DOI:10.24996/ijs.2018.59.1C.16

ISSN: 0067-2904

Hybrid Approach of Prediction Daily Maximum and Minimum Air Temperature for Baghdad City by Used Artificial Neural Network and Simulated Annealing

Hind Saleem Ibrahim Harba
Department of Atmospheric Science, University of Mustansiriyah, Iraq, Baghdad, Iraq.

Abstract

Temperature predicting is the utilization to forecast the condition of the temperature for an upcoming date for a given area. Temperature predictions are done by gathering quantitative data in regard to the current state of the atmosphere. In this study, a proposed hybrid method to predication the daily maximum and minimum air temperature of Baghdad city which combines standard backpropagation with simulated annealing (SA). Simulated Annealing Algorithm are used for weights optimization for recurrent multi-layer neural network system. Experimental tests had been implemented using the data of maximum and minimum air temperature for month of July of Baghdad city that got from local records of Iraqi Meteorological Organization and Seismology (IMOS) in period between 2010 to 2016. The results show that the proposed hybrid method got a high accuracy prediction results that reach nearly from real temperature records of desired year.

Keywords: Artificial Neural Network, Backpropagation Neural Network, Simulated Annealing, Prediction Air Temperature.

أسلوب الدمج للتنبؤ بدرجات حرارة الهواء الكبرى والصغرى لمدينة بغداد باستخدام تقنيتي الشبكات العصبية الصناعية مع طريقة محاكاة التلدين

هند سليم أبراهيم حربه
قسم علم الجو، كلية العلوم، الجامعة المستنصرية، بغداد، العراق

الخلاصة

التنبؤ بدرجات الحرارة يستخدم لتوقع شروط التغير في درجات الحرارة للأيام القادمة للمنطقة المحددة. ان التنبؤ بدرجات الحرارة يتم من خلال جمع البيانات الكمية الخاصة بالحالة الراهنة للغلاف الجوي. في هذه الدراسة، تم تصميم طريقة هجينة لغرض التنبؤ بدرجات الحرارة اليومية الكبرى والصغرى لمدينة بغداد والتي تتضمن دمج طريقة الانتشار الخلفي (Backpropagation) مع طريقة محاكاة التلدين simulated (annealing). ان خوارزمية محاكاة التلدين تستخدم لتحسين الأوزان لنظام الشبكة العصبية العكسية متعددة الطبقات. تم تحضير الاختبارات المختبرية من خلال استخدام بيانات درجات حرارة الهواء الكبرى والصغرى لشهر تموز لمدينة بغداد والتي تم الحصول عليها من التسجيلات المحلية للهيئة العامة للأنواء الجوية والرصد الزلزالي في الفترة من سنة (2010) الى سنة (2016). أظهرت النتائج ان الطريقة الهجينة المقترحة حصلت على نتائج تنبؤ عالية الدقة والتي اقتربت من القراءات الحقيقية لدرجات الحرارة للسنة المطلوبة.

Email: hindharba76@yahoo.com

1. Introduction

Artificial neural networks (ANN) are one of the best learning methods known in the present day to solve the certain types of problems, that gives a powerful strategy to estimating vector-valued, discrete-valued and real-valued target functions. Backpropagation neural network (BPNN) is the most common method of ANN technologies. BPNN is a type of multi-layer feed forward neural network model, will be able to learn a wide range of model mapping relationship, which simulate the intelligent behavior of human brain, and is commonly used in the area of classification and pattern recognition, as well as prediction and many other various fields, this because it has strong adaptive ability. Nevertheless, the BPNN is dependent upon the error gradient descent process that the weight inevitably is categorized as local minimum points, besides slow convergence speed and may easy to cause faults for example shock. Simulated Annealing (SA) is great at global searching, and search for accuracy seems to be partial capacity Limited. SA offers a significant advantage over alternative methods is the possibility to avoid being trapped at local minimum. SA algorithm uses a random search that not only allows changes that reduce objective function, as well as several changes that raise it. The SA algorithm works effectively on a neighborhood search within solution space, acceptance probability, and inferior ways to avoid trap (i.e., local optimal solution)] [1, 2].

