# Artificial Neural Network Application in Prediction of Concrete Embedded Antenna Performance

Ju Tan*, Yu Shao*†, Jiliang Zhang*, Jie Zhang*‡
*Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, S10 2TN, UK.
†Chongqing University of Posts and Telecommunications, Chongqing, China.
‡Ranplan Wireless Network Design Ltd., Cambridge, CB23 3UY, UK.
*jtan15@sheffield.ac.uk, y.shao@sheffield.ac.uk, jiliang.zhang@sheffield.ac.uk, jie.zhang@sheffield.ac.uk

***Abstract*—Artificial Neural Network (ANN) has been extensively applied to microwave device modeling, design and simulations. In the present paper, the prediction of concrete embedded antenna performance using ANN is presented. The ANN model takes antenna embedded depth and concrete dielectric constant as inputs and gives antenna radiation efficiency, gain and input impedance as outputs. The Particle Swarm Optimisation (PSO) is employed to search the global optimal weights and bias for ANN, then Bayesian Regularisation (BR) is used to train the ANN for overcoming the overfitting issue. It is found that the PSO computation iteration for optimal network weights and bias searching is less than gradient descent algorithm. A PSO-BR neural network (PSO-BRNN) and back-propagation neural network (BPNN) are trained to compute and predict the antenna performance. The PSO-BRNN performance is better than BPNN in terms of accuracy and generalisation.**

***Index Terms*—Artificial neural network, Bayesian Regularisation, concrete embedded antenna, particle swarm optimisation.**

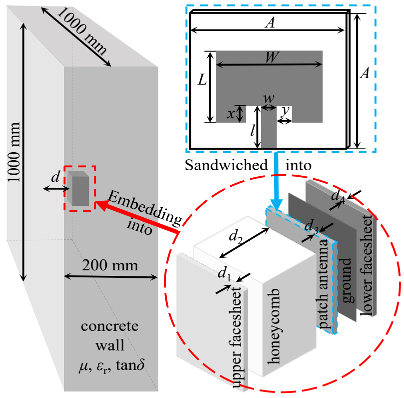


Fig. 1. Antenna model geometry. The parameters of antenna unit are $A$=60 mm, $W$=33.85 mm, $L$=28.39 mm, $x$=8 mm, $y$=2.69 mm, $w$=3.12 mm, and $l$=23.805 mm

## I. Introduction

Mobile operators have experienced an exponential traffic growth in their network in the last decade. Ultra dense cell deployment is considered as a promising way to fulfil the large amount of traffic demand that takes place indoors, and the dense deployment of small cells in buildings facilitates the improvement of throughput in the next generation of cellular communication [1]. However, the physical dimension of indoor small cells can lead to the extra space occupation and disfunction of the building [2]. A feasible solution for these issues is to integrate antennas with the building materials, such as embedding antennas into concrete walls. In fact, due to the strong coupling between antennas and concrete, it is challenging to calculate and predict the antenna performance once antennas are embedded. The numerical method based on full-wave simulations (e.g. method of moments, finite element analysis) can provide rigorous solution to the antenna performance in concrete wall. However, the full-wave simulations are computation-intensive, and requiring a large amount of computation time and computer memory. As a result, time-saving and fast surrogate models are required to address high-dimensional and nonlinear electromagnetic problems.

Artificial neural network (ANN) has already been recognised as a feasible tool for microwave modelling and simulation in recent years [3], which can learn and solve the complex and nonlinear problems in a relative short time. To a certain extent, ANN could be used as a surrogate model that substitutes the computationally intensive EM simulation solver. By far, ANN have been successfully applied to various antenna applications, such as antenna optimisation [4][5] and antenna analysis and synthesis [6]. In [7], the ANN based models were presented to compute the resonant frequency of antenna with lower error. Generally, the gradient based training algorithm such as back-propagation (BP) algorithm is used in ANN training process. However, the slow convergence ratio and local optimum issues are the main drawbacks of gradient based algorithm [8]. As a result, the optimisation algorithm such as particle swarm optimisation (PSO) [9] and genetic algorithm (GA) [10] are combined with ANN and can significantly improve the performance of ANN. In addition, the PSO could also be applied to optimise the topology and parameters in the ANN [8]. On the other hand, the overfitting issue and generalisation capability are main concerns for ANN model. In order to improve the accuracy of ANN, Early-Stopping strategy and Bayesian Regularisation (BR) [11] are introduced to address these issues, which prevents the over-training occurring and effectively improves the generalisation capability of network.

