A new modified particle swarm optimization algorithm for adaptive equalization

Ali T. Al-Awami, Azzedine Zerguine, Lahouari Cheded, Abdelmalek Zidouri, Waleed Saif

Abstract

In this paper, we present a novel modification to the standard particle swarm optimization (PSO) technique and illustrate the superiority of the proposed modified technique over other PSO-based techniques, with an application to the important area of adaptive channel equalization. Different published versions of the original PSO algorithms are first reviewed and the new proposed technique discussed in the context of the design of adaptive channel equalizers. An exhaustive simulation-based sensitivity analysis of the proposed PSO algorithm, with respect to its underpinning parameters, is carried out here so as to select the “best” (or near optimal) values of these parameters. The performance of various PSO algorithms, including our proposed algorithm, is compared in the context of adaptive channel equalization to that of the LMS algorithm through extensive simulations. This detailed comparison revealed the superior performance of our proposed PSO-based adaptive channel equalizer over both its LMS-based counterpart and other adaptive equalizers based on the published PSO algorithms. This superior performance was exhibited on both linear and nonlinear channels.

Keywords:
Particle swarm optimization (PSO)  Adaptive channel equalization  LMS algorithm  Constriction factor

1. Introduction

Adaptive equalization (AE) [1] is important due to the need for mitigating the effect of intersymbol interference (ISI) in digital communication systems where an adaptive algorithm is used to adjust the equalizer's coefficients. Many efficient adaptive algorithms such as the least mean squares (LMS) algorithm [2] have been developed. However, the use of a linear adaptive algorithm was only occasionally successful because the assumption that the output is a linear function of the inputs is rarely true in practice. To this end, nonlinear adaptive equalization techniques were required and a number of such techniques have already been proposed in the literature [3]. Alternatively, heuristic techniques have also been employed for AE and in particular, the use of particle swarm optimization (PSO) in adaptive IIR phase equalization [4] and in a recent work on interference cancellation in CDMA systems [5]. Judging by its successful applications so far, PSO will undoubtedly continue to enjoy more successes in the general area of optimizing engineering systems.

Particle swarm optimization was first introduced by Kennedy and Eberhart [6]. This new approach features many advantages, the most important ones being that it is simple and fast, can be coded in few lines, and requires minimal storage.
Moreover, this approach is advantageous over evolutionary algorithms in more than one way. One key advantage is that PSO has memory, i.e., every particle remembers its best solution (local best) as well as the group’s best solution (global best). As such PSO is well suited to tackling dynamical problems, another advantage of PSO is that its initial population is maintained fixed throughout the execution of the algorithm, and so, there is no need for applying operators to the population, a process which is both time- and memory-storage-consuming. In addition, PSO is based on a “constructive cooperation” between particles, as opposed to the other artificial algorithms which are based on “the survival of the fittest” [6].

This work investigates the application and effectiveness of particle swarm optimization techniques in adaptive channel equalization. Different versions of the PSO algorithms are first discussed and a modified version proposed. Then to ensure that the proposed PSO algorithm performs at (or at least near) its best, an exhaustive simulation-based sensitivity analysis is conducted with respect to the algorithm’s key parameters. An extensive simulation work has also been carried out to test the proposed algorithm with both linear and nonlinear channels to show the superior performance of the various PSO algorithms used here over the conventional LMS algorithm in the context of adaptive channel equalization. More importantly, amongst all of these PSO algorithms, our proposed algorithm clearly show the best performance for both types of channels.

2. Particle swarm optimization algorithms

2.1. Common PSO algorithms

PSO starts with a population of random solutions (particles) in a D-dimension space. The ith particle is represented by \( \mathbf{X}_i = (x_{i1}, x_{i2}, \ldots, x_{iD}) \). Each particle keeps track of its coordinates in hyperspace which are associated with the fittest solution it has achieved so far. The value of the fitness for particle \( i \) (\( p_{\text{best}} \)) is also stored as \( \mathbf{P}_i = (p_{i1}, p_{i2}, \ldots, p_{iD}) \). The global version of the PSO keeps track of the overall best value (\( g_{\text{best}} \)), and its location, obtained thus far by any particle in the population.