In this study, a temperature predicting models has been designed for predicting the minimum and maximum air temperature for month of Juley by using a combination of artificial neural network (ANN) and Simulated Annealing (SA). Minimum and maximum temperature for lead seven days were taken into account in this analysis Iraq Meteorological Organization and Seismology (IMOS) provides forecast of deviation of minimum and maximum temperature for one week and forecast of minimum and maximum temperature are very essential whenever there was a high or low temperature epoch.

2. Statement of Problem

BP is a typical approach to teaching ANN the way to perform a specified task. It is classified as supervised learning procedure, which is a generalization of the delta rule. BP needs a teacher that knows, or will be able to calculate, the required output for every input in the training set. BP is very useful for feedforward networks (FFNN) The name is an acronym with regard to "backward propagation of errors". However, this method provides a produced with promising results. The SA has not fully realized neural network ability to estimate any unidentified function for any preferred degree of accuracy, since of its tendency being trapped in local optima. Studies have used a range of methods attempt to adjust for this characteristic of BP. As an example, algorithm parameters can be modified to affect the step size or momentum of the search which means the search will get out of locals and go to the global solution. The appropriate value of such parameters, however, isn't known a priori which is usually problem specific. Thus, for any given problem, a number of parameters should be tried to create confidence that a global solution will found. Because this algorithm was made to converge on local solutions, the dependency takes place between the initially randomly starting points and the located solutions. However, several initial points are often tried, there isn't any known strategy, beyond mere chance, for locating a global solution with BP [3, 4].

SA is the more useful method could be used to replace the gradient based search method with a global search method. Many studies have been done in field of predicating air temperature using ANN, some dependent on local search such as used BPNN, some on global search techniques such as genetic algorithm, and some hybrid local and global search such as hybrid genetic to persistently find superior solutions in comparison with BP for training ANN. In this work, we investigate the use of simulated annealing (SA) with BP for ANN training.

3. Search Algorithms

The next sections give a historical background of the SA and BP algorithms in addition to a general explanation of these algorithm that used in this research. The compared between these algorithms are dependent on their suitability for training ANN, as well as the need for parameter adjustment for locating superior solutions

3.1 Backpropagation Neural Network

Predicting with ANN is common due to its capability in estimating any nonlinear function and finding prediction via learning process. In this particular analysis, we focused on prediction network that used feed-forward BPNN [5].

A BP network is consisted of at minimum three layers (multilayer perception): input layer, hidden layer, and output layer. As opposed to Hopfield networks and Interactive Activation and Competition

(IAC) ANN, the connection weights in BPNN are one way. Generally, units are linked in a feed-forward model with input units completely linked to units in the hidden layer, then these hidden neurons are fully connected to the output neurons. When a BPNN is cycled, it propagated the input pattern toward the output units via the intervening hidden to output and input to hidden weights. AS the BP algorithm name means, the errors (and so the learning) is propagates backwards from output layer nodes to the inner layer nodes. So theoretically, BP is utilized to evaluate the gradient of the error of the network according to the network modifiable weights. Gradient is often utilized in a basic stochastic gradient descent algorithm for getting weights that reduce the error. BP generally allows for rapid convergence on satisfactory local minima for error in the type of networks that it is suitable [6].
The BP algorithm can be applied as follows [7]:
1. Initialize every bias and weights and normalize the data of training;
2. Computing the output of hidden layer neurons and in the output layer using

$$n_i = \sum w_{ij}x_i + \theta_i \quad (1)$$

Where: x_j is the transformation function**,** θ_i is the bias, w_{ij}, the weight of the node (j) that can be calculated as follows:

$$w_{ij}(n+1) = w_{ij}(n) + \alpha\delta_i(n)x_j(n) \quad (2)$$

Where: $\boldsymbol{\alpha}$ is the step size and the $\boldsymbol{\delta_i(n)}$ is the weighted sum of error, and;
3. Computing the output of hidden layer neurons and in the output layer by use modified delta rule;
4. Update all bias and weights.
5. Repeat Steps 2 and 3 for all training data;
6. Repeat Steps 2 to 4 until the error converges to a suitable level.

