Adaptive equalisation using particle swarm optimisation for uplink SC-FDMA

N. Iqbal, A. Zerguine and N. Al-Dhahir

An adaptive frequency-domain equaliser for the single carrier frequency division multiple access (SC-FDMA) system using the particle swarm optimisation (PSO) technique is proposed. Unlike the stochastic gradient and recursive least squares algorithms, the PSO is known to have fast convergence which does not depend on the underlying structure. The cost function used in a PSO is formulated based on the respective structure of the equaliser, whether it is a linear equaliser or a decision feedback equaliser. The robustness of the proposed PSO algorithm is demonstrated on a high Doppler scenario. Furthermore, it is shown that the performance improves more when using re-randomisation. Finally, it is shown that the PSO-based frequency-domain equaliser is more computationally efficient than its time-domain counterpart.

Introduction: The most used algorithms for adaptive equalisation to deal with time-varying channels are the least mean squares (LMS) and recursive least squares (RLS) [1]. However, their performance degrades in channels having a large eigenvalue spread. Recently, particle swam optimisation (PSO) has been used for adaptive estimation/equalisation problems and showed its improved performance when compared with other conventional algorithms [2–4]. The PSO algorithm does not assume any underlying model; therefore, its performance is independent of the characteristics of the system used. For this reason, PSO is expected to perform well in a channel with a large eigenvalue spread.

The PSO is a robust algorithm with fast convergence [5]. It is used to minimise continuous and real-valued function in an l-dimensional space. The swarm, also known as population, is comprised of particles that move in a predefined search space. The position of each particle within the search space represents the potential solution to the problem.

In this Letter, PSO is used in an adaptive frequency-domain (FD) equaliser for the uplink single carrier frequency division multiple access (SC-FDMA) system and it is shown to have less computational complexity as compared with PSO applied to adaptive time-domain (TD) equalisation. More importantly, in the case of a decision feedback equaliser (DFE), separate fitness functions, for the feedforward and the feedback filters, are developed entirely in the FD.

System description: We consider an uplink SC-FDMA system having K users and a total of N subcarriers, with each user assigned M subcarriers, i.e. $N = KM$. The transmitter of an SC-FDMA system consists of an M-point DFT, a subcarrier mapping, an N-point IDFT and a cyclic prefix insertion operation [6]. At the receiver, the reverse operations are performed. Let M be the modulated data symbols of the mth user which are grouped to form a block $\boldsymbol{x}$. Now, the mth user's received signal after demapping is given as

$$\mathcal{Y} = \Lambda\mathcal{X} + \mathcal{N} \tag{1}$$

Λ is the $M \times M$ channel matrix having a diagonal structure, $\mathcal{N}$ is the noise vector having variance $\sigma_{\mathcal{N}}\boldsymbol{I}_N$, and $\mathcal{X} = F_M(\boldsymbol{x})$ ($\boldsymbol{F}_M$ is the $M \times M$ DFT matrix). Let $\mathcal{Z} = \mathrm{diag}(\mathcal{Y})$ and the weight vector of the linear equaliser (LE) be denoted by $\mathcal{W}$, then the output of the equaliser, $\check{\mathcal{X}}_k$, in the FD at instant k is given by

$$\check{\mathcal{X}}_k = \mathcal{Z}_k\mathcal{W}_{k-1} \tag{2}$$

In the case of the DFE, the output of the equaliser is

$$\check{\mathcal{X}}_k = \mathcal{Z}_k\mathcal{F}_{k-1} + \mathcal{D}_k\mathcal{B}_{k-1} \tag{3}$$

where $\mathcal{F}_{k-1} = \left[\mathcal{F}(0)_{k-1},\ \mathcal{F}(1)_{k-1},\ \ldots,\ \mathcal{F}(M-1)_{k-1}\right]^{\mathrm{T}}$ and $\mathcal{B}_{k-1} = \left[\mathcal{B}(0)_{k-1},\ \mathcal{B}(1)_{k-1},\ \ldots,\ \mathcal{B}(M-1)_{k-1}\right]^{\mathrm{T}}$ are the feedforward and the feedback filters of the DFE, respectively. Note that the exact solution of these filter coefficients is not needed in the case when using an adaptive algorithm. The decision matrix $\mathcal{D}_k$ is defined as

$$\mathcal{D}_k = \begin{cases} \mathrm{diag}(F_M(\boldsymbol{x}_k)) & \text{for training} \\ \mathrm{diag}(F_M(\hat{\boldsymbol{x}}_k)) & \text{for decision-directed} \end{cases}$$

where $\hat{\boldsymbol{x}}_k$ is the decision on $\check{\boldsymbol{x}}_k$ which is given as

$$\check{\boldsymbol{x}}_k = F_M^{\mathrm{H}}\check{\mathcal{X}}_k \tag{4}$$

Finally, the error signal, $\boldsymbol{e}_k = [e_k(0),\ \ldots,\ e_k(M-1)]^{\mathrm{T}}$, is given as

$$\boldsymbol{e}_k = \begin{cases} \hat{\boldsymbol{x}}_k - \boldsymbol{x}_k & \text{for training} \\ \hat{\boldsymbol{x}}_k - \check{\boldsymbol{x}}_k & \text{for decision-directed} \end{cases} \tag{5}$$