In the present work, ANN is utilised to compute and predict the performance of a concrete embedded antenna for indoor

TABLE I
ELECTRICAL PROPERTIES AND THICKNESS FOR EACH LAYER

| Layer | Material | $\epsilon_r$ | $\tan\sigma$ | Thickness (mm) |
|---|---|---|---|---|
| UF | Roger 3003 | 3 | 0.001 | $d_1 = 0.25\text{mm}$ |
| HC | Air | 1 | 0 | $d_2 = 10mm$ |
| Substrate | Roger 5880 | 2.2 | 0.0009 | $d_3 = 1mm$ |
| LF | Roger 3003 | 3 | 0.001 | $d_4 = 0.25mm$ |

communications. A hybrid ANN model with PSO, BP and Bayesian Regularisation (PSO-BRNN) and a classic back-propagation neural network (BPNN) are developed for the computation and prediction. The performance of PSO-BRNN in terms of accuracy and computational savings are compared with BPNN, and the generalisation capabilities of PSO-BRNN and BPNN are tested.

## II. ANTENNA SYSTEM MODEL

An structurally integrated antenna with multi-layer configuration that proposed in [12] is selected because of its excellent mechanical and electrical performances. The antenna is fully embedded in a solid concrete slab as shown in Fig. 1, and the concrete has a dimension of $1000\,\text{mm} \times 1000\,\text{mm} \times 200\,\text{mm}$. The embedding depth *d* of antenna is measured as the distance between the top concrete-air interface and the top surface of the antenna. The effect embedding depth *d* and concrete dielectric constant $\epsilon_\text{r}$ on antenna performance are going to be investigated, thus other electrical property such as loss tangent is fixed to 0.03 ($\tan\delta = 0.03$). The proposed antenna is optimised to operate at 3.5 GHz, and it is sandwiched among lower facesheet (LF), a honeycomb (HC) structure and an upper facesheet (UF) for obtaining better electrical and mechanical characteristics in the concrete wall, the electrical properties and thickness of each layer are listed in Table I.

## III. ANN ARCHITECTURE AND TRAINING

### A. ANN construction and data preparing

Neural network is a powerful tool to map the nonlinear and complicated relationship between inputs and outputs. In the present work, ANN model is used for the prediction of proposed antenna performance while embedded in the concrete wall. For the concerned input variables, the antenna embedding depth *d* and the concrete dielectric constant $\epsilon_\text{r}$ are selected, while the antenna's radiation efficiency $\eta_\text{rad}$, gain $G$, input resistance $R_\text{in}$ and input reactance $X_\text{in}$ are considered as outputs. Therefore, the suggested network architecture consists of one hidden layer, 2 input neurons and 4 output neurons as shown in Fig. 2. The hidden layer consists of 75 neurons which are fully connected to the output layer that gives the desired values of antenna performance. The activation function used in hidden layer is tangent sigmoid, while simple linear function is used in the output layer.

Given a data set $\boldsymbol{D} = [\boldsymbol{x}_j, \boldsymbol{y}_j]^T$ consists of inputs vector $\boldsymbol{x}_j$ and outputs vector $\boldsymbol{y}_j$, a supervised nonlinear regression task is going to be solved by ANN. The relationship between the inputs vector and the outputs vector could be written as:

$$\boldsymbol{y}_j = f(\boldsymbol{x}_j), \tag{1}$$

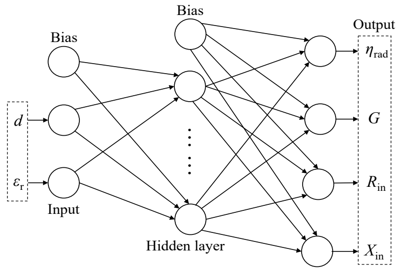


Fig. 2. Architecture of neural network model

where the corresponding inputs and outputs of ANN model are:

$$\boldsymbol{x}_j = [d, \epsilon_\text{r}]^T, \tag{2}$$

$$\boldsymbol{y}_j = [\eta_\text{rad}, G, R_\text{in}, X_\text{in}]^T. \tag{3}$$

The range of inputs are 0.001 m to 0.189 m with step width of 0.002 m for the embedding depth *d*, and 4 to 9 with step width of 1 for the concrete dielectric constant $\epsilon_\text{r}$. The data set $\boldsymbol{D}$ is generated using Computer Simulation Technology (CST) Studio. The length of $\boldsymbol{D}$ is 570, all the obtained data have been normalised between 0 and 1 for avoiding the error caused by different order of magnitude. The data set $\boldsymbol{D}$ is randomly divided as training set and testing set, wherein, 85% data are classified as training set and the rest of 15% are testing set. All the optimisations and ANN trainings are performed on an Intel Xeon W2135 3.70 GHz machine with 32 GB RAM.

### B. PSO and ANN

The conventional ANN utilises the gradient based method to train generally, and the convergence of ANN strongly depends on the initial guess of weights and bias. BP algorithm is a well known training method for neural networks, it is based on the gradient descent algorithm. Hence, the initial point of weights and bias is essential for the BP training, if the weights and bias are not initialised properly, the results are likely to get stuck in a local optimum and consequently the solution is not the best.

PSO is a random search algorithm based on group cooperation, which is developed by emulating the foraging behavior of birds. It is an effective evolutionary algorithm that can find the global maximum or minimum of target function. In this study, the mean square error of neural network is taken as the evaluated fitness in PSO, which is calculated as:

$$E = \frac{1}{N}\sum_{1}^{N}\sum_{1}^{k}(\widehat{\boldsymbol{y}}_j - \boldsymbol{y}_j)^2, \tag{4}$$

where $\widehat{\boldsymbol{y}}_j$ is the network outputs vector, $\boldsymbol{y}_j$ is the outputs vector of data set $\boldsymbol{D}$, *N* is the total number of data, *k* is the total number of output. In the present work, the *N* and *k* are 570 and 4, respectively.

(4) is the target function that needs to be optimised in PSO. Since the neural network learning process is mainly to update the weights and bias, thus the location of the particles in PSO are corresponding to all weights and bias in the network. *E*

TABLE II
PSO PARAMETERS

| Parameter | value |
|---|---|
| Number of particle | 500 |
| Position boundry | [-1,1] |
| Velocity boundry | [-0.8,0.8] |
| inertial weight | [0.2,1] |
| learning factor $c_1$ | 2 |
| learning factor $c_2$ | 2 |
| Maximum iteration | 700 |

is taken as the fitness function of the PSO algorithm, and all the weights and bias are optimised by PSO algorithm in order to obtain the global minimum of the fitness function. In each iteration, the fitness function of each particle is calculated, and the corresponding position $P_i$ and velocity $V_i$ are updated according to the calculated value of fitness function, personal best $p_{\text{best}}$ and global best $g_{\text{best}}$, and the updated regulations are:

$$V_i = \omega V_i + c_1\phi_1(p_{\text{best}} - P_i) + c_2\phi_2(g_{\text{best}} - P_i), \quad (5)$$

$$P_i = P_i + V_i, \quad (6)$$

where $c_1$ and $c_2$ are acceleration coefficients, $\phi_1$ and $\phi_2$ are random and positive number with uniform distribution ranged between 0 and 1, $p_{\text{best}}$ is the personal best position of particle, and $g_{\text{best}}$ is the global optimum position of particle. $\omega$ is the inertial weight, the linear decline weight (LDW) strategy is used to manipulate $\omega$ for the optimum solution search. The larger $\omega$ facilitates global searching, while the smaller $\omega$ is beneficial to precise local searching. The LDW strategy is expressed as:

$$\omega = \omega_{\max} - \frac{t \times (\omega_{\max} - \omega_{\min})}{t_{\max}}, \quad (7)$$

where $\omega_{\max}$ is the maximum inertial weight, $\omega_{\min}$ is the minimum inertial weight, $t$ is current iteration, and $t_{\max}$ is the maximum iteration of PSO.