At each step, PSO consists of changing the velocity of each particle toward its \( p_{\text{best}} \) and \( g_{\text{best}} \) according to [6,7]:

\[
v_{id} = w * v_{id} + c_1 * \text{rand}() * (p_{id} - x_{id}) + c_2 * \text{rand}() * (p_{gd} - x_{id}), \quad 1 \leq d \leq D, \tag{1}
\]

where \( w \) is the inertia weight, \( c_1 \) and \( c_2 \) are constants, \( \text{rand}() \) is a uniformly-distributed random number in the range [0,1], and \( p_{id} = p_{\text{best}} \) and \( p_{gd} = g_{\text{best}} \). It is instructive to briefly explain the role and influence of some of the PSO’s key parameters. This will provide some insight into the inner workings of the PSO algorithm. Here the inertia weight (\( w \)) in effect controls a “momentum” term \( w * v_{id} \) which in turn denotes the influence of the previous velocity of the ith particle upon its current velocity. In essence, this means that the bigger \( w \) is, the bigger the current velocity \( v_{id} \) and the search space for the particles become, thus helping in discovering new solution spaces, but at the cost of a slower convergence rate. However the smaller \( w \), the smaller \( v_{id} \) becomes, thus helping find a better solution in the current search space. The acceleration constants \( c_1 \) and \( c_2 \) control the rates at which a particle accelerates toward its individual local and global bests, respectively. If \( c_1 = 0 \), then each particle enjoys only a “global experience” whereby all particles in a swarm move freely and the probability of finding a global solution is rather small. However if \( c_2 = 0 \), each particle is then limited to its “self experience” or “self knowledge”. It may converge fast but is liable to easily getting trapped in a local optimum. In the case of \( c_1 = c_2 = 0 \), all particles are deprived of any experience, be it individual or social, with all particles in the swarm moving in a rather disorderly or chaotic manner. To prevent all particles from flying out of the solution space, their velocities are constrained within a chosen range \([-v_{\max}, v_{\max}]\). The velocity vector of the ith particle is represented as \( \mathbf{v}_i = (v_{i1}, v_{i2}, \ldots, v_{iD}) \). Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward \( p_{\text{best}} \) and \( g_{\text{best}} \). The position of the ith particle is then updated according to [6]:

\[
x_{id} = x_{id} + v_{id}, \quad 1 \leq d \leq D. \tag{2}
\]

The PSO algorithm is further improved through using a time-decreasing inertia weight, which leads to a reduction in the number of iterations [7,8]. The performance of this modified algorithm depends on the method of tuning the inertia weight. In this work, a linearly-decreasing time-dependent inertia weight proposed in [8] has been implemented, according to the following update law:

\[
w_t = (w_i - w_f) \frac{m - n}{m - 1} + w_f \tag{3}
\]

where \( w_i = \text{initial weight}, w_f = \text{final weight}, m = \text{maximum iteration value and } n = \text{variable iteration index} \).

Note here that the inertia weight \( w \) plays an important role in the convergence of the PSO algorithm to the global optimal solution and hence has an influence on the time taken for a simulation run. Recall here that the weight factor is used to control the influence of the previous history of the particle velocities on both the current velocity and the local and global exploration capabilities of the PSO algorithm. It thus follows that the reason for using a linearly-decreasing-in-time inertia weight parameter \( w \) is that larger values of \( w \) tend to be used at the start of the search to enable the PSO algorithm to explore globally the solution space, whereas smaller values of \( w \) are used toward the end of the search to enable the PSO algorithm to explore locally around the global optimum before finally homing in onto it.
Table 1
Steps of the PSO algorithm.