PSO-based adaptive equalisation: In this Section, a PSO-based adaptive equalisation algorithm is devised for the SC-FDMA system. For this, the filter coefficients in (2) and (3) are calculated adaptively using PSO [4]. We denote the M-dimensional position and velocity vectors of the ith particle at the instant k as $\boldsymbol{p}_{i,k} = [p_{i,k}(0),\ p_{i,k}(1),\ \ldots,\ p_{i,k}(M-1)]$ and $\boldsymbol{v}_{i,k} = [v_{i,k}(0),\ v_i(1),\ \ldots,\ v_{i,k}(M-1)]$, respectively, where $p_{i,k}(l)$ represents the ith particle position having velocity $v_{i,k}(l)$ in the lth-dimension. Each $p_{i,k}(l)$ and $v_{i,k}(l)$ are clamped in the range $[-p_{\max},\ +p_{\max}]$ and $[-v_{\max},\ +v_{\max}]$, respectively, where $v_{\max} = v_c p_{\max}$ and v_c is the velocity constraint factor. A fitness function (cost function), developed later, is minimised to reach the global minimum. The local and global bests in a conventional PSO are found as follows. For the ith particle, among all the particle's visited positions up to the instant k, the one that gives the lowest value of the cost function is the local best of the ith particle denoted by $\boldsymbol{pbest}_{i,k}$. Similarly, for the whole swarm and among all the swarm's visited positions up to the instant k, the one that gives the lowest value of the cost function is the global best of the swarm abbreviated as $\boldsymbol{gbest}_k$. The velocity update equation is given as

$$\begin{aligned} \boldsymbol{v}_{i,k+1} = \gamma(k)\big[&\boldsymbol{v}_{i,k}c_1 * \mathbf{rand}_1 * \left(\boldsymbol{pbest}_{i,k} - \boldsymbol{p}_{i,k}\right) \\ &+ c_2 * \mathbf{rand}_2 * \left(\boldsymbol{gbest}_k - \boldsymbol{p}_{i,k}\right)\big] \end{aligned} \tag{6}$$

where $\mathbf{rand}_j = \left[\mathrm{rand}_{0,j},\ \mathrm{rand}_{1,j},\ \ldots,\ \mathrm{rand}_{M-1,j}\right]^{\mathrm{T}}$, $j = 1, 2$ and the lth element $\mathrm{rand}_{l,j}$ is a uniformly distributed number in the range [0, 1], $\gamma(k)$ is the time-varying constriction factor defined in [4] and c_1 and c_2 are the positive acceleration constants satisfying $c_1 + c_2 > 4$. After updating (6), the ith particle's position is changed according to $\boldsymbol{p}_{i,k+1} = \boldsymbol{p}_{i,k} + \boldsymbol{v}_{i,k}$.

Fitness function: A PSO is more effective in offline applications where the whole data are available; however, our case is an online one; therefore, instead of using the whole data, a block of data, i.e. one SC-FDMA symbol is used. The fitness function (cost function) used in the minimisation procedure at the kth iteration is given as

$$J(k) = \sum_{j=0}^{M-1} |e_k(j)| \tag{7}$$

where $e_k(j)$ is the jth error at the kth instant and it is obtained from (5). By taking the DFT, the FD version of the error is given as

$$\mathrm{E}_k(l) = \sum_{j=0}^{M-1} e_k(j)\exp\left(-\sqrt{-1}2\pi lj/M\right), \quad l = 0,\ 1,\ \ldots,\ M-1 \tag{8}$$

and also

$$\mathcal{E}_k = \boldsymbol{D}_k - \check{\mathcal{X}}_k \tag{9}$$

Minimising (7) in the TD is equivalent to minimising (9) in the FD as (7) depends on $\sum_{j=0}^{M-1}|e_k(j)|$ and so does $\mathcal{E}_k$. Therefore, the absolute value of $\mathcal{E}_k$ will be our fitness function. As $\mathcal{E}_k = [\mathcal{E}_k(0),\ \ldots,\ \mathcal{E}_k(M-1)]$ hence unlike the conventional PSO, we define a vector of fitness functions of length M. In other words, the value of the fitness function is different for each dimension. In this way, instead of comparing particle positions to constitute $\boldsymbol{pbest}_{i,k}$ for the ith particle, we find the best value of each dimension and $\boldsymbol{pbest}_{i,k}$ will be the amalgamation of each best dimension up to the instant k. Similarly, $\boldsymbol{gbest}_k$ is the combinations of each best dimension among all $\boldsymbol{pbest}_{i,k}$, $i = 1, 2, \ldots n$, where n is the swarm size.

