

A NOVEL ULTRA-FAST ULTRA-SIMPLE ADAPTIVE BLIND BEAMFORMING ALGORITHM FOR SMART ANTENNA ARRAYS

A. N. Jabbar*

University of Babylon, Hilla, Babylon, Iraq

Abstract—This research proposes a new blind tracking algorithm for smart antenna arrays by switching the main beam iteratively using the cost calculated from the received and predicted symbol. This algorithm will be called Cost Steering Algorithm Using Demodulation-Remodulation Technique *COSTAUS/DRT*. It is completely independent of the Least Mean Squares (*LMS*) or any derived version from it, and it does not need to investigate the cyclostationary properties of the incoming signal. A complete derivation and analytical model with simulation using Simulink is given in this research. The algorithm was tested under three different target motions which are a triangular motion (linear), sinusoidal motion (circular) and saw tooth motion which is an adverse case when the linear motion changes its path suddenly. The transmitter uses 16-level PSK signal with no Forward Error Correction code (*FEC*) in order to test the algorithm under the worst situation. The algorithm is tested under different noise power levels. The antenna array is a linear array with 16-elements.

1. INTRODUCTION

Smart antenna arrays have become an essential part in modern wireless communication systems. The power of smart antenna comes from the fact that it can steer and reshape its radiation pattern to maximize Signal to Interference and Noise Ratio (*SINR*) [1–5]. This is done electronically using beamforming algorithms without the involvement of the mechanical parts to steer the array [6–10]. The beamforming algorithms are classified mainly into three types: (1) estimating the Angle Of Arrival (*AOA*) algorithms like MUSIC and

Received 15 September 2011, Accepted 20 October 2011, Scheduled 2 November 2011

* Corresponding author: Ahmed N. Jabbar (Ahmed_Aljafari@Yahoo.com).

ESPRIT [9, 11–13], (2) non-blind algorithms where the transmitter and the receiver agree to certain code to detect the location of the transmitter to steer the main beam like Maximum Likelihood (*ML*), and Minimum Variance Distortionless Response (*MVDR*) [7], (3) blind beamforming algorithm. These algorithms are widely used and difficult to be built and developed. In the blind algorithms, the receiver has no idea or clue on how and where the transmitter is located. The receiver should estimate and steer blindly towards the transmitter. These algorithms are based on the maximization of *SINR*, or sometimes called training, by detecting the cyclostationary or modulus properties for a vector of stored transmitted symbols to predict *AOA* [14–17]. Least Mean Square *LMS* algorithm is the base for most of the known adaptive beamformers. This algorithm suffers from high computation load because *LMS* needs, theoretically, infinite vector length, high complexity, time consuming, and singularities. Modified algorithms were developed to reduce the computational load by reducing the required vector length such as the Recursive *LMS* (*R-LMS*). Nevertheless, this algorithm should wait until the required number of symbols is assembled to perform training [9, 18]. Combining this algorithm with modulus detecting algorithms such as Constant Modulus Algorithm (*CMA*), Decision Direct Algorithm (*DDA*), or Conjugate Gradient Method (*CGM*) will produce the blind adaptive beamforming algorithms such as *LMS-CMA*, *LMS-DDA*, *RLMS-CMA*, ... [16–19].

In sensitive communication systems, such as military, these difficulties represent obstacles that should be solved by simplifying the beamforming algorithm. Hence, the need to develop new approaches becomes persistent. Researchers proposed new systems that are not based on *LMS* but still they need to store a vector of symbols to perform beamformation [20–27]. One of the interesting approaches uses the Particle Swarm Optimization (*PSO*) to accelerate the beamforming algorithm training by maximizing a suitable fitness function to find the required phase weights to steer the main beam [28–30]. Various versions of the *PSO* were developed to increase the speed, reduce the required memory, and maximize *SINR* to accommodate the requirements of real-time applications [31–33].

The algorithm presented in this research is completely independent of the *LMS*. The approach is simply based on the difference or the cost between the received and detected symbols at the receiver without the need to store a vector of symbols.

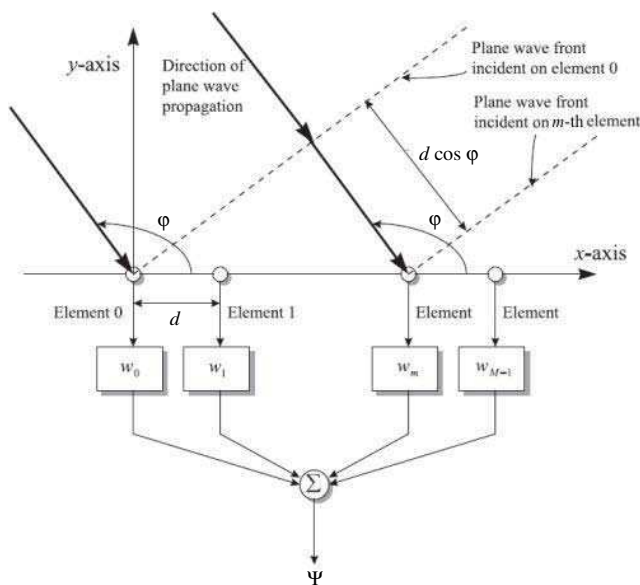


Figure 1. Linear M -elements array antenna.

2. FORMULATING THE ALGORITHM

2.1. The Misalignment Effect

Assume a receiver that implements smart antenna to receive a transmitted signal as shown in Figure 1. The linear antenna array consists of M number of elements with inter-element distance d , and an incident wave at an angle of φ .

The Array Factor AF for this array is [18–20]

$$\mathbf{AF} = \sum_{j=0}^{M-1} e^{i(M-j-1)kd \cos \varphi} \tag{1}$$

where k is the wavenumber $= \frac{2\pi}{\lambda}$.