3.2 Simulated Annealing Algorithm (SA)

SA is a common probabilistic meta algorithm used for the global optimization issue, specifically finding a good approximation for the global optimum of the given function in any large search space. SA, has been inspired by Boltzmann equation and the physical annealing process. It has been suggested by Kirkpatrick et al. [8] to be an algorithm for solved the combinatorial optimization problems. In addition to it is easy implementation, this algorithm presents the wonderful option of finding global optimal solutions due to the flexibility to analyze suboptimal solutions over the method execution. In opposition to SA, BP algorithm using the Delta Rule [9] is a local search algorithm, therefore it is not capable of locating global optimal solutions [10].

SA can solve the local optimum problem by including a few amounts of randomness that allows to allow lesser states earlier in the search using the intent of allowing to overcome local or suboptimal optimum solutions. It could be noticed that through the iterative state search, the optimal solution is always remembered, as a result even when accepting a weak solution will not eventually make a better solution the algorithm continues to maintain a better solution. The SA standard algorithm is the Kirkpatrick SA (KSA), which included a temperature schedule for effective searching. KSA solves the local optimum problem via integrating a few amounts of randomness generic the Metropolis Monte Carlo integration algorithm (MMCI) [11]. The randomness permits to accept a weak state early in the search with the intent of allows to overcome suboptimal or local optimum solutions. A sufficiency proof was then proven to place on lower bound on that schedule as 1/ log (t), in which it is an artificial time measure of the annealing schedule [12].

The Simulated Annealing algorithm can be can be described as in follows [5]:
- At First, set the initial temperature and create a random initial solution.
- Start looping until stop condition is met. Usually either the system has sufficiently cooled, or a good-enough solution has been found.
- Select a neighbor by making a small change to current solution.
- Make decision whether to move to that neighbor solution.
- Finally, decreased the temperature and continue looping.

SA for combinatorial optimization has a generator, which outputs a randomly-chosen state from any input state. The set of output states produced from input states is called the *neighborhood* of **s**. The algorithm randomly chooses a first state, and then starts a loop. The loop calls the generator to obtain a trial-state from the current-state. If the trial-state has lower cost, the algorithm selects the trial-state as its new current-state, otherwise, is selects the old current-state. The SA continues its loop, generating trial-states and selecting the next state until some stopping criteria is met, usually when it sees no further improvement for several iterations. The SA algorithm then returns the current-state as its outcome.

SA augments the greedy algorithm with a random escape from local minima. The escape is controlled through a value called "temperature". Higher temperatures make the algorithm more likely to increase cost when selecting a trial-state. In this way, SA can "climb out" of a local minimum. Figure-1 illustrated the algorithm and details of algorithm.

```
1.  T ←— T0;
2.  s ←— starting – state;
3.  E ←— C(s);
4.  while not stopping-criteria()
5.      s' ←— generate (s) with probability Gss';
6.      E' ←— C(s');
7.      Δ ←— E' – E;
8.      if (Δ ≤ 0) ∨ (random () < e^(-Δ/T) )
9.          s ←— s';
10.         E ←— E';
11.     T ←— reduce-temperature (T);
12.  end while;
```

Figure 1-The SA algorithm [13].

As shown from figure 1, the line 1 sets the initial temperature to T_0. Lines 2 and 3 set the current-state **s** and its cost *E*. the loop at lines 4-12 generates a trial-state'**s**', evaluates the change in cost **Δ**, selects the next current-state, and reduces the temperature until the stopping criteria is met. Line 8 shows how simulated annealing accepts a trial state. The first term, ($\boldsymbol{\Delta} \leq o$), expresses greedy it always accepts a lower-cost trial state. The random function returns a uniformly distributed random value between 0 and 1. The second term of line 6, (random () < $e^{-\Delta/T}$), expresses the likelihood of accepting a costlier trial-state. When the stopping criterion is met, SA returns current-states as its outcome.