The above fitness function is valid for LE, but in the case of DFE we have to find the coefficients of both the feedforward and the feedback filters which is not possible by using the same fitness function. Therefore, we put a constraint on the feedback filter which cancels out the pre- and post-cursors, but not the desired component as in [7]. We use this constraint to formulate our fitness function for DFE. The constraint-based problem for each frequency bin is given as follows:

$$\min_{\mathcal{F}(l),\mathcal{B}(l)} |\mathcal{E}_k(l)|^2 \text{ subject to } \sum_{j=0}^{M-1} \mathcal{B}_k(j) = 0, \quad l = 0,\ 1,\ \ldots,\ M-1 \tag{10}$$

By using a Lagrange multiplier, we obtain

$$f(k) = |\mathcal{E}_k(l)|^2 + \alpha_k \sum_{j=0}^{M-1} \mathcal{B}_k(j) \tag{11}$$

Now, the gradient of (11) with respect to $\mathcal{F}(l)$ and $\mathcal{B}(l)$ is equal to

$$f(k)_{\mathcal{F}(l)} = \mathcal{Y}_k(l)|\mathcal{E}_k(l)| \tag{12}$$

and

$$f(k)_{\mathcal{B}(l)} = \mathcal{Y}_k(l)|\mathcal{E}_k(l)| + \alpha_k \tag{13}$$

PSO compares the fitness value of all the particles and picks the one that gives the lowest value. As the term $\mathcal{Y}_k(l)$ in (12) is common for all the particles, hence we ignore this term and the fitness function for the feed-forward filter is $f(k)_{\mathcal{F}(l)} = |\mathcal{E}_k(l)|$, which is the same as for the LE. In (13), α_k is updated according to the stochastic gradient method

$$\alpha_{k+1} = \alpha_k + \mu \sum_{j=0}^{M-1} \mathcal{B}_k(j) \tag{14}$$

where μ is the step size.

Re-randomisation: One of the problems of the PSO is that once a ***g*best** is found, then all particles start to move towards it and hence become stagnant around the global minimum leaving empty spaces in the search space. In our scenario, we are using PSO not only to avoid providing channel information needed at the SC-FDMA receiver but also to track the variations in the channel. Therefore, due to the time-varying nature of the problem, the values of the equaliser taps are not fixed and if the particles become stagnant in one place, then the PSO will not be able to find a plausible solution. To tackle this issue, a re-randomisation is proposed. In this method, the particles are re-randomised around $\mathbf{gbest}_k$ after certain time instants, except during the training phase. In the training phase, the main objective is to enable the particles to search for the global minimum quickly. However re-randomisation during this phase will slow down the speed of convergence of the global minimum search process. The benefit of re-randomisation is its capability to allow the particles a higher probability of finding the best solution in time-varying environments. As such it also yields a better bit error rate (BER). Re-randomisation here can be thought of retraining of the RLS/LMS schemes which is used to avoid divergence in these algorithms. Therefore, re-randomisation will not only improve the performance but also reduce the overhead that would otherwise be required in retraining blocks needed in RLS/LMS.

Complexity analysis: In this Section, the computational complexity of the PSO algorithm operating in the FD (FD-PSO) is compared to that of the PSO operating in the TD (TD-PSO) [4] for a DFE. The comparison is based on the total number of complex multiplications and additions. Table 1 summarises the computational complexity of both algorithms. To illustrate the computational complexity, for $M = 512$, FD-PSO is 128 times faster than TD-PSO in terms of multiplication and 103 times faster in terms of addition. Although the FD-PSO is computationally heavier than the RLS, it is nevertheless still preferred to use the PSO because of its superior performance over that of the RLS. Moreover, as processing is carried out in the base station (the uplink scenario), the computational complexity is not problematic any more for the FD-PSO algorithm.