If a transmitted message $\alpha(t)$ is mapped into a complex symbol $s = s_R + s_I$, where s_R is the real part and s_I the imaginary part, the complex symbol is received at an angle φ , then the output of the above array is (neglecting the noise)

$$\Psi = (s_R + s_I) \cdot (\mathbf{w}^H \cdot \mathbf{AF}) \tag{2}$$

The weighting factor or phase shifters w_j are used to neutralize the effect of φ or in other words to steer the main lobe to the direction

of arrival. These values are calculated as the Hermitian (complex conjugate referred to by H) of an angle φ_w as follows:

$$w_j = \sum_{j=0}^{M-1} e^{-i(M-j-1)kd \cos \varphi_w} \tag{3}$$

If $\varphi_w = \varphi$ then $(\sum \mathbf{w} \cdot \mathbf{A}\mathbf{F} = 1)$ and leaves only the complex symbol which is $s = s_R + s_I$. Assume that $\varphi_w \neq \varphi$, then Equation (2) becomes

$$\begin{aligned} \Psi &= (s_R + s_I) \cdot \sum_{j=0}^{M-1} e^{-i(M-j-1)kd \cos \varphi_w} \cdot \sum_{j=0}^{M-1} e^{i(M-j-1)kd \cos \varphi} \\ &= (s_R + s_I) \cdot \sum_{j=0}^{M-1} e^{i(M-j-1)kd(\cos \varphi - \cos \varphi_w)} \\ &= (s_R + s_I) \cdot (A_R + A_I) = s_R^D + s_I^D \end{aligned} \tag{4}$$

where A_R and A_I are the real and imaginary parts of the array residuals, and s_R^D, s_I^D are the real and imaginary parts of the received symbol s^D at the array output that will be the fed to the demodulator in the receiver.

2.2. Cost Calculation Using Demodulation Remodulation Technique

The algorithm is shown as a block diagram in Figure 2.

The demodulator output is the most probable message $\alpha^D(t)$ corresponding to the array output Ψ as shown in Figure 3.

The error added by the misalignment of the array with the transmitter angle will lead to an erroneous detection of the received symbol ($\alpha^D \neq \alpha$). The procedure to calculate the cost is as follows: (1) calculate Ψ to identify α^D , (2) remodulate α^D using the same array at the receiver to generate Ψ_D , and then subtract Ψ_D from Ψ and take the absolute value to find the cost.

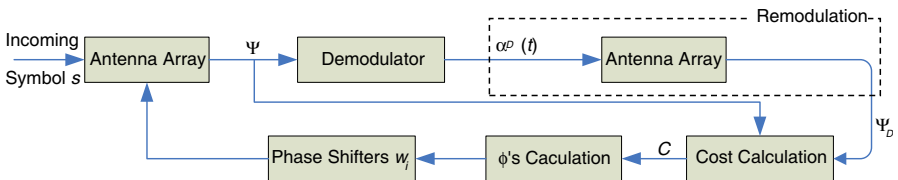


Figure 2. The algorithm block diagram.

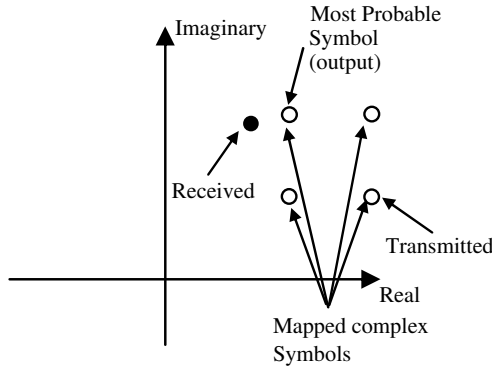


Figure 3. The error in detection process due to misalignment.

Converting the above block diagram into a mathematical model

$$\alpha^D \text{ remodulate} \Rightarrow s_R^{Dr} + s_I^{Dr} \tag{5}$$

where s_R^{Dr} and s_I^{Dr} are the real and imaginary parts of the remodulated message.

Multiplying Equation (5) by Equation (3) results in

$$\Psi_D = (s_R^{Dr} + s_I^{Dr}) \cdot \sum_{j=0}^{M-1} e^{-i(M-j-1)kd \cos \varphi_w} = S_R^{Dr} + S_I^{Dr} \tag{6}$$

To calculate the cost C

$$\begin{aligned} C &= \sum_{\text{all elements}} |\Psi - \Psi_D| = |(s_R^D + s_I^D) - (S_R^{Dr} + S_I^{Dr})| \\ &= |((s_R^D - S_R^{Dr}) + (s_I^D - S_I^{Dr}))| \end{aligned} \tag{7}$$

If a proper alignment is acquired ($\varphi_w = \varphi$) then $C = 0$ otherwise $C \neq 0$. The value of C will play a central role in switching the beam gradually towards the direction of transmission.

2.3. The Generation of Φ Set

In order to steer the beam iteratively in the direction of the transmitter, three new angles will be generated using the recent value of φ_w that are $\Phi = \varphi_w$, $\Phi^+ = \varphi_w + C$, and $\Phi^- = \varphi_w - C$. After this step Equation (7) is used to calculate the cost C , C^+ , and C^- corresponding to Φ , Φ^+ , and Φ^- , respectively. Any angle of the calculated Φ s with the lowest cost indicates that it is the nearest angle to the transmitter direction. Take the value of the angle with the lowest cost and let ($\varphi_w = \Phi$).

Next, find a new set of three angles Φ , Φ^+ , and Φ^- . Repeating this process iteratively will steer the beam towards the transmitter because the nearest angle will always be chosen. When total alignment occurs, then C will be zero, and all the angles will merge into one value Φ which is the angle of the transmitter $\Phi = \varphi$.

2.4. Dead Locks or Trap States

Although the above algorithm seems to be straightforward and simple to be implemented, it suffers from dead locks or trapping states. To explain a trapping state, assume that the current Φ has the lowest cost and $\Phi \neq \varphi$. At the same time, Φ^+ and Φ^- having C^+ and C^- , respectively, are both greater than C . Thus the next iteration will select the value of Φ as the nearest angle to the transmitter, and the same values for Φ^+ and Φ^- will be recalculated. This situation will continue indefinitely, which is called a trapping state or a dead lock.

To step out this trapping state, a simple modification in calculating the value of C is introduced to ensure that the algorithm will not fall in a dead lock. This modification is accomplished by generating a random variable ζ provided that $(0 \leq \zeta \leq \pi)$ and adding it to the calculated value of C in the following manner

$$C' = C \cdot \zeta \quad (8)$$

Equation (8) ensures that the next decision for the angles differs from the previous one. Also, the value of ζ is related to C in such a way that as $(\Phi \rightarrow \varphi)$ then $(C \cdot \zeta \rightarrow 0)$. The calculation of the angles set, $\Phi = \varphi_w$, $\Phi^+ = \varphi_w + C'$, and $\Phi^- = \varphi_w - C'$, will also coincide with one angle that is φ .