At the beginning, the ANN is building with specific topology, thus the dimension of particle can be determined. The dimension of each particle equals to the total number of weights and bias in the network, in this work we implement a PSO to optimise 529 weights and bias in total. 500 particles are employed and the computation iterates 700 times. Firstly, the number of particle is selected, then the particle positions and velocities are randomly initialised, each particle $i$ is characterised by its position vectors $X_i$ and velocity $V_i$. The position boundry [$-X_{\max}$, $X_{\max}$], velocity boundry [$-V_{\max}$, $V_{\max}$], inertial weight range [$\omega_{\min}$, $\omega_{\max}$], acceleration coefficients $c_1$ and $c_2$, and the maximum iteration are defined, and these parameters are presented in Table II.

In each iteration of PSO, the value of fitness function of individual particle is calculated, and the velocity and position of all particles is updated using (5) and (6). Once the optimisation criteria meets, the PSO iteration terminates. $g_{\text{best}}$ stores the global optimum solution for the network weights and bias, then the $g_{\text{best}}$ is reshaped and assigned according to the topology of network which is prepared to be trained.

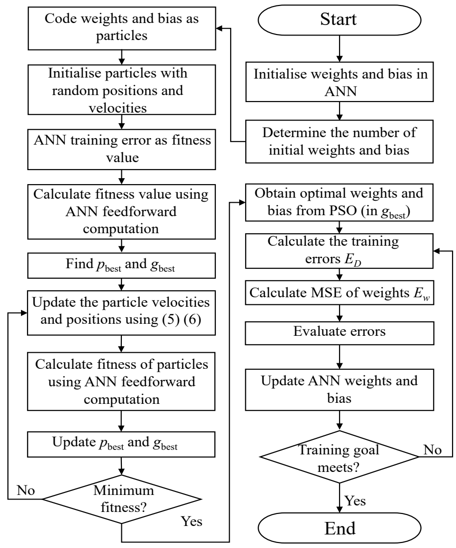


Fig. 3. The training process of PSO-BRNN

### C. Bayesian regularisation of ANN

The Bayesian regularisation (BR) is used to mitigate the potential overfitting problem that may occurs in ANN training process. The overfitting and overtraining can lead to the loss of regression accuracy and generalisation of the network. For the purpose of overcoming overfitting issue, the BR adds an addition regularisation term to the object function as:

$$F = \beta E_D + \alpha E_w, \quad (8)$$

where the $F$ is the objective fucntion, $E_D$ is the sum of squared errors of network, $E_\omega = \frac{1}{m}\sum_{i=1}^{m} w_i^2$ is the sum of squared errors of the weights in network, $m$ is the total number of weights. $\alpha$ and $\beta$ are the hyperparameters that need to be estimated in the training process. Network weights $\omega$ are regarded as random variables and the density function could be written as:

$$P(w|D,\alpha,\beta,M) = \frac{P(D|w,\beta,M)P(w|\alpha,M)}{P(D|\alpha,\beta,M)}, \quad (9)$$

where $D$ represents data set, and $M$ is the ANN topology; $P(w|D,\alpha,\beta,M)$ is the posterior distribution of ANN weights, $P(D|w,\beta,M)$ is the likelihood function represents the training data occurrence probability with given weights, $P(w|\alpha,M)$ is the prior density of weights before data is fed. The BR algorithm is explained in detail in [10]. In general, all the noise in data is assumed to be Gaussian additive noise, with this assumption the probability density function of weights in (9) could be estimated. Then the hyperparameters $\alpha$ and $\beta$ are determined by solving the Hessian matrix of $F$ at the minimum point. Gauss-Newton approximation is used to solve Hessian matrix while the Levenburg-Marquardt (LM) training algorithm is used to search the minimum point, the training process terminates once the training goal is met. The flow chart summarizing major step of PSO-BRNN training is shown in Fig. 3.

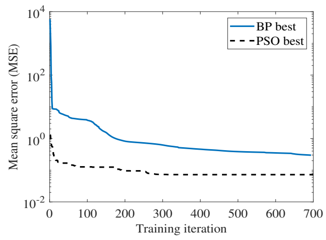


Fig. 4. Comparison between PSO and BP algorithm

TABLE III
TRAINING METRICS OF THE DIFFERENT ALGORITHM

| Algorithm | MSE | MAPE (%) | Iteration | Time (s) |
|---|---|---|---|---|
| BPNN | 0.15 | 3.92 | 1984 | 270 |
| PSO-BRNN | 0.0002 | 1.79 | 1076 | 151 |