Table 1: Computational complexity of PSO algorithms

Algorithm	Multiplications	Additions
TD-PSO	$2M^2n + 6Mn + 6$	$2M^2n + 12Mn - 2n + 6$
FD-PSO	$8Mn + 6$	$10Mn + 6$

Simulation results: In this Section, the FD-PSO algorithm is tested on a long term evolution system (uplink SC-FDMA) with a carrier frequency of 2 GHz and a bandwidth of 5 MHz. Quadrature phase shift keying modulation with interleaved mapping on a three-path Rayleigh fading channel is used. $M = 512$, $n = 40$ and the rest of the parameters are similar to those used in [4]. Fig. 1 reports on the performance brought about by the use of the FD-PSO. The product of Doppler frequency, f_d, and the sampling time, T_s, is $f_d T_s = 0.0001$. Almost a 2 dB improvement in BER, at BER $= 10^{-3}$, has been achieved by the FD-PSO over the RLS algorithm. The impact of the number of the particles and re-randomisation is depicted in Fig. 2. By increasing the swarm size, we can improve the performance of the proposed algorithm, this being due to the fact that the particles will cover all the search space, thus allowing the PSO to easily find the potential solution. Similarly, re-randomisation prevents all the particles from converging to a single point, thus improving performance. At a high Doppler, the PSO performs much better than the RLS, as shown in Fig. 2.

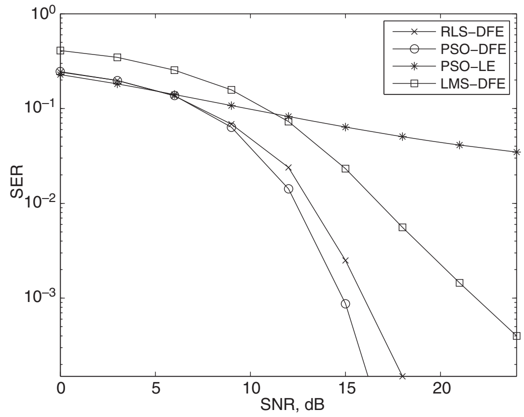

Fig. 1 *Comparison of PSO algorithm with RLS and LMS for $f_d T_s = 0.0001$*

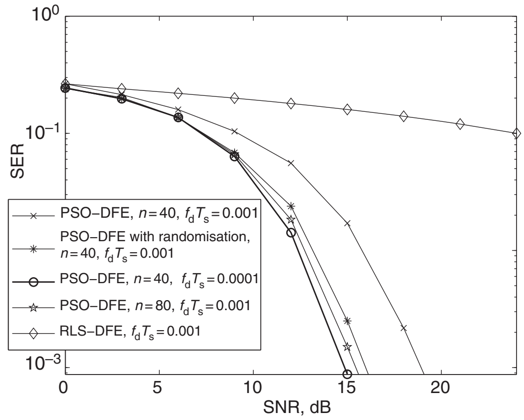

Fig. 2 *Effect of Doppler on PSO algorithm*

Conclusion: An FD-PSO-based adaptive equaliser in an SC-FDMA system is proposed. Simulation results verified that our devised adaptive equalisation scheme has a better performance than that of RLS and LMS. Furthermore, in the case of a high Doppler, the performance of the FD-PSO algorithm can be improved by increasing the number of particles and the use of re-randomisation.

Acknowledgment: This work was supported by grant RG1216, DSR, KFUPM and NPRP no. 6-070-2-024 from QNRF.

15 December 2013
doi: 10.1049/el.2013.4091

N. Iqbal and A. Zerguine (*Department of Electrical Engineering, King Fahd University of Petroleum & Minerals, Dhahran 31261, KSA*)

E-mail: naveediqbal@kfupm.edu.sa

N. Al-Dhahir (*Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX 75083, USA*)

References

1 Haykin, S.: 'Adaptive filter theory' (Prentice-Hall, Upper-Saddle River, NJ, 2002, 4th edn)

2 Krusienski, D.J., and Jenkins, W.K.: 'The application of particle swarm optimization to adaptive IIR phase equalization'. ICASSP, Montreal, Quebec, Canada, May 2004, pp. 693–696

3 Liu, H., and Li, J.: 'A particle swarm optimization-based multiuser detection for receive-diversity-aided STBC systems', *IEEE Signal Process. Lett.*, 2008, **15**, (3), pp. 29–32

4 Al-Awami, A.T., Zerguine, A., Cheded, L., Zidouri, A., and Saif, W.: 'A new modified particle swarm optimization algorithm for adaptive equalization', *Digital Signal Process.*, 2011, **21**, pp. 195–207

5 Kennedy, J., and Eberhart, R.: 'Swarm intelligence' (Academic Press, London, UK, 2001)

6 Zhang, C., Wang, Z., Yang, Z., Wang, J., and Song, J.: 'Frequency domain decision feedback equalization for uplink SC-FDMA', *IEEE Trans. Broadcast.*, 2010, **56**, pp. 253–257