SA has a useful property at a fixed temperature, it "equilibrates". That is, it approaches a stationary probability distribution, or "equilibrium" temperature changes are usually chosen to keep transient distributions close to equilibrium. SA equilibrium is the "Boltzmann distribution", a probability distribution dependent solely on the cost-function. These terms-annealing, equilibrium, temperature, Boltzmann distribution, etc.– come from thermodynamics. SA solves the local optimum problem by incorporating a small amount of randomness. The randomness allows us to accept poorer states earlier in the search with the intent of allowing us to overcome suboptimal or local optimum solutions. It should be noted that during the iterative state search, the best solution is always remembered, so even if accepting a poor solution does not eventually result in a better solution the algorithm still maintains the better solution. The standard of SA is the Kirkpatrick algorithm to include a temperature schedule for efficient searching Kirkpatrick SA solves the local optimum problem by incorporating a small amount of randomness generalized the Metropolis Monte Carlo integration algorithm [11]. The randomness allows us to accept poorer states earlier in the search with the intent of allowing us to overcome suboptimal or local optimum solutions. It should be noted that during the iterative state search, the best solution is always remembered, so even if accepting a poor solution does not eventually result in a better solution the algorithm still maintains the better solution. The standard of SA is the Kirkpatrick algorithm to include a temperature schedule for efficient searching generalized the Metropolis Monte Carlo integration algorithm[11]. A sufficiency proof was then shown to put on lower bound on that schedule as 1/ log (t), where t is an artificial time measure of the annealing schedule [14] However, independent credit usually goes to several other authors for independently developing the algorithm that is now recognized as SA[15,16]. and hence faster convergence. [17].

The default parameter-sampling distribution permits temperature schedules exponentially faster ($T=T_0.exp(-C_iK)^{1/D}$) than a Cauchy distribution ($T=T_0/k$), which is in turn exponentially faster than a Gaussian/Boltzmann distribution ($T=T_0/lnk$), according to each distribution's proof for (weakly) ergodic sampling of the parameter space [17, 18].

ASA algorithm permits an annealing schedule for "temperature" T decreasing exponentially in annealing-time K. Optimization using VFSR and ASA has been effectively applied in wide variety pf situations, including three-dimensional image compression [19], modeling of financial markets [20, 21], dairy forming [22], neural networks [23, 24], geophysical inversion [25], electroencephalography ([26, 27]), and combat simulation [28].

4. Proposed Method

In this work, we proposed a hybrid approach for global optimization of neural networks' topology and weight adjustment using SA for the former and standard backpropagation for the latter, until convergence of the training process. This method comprises a series of cycles where SA produces candidate network topologies by activating or deactivating neurons with all their associated connections and the standard backpropagation algorithm gradually adjusts, during a short-predefined number of epochs, the connections' weights. These cycles continue until the optimal topology for the architecture and the optimal weight adjustment for the connections is found. Figure-2 shows the block diagram of proposed system.