<table>
<thead>
<tr>
<th>Step #</th>
<th>Action taken</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Define the problem space and set the boundaries, i.e., inequality constraints of the tap weights defined by their maximum and minimum limits.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Initialize an array of particles with random positions (tap weights) and their associated velocities inside the problem space.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Check if the current position is inside the problem space or not. If not, adjust the positions so as to be inside the problem space.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Evaluate the fitness value (MSE) of each particle.</td>
</tr>
<tr>
<td>Step 5</td>
<td>Compare the current fitness value with the particles’ previous best value (pbest). If the current fitness value is better, then assign the current fitness value to pbest and assign the current coordinates to pbest coordinates.</td>
</tr>
<tr>
<td>Step 6</td>
<td>Determine the current global minimum among particle’s best position.</td>
</tr>
<tr>
<td>Step 7</td>
<td>If the current global minimum is better than gbest, then assign the current global minimum to gbest and assign the current coordinates to gbest coordinates.</td>
</tr>
<tr>
<td>Step 8</td>
<td>Change the velocities according to (1) or (4).</td>
</tr>
<tr>
<td>Step 9</td>
<td>Move each particle to the new position according to (2) and return to Step 3.</td>
</tr>
<tr>
<td>Step 10</td>
<td>Repeat Step 3–Step 9 until a stopping criterion is satisfied.</td>
</tr>
</tbody>
</table>

Recent works in [9] and [10] indicate that the use of a “constriction factor” may be necessary to insure convergence of the PSO. A simplified method of incorporating a constriction factor is represented in [9]:

\[ v_{id} = K \left[ v_{id} + c_1 \times \text{rand()} \times (p_{id} - x_{id}) + c_2 \times \text{rand()} \times (p_{gd} - x_{id}) \right], \ 1 \leq d \leq D, \tag{4} \]

where \( K \) is a function of \( c_1 \) and \( c_2 \) as illustrated by the following equation [9]:

\[ K = \frac{k}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|}, \tag{5} \]

where \( k = 2, \varphi = c_1 + c_2, \) and \( \varphi > 4. \) In [9], the performance of PSO using an inertia weight was compared with the PSO performance using a constriction factor. It was concluded that the best approach is to use a constriction factor while limiting the maximum velocity \( v_{\text{max}} \) to the dynamic range of the variable \( x_{\text{max}} \) in each dimension. It was also shown in [9] that this approach provides a performance superior to any similar technique reported in the literature.

2.2. Proposed PSO algorithm

Building on the successful use of the constriction factor-based technique [9], we propose in this work, a new modification of the constriction factor-based technique by introducing a time-dependent linearly-decreasing \( K \), instead of a fixed one. This is done by adjusting \( K \) at every iteration according to the following recursion:

\[ k_n = k_{\text{min}} + (k_{\text{max}} - k_{\text{min}}) \frac{m - n}{m - 1}, \tag{6} \]

where \( m \) is the maximum number of iterations and \( n \) is the current iteration. Note here that the use of a constriction factor ensures computational stability of the PSO algorithm. The proposed modification of \( K \) can be intuitively explained by noting that as the particle is getting closer to its optimal position, it undergoes a process similar to a “cooling” one which results in a stabilizing effect on the swarm and which therefore calls for the use of a lower value of the constriction factor.

In this work, PSO is employed to search for the optimum tap weights so as to minimize the mean square error (MSE). PSO is most efficient for batch-type optimization. However, due to practical constraints, the entire input data is not available to the equalizer. Therefore, only a block, or a window, of the input data is proposed at every iteration. Consequently, the objective function considered in every iteration represents an estimate of the MSE over the input window used in that iteration. At the \( n \)th iteration, this estimate of MSE for the \( i \)th particle is given by:

\[ J_i(n) = \frac{1}{N} \sum_{j=1}^{N} [e_{ji}(n)]^2, \tag{7} \]

where \( N \) is the length of the window of the input data and \( e_{ji}(n) \) is the \( j \)th error for the \( i \)th particle.

