Estimation of Time-Varying Channel State Transition Probabilities for Cognitive Radio Systems by means of Particle Swarm Optimization

Ahmet AKBULUT, Tugrul ADIGUZEL, Asim Egemen YILMAZ

Dept. of Electronics Engineering, Ankara University, Tandogan, 06100 Ankara, Turkey

{aakbulut, adiguzel, aeyilmaz}@eng.ankara.edu.tr

Abstract. *In this study, Particle Swarm Optimization is applied for the estimation of the channel state transition probabilities. Unlike most other studies, where the channel state transition probabilities are assumed to be known and/ or constant, in this study, these values are realistically considered to be time-varying parameters, which are unknown to the secondary users of the cognitive radio systems. The results of this study demonstrate the following: without any a priori information about the channel characteristics, even in a very transient environment, it is quite possible to achieve reasonable estimates of channel state transition probabilities with a practical and simple implementation.*

Keywords

Cognitive radio, channel state transition probability, partially observable Markov decision process, particle swarm optimization.

1. Introduction

Cognitive radio is a novel concept proposed for improvement in the utilization of the radio electromagnetic spectrum. Built on the concept of the software-defined radio, cognitive radio can be considered as an intelligent wireless communication system, which is aware of its environment and which uses the methodology of learning from the environment and adapting to statistical variations in the inputs with two primary objectives:

- highly reliable communication whenever and wherever needed;
- efficient utilization of the radio spectrum [1].

In current spectrum allocation systems, any frequency spectrum is usable for licensed users only. Although the spectrum gets more crowded day-by-day, extensive measurements show that a large portion of licensed spectrum lies unused at any given time and location. Cognitive radio, where secondary users opportunistically share spectrum with primary users (i.e. licensees), is one of the approaches envisioned for solving this problem.

One of the main concepts introduced via the cognitive radio concept is the sensing strategy for secondary users. A secondary user may not be able to sense all channels in the spectrum simultaneously due to various limitations (i.e. hardware oriented constraints, energy limitations).

One of the approaches for the design and development of the sensing strategy might be modeling of the channel occupancy of primary users by means of Markov processes. After that, the sensing strategy can be formulated as a partially observable Markov decision process (POMDP) [2], [3].

Up to now, most of the POMDP based spectrum-sensing strategy research is based on the assumption that the channel state transition probabilities are known by the secondary users, which is in most cases not true in real world. Hence, even though the concept of sensing is valid literally, its practical application is severely limited [4-8].

In [4], quite realistically, the channel state transition probabilities are assumed to be unknown. On the other hand, during the whole operation time, the channel state transition probabilities are assumed to be constant; and these values are estimated by means of a maximum likelihood estimator in [4]. Considering the usage demand of the radio spectrum (and relevant load pattern), the "channel state transition probability" assumption is not a valid assumption.

In this paper, based on the POMDP formulation, a method for estimation of the channel state transition probabilities is proposed. During this study, the channel state transition probabilities are assumed to be unknown similar to [4], and additionally these values are assumed to be time-varying. Consequently, instead of the concrete maximum likelihood estimator, a bio-inspired algorithm (Particle Swarm Optimization) is applied to overcome the difficulties due to the additional complexity of this new problem. In other words, to our belief, the originality of this paper lies in the following points: i) The assumption of unknown and time-varying the channel state transition probabilities; ii) application of the Particle Swarm Optimization for the solution.

This paper is organized as follows: After this introductory section, the network model and the corresponding formulation are presented in Section 2. In Section 3, Particle Swarm Optimization is presented after some discussion regarding the metaheuristics and bio-inspired optimization methods. In Section 4, together with the simulation setup for verification/validation of the proposed approach, the results obtained are presented. Finally, Section 5 includes the concluding remarks and discussions about the potential future work.