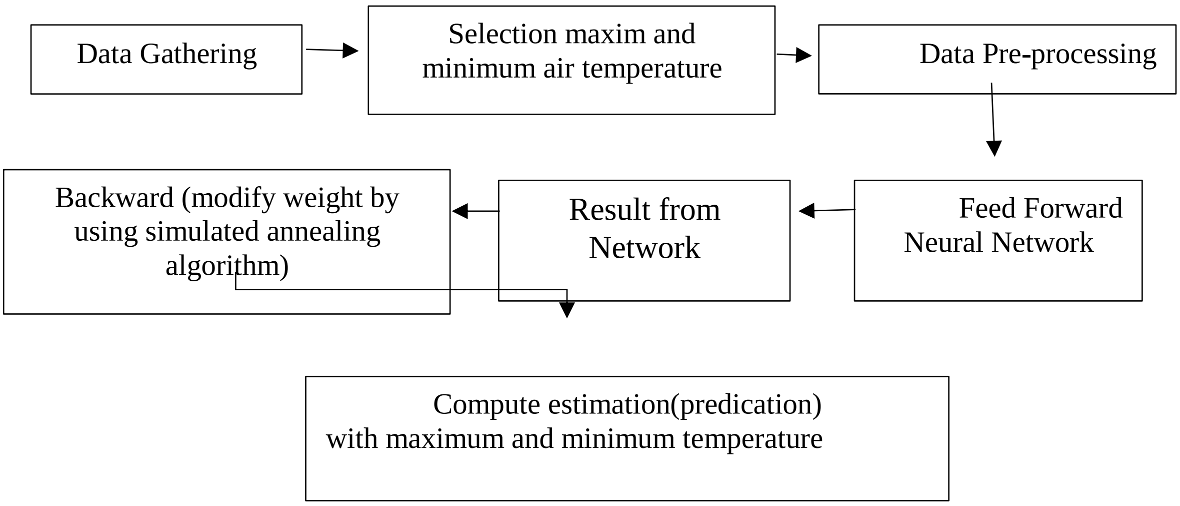

Figure 2- Proposed System Block Diagram

4.1 Study Area (Data Collection)

The area selected for this study is the city of Baghdad, the capital of Iraq. The Baghdad is a densely populated city which has desert climate. July is the hottest month of the year in Baghdad, where the average temp.is to 34.75°C (94.6°F), but sometimes rise up to reach 44°C (111.2°F). In spite of this, January is the coldest month of the year in Baghdad, where the average temperature is 9.65°C (49.4°F), but may get lower to 3.8°C (49.4°). The observed predictor dataset for study the maximum and minim air temperature of Baghdad has been obtained from Iraqi Meteorological Organization and Seismology (IMOS) and form underground weather site which stores data of major weather observing stations (www.undergroundweather.com). Dataset of month of July for period of 12 years (2005-2016) was taken for analysis out for forecasting one year (2016) to determine the efficiency of the derived models.

4.2 Hybrid BPNN and SA Algorithm

The proposed model algorithm of hybrid BPNN with SA are described in follows:

A. **Initialization:** set all biases and weights of the network to small random numbers. Note that the activation of node biases is fixed at 1.

B. Forward Pass:

1) The activation level of an input unit (Si) is determined by the instance present to:

$$Si = Xi, (0 \leq i \leq nin) \quad (3)$$

Where: Xi: input nodes (i) input the layer including bias node(X_0=1), nin: number of input nodes.

2) The activation level h_j of a hidden unit and the output(ok) of an output unit is determined by the equations given (Unipolar Sigmoid function):

$$f(net) = \frac{1}{1+e^{-\lambda net}} \quad \text{where } \lambda \geq 1 \tag{4}$$

$$h_j = \mathrm{f}(net_j) = f\left(\sum_{i=0}^{n_{in}} V_{ij} X_i\right) \tag{5}$$

Where:

h_j: output of the node(j) in the hidden layer including bias node(h_0=1), net_j=the activation output of the hidden node(j) in the hidden layer, n_{hid}=number of hidden neurons, V_{ij}= the weight of connection, which is from node(i)that is in input layer toward the node(j) in the hidden layer including bias node weight (v0j)

$$o_k = f(net_k) = f(\textstyle\sum_{j=0}^{n_{hid}} W_{jk} h_j) \text{ where } 1 \leq \text{k} \leq \text{nout} \tag{6}$$

Where:

net_k= the activation output of the output node(j)in the output layer nout= number of output nodes

o_k= output of the node(k)in the output layer

W_{jk}=the weight of connection from node(j)in the hidden layer to node(k) in the output layer including bias node weight (W_{0j})

3) Equation for compute error (Performance Measurement): The error (predication) of neural network is compute by the equation:

$$P = (dk - ok) \tag{7}$$

$$error = \sum (d_k - o_k)/d_k \tag{8}$$

Where: dk the desire output for k cell, o_k the actual output for k cell

C. Back word Pass (Weight modify by simulated annealing)

1- Initialize the temperature

2- Generate two individuals from the network depending on the network weights (w, v) of the pervious stage feed forward

3- Compute error

4- loop (start the highest iteration (which represents the temperature)

5- Generation of two new individuals (depending on generate random weights in addition to the network weights t

The activation level (hj) of a hidden unit and the output(ok) of an output unit is determined by the equations given (Unipolar Sigmoid function)

$$f(net) = \frac{1}{1+e^{-\lambda net}} \quad \text{where } \lambda \geq 1 \tag{9}$$

Where:

$$h_j = \mathrm{f}(net_j) = f\left(\sum_{i=0}^{n_{in}} V_{ij} X_i\right) \tag{10}$$

netj=the activation output of the hidden node(j)in the hidden layer

nhid=number of hidden nodes vij=the weight of connection from node(i)in the input layer to node(j) in the hidden layer including bias node weight(v0j)

$$o_k = f(net_k) = f(\textstyle\sum_{j=0}^{n_{hid}} W_{jk} h_j) \text{ where } 1 \leq \text{k} \leq \text{nout} \tag{11}$$

netk= the activation output of the output node(j)in the output layer.

