M

MREM, Discrete Recurrent Network for Optimization

Enrique Mérida-Casermeiro
University of Málaga, Spain

Domingo López-Rodríguez
University of Málaga, Spain

Juan M. Ortiz-de-Lazcano-Lobato
University of Málaga, Spain

INTRODUCTION

Since McCulloch and Pitts' seminal work (McCulloch & Pitts, 1943), several models of discrete neural networks have been proposed, many of them presenting the ability of assigning a discrete value (other than unipolar or bipolar) to the output of a single neuron. These models have focused on a wide variety of applications. One of the most important models was developed by J. Hopfield in (Hopfield, 1982), which has been successfully applied in fields such as pattern and image recognition and reconstruction (Sun et al., 1995), design of analogdigital circuits (Tank & Hopfield, 1986), and, above all, in combinatorial optimization (Hopfield & Tank, 1985) (Takefuji, 1992) (Takefuji & Wang, 1996), among others.

The purpose of this work is to review some applications of multivalued neural models to combinatorial optimization problems, focusing specifically on the neural model MREM, since it includes many of the multivalued models in the specialized literature.

BACKGROUND

In Hopfield and Tank's pioneering work (Hopfield & Tank, 1985), neural networks were applied for the first time to solve combinatorial optimization problems, concretely the well-known travelling salesman problem. They developed two types of networks, discrete and continuous, although the latter has been mostly chosen to solve optimization problems, adducing that it helps to escape more easily from local optima. Since then, the search for better neural algorithms, to face the diverse problems of combinatorial optimization (many of them belonging to the class of NPcomplete problems), has been the objective of researchers in this field.

This method of optimization consists of minimizing an energy function, whose parameters and constraints are obtained by means of identification with the objective function of the optimization problem. In this case, the energy function has the form:

$$E(\boldsymbol{S}) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{i,j} s_i s_j + \sum_{i=1}^{N} \theta_i s_i$$

where N is the number of neurons of the network, $w_{i,j}$ is the synaptic weight between neurons j and i, and θ_i is the threshold or bias of the neuron i.

In the discrete version of Hopfield's model, component s_i of the state vector $\boldsymbol{S} = (s_1,\ldots,s_N)$ can take values in $\mathcal{M} = \{-1,1\}$ (constituting the bipolar model) or in $\mathcal{M} = \{0,1\}$ (unipolar model). In the continuous version, $\mathcal{M} = [-1,1]$ or $\mathcal{M} = [0,1]$. This continuous version, although it has been traditionally the most used for optimization problems, presents certain inconveniences:

- Certain special mechanisms, maybe in form of constraints, should be contributed in order to get that, in the final state of the network, all the components of state vector $\boldsymbol{S}$ belong to $\{-1, 1\}$ or $\{0,1\}$.
- The traditional dynamics used in this model, implemented in a digital computer, does not guarantee the decrease of the energy function in every iteration, so it is not ensured that the final state is a minimum of the energy function (Galán-Marín, 2000).

However, the biggest problem of this model (the discrete as well as the continuous one) is the possibility to converge to a non feasible state, or to a local (not global) minimum. Wilson and Pawley (1988) demonstrated, through massive simulations, that, for the travelling salesman problem of 10 cities, only 8% of the solutions were feasible, and most not good. Moreover, this proportion got worse when problem size was increased.

After this, many works were focused on improving Hopfield's network:

- By modifying the energy function (Xu & Tsai, 1991).
- By adjusting the numerous parameters present in the network, as in (Lai & Coghill, 1988).
- By using stochastic techniques in the dynamics of the network (Kirkpatrick et al., 1983) (Aarts & Korst, 1988).

Particularly, researchers tried to improve the efficiency of Hopfield's network for the travelling salesman problem, achieving acceptable results, but inferior to Operations Research techniques (Takahashi, 1997). The reason for these disappointing results is that the linear formulation used by these techniques is a great advantage in comparison with neural networks, which unavoidably use a quadratic energy