2. The Network Model and the Formulation

In our framework, receiver senses through a sensing time with the intention of acquiring the intended level of detection quality. On that account, we assume the sensing error is negligible. We start the formulation by considering a primary network consisting of N synchronous channels with bandwidths B_j ($j = 1,\ldots,N$). For each channel, the channel occupation can be represented via an independent discrete-time Markov process with two states (0 for occupied, and 1 for idle state) as seen in Fig. 1. For a given channel, at time t_i, the state transition probabilities are represented by $(\alpha(t_i),\beta(t_i))$. For purity and simplicity of notation, from now on, $(\alpha(t_i),\beta(t_i))$ will be simply denoted as (α_i,β_i). From the probability theory, the probability that the channel state is idle can be found as:

$$\mu = \frac{\alpha_i}{\alpha_i + \beta_i} \tag{1}$$

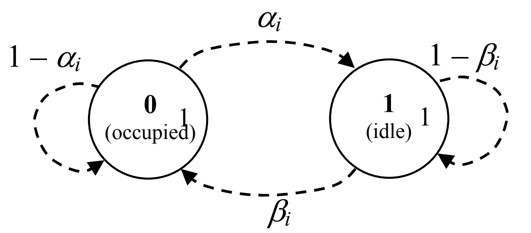

Fig. 1. The Markov channel model for occupancy and relevant transitions.

In practice, a secondary user senses a set of channels, and then utilizes the idle channels for data transmission. Hence, for such a setup described above, it is quite critical for the secondary users to sense which channels are available for usage. This operation, which is called as the sensing strategy, can be formulated as a partially observable Markov decision process (POMDP) as in [2-4].

Depending on the most recent channel state samples obtained from the slots of a fixed channel, a likelihood function can be defined as follows:

Let $\mathbf{Z} = (Z_{k-W+1}, \ldots, Z_k)$ denote the most recent channel state observation samples, where W is the observation window size (i.e. number of observation samples to be considered), and Z_j ($k-W+1 \le j \le k$) take the values 0 (for occupied state) or 1 (for idle state). Assuming that during the whole observation window of size W, the channel state transition probabilities (α_i,β_i) are constant; and by setting $\boldsymbol{\theta}=(\alpha_i,\beta_i)$, the likelihood function can be stated as:

$$L(\boldsymbol{\theta}) = P(\mathbf{Z};\boldsymbol{\theta}) = \\ P(Z_{k-W+1} = z_{k-W+1};\boldsymbol{\theta}) \prod_{j=k-W+2}^{k} P(Z_j = z_j \big| Z_{j-1} = z_{j-1};\boldsymbol{\theta}) \tag{2}$$

Since the possible transitions can be (0,0), (0,1), (1,0), and (1,1); or mathematically stating $(z_{j-1},z_j) \in \{(0,0), (0,1), (1,0), (1,1)\}$, equation (2) can be rewritten as:

$$L(\boldsymbol{\theta}) = P(\mathbf{Z};\boldsymbol{\theta}) = \\ P(Z_{k-W+1} = z_{k-W+1};\boldsymbol{\theta})((1-\alpha_i)^{n_0}\alpha_i^{n_1}(1-\beta_i)^{n_2}\beta_i^{n_3}) \tag{3}$$

where n_0, n_1, n_2 and n_3 represent the number of occurrence of each transition type in a window with size W. Certainly:

$$n_0 + n_1 + n_2 + n_3 = W - 1 . \tag{4}$$

With this formulation, the problem is nothing but a maximization problem (i.e. to find the (α_i,β_i) pair that maximizes the $L(\boldsymbol{\theta})$ value given in (3)). Hence, a reasonable approach for estimation of the channel state transition probabilities (α_i,β_i) would be getting a set of observations (i.e. $\mathbf{Z}$) and to apply an optimization algorithm for the solution of the aforementioned maximization problem.

3. The Solution Procedure

For the solution of the problem mentioned in the previous section, it seems that conventional optimization methods might not work due to additional complexity introduced by additional assumptions. Considering that more assumptions to be made in the future would yield more complications and require more degrees of freedom, we assessed the implementation of a generic framework compulsory. To have more flexibility and modularity for probable ongoing research of this sort, we considered that a bio-inspired optimization algorithm would be more appropriate. In the following, after the advantages of the bio-inspired optimization methods (more generally metaheuristics), the definition and formulation of the Particle Swarm Optimization algorithm is presented.