2.5. Number of Decisions Per Symbol

The algorithm should ensure a total alignment with the transmitter. If a single decision is made for the received symbol, the probability of having an erroneous decision will increase and affect the future received symbol because misalignment adds extra (real and imaginary) error to s^D as in Equation (4). To avoid this case, the algorithm should make enough iterations per symbol which means repeating Equations (5) to (7) for N number of times such that $N > 1$. The repetition ensures that the main beam shifts from its current position to meet the transmitter angle as fast and accurate as possible. Approaching transmitter angle ensures a maximum *SINR*.

The value of N should be chosen carefully depending on the environment that the receiver is working within to maximize the probability of the last decision to be correct. If a faulty value of N

is chosen (high or low) the algorithm may either works slowly, or the shift towards the transmitter is not enough to minimize the Symbol Error Rate *SER*.

3. THE MODEL OF THE ALGORITHM

The simulated system containing the algorithm is shown in Figure 4. It consists of three main parts: (1) a transmitter, (2) an antenna array, and (3) the receiver that hosts the proposed algorithm.

The transmitter section generates data messages $\alpha(t)$, then encodes them using PSK modulation scheme, generates the complex symbol $s = s_R + s_I$, and adds the AWGN noise to the transmitted symbol. The resultant transmitted symbol appears at (SymGen1) pin. The motion of the transmitter is fed to the next section, (Smart Antenna Array), using (MRA out) pin. The (Smart Antenna Array) generates the added phases corresponding to every element and feed

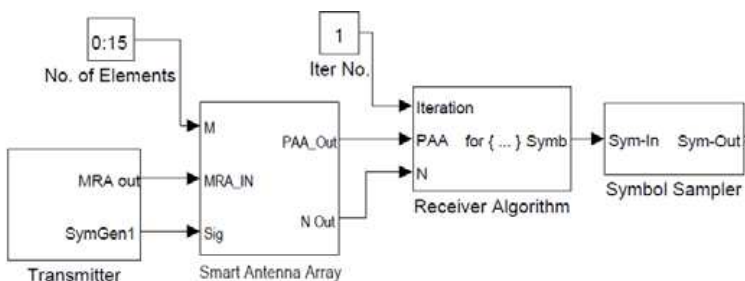


Figure 4. The proposed system simulink model.

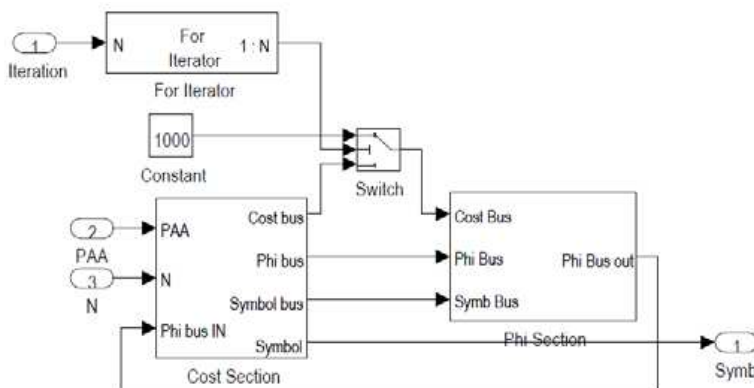


Figure 5. Algorithm simulation.

them to the receiver using (PAA_Out).

The model of the algorithm is shown in Figure 5, which consists of two sections: (1) cost calculation and (2) Phi calculation. The value of the iterations N is set externally as shown in Figure 4 (Iter No.). The switch is used to reset the system after each symbol period in order to clean any residuals from the previous calculations and starts a new estimation.

The inside of the cost section is shown in Figure 6.

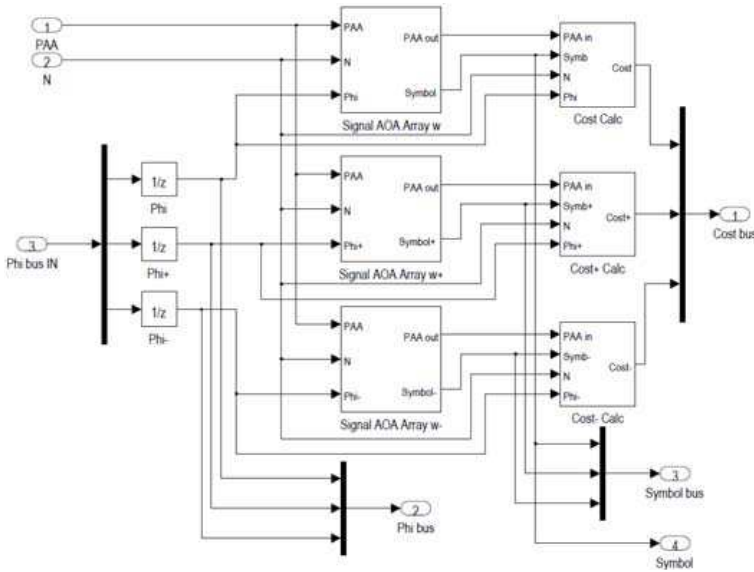


Figure 6. Cost calculation section.