nout= number of output nodes

Ok= output of the node(k)in the output layer

6- Equation for compute error (Performance Measurement)

The error (predication) of neural network is compute by the equation (7) and (8)

7- Calculate the difference in errors (delta)

Δ = E' – E

8- If ($\boldsymbol{\Delta} \leq 0$) $\vee$ (random () $e^{-\Delta T}$1111111

9- If the condition meets the new individuals replace e for the old with error
10- reduce temperature (the most repeated temperature value)
11- end

5. Results and Discussions

Forecast precision for the proposed methods for month of July is arranged in table 1 (a) and (b), where the forecast precision of the proposed model is compared with real readings of Baghdad temperature in 2016.

Table 1-Predication results of hybrid model for month of July, (a) Minimum temperature (b) Maximum temperature.

(a)

Days	Desir	Predication
1	24	23.74811064
2	24	23.73231388
3	26	25.62602442
4	27	26.69454052
5	26	25.40294015
6	26	25.64774025
7	25	24.75527308
8	26	25.41391064
9	25	24.36750317
10	24	23.64916034
11	24	23.75942449
12	24	23.72187773
13	23	22.65661316
14	24	23.55390297
15	23	22.71268227
16	22	21.55292663
17	26	25.50395089
18	23	22.72675814
19	23	22.76462504
20	26	25.68885132
21	25	24.70843452
22	25	24.6626807
23	25	24.61309807
24	24	23.77805728
25	26	25.71314281
26	26	25.4441613
27	25	24.79621576
28	24	23.40754362
29	25	24.49249914
30	25	24.60851265
31	24	23.55527841

(b)

Days	Desir	Predication
1	36	35.485543
2	37	36.59131613
3	34	33.72171822
4	34	33.52718969
5	36	35.71333278
6	34	33.62872587
7	36	35.65807547
8	36	35.58355363
9	37	36.36850651
10	34	33.48879768
11	33	32.37164319
12	34	33.72961894
13	33	32.6075333
14	34	33.7991902
15	34	33.49129191
16	34	33.56572725
17	33	32.41954753
18	37	36.38943928
19	37	36.40512791
20	37	36.649329
21	36	35.82526383
22	33	32.57031783
23	33	32.72824832
24	33	32.71914689
25	35	34.70781503
26	32	31.77280171
27	34	33.46906021
28	34	33.49688991
29	34	33.60786094
30	35	34.6808597
31	36	35.43782259

As appeared in Table-1, the proposed hybrid model is noticed to possess greater forecast precision for seasons with low error (less than 0.05 degree).

6. Conclusion

The paper studies the performance of hybrid method of BP and in the area of forecasting daily maximum and minimum air temperature. In this paper we have proposed a model that based on BP and SA method, where we exposed to achieve good forecasting results when combined it with regular BPNN. For experimental level, we have trained network with large scale of data, that is the temperature records for month of July for six years (2010-2015) then we predicate the temperature of July for year of 2016, then compered the predicated temperature with actual temperature from IMOS records in 2016. The results show that this hybrid method have got an excellent forecasting temperature reach near actual recorders. This study provides it can achieve a good forecasting results by hybrid SA and BP.