3.1 Bio-Inspired Optimization Methods

Bio-inspired (or nature-inspired) optimization methods fall into the class of metaheuristics. These are nothing but some methods influenced by the existing behaviors/phenomena for the solution of an optimization-like problem in nature. A very simple source of inspiration is the behavior of a colony or a swarm while searching for food.

In computer science, the term metaheuristic is used for description of a computational method, which optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. In other words, such methods are nothing but systematical trial-and-error approaches. Metaheuristics (sometimes also referred to as derivative-free, direct search, black-box, or indeed just heuristic methods) make few or no assumptions about the problem (such as modality or dimension) being optimized and can search very large spaces of candidate solutions.

However, it should be noted that metaheuristics do not guarantee an optimal solution is ever found. On the other hand, for each algorithm, the literature is full of a plethora of empirical studies, which have been carried out in order to understand how the algorithm parameters should be adjusted in order to increase the performance.

Metaheuristics, more specifically bio-inspired optimization algorithms constitute powerful means for the solution of existing problems in real life. The main factors promoting the usage of such algorithms can be summarized as follows [9]:

- The algorithms make no assumptions (or require no a priori information) about the objective function.
- They do not require the objective function to be continuous or differentiable.
- They can handle complicated models with constraints.
- Almost all of them support parallelization, which yields the solution of very large-scale problems.

3.2 Particle Swarm Optimization

Particle Swarm Optimization is a method proposed by Eberhart and Kennedy [10] after getting influenced by the behaviors of the animals living as colonies/swarms. Similar to the members of the swarms individually searching for the best place for nutrition in 3-dimensional space, the method depends on motions of particles (swarm members) searching for the global best in N-dimensional continuous space. The position of each particle is a solution candidate, and every time the fitness of this solution is recomputed. In addition to its exploration capability (i.e. the tendency for random search throughout the domain), each particle has a cognitive behavior (i.e. remembering its own good memories and having tendency to return there); as well as a social behavior (i.e. observing the rest of the swarm and having tendency to go where most other particles go). The parametric representation of all these tendencies and the balance among them are the keys for the success and the power of the method.

So far, the method has been successfully applied to various multidimensional continuous and discontinuous problems; a recent review article demonstrates that the application spectrum of the method currently is quite wide [11].

All particles are initialized at random positions inside the search space with random velocities. Until the exit criterion is met, the position and velocity of each particle are updated as follows:

$$v_n(k+1) = wv_n(k) + c_1u_1(pbest(k) - x_n(k)) + c_2u_2(gbest(k) - x_n(k)) \tag{5}$$

$$x_n(k+1) = x_n(k) + \Delta t v_n(k) \tag{6}$$

where v_n is the velocity of the particle in the nth dimension, and x_n is its coordinate at the nth dimension; where these two operations are repeated for all dimensions in a multi-dimensional problem. In these equations;

- the so-called inertial weight w is a measure indicating the tendency to preserve the velocity along the previous course. The term inertial weight was not included in the original PSO paper; it was introduced later in [12] in order to improve the performance of the method. Moreover, in a following study they showed that the ideal choice for the inertial weight is to decrease it linearly from 0.95 to 0.4 [13].
- c_1 and c_2 are measures indicating the tendencies of approaching to *pbest* and *gbest*, which are the personal and global best positions, respectively. For recent PSO works, 1.494 seems to be the most preferred value for c_1 and c_2. c_1 represents the tendency of a particle to revisit the position regarding its own personal good memories, hence it is also referred as the "cognitive" factor or tendency. c_2 represents the tendency of a particle to track and pursue the positions of other swarm members, hence it is also referred as the "social" factor or tendency.
- u_1 and u_2 are random numbers between 0.0 and 1.0; and the time step size Δt is usually taken to be unity for simplicity.

The exit criteria for the termination of iterations might be of various forms such as a predefined total iteration number, total number of fitness computations, saturation in improvement in *gbest*, etc.