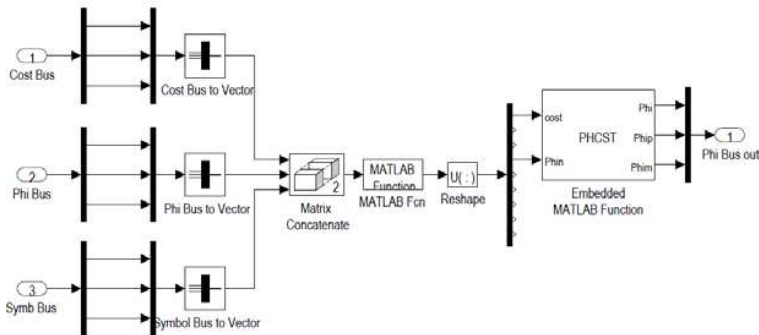
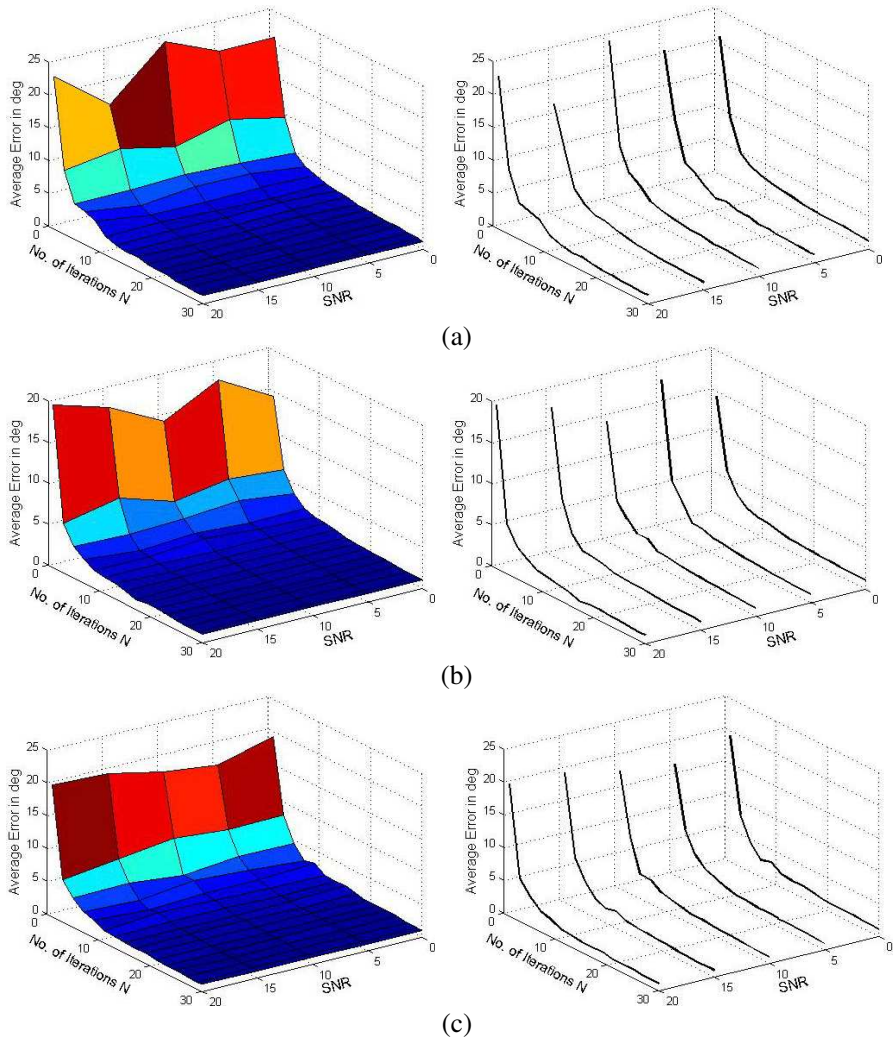


Figure 7. Φ -calculation section.

The first three blocks (Signal AOA Array w) are identical and used for the demodulation. The next three blocks (Cost Calc) are used to calculate the cost for each generated Φ with its corresponding decoded symbol. Each one consists of a remodulator which remodulates the input symbol (Symb) using the generated set of Φ s (Phi) and applies Equations (7) and (8) to calculate the cost for the next iteration. All costs are multiplexed on a single bus in order to be fed to the next section which is shown in Figure 7.

The Φ calculation section is shown in Figure 7. The cost, Phi,



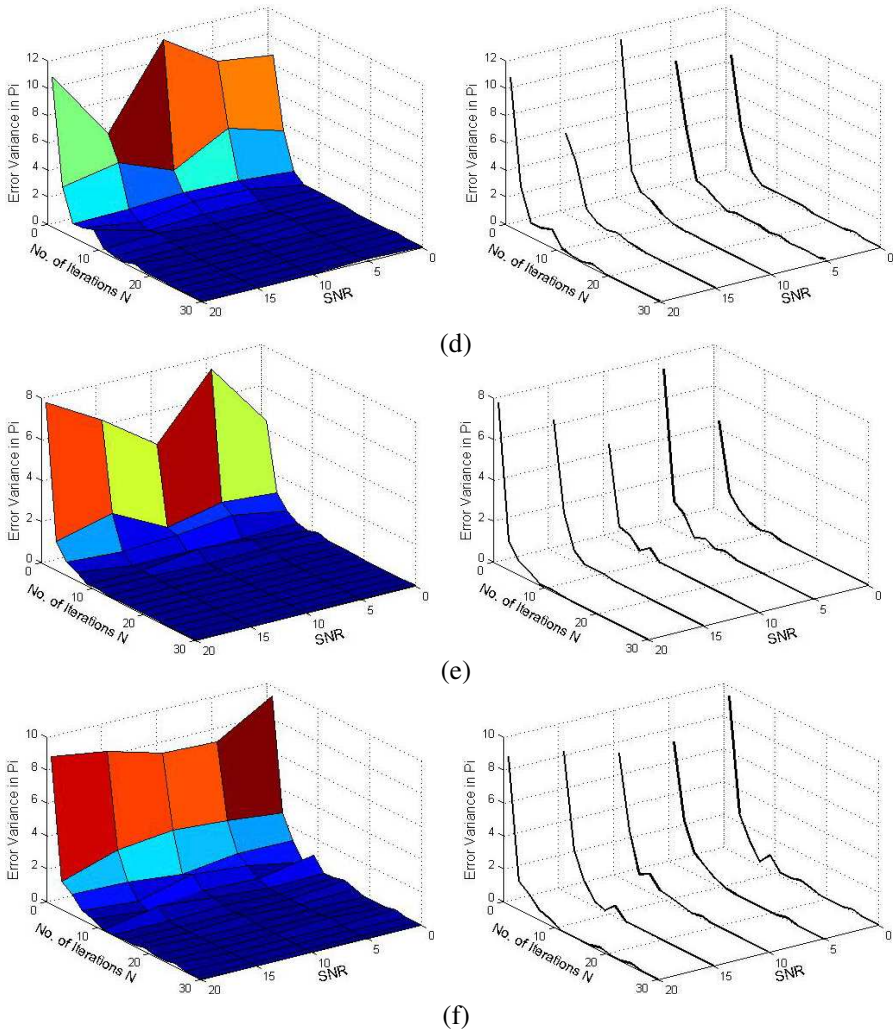


Figure 8. Statistical results for the algorithm performance. (a)–(c) Average misalignment error for the circular, linear and saw tooth motion as a surface and curves. (d)–(f) Dispersion measure (variance) of the misalignment for the circular, linear and saw tooth motion as a surface and curves.

and symbol buses are concatenated and sorted ascendingly according to the cost. The value of Φ corresponding to the lowest cost is fed to PHCST block to calculate the Φ set for the next iteration (Section 2.4). Here $\Phi \equiv \Phi$, $\Phi^+ \equiv \Phi^+$, and $\Phi^- \equiv \Phi^-$.