## IV. RESULT AND DISCUSSION

PSO are used for ANN learning process, weights and bias are adjusted in order to reach the minimum of the error between ANN estimation and actual values. The training effect of PSO is compared with the BP algorithm, the comparisons of convergence and regression accuracy are exhibited. Fig. 4 presents the comparison of PSO and BP algorithm in terms of convergence rate. It can be observed that PSO performs better than the BP algorithm. PSO converges faster than BP algorithm, and the iteration is terminated with a lower mean square error (MSE) and mean absolute percentage error (MAPE) which is calculated in (10). The training metrics such as performance, accuracy and training time are illustrated in Table III. It apparently shows that the MSE of PSO-BRNN is much lower than the classic BPNN, with 0.0002 to 0.15, as well as the error ratio, 1.79% to 3.92%. In addition, by applying the PSO and Bayesian regularisation, the iteration times of convergence is lower than BPNN, thus result in the reduction of training time of PSO-BRNN (151 seconds) than BPNN (270 seconds).

$$MAPE = \frac{1}{N}\sum_{1}^{N}\sum_{1}^{k}\left|\frac{\widehat{\boldsymbol{y}}_j - \boldsymbol{y}_j}{\boldsymbol{y}_j}\right| \times 100\%, \qquad (10)$$

Fig. 5 presents the antenna performance prediction results of PSO-BRNN and BPNN with the actual value as reference. It can be observed that the learning accuracy of PSO-BRNN is better than the BPNN, the BPNN cannot map the fluctuation as the embedded depth increases. This problem is caused by the local minima issue, once a network is trained with gradient descent based algorithm, the local minima is likely to be considered as the best result by network. Therefore the error between network estimation and actual value cannot be further minimised, then the weights and bias in the network stop adjusting and maintain in a plateau. While the PSO-BRNN is trained with optimum weights and bias, so it can map the nuanced fluctuation of the antenna performance, which indicates the learning ability of PSO-BRNN is better.

The generalisation capability is essential for networks, and the performance of network is mainly measured by its generalisation capability. For testing the generalisation capability of trained ANN model, the data other than that used in training process is introduced. The selected embedded depth $d$ and dielectric constant $\epsilon_r$ are exclusive from the data set $\boldsymbol{D}$, the $d$ is sampled with step width of 0.004 m and ranged from 0.001 m to 0.189 m, and the $\epsilon_r$ is 4.5. The outputs of different networks are obtained and depicted in Fig. 6. The PSO-BRNN gives rational responses to the new inputs vector, the antenna performance curve tendency is agree with the actual values. The BPNN performs poorly when a novel inputs is fed, the regression curves of BPNN are diverged beyond the point where $d$ approximates to 0.07 m, and its MSE of generalisation is larger than PSO-BRNN, which are 30.46 and 12.64, respectively.

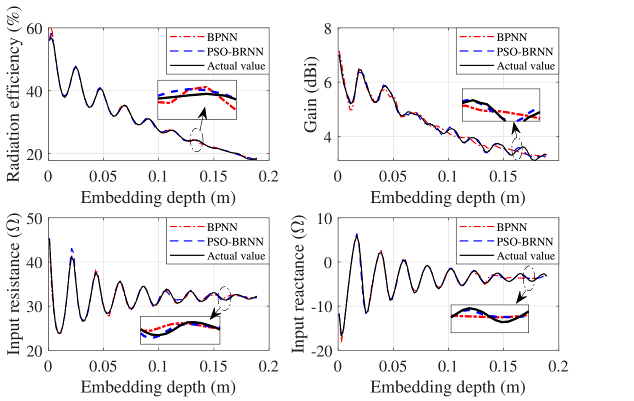


Fig. 5. Comparision of BPNN prediction, PSO-BRNN prediction and actual value for the antenna performances