At each dimension, some restrictions can be defined applied on the movement of the particles, which are referred to as boundary conditions [14]. By means of these rules, which are revisited and re-defined in [15], in a controlled manner, all the particles are inhibited to go out of the search space.

4. Results

In order to verify the applicability of the proposed method the simulator illustrated in Fig. 2 is developed. The simulator is nothing but a channel state modeler, which generates channel states (i.e. responses for interrogations of a cognitive radio of a secondary user) according to a channel state transition probability profile. The channel state transition probability profiler is an engine, which constructs time-varying channel state transition probability profiles (corresponding to the changes in the load pattern), and generates instantaneous channel state transition probability values (i.e. $((\alpha_i, \beta_i))$. The proposed estimation method is implemented in the Channel State Estimator box seen in Fig. 2, with its details illustrated in Fig. 3. In Fig. 2 and 3, $(\hat{\alpha}_i, \hat{\beta}_i)$ denotes the estimate of (α_i, β_i).

With these definitions and formulations, the optimization search space is a two-dimensional subspace, which corresponds to valid α and β values, namely $[0,1]\times[0,1]$. Hence, the position of a particle in PSO implementation corresponds to the estimated channel transition probabilities, $(\hat{\alpha}_i, \hat{\beta}_i)$; and the velocity of a particle corresponds to the rate of change in these estimated channel transition probabilities.

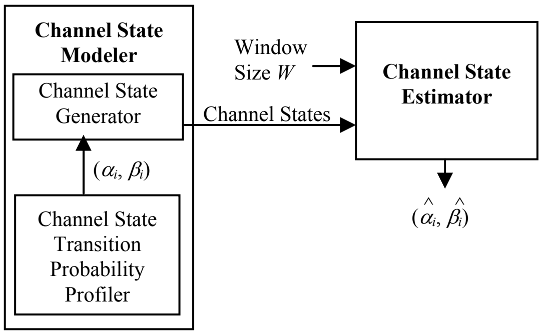

Fig. 2. The simulation setup constructed for verification/ validation of the proposed method.

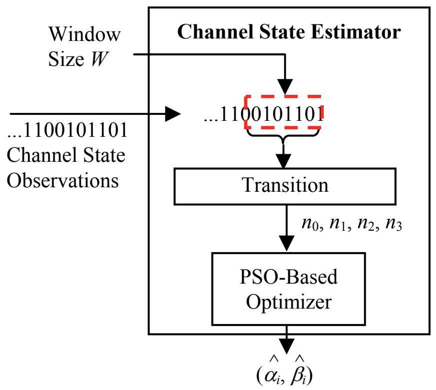

Fig. 3. The structure of the estimator, inside which the proposed method is implemented.

A scenario with the channel state transition probability profile seen in Fig. 4 is constructed throughout the simulations in this study. For the implementation of PSO, the values seen in Tab. 1 are used. In addition, reflecting boundary conditions (according to the definition given in [15]) are implemented in order to keep the particles inside the search domain throughout the whole process.

By means of the simulation setup, first, the impact of the window size on the estimation performance is investigated. Rough and qualitative remarks about the window size can be made as follows: In order to have more accurate estimates, the window size shall be kept sufficiently large in order to consider the contribution of more samples for the solution. On the other hand:

- During the derivation of (2) and (3), the channel state transition probabilities (α_i, β_i) are assumed to be constant along the observation window of size W. Hence, as the window size W increases, this assumption becomes unrealistic. Hence, the window size shall be selected sufficiently small in order to make this assumption valid.

Swarm Size	25
Number of Iterations	50
Cognitive tendency, c_1	1.494
Social tendency, c_2	1.494
Inertial weight, w	0.95 linearly down to 0.4
Search space	$0 \le \alpha_i \le 1$; $0 \le \beta_i \le 1$ (Fig. 5)

Tab. 1. PSO parameters for all solution setups.