4. SIMULATION SETTINGS AND RESULT

The settings for the simulated system are as follows: (1) the data source is a uniform random integer generator with 16 M-ary (0–15) used to generate $\alpha(t)$, (2) a PSK modulator has 16 levels, (3) the channel is AWGN, (4) the array is linear with 16 elements with $d = \lambda/2$, (5) the angle of motion is restricted between 10° to 170° , and (6) the value of φ (MRA) is set to take three motion types which are triangular, sinusoidal, and saw tooth as an extreme scenario.

Table 1. The average misalignment error and variance numerical value for the three motion types at different SNR levels and iterations values N .

Motion types	Misalignment average error in (deg)			
	SNR 0 dB		SNR 20 dB	
	$N = 1$	$N = 30$	$N = 1$	$N = 30$
Circular	20.914	1.154	23.212	1.279
Linear	14.378	1.037	19.972	1.044
Saw tooth	19.209	0.996	20.142	1.065

Motion types	Misalignment error variance in (180°)			
	SNR 0 dB		SNR 20 dB	
	$N = 1$	$N = 30$	$N = 1$	$N = 30$
Circular	8.699	0.01453	11	0.05383
Linear	4.374	0.01011	7.943	0.01004
Saw tooth	9.38	0.0085	9.009	0.0098

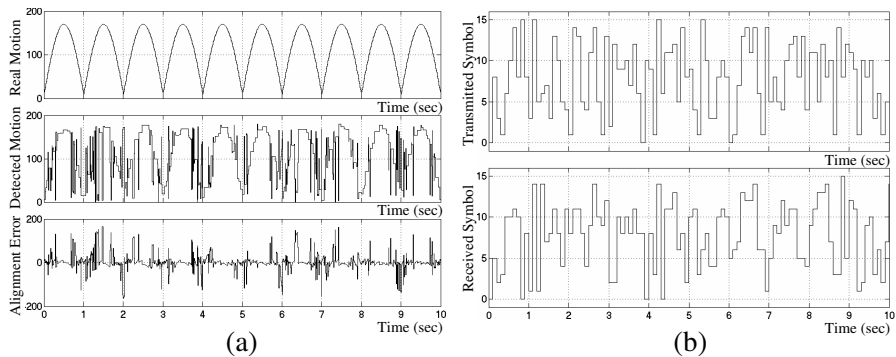


Figure 9. System performance at $SNR = 0$, $N = 1$, $SER = 80$. (a) Motion in degrees. (b) Symbols.

The algorithm was tested under different levels of SNR (0–20 dB). At each value of SNR , the number of iterations N is varied from 1–30. The average error (in degrees) and error variance (in π , 180°) were calculated using 10,000 samples for each point. The algorithm performance for the three motion types is shown in Figure 8.

Figure 8 shows that the algorithm performance is independent of SNR level because the average error and variance are almost constant along the SNR axis at the same N value. While the average error and variance drop significantly with the increase of N value. Also, we can see that the motion type has no effect on the algorithm performance because the performance planes (Figure 8) for each motion type are approximately the same. Table 1 gives a numerical comparison for the three motion types.

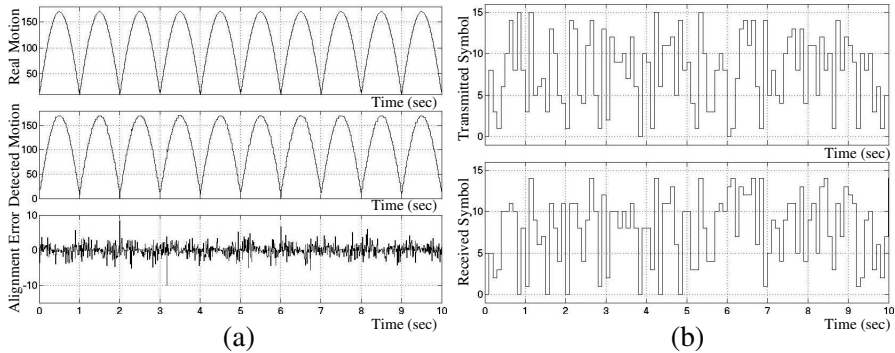


Figure 10. System performance at $SNR = 0$, $N = 30$, $SER = 74$. (a) Motion in degrees. (b) Symbols.

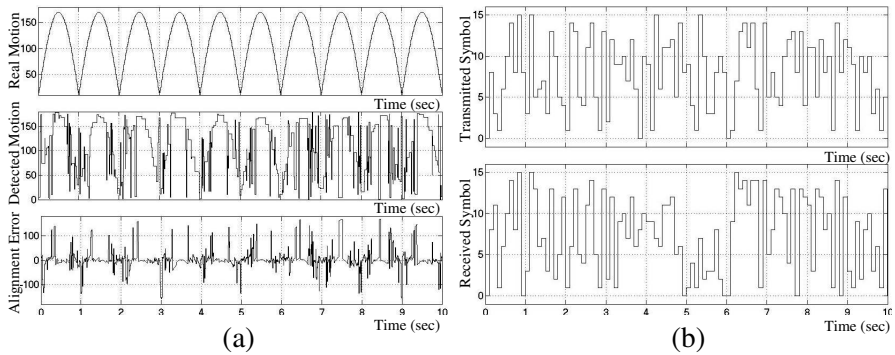


Figure 11. System performance at $SNR = 20$, $N = 1$, $SER = 33$. (a) Motion in degrees. (b) Symbols.

The simulated tracking results for the algorithm are shown in Figures 9 to 20 with SE_R for each case. The total transmitted symbols are 100 with a symbol period 0.1.

4.1. Circular Motion

Circular motion is simulated at the transmitter using a periodic half sine wave. The results are shown in Figures 9 to 12.

4.2. Linear Motion

This motion is simulated using a simple triangular motion for the transmitter. The results are shown in Figures 13 to 16.

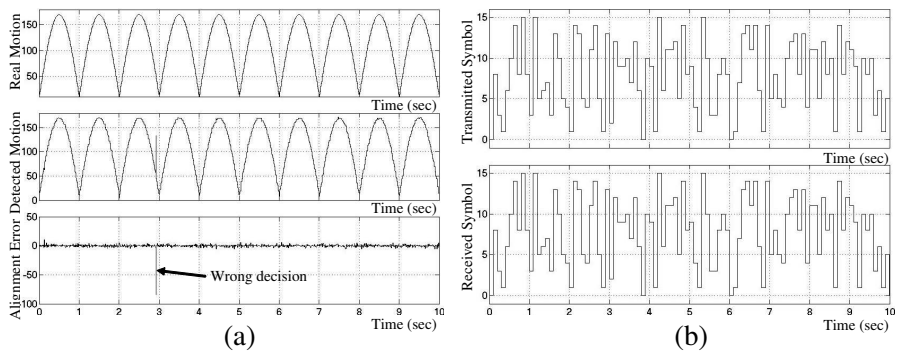


Figure 12. System performance at $SNR = 20$, $N = 30$, $SE_R = 0$. (a) Motion in degrees. (b) Symbols.

