and correlated channels with various degrees of correlation was presented. The cuckoo search detector performs better than the existing methods, in particular better than the lattice reduction aided
detection i.e. LR-MMSE-SIC.

In large-MIMO systems, in uncorrelated channels, the CS detector ensures a minimum gain of $1 \, \text{dB} - 2 \, \text{dB}$ in $8 \times 8$ MIMO system. The trend is encouraging with the gain of CS detector increasing with increasing number of antennas approaching a gain slightly higher than $3 \, \text{dB}$ in $16 \times 16$ systems and nearly $4 \, \text{dB}$ in $32 \times 32$ MIMO system.

In correlated channels, the results suggest that an average gain of around $3 \, \text{dB}$ is expected in CS detector in correlated channels for an $8 \times 8$ MIMO system. The CS detector consistently outperforms the LR-MMSE-SIC detector with a minimum gain of $4 \, \text{dB}$ in $16 \times 16$ MIMO. In $32 \times 32$ MIMO system, the performance gap between CS and LR-MMSE-SIC increases in the case of correlated channels and increase in SNR. In addition, it is observed that the degradation in performance is at a slower rate in CS detector compared to LR-MMSE-SIC detector.

Based on the simulation results, the cuckoo search detector is a commendable choice, especially in spatially correlated MIMO channels. It can replace the semidefinite relaxation detector in terms of complexity and performance particularly in high correlated channels. The cuckoo search can be fused with other metaheuristics algorithm like tabu search to further reduce the complexity. Currently the CS detector is presented for 4-QAM constellations, and the performance in higher order constellations needs to be verified. As the constellation size increases, the dimension of the neighborhood matrix increases. Limiting the neighborhood vector size limits the performance of the CS detector. This aspect needs further investigation.