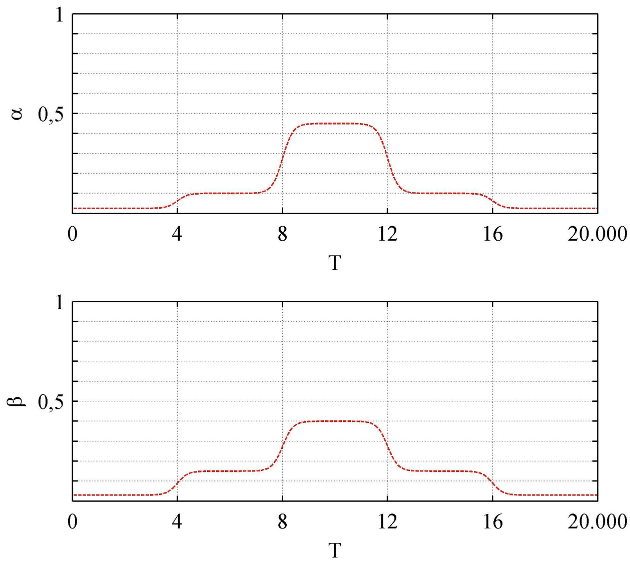

Fig. 4. Channel state transition probability profile used throughout the example (i.e the change in α with respect to time, the change in β with respect to time).

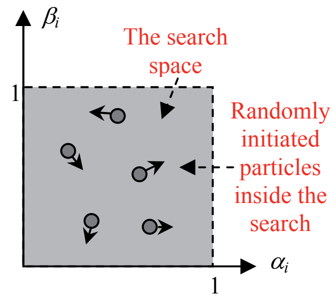

Fig. 5. The search space and initialization of the particles inside it.

- In case that any of the channel state transition probability values changes in a sudden and dramatic manner, large window sizes make it difficult to track the change. In other words, the time-behavior of the estimates (i.e. $(\hat{\alpha}_i, \hat{\beta}_i)$ values) will have some lag relative to the time-behavior of the truth (i.e. (α_i, β_i) values). As the window size W increases, the amount of this lag will increase.

Throughout Fig. 6 to Fig. 8, the performance of the proposed method is illustrated. In Tab. 2, the impact of the window size is presented via some performance metrics (mean absolute error; root mean square error) for the PSO and the Maximum Likelihood (ML) methods. These metrics show that the overall estimation performance is almost equal for varying window sizes. The reason for this can be qualitatively explained as follows: For the regions where the parameter to be estimated is constant, smaller windows size estimates have more ripples. On the other hand, for the regions where the parameter to be estimated is varying, larger windows size estimates have much more lags. Consequently, these two effects yield almost similar overall performances.

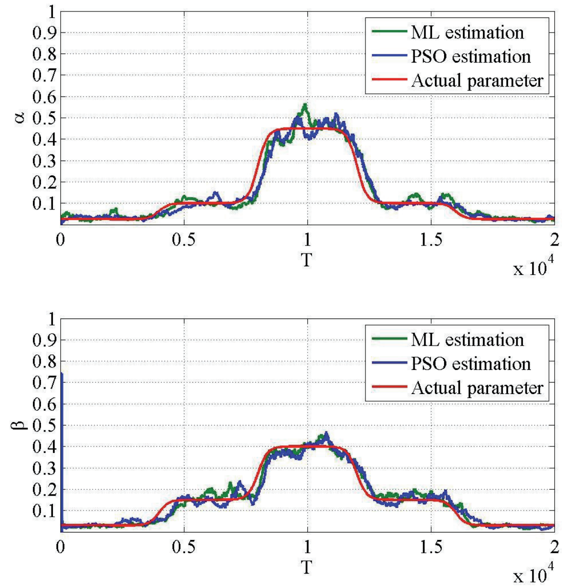

Fig. 6. Estimated channel state transition probability profiles for window size 550.