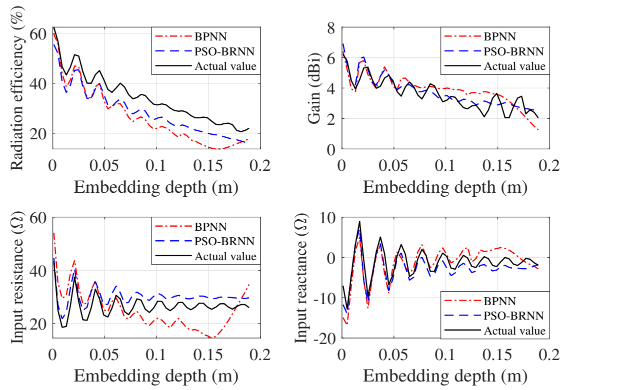


Fig. 6. Generalisation capability comparision of BPNN and PSO-BRNN

## V. CONCLUSION

In this paper, the ANN-based method has been presented to predict the performance of concrete embedded antenna. A hybrid ANN (PSO-BRNN) is trained to predict the performance of concrete embedded antenna, and the training metrics are compared to the BPNN. The PSO algorithm is utilised to search for the global optimum weights and bias for ANN, and the BR algorithm is employed to overcome the overfitting issue of ANN. Compared to BPNN, PSO-BRNN exhibits an more accurate and efficient manner in computation and prediction, and leads to a reduction in MSE and iteration times. The generalisation capability of different networks is tested with the new inputs vector, the outputs of PSO-BRNN reveals an excellent generalisation capability, and its learning ability excels BPNN. The result indicates that the PSO-BRNN is an effective method for the concrete embedded antenna performance prediction for indoor communication.

## ACKNOWLEDGMENT

The research is funded in part by the European Union's H2020-MSCA-IF-AceLSAA (752644) and in part by the European Union's H2020-MSCA-IF-GATE (843133).

## REFERENCES

[1] J. Zhang, A. A. Glazunov and J. Zhang, "Wireless energy efficiency evaluation for building under design based on analysis of interference gain," *IEEE Trans. Veh. Tech.*, vol. 69, no. 6, pp. 6310-6324, 2020.

[2] Y. Zhang, C. Chen, S. Yang, J. Zhang, X. Chu and J. Zhang, "How friendly are building materials as reflectors to indoor LOS MIMO communications?" *IEEE Internet Things J.*, vol. 7, no. 9, pp. 9116-9127, 2020.

[3] Q. Wu, Y. Cao, H. Wang and W. Hong, "Machine-learning-assisted optimization and its application to antenna designs: opportunities and challenges," *China Commun.*, vol. 17, no. 4, pp. 152-164, 2020.

[4] L. Xiao, W. Shao, F. Jin and B. Wang, "Multiparameter modeling with ANN for antenna design," *IEEE Trans. Antennas Propag.*, vol. 66, no. 7, pp. 3718-3723, 2018.

[5] Q. Wu, H. Wang and W. Hong, "Multistage collaborative machine learning and its application to antenna modeling and optimization," *IEEE Trans. Antennas Propag.*, vol. 68, no. 5, pp. 3397-3409, 2020.

[6] Z. Wang, S. Fang, Q. Wang and H. Liu, "An ANN-based synthesis model for the single-feed circularly-polarized square microstrip antenna with truncated corners," *IEEE Trans. Antennas Propag.*, vol. 60, no. 12, pp. 5989-5992, 2012.

[7] S. Sagiroglu and A. Kalinli, "Determining resonant frequencies of various microstrip antennas within a single neural model trained using parallel tabu search algorithm," *Electromagnetics.*, vol. 25, pp. 551-565, 2005.

[8] E. A. Grimaldi, F. Grimaccia, M. Mussetta and R. E. Zich, "PSO as an effective learning algorithm for neural network applications," in *Proc. 3rd Int. Conf. Comput. Electromagn. Appl.*, pp. 557-560, 2004.

[9] H. Moayedi, A. Moatamediyan, H. Nguyen, X. Bui, D. T. Bui and A. S. A. Rashid, "Prediction of ultimate bearing capacity through various novel evolutionary and neural network models," *Engineering with Computers*, vol. 36, no. 2, pp. 671-687, 2020.

[10] D. F. Cook, C. T. Ragsdale and R. L. Major, "Combining a neural network with a genetic algorithm for process parameter optimization," *Eng. Appl Artif. Intell.*, vol. 13, pp. 391-396, 2000.

[11] D. J. MacKay, "A practical bayesian framework for backpropagation networks," *Neural computat.*, vol. 4, no. 3, pp. 448-472, 1992.

[12] C. S. You, M. M. Tentzeris and W. Hwang, "Multilayer effects on microstrip antennas for their integration with mechanical structures," *IEEE Trans. Antennas Propag.*, vol. 55, no. 4, pp. 1051-1058, 2007.