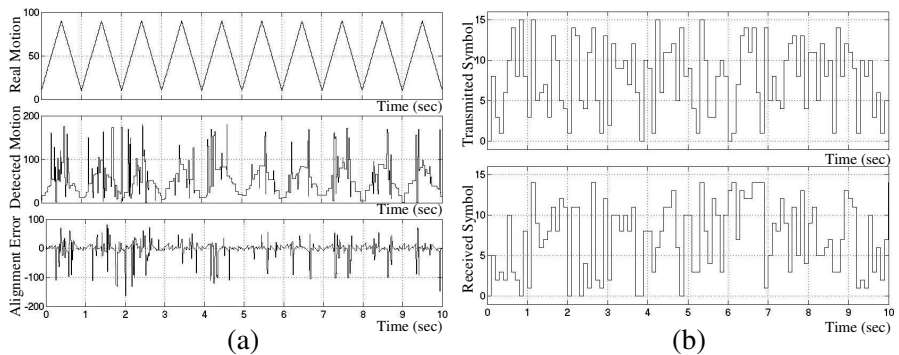


Figure 13. System performance at $SNR = 0$, $N = 1$, $SE_R = 76$. (a) Motion in degrees. (b) Symbols.

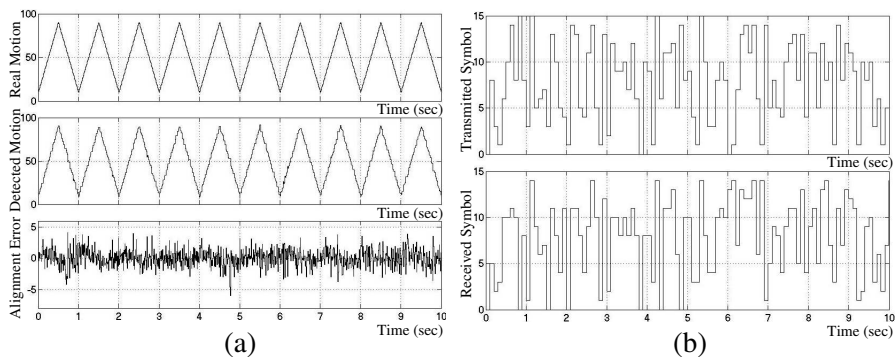


Figure 14. System performance at $SNR = 0$, $N = 30$, $SER = 74$. (a) Motion in degrees. (b) Symbols.

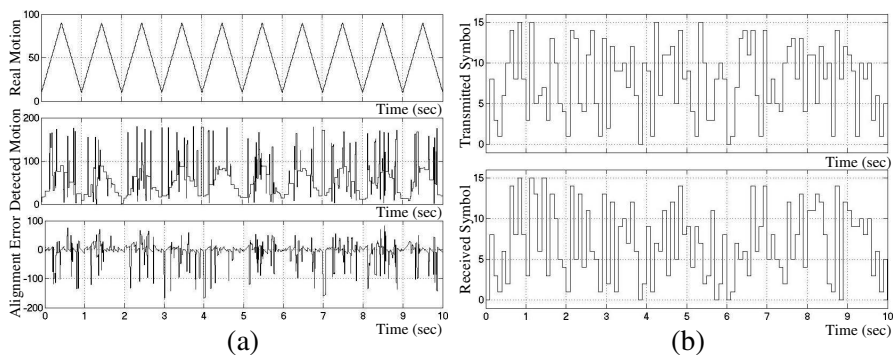


Figure 15. System performance at $SNR = 20$, $N = 1$, $SER = 26$. (a) Motion in degrees. (b) Symbols.

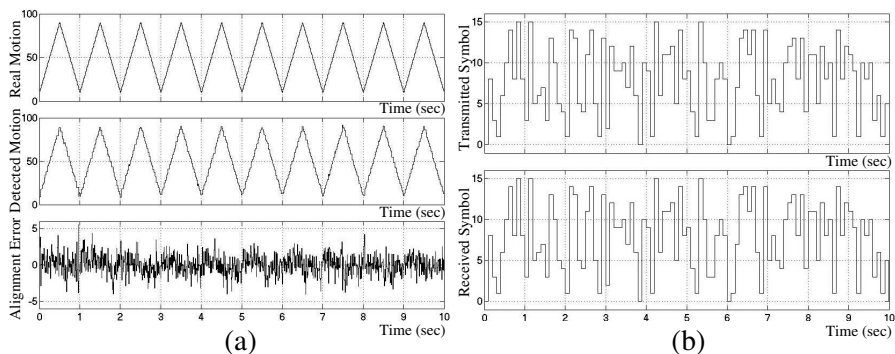


Figure 16. System performance at $SNR = 20$, $N = 30$, $SER = 0$. (a) Motion in degrees. (b) Symbols.

4.3. Saw Tooth Motion

This motion is a severe case that is used to simulate a very fast position changing objects. It is used to test the algorithm to track illusive objects. The results are shown in Figures 17 to 20.

In all types of motions and SNR values, the single decision ($N = 1$) has the worst tracking accuracy. The misalignment error bounds are $\pm 170^\circ$ while at ($N = 30$) the misalignment error bounds are at maximum of $\pm 5^\circ$ which confirms that N should be much higher than 1, as suggested in Section 2.5. Figure 12(a) shows an event of wrong decision. If this event occurs at the last iteration, then a wrong value for Φ will be assumed, leading to steer the beam in the wrong direction. But this wrong decision has happened during the estimation period, thus it did not affect the SER value because the next iteration has corrected this decision.