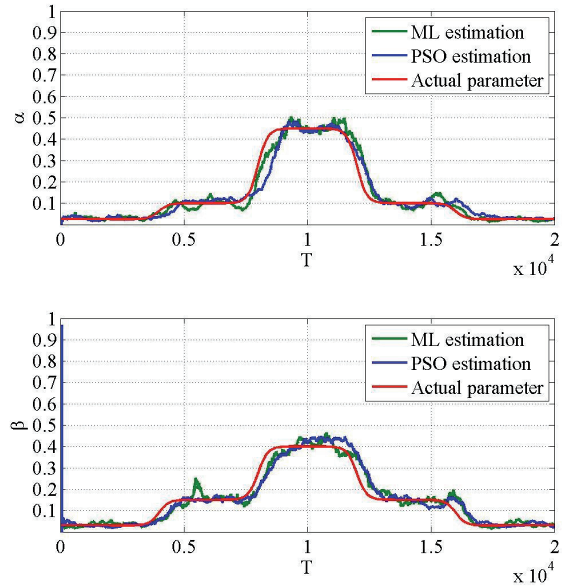

Fig. 7. Estimated channel state transition probability profiles for window size 800.

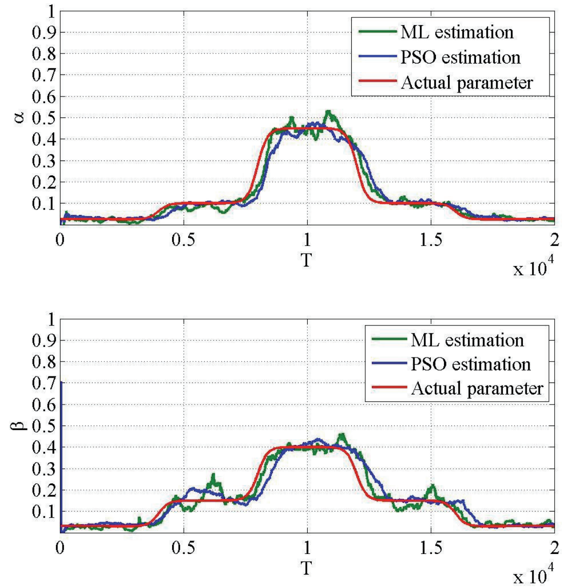

Fig. 8. Estimated channel state transition probability profiles for window size 1050.

Window Size (W)		MAE		RMSE	
		PSO	ML	PSO	ML
550	α	0.0214	0.0223	0.0338	0.0378
	β	0.0233	0.0237	0.0474	0.0333
800	α	0.0220	0.0268	0.0405	0.0737
	β	0.0245	0.0205	0.0364	0.0306
1050	α	0.0234	0.0267	0.0416	0.0508
	β	0.0266	0.0245	0.0431	0.0383

Tab. 2. Performance metrics and the impact of the window size (MAE: Mean Absolute error; RMSE: Root Mean Square Error).

5. Discussions and Conclusion

PSO is an easy-to-implement but powerful tool for the solution of complex multidimensional optimization problems both in continuous and discrete domains. The results of this study demonstrate that it can be applied for estimation of time-varying and unknown channel state transition probabilities, which is practically a quite significant need for secondary users of cognitive radio systems. The results of this study show that it is quite possible to achieve reasonable estimates of channel state transition probabilities with such a practical and simple implementation of the proposed method. Moreover, the proposed approach yields reasonable results even in a very transient environment, and it requires no a priori information about the channel characteristics.

The maximum likelihood may yield good performance, but when the cost function is multimodal, high dimensional and nonlinear the PSO approach will be much better as a fast and robust global search tool. For the time being, during the formulation of the problem, the channel state transition probabilities along a window was assumed to be constant; which yielded simplification in the formulation (and a 2-degree-of-freedom-optimization problem) while sacrificing some accuracy. As a future work, the channel state transition probabilities will also assumed to be varying along a window; which would yield much more degrees-of-freedom. Nevertheless, so far, PSO has proved to be quite successful for optimization problems with high degrees-of-freedom.

In the current work, subsequent $(\hat{\alpha}_i, \hat{\beta}_i)$ estimates are computed independently. Another potential future work is about the performance improvement by combining subsequent estimates in an intelligent manner.

References

[1] HAYKIN, S. Cognitive radio: brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications*, 2005, vol. 23, no. 2, p. 201 – 220.

[2] ZHAO, Q., TONG, L., SWAMI, A., CHEN, Y. Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: a POMDP framework. *IEEE Journal on Selected Areas in Communications*, 2007, vol. 25, no. 3, p. 589 – 600.