Finally, Table 1 details the steps of the proposed PSO algorithm.
3. Application of PSO algorithms to adaptive equalization

The performance of three different PSO algorithms, including our proposed PSO with a variable constriction factor (PSO-VCF), is compared to that of the LMS algorithm in the context of channel equalization. The digital message applied to the channel is a binary sequence of uniformly-distributed random numbers taking on the values \((-1, 1)\). The channel noise is taken to be additive white Gaussian noise with a signal-to-noise ratio of 20 dB. Two linear time-invariant channel models are used in the simulation and are described by their following transfer functions

\[ H_1(z) = 0.2602 + 0.9298z^{-1} + 0.2602z^{-2}, \]

and

\[ H_2(z) = 0.408 + 0.816z^{-1} + 0.408z^{-2}. \]

The eigenvalue spread for the first and second channel are 11 and 81, respectively.

3.1. Sensitivity analysis of the proposed PSO-VCF algorithm

For the proposed PSO algorithm to have a near-optimal performance, a careful selection of the values of its key parameters is in order. A judicious selection of the values of the algorithm’s key parameters was achieved through a thorough experimental (i.e. simulation-based) sensitivity analysis of the proposed PSO algorithm with respect to six (6) key parameters, namely the population size \(n\), the data window size \(N\), the acceleration parameters \(c_1\) and \(c_2\), the range of the maximum velocity \(v_{\text{max}}\), the limits \(k_{\text{min}}\) and \(k_{\text{max}}\) of the time-dependent factor \(k_n\) which directly controls the constriction factor \(K\), and finally the number of taps used in the adaptive channel equalizer.

The results of this experimental sensitivity analysis are all shown in Figs. 1–6 for channel \(H_1(z)\). With respect to the population size \(n\) and regardless of the complexity of the problem at hand, it is expected that a large population size will provide a better search and a faster convergence rate on average because of the large number of estimates used at each epoch. Fig. 1 shows that a population size of \(n = 40\) is sufficient to avoid local minima in this particular problem and that any larger population will hardly improve the convergence rate.

Fig. 2 shows that there is no additional benefit gained by choosing a window size beyond \(N = 200\) for the simulated linear channel. This choice is clearly problem-dependent.

As mentioned earlier, the acceleration constants \(c_1\) and \(c_2\) control the rate at which the local and global optima are reached, respectively. In general, setting these acceleration rates at a small value will enable a thorough search of a complex error surface, but at the cost of a slower convergence. In this case, the swarm may not converge quickly enough around the global optimum to perform a fine-tuned local search, hence possibly leading to low-accuracy results. It may also occur that particles may remain confined in a particular region with no real improvement in their position, thus increasing the likelihood of getting trapped in a local minimum. In our experiment, we used equal acceleration constants whose values range from 3 to 6 so as to have a balance between a local and a global search. As Fig. 3 shows, the best value for both acceleration constants is 4 as, at this value, the PSO algorithm converges fastest to its lowest MSE value of \(-30\) dB. The smaller values of 3 and 3.5 either show a slower convergence or a lower accuracy, respectively. However values larger than 4 (e.g. 5 and 6) result in a lower MSE and hence in a lower-accuracy result.
Fig. 2. Effect of the window size $N$ on the convergence rate.

Fig. 3. Effect of the acceleration constants $c_1$ and $c_2$ on the convergence rate.

Fig. 4 clearly shows that, within the displayed experimental range of the maximum velocity, i.e. $v_{\text{max}} \in [0.05x_{\text{max}}, 0.75x_{\text{max}}]$ where $x_{\text{max}}$ is the particle’s maximum position, the largest maximum velocity value of $v_{\text{max}} = 0.2x_{\text{max}}$ is the one that leads to the lowest MSE value with the fastest convergence. Although not shown in Fig. 4, velocity values higher than $v_{\text{max}} = 0.75x_{\text{max}}$ yielded no real improvement. This is somewhat expected as larger $v_{\text{max}}$ values would allow the swarm to migrate faster toward the global optimum, thus enabling the local search to utilize more information provided by the larger number of particles positioned in the vicinity of the global minimum, and in the process provide a more accurate estimate of the global solution.
With respect to the sensitivity analysis regarding the limits $k_{\text{min}}$ and $k_{\text{max}}$ of the parameter $k_n$ which controls the constriction factor, these are all shown in Fig. 5 where it is clear that the best values amongst the simulated values are $k_{\text{min}} = 4$ and $k_{\text{max}} = 6$. No real performance improvement was obtained beyond these two values. Note here that the constriction factor has a similar “velocity-clamping” effect to that of the maximum velocity $v_{\text{max}}$. Higher constriction factor values favor a global search and help to avoid a “swarm explosion” which may lead to algorithmic instability.