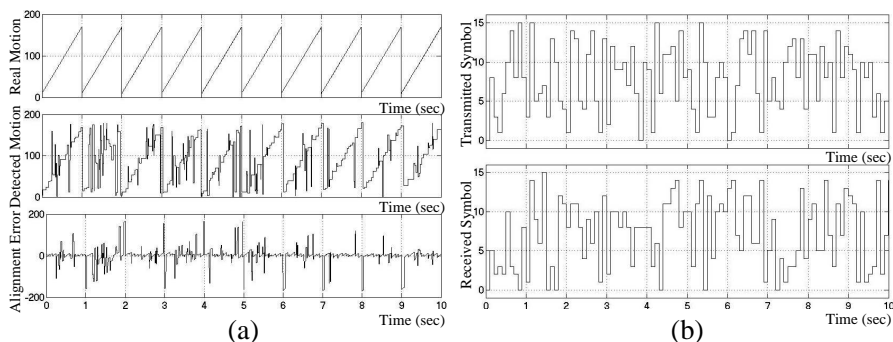


Figure 17. System performance at $SNR = 0$, $N = 1$, $SER = 80$. (a) Motion in degrees. (b) Symbols.

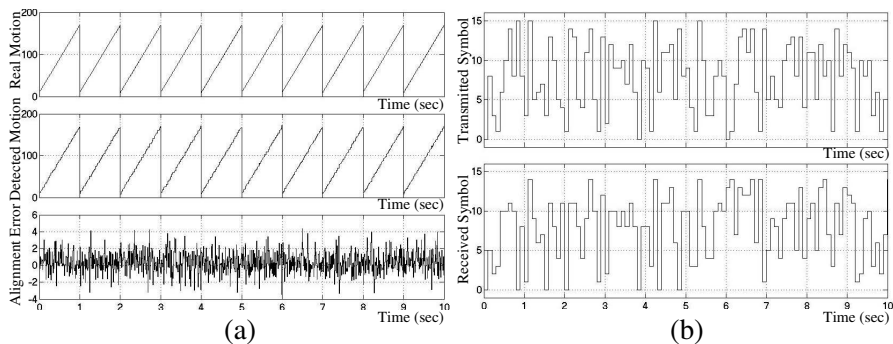


Figure 18. System performance at $SNR = 0$, $N = 30$, $SER = 74$. (a) Motion in degrees. (b) Symbols.

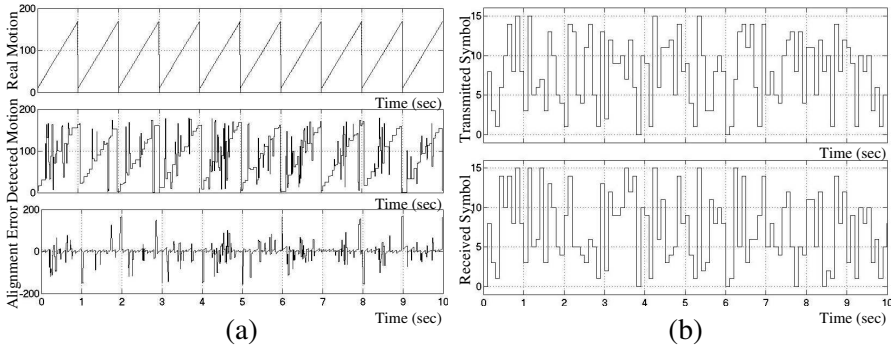


Figure 19. System performance at $SNR = 20$, $N = 1$, $SER = 24$. (a) Motion in degrees. (b) Symbols.

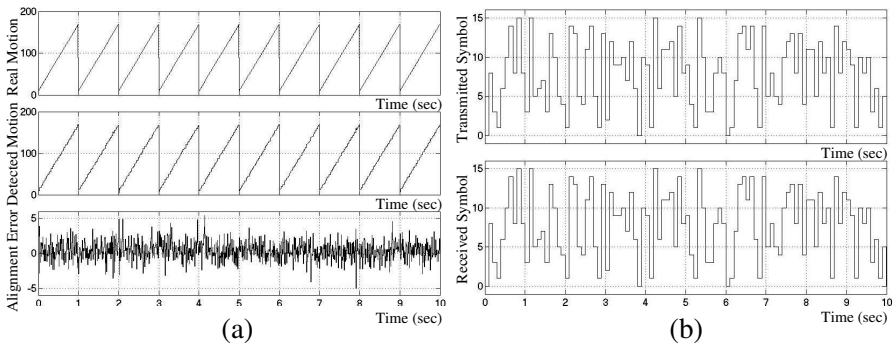


Figure 20. System performance at $SNR = 20$, $N = 30$, $SER = 0$. (a) Motion in degrees. (b) Symbols.

5. CONCLUSIONS

From the above, it can be concluded that the *LMS* is not necessary to steer the main beam blindly to track the transmitter movement. The cost is an alternative efficient method to be used for blind tracking. This algorithm does not suffer from the drawbacks of the *LMS* algorithm or its derivatives such as complexity, high computation load, and singularities that cripple the performance of the smart antenna because it does not store a symbol vector. This algorithm also decodes the incoming stream symbol by symbol unlike the *LMS* that needs to store enough symbols to perform maximization.

The increase in the number of array elements leads to a narrower beamwidth, thus the iterations should also be increased to ensure a precise alignment and minimum *SER*.

The algorithm is able to track objects that change their position abruptly (very high speed moving object) with high accuracy. For this type of motion, any symbol vector based algorithm might easily fail to track such objects because of the need for assembling the symbol vector.

The algorithm tracking ability is not affected by the high noise level. It can be seen that the algorithm can track the transmitter with high accuracy (with an average error 1°) whether *SNR* is high or low. This means that the algorithm can perform well in harsh environments without being affected by the high noise level. The tracking accuracy can be increased by increasing the number of iterations. The higher is the iterations the better is the tracking accuracy to a certain limit when higher iterations do not add significant improvement while increasing the computation load and decision delay.

The last decision in the iteration section has a major effect on the *SER* level. Thus setting the adequate iterations is vital to decreasing *SER*.

The type of motion of the transmitter has a minor effect on the tracking precision of the algorithm because the algorithm can make many decisions at each symbol and select the angle with least cost. Hence it can chase any sudden change in the object course.