7. Reference

1. Huawang S. **2010**. Evolving Artificial Neural Networks Using Simulated Annealing-based Hybrid Genetic Algorithms. *Journal of Software*, **5**(4): 353-360.
2. Lin L. and Chen F. **2012**. The Simulated Annealing Algorithm Implemented by the MATLAB, Journal of Software. *International Journal of Computer Science Issues* (IJCSI), **9**(6, No.2): 357-360.
3. Maung K. S., Bjørn-Erik M., Tore U. Ø., Wita W., W. Gregory W., Joze Z. and Stanislaw W. **2012**. *Contemporary Trends in Systems Development*. Springer Science & Business Media, ISBN: 9781461513414.
4. Bernal J. and Torres-Jimenez J. **2015**. SAGRAD: A Program for Neural Network Training with Simulated Annealing and the Conjugate Gradient Method. *Journal of Research of the National Institute of Standards and Technology*, **120**: 113-128.
5. Jyosthna D., Syam P. R., Vagdhan K., Musala R. and Raja N. **2012**. ANN Approach for Weather Prediction using Back Propagation. *International Journal of Engineering Trends and Technology*, **3**(1): 19-23.
6. Shereef I. K. and Baboo S. S. **2011**. A New Weather Forecasting Technique using Back Propagation Neural Network with Modified Levenberg-Marquardt Algorithm for Learning. *International Journal of Computer Science Issues* (IJCSI), **8**(6, No 2): 153-160.
7. Irena K. **2003**. Neural Networks: Introduction to Neural Networks and Machine Learning, COMP4302/5322 Neural Networks, w1, s2, School of Information Technologies, University of Sydney.
8. Kirkpatrick S., Gellat Jr. C.D. and Vecchi M. P. **1983**. Optimization by Simulated Annealing, *Science*, **220**(4598): 671-680.
9. Rumelhart D. E., G.E. and Williams R. J. **1986**. *Learning internal representations by error propagation*. Parallel Distributed Processing, Cambridge, MIT Press, **1**: 318-362.
10. Alexander G. N. and Sheldon H. J. **2010**. *Simulated Annealing, Handbook of Metaheuristics*. Springer Science+Business Media, LLC.
11. Metropolis N., Rosenbluth A. W., Rosenbluth M. N., Teller A. H. and Teller E. **1953.** Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, **21**(6): 1087-1092.
12. Darrall, H., Sheldon H. J. and Alan W. J. **2003**. *Handbook of Metaheuristics, The Theory and Practice of Simulated Annealing*. Springer US, 287-319.
13. Greening, D. R. **1995**. *Simulated Annealing with Errors*. University of California, Los Angeles.
14. Geman, S. and Geman, D. **1984.** Stochastic relaxation, Gibbs distribution and the Bayesian restoration in images. *IEEE Trans. Patt. Anal. Mac. Int*. **6**(6): 721-741.
15. CERNY, V. **1982.** 'A thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. Report, Bratislava, Czechoslovakia, Comenius University.
16. PINCUS, M. **1970.** A Monte Carlo method for the approximate solution of certain types of constrained optimization problems. *Oper. Res*. **18**:1225.
17. Boese K. D. and Kahng A. B. **1993**. Simulated Annealing of Neural Networks: the "Cooling" Strategy Reconsidered. UCLA Computer Science Dept., Los Angeles, CA 90024-1596.
18. Atiya A. F., Parlos A. G. and Ingber L. **2003**. A Reinforcement Learning Method Based on Adaptive Simulated Annealing'. Dept Mechanical Engineering, Texas A&M University, College Station
19. Forman, M. C. **2000**. Compression of Integral Three-Dimensional Television Pictures. PhD thesis, De Montfort University, Leicester, United Kingdom, 2000.
20. Ingber L. and Wilson J. K. **2000**. Statistical mechanics of financial markets: Exponential modifications to Black-Scholes. *Mathematical Computer Modelling*, **31**(8/9):167–192.
21. Sakata S. and White H. **1998**. High breakdown point conditional dispersion estimation with application to S&P 500 daily returns volatility'. Econometrica, **66**: 529–567.
22. Mayer, D. G., Belward, J. A. and Burrage, K. **1996**. Use of advanced techniques to optimize a multi-dimensional dairy model. *Agricultural Systems*, **50**: 239–253.
23. Cozzio-Bueler, R.A. **1995**. The design of neural networks using a priori knowledge'. PhD thesis, Swiss Federal Institute of Technology, Zurich, Switzerland.

24. Cohen B. **1994**. Training synaptic delays in a recurrent neural network. MSc. thesis, Tel-Aviv University, Tel-Aviv, Israel.

25. Sen M. K. and Stoffa P. L. **1995**. Global Optimization Methods in Geophysical Inversion'. Elsevier, The Netherlands, ISBN: 0-444-81767-0.

26. Kirkpatrick S., Gelatt C.D. and Vecchi M.P. **1983.** Optimization by simulated annealing', *Science*, **220**: 671-680.

27. Ingber L. **1997**. Statistical mechanics of neocortical interactions: Canonical momenta indicators of electroencephalography. *Physical Review E*, **55**(4): 4578–4593.

28. Bowman M. and Ingber L. **1997**. Canonical momenta of nonlinear combat. In Proceedings of the Simulation Multi-Conference, 6-10 April 1997, Atlanta, GA, San Diego, CA.