The final sensitivity analysis results are shown in Fig. 6 where, out of the four numbers of taps tested, it emerged that nine taps are best suited for this problem. The number of taps is clearly a parameter that is problem-dependent and may vary from channel to channel. Since the adaptive equalizer here attempts to capture (or model) as much of the swarm
dynamics as possible, it is then logical to expect that the higher the number of taps, the better the modeling performance. As seen from Fig. 6 and for this particular problem, the choice of seven taps seems to indicate a case of channel under-modeling, whereas especially the choice of thirteen taps refers to the case of channel over-modeling.

3.2. Application of the proposed PSO algorithm to linear channels

The three different PSO algorithms investigated in this study are: the PSO with a time-dependent linearly-decreasing inertia weight (PSO-W) [11–15], the PSO with a constant constriction factor (PSO-CCF) [15,16], and our proposed PSO with a variable constriction factor (PSO-VCF). All of these PSO algorithms share the following parameters: \( x_{\text{min}} = -2, x_{\text{max}} = 2 \), the number of particles is 40, the number of iterations equals 500 and the input window length, \( N \), is chosen to be 200. In each simulation, the estimated MSE, defined in (6), is averaged over 10 runs. The remaining specifications of each algorithm are \( c_1 = c_2 = 1.5, w_{\text{max}} = 1, w_{\text{min}} = 0.6, \) and \( v_{\text{max}} = 0.07x_{\text{max}} \) for PSO-W; while for PSO-CCF, they are \( c_1 = c_2 = 4, k = 5, \) and \( v_{\text{max}} = 0.20x_{\text{max}} \); and finally \( c_1 = c_2 = 4, k_{\text{min}} = 4, k_{\text{max}} = 6, \) and \( v_{\text{max}} = 0.20x_{\text{max}} \) for PSO-VCF. It is worth pointing out here that in order to carry out a fair comparison between our proposed algorithm (PSO-VCF) and the other two algorithms (PSO-W and PSO-CCF), extensive simulations were carried out to carefully select the parameters for both of these algorithms, i.e., PSO-W and PSO-CCF. The selected parameters where those corresponding to the best performance exhibited by these two algorithms (PSO-W and PSO-CCF) in all of our simulations. The step size for the LMS algorithm used is 0.025 and the learning curves obtained are averaged over 500 independent runs. All of these parameters are chosen such that all the algorithms converge to the same steady state. Finally, the equalizer considered here has nine taps and a delay of \( D = 11 \).

Fig. 7 depicts the convergence behavior of the four algorithms for the first channel. As shown in Fig. 7, an improvement in both the convergence time and the steady-state MSE has been achieved by all three PSO algorithms, over the LMS one, in adaptively equalizing the channel.

A similar improvement was also achieved for the second channel, which has a larger eigenvalue spread of 81, as shown in Fig. 8. In this figure, the difference in convergence time between the PSO-VCF and the LMS is now more pronounced than in the case of channel \( H_1(z) \). Moreover, the insensitivity of the PSO algorithms to the eigenvalue spread, is also very clear for both channels as depicted in Figs. 7 and 8.

Figs. 9 and 10 depict the bit-error rate (BER) performance of the PSO-VCF- and LMS-based equalizers for the first and second channel, respectively. The consistency in performance of the proposed algorithm in both channels is very distinctive. About a 2 dB-improvement in SNR was achieved by the new proposed algorithm over the LMS, at a BER of \( 10^{-3} \) for the first channel. A similar improvement in SNR was also achieved by the proposed PSO-VCF algorithm over the LMS, at the same BER level and for the second channel.
3.3. Application of the proposed PSO algorithm to nonlinear channels