REFERENCES

1. Sundaresan, K. and R. Sivakumar, "Cooperating with smartness: Using heterogeneous smart antennas in multi-hop wireless networks," *IEEE Transactions on Mobile Computing*, Vol. PP, No. 99, 2011.
2. Wang, M., et al., "Design and simulation of dipole and cable-FED network of TD-SCDMA smart antenna," *International Conference on Communications and Mobile Computing CMC'09 WRI*, Vol. 1, 65–69, 2009.
3. Mouhamadou, M., P. Vaudon, and M. Rammal, "Smart antenna array patterns synthesis: Null steering and multi-user beamforming by phase control," *Progress In Electromagnetics Research*, Vol. 60, 95–106, 2006.
4. Zhang, X., X. Gao, and Z. Wang, "Blind parallel multiuser detection for smart antenna CDMA system over multipath fading channel," *Progress In Electromagnetics Research*, Vol. 89, 23–38, 2009.
5. Gozash, F., G. R. Dadashzadeh, and S. Nikmehr, "A comprehensive performance study of circular and hexagonal array

- geometries in the LMS algorithm for smart antenna applications,” *Progress In Electromagnetics Research*, Vol. 68, 281–296, 2007.
6. LaMar, S., et al., “Beamforming solutions for interference reduction for high altitude airborne CDMA systems,” *IEEE Aerospace Conference*, 1–7, 2011.
 7. Gross, F. B., *Smart Antennas for Wireless Communications*, McGraw-Hill, New York, 2005.
 8. Varlamos, P. K. and C. N. Capsalis, “Electronic beam steering using switched parasitic smart antenna arrays,” *Progress In Electromagnetics Research*, Vol. 36, 101–119, 2002.
 9. Sun, C., J. Cheng, and T. Ohira, *Handbook on Advancements in Smart Antenna Technologies for Wireless Networks*, Information Science Reference, New York, 2009.
 10. Rivas, M., S. Xie, and D. Su, “A wideband beamformer with interference and noise suppression capabilities employing only spatial signal processing,” *International Workshop on Smart Antennas (WSA)*, 1–5, ITG, 2011.
 11. Kim, S.-T., J.-T. Kim, and Y.-S. Choi, “DOA estimation using time-frequency conversion pre-processing method,” *Progress In Electromagnetics Research C*, Vol. 24, 13–24, 2011.
 12. Yu, J., H. Li, and Z. He, “A novel method of DOA estimation based on subarray beamforming for uniform circular arrays,” *International Conference on Signal Acquisition and Processing ICSAP'10*, 80–84, 2010.
 13. Ozaki, Y., J. Ozawa, E. Taillefer, J. Cheng, and Y. Watanabe, “A simple DOA estimator using adjacent pattern power ratio with switched beam antenna,” *Progress In Electromagnetics Research C*, Vol. 22, 55–71, 2011.
 14. Zhang, X., X. Gao, and Z. Wang, “Blind paralind multiuser detection for smart antenna CDMA system over multipath fading channel,” *Progress In Electromagnetics Research*, Vol. 89, 23–38, 2009.
 15. Shetty, K. K., “A novel algorithm for uplink interference suppression using smart antennas in mobile communications,” M.Sc. Thesis, Department of Electrical and Computer Engineering, Florida State University, USA, 2004.
 16. Biedka, T. E., “Analysis and development of blind adaptive beamforming algorithms,” Doctoral Dissertation, Electrical Engineering, Virginia, 2001.
 17. Yan, H., “Cyclostationarity based DOA estimation and tracking,” Doctoral Dissertation, Electrical and Computer Engineering and

- Computer Science (ECECS), University of Cincinnati, 2005.
18. Abdullah, N. and Y. Kuwahara, "Steerable antenna using algorithm based on downhill simplex method," *Progress In Electromagnetics Research C*, Vol. 22, 23–34, 2011.
 19. Zhang, J., "Blind adaptive cyclic filtering and beamforming algorithms," Doctoral Dissertation, McMaster University, Canada, 2001.
 20. Bavand, M. and P. Azmi, "A novel interference cancellation aided minimum bit error rate beamforming," *International Symposium on Computer Networks and Distributed Systems (CNDS)*, 35–39, 2011.
 21. Huang, Z. and Y. Li, "Optimization and simulation of the fuze antenna beamforming algorithm based on LCMV criterion," *International Conference on Intelligent Control and Information Processing*, 144–147, Dalian, China, Aug. 2010.
 22. Inoue, Y. and K. Cho, "New smart antenna algorithm applied to autonomous area control for mobile radio network," *3rd European Conference on Antennas and Propagation EuCAP*, 2050–2054, 2009.
 23. Zarb-Adami, K., et al., "Beamforming techniques for large-N aperture arrays," *IEEE International Symposium on Phased Array Systems and Technology (ARRAY)*, 883–890, 2010.
 24. Rivas, M., S. Xie, and D. Su, "An adaptive wideband beamformer using Kalman filter with spatial signal processing characteristics," *13th International Conference on Advanced Communication Technology (ICACT)*, 321–325, 2011.
 25. Yu, Y., "Comparison of radiation patterns of smart antenna arrays," *China-Japan Joint Microwave Conference Proceedings (CJMW)*, 1–4, 2011.
 26. Fenn, A. J., *Adaptive Antennas and Phased Arrays for Radar and Communications*, Massachusetts Institute of Technology, Artech House, 2008.
 27. Lee, M.-S., "A low-complexity planar antenna array for wireless communication applications: Robust source localization in impulsive noise," *ETRI Journal*, Vol. 32, No. 6, Dec. 2010.
 28. Mahmoud, K. R., M. El-Adawy, S. M. M. Ibrahim, R. Bansal, and S. H. Zainud-Deen "A comparison between circular and hexagonal array geometries for smart antenna systems using particle swarm optimization algorithm," *Progress In Electromagnetics Research*, Vol. 72, 75–90, 2007.

29. Benedetti, M., G. Oliveri, P. Rocca, and A. Massa, "A fully-adaptive smart antenna prototype: Ideal model and experimental validation in complex interference scenarios," *Progress In Electromagnetics Research*, Vol. 96, 173–191, 2009.
30. Benedetti, M., et al., "PSO-based real-time control of planar uniform circular arrays," *IEEE Antennas and Wireless Propagation Letters*, Vol. 5, No. 1, 545–548, Dec. 2006.
31. Poli, L., et al., "Adaptive nulling in time-modulated linear arrays with minimum power losses," *IET Microwaves, Antennas and Propagation*, Vol. 5, No. 2, 157–166, Jan. 2011.
32. Benedetti, M., R. Azaro, and A. Massa, "Memory enhanced PSO-based optimization approach for smart antennas control in complex interference scenarios," *IEEE Transactions on Antennas and Propagation*, Vol. 56, No. 7, 1939–1947, Jul. 2008.
33. Manica, L., P. Rocca, and A. Massa, "On the synthesis of sub-arrayed planar array antennas for tracking radar applications," *IEEE Antennas and Wireless Propagation Letters*, Vol. 7, 599–602, 2008.