[3] ZHAO, Q., TONG, L., SWAMI, A. Decentralized cognitive MAC for dynamic spectrum access. In *Proceedings of the First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN 2005)*. Baltimore (MD, USA), 2005, p. 224 – 232.

[4] LONG, X., GAN, X., XU, Y., LIU, J., TAO, M. An estimation algorithm of channel state transition probabilities for cognitive radio systems. In *Proceedings of the 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom 2008)*. Singapore (Singapore), 2008, p. 1 – 4.

[5] UNNIKRISHNAN, J., VEERAVALLI, V. V. Algorithms for dynamic spectrum access with learning for cognitive radio. *IEEE Transactions on Signal Processing*, February 2010, vol. 58, no. 3, p. 750-760.

[6] FILIPPI, S., CAPPE, O., CLEROT, F., MOULINES, E. A Near optimal policy for channel allocation in cognitive radio. In *Proc. EWRL*, 2008, p. 69-81.

[7] JANG, J., LEE, K. B. Transmit power adaptation for multiuser OFDM systems. *IEEE JSAC*, Feb. 2003, vol. 21, p. 171–178.

[8] KIM, Y. H., SONG, I., YOON, S., PARK, S. R. A multicarrier CDMA system with adaptive subchannel allocation for forward links. *IEEE Trans. on Vehicular Tech.*, Oct. 1999, vol. 48, p. 1747–1758.

[9] YILMAZ, A. E., WEBER, G.-W. Why you should consider nature-inspired optimization methods in financial mathematics. In *CD-ROM Proceedings of the3rd Conference on Nonlinear Science and Complexity (NSC'10)*. Ankara (Turkey), 2010.

[10] KENNEDY, J., EBERHART, R. C. Particle swarm optimization. In *Proceedings of the IEEE Conference on Neural Networks*. Perth (Australia), 1995, p. 1942 – 1948.

[11] POLI, R. Analysis of the publications on the applications of particle swarm optimisation. *Journal of Artificial Evolution and Applications*, 2008, Article ID: 685175, doi:10.1155/2008/685175.

[12] SHI, Y., EBERHART, R. C. A modified particle swarm optimizer. In *Proceedings of the IEEE International Conference on Evolutionary Computation*. Anchorage (AK, USA), 1998, p. 69 – 73.

[13] SHI, Y., EBERHART, R. C. Empirical study of particle swarm optimization. In *Proceedings of the Congress of Evolutionary Computing*. Piscataway (NJ, USA), 1999, p. 1945 – 1950.

[14] ROBINSON, J., RAHMAT-SAMII, Y. Particle swarm optimization in electromagnetics. *IEEE Transactions on Antennas and Propagation*, 2004, vol. 52, no. 2, p. 397 – 407.

[15] XU, S., RAHMAT-SAMII, Y. Boundary conditions in particle swarm optimization revisited. *IEEE Transactions on Antennas and Propagation*, 2007, vol. 55, no. 3, p. 760 – 765.

About Authors ...

Ahmet AKBULUT received his B.S. degree in Electronics Engineering from the Ankara University in 1998. He received his M.S. and Ph.D. degrees in Electronics Engineering from the same university in 2000 and 2006, respectively. His research interests include digital communication, optical communication and mobile robotics.

Tugrul ADIGUZEL received his B.S. degree in Electronics Engineering from the Ankara University in 1999. He received his M.S. and Ph.D. degrees in Electronics Engineering from the same university in 2001 and 2008, respectively. His research interests include nonlinear control theory and robotics in general.

Asim Egemen YILMAZ received his B.Sc. degrees in Electrical-Electronics Engineering and Mathematics from the Middle East Technical University in 1997. He received his M.Sc. and Ph.D. degrees in Electrical-Electronics Engineering from the same university in 2000 and 2007, respectively. He is currently with the Department of Electronics Engineering in Ankara University. His research interests include computational electromagnetic, nature-inspired optimization algorithms, knowledge-based systems; more generally software development processes and methodologies.