In satellite communications, the traveling wave tube amplifier in satellite usually operates in the saturated region because of several signal attenuations involved and the limited power sources utilized by the satellite. Such operation results in severely nonlinear channels. The nonlinear effect of the channel not only spreads the signal spectrum, but also introduces nonlinear amplitude and phase distortions to the channel's in-band signal. Besides satellite links, this model has also been applied to voice-band telephone channels. The first general but simple third-order Volterra model for the voice-band channel
was proposed by Falconer [17]. A more simplified version of Falconer’s model was used by Biglieri et al. [18] as shown in Fig. 11, where the output $y(n)$ is related to the input $x(n)$ through the following relation:

$$y(n) = a_1 x(n) + a_2 x^2(n) + a_3 x^3(n).$$

(8)

The coefficients $a_1$, $a_2$, and $a_3$ in (8) are scalars, which control the degree of nonlinearity.

Two nonlinear channels with different degrees of nonlinearity are used. The first channel, denoted as nonlinear channel 1, uses the linear channel $H_1(z)$ as the FIR filter in the nonlinear channel. Whereas, for the second nonlinear channel, denoted as nonlinear channel 2, the linear channel $H_2(z)$ is used as the FIR filter. The system is tested by setting the first-, second- and third-order coefficients, $a_1$, $a_2$, $a_3$ to 1, 0.1 and 0.05 as given in [19], respectively.

Fig. 12 depicts the convergence behavior of the four algorithms for channel 1. As shown in Fig. 12, an improvement in both the convergence time and the steady-state MSE has been achieved by all three PSO algorithms, over the LMS one, in adaptively equalizing the channel. Again, similar behavior is noticed, here, as in the case of the linear channel where the
PSO-VCF algorithm exhibits consistent performance even in nonlinear systems as it is depicted in the comparison of Fig. 7 against Fig. 12.

To further investigate the consistency in performance of the PSO-VCF algorithm, BER performances are studied for the two equalizer configurations and are shown in Figs. 13–14. As expected, the LMS-based equalizer shows a very poor performance, whereas the PSO-VCF equalizer performs better than the LMS-based equalizer.

Fig. 13, which corresponds to channel 1 shows that a gain of about 2 dB is made by the PSO-VCF at a BER of $10^{-3}$ when compared to the LMS algorithm. This shows the impact of particle swarm strength over the traditional LMS algorithm. Almost a similar performance is achieved by the PSO-VCF algorithm in both linear and nonlinear channels.

The situation deteriorates even further for the LMS algorithm in the case of the second nonlinear channel, channel 2, as shown in Fig. 14. On the other hand, the PSO-VCF algorithm is still consistently behaving even in this severe scenario.

In conclusion, our proposed PSO-VCF algorithm outperformed both the LMS algorithm and the other PSO-based algorithms. More importantly, the PSO-based algorithms are in general insensitive to the eigenvalue spread of the channel autocorrelation matrix of the input signal and behave better than the LMS algorithm in nonlinear environments.

4. Conclusion

This paper has presented the results of the first application of particle swarm optimization techniques to channel adaptive equalization. The extensive simulation work, carried out here, has clearly shown that PSO not only improves the
This document discusses the BER performance of PSO-VCF and LMS algorithms in different channels. Fig. 13 shows the BER performance in channel 1 with $a_1 = 1$, $a_2 = 0.1$, and $a_3 = 0.05$. Fig. 14 presents the performance in channel 2 with the same parameters. The results indicate that the PSO-VCF algorithm converges faster and improves the BER performance more efficiently in heavily-distorting channels with a large eigenvalue spread of the channel autocorrelation matrix of the input signal. An extensive simulation-based sensitivity analysis was conducted to select the best key parameters for these algorithms. This approach not only reduces the convergence time of the equalizer but also enhances the BER performance significantly.
posed PSO-VCF algorithm. Based on the best values for these key parameters, an extensive simulation work was conducted and showed that, overall, our proposed new PSO-VCF has a much better performance than both the LMS and the other two PSO algorithms (PSO-W and PSO-CCF). It is also to be noted that the improved performance of the proposed structure is obtained without any additional increase in complexity. Finally, the results achieved here provide us with ample encouragement to further explore the use of PSO in other applications in adaptive filtering such as echo cancellation in telephony and adaptive noise cancellation in industrial settings.

Acknowledgment

The authors acknowledge the support of King Fahd University of Petroleum & Minerals.

References


Ali Al-Awami earned his B.Sc. and M.Sc. degrees in Electrical Engineering from King Fahd University of Petroleum & Minerals, Saudi Arabia, in 2000 and 2005, respectively. In 2001 he joined the Saudi Electricity Company as a control engineer in the System Operation Dept. In 2002, he joined the EE Department at KFUPM where he is currently a Lecturer. He is now working towards his Ph.D. at the University of Washington, Seattle, USA. His research interests include wind integration into the smart grid, power system operation and optimization, and artificial intelligence applications in power systems and signal processing.

Azzedine Zerguine received the B.Sc. degree from Case Western Reserve University, Cleveland, OH, in 1981, the M.Sc. degree from King Fahd University of Petroleum & Minerals (KFUPM), Dhahran, Saudi Arabia, in 1990, and the Ph.D. degree from Loughborough University, Loughborough, UK, in 1996, all in electrical engineering. Dr. Zerguine is presently an Associate Professor, Electrical Engineering Department, KFUPM, working in the areas of signal processing and communications. His research interests include adaptive filtering, signal processing for communications, neural networks, multiuser detection, and interference cancellation and he is currently serving as an associate editor with the EURASIP Journal on Advances in Signal Processing. Finally, Dr. Zerguine is the recipient of two KFUPM best teaching awards in 2000 and 2005.

Laouhouri Cheded gained his BS (Honors) in Physics from Oran University (Algeria), his MS in Electronic Control Engineering from Salford University (UK) and his PhD in Signal Processing from the University of Manchester (ex-UMIST), UK, in 1975, 1979 and 1988, respectively. While at UMIST, he taught physics to first-year students at the Textile Technology Department (1980–1981) and was a teaching and research assistant at the Department of Instrumentation and Analytical Science (DIAS) during the period 1981–1984. He has been with the Systems Engineering Department of the King Fahd University of Petroleum and Minerals in the Kingdom of Saudi Arabia (KSA) since September 1984, where he is currently an Associate Professor. He was a visiting researcher in DIAS during summer 1996. His research interests include digital signal processing (theory and applications) and its many interactions with control, detection and estimation, communication and soft computing. Besides teaching and research, he is also involved in undergraduate and graduate students supervision, paper reviewing for both international conferences and journals (including IEEE Trans. Signal Process.), review of
grant proposals for grant-awarding agencies and administrative work at the departmental, college and university levels. He has also organized and taught short courses in Control and Signal Processing to local and national companies in KSA and has been involved in the organization of the international conference ISSPA in 2001 and other national workshops.

He is currently a Senior Member of the IEEE, a member of the IET and a past member of EURASIP.

He served as a member of the editorial board of the International Journal of Electrical Engineering Education during the period 1994–2005.

Abdelmalek Zidouri is a Senior IEEE member and an Assistant Professor in the Department of Electrical Engineering at King Fahd University of Petroleum and Minerals, Dhahran Saudi Arabia. He holds a Master of Science in Control Engineering from Bradford University, UK (1984) and a Doctor of Engineering in Applied Electronics from Tokyo Institute of Technology Japan (1995). His research interests are in the field of signal processing and pattern recognition, in particular character recognition and document image analysis. Dr. Zidouri has published many refereed journal and conference papers. He has supervised many projects and theses.

He was the head of the Digital Signal Processing Group for 3 years and the research coordinator of the EE Department for 2 years. He is a member of the Engineering Education Society, Signal Processing Society and Communications Society.

Waleed Saif received the B.Sc. and M.Sc. degrees from King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia, in 2000 and 2003, respectively. Since 2007 he is pursuing his Ph.D. degree at the University of Leeds, Leeds, UK. Mr. Saif research interest includes wireless communications, multiuser detection, and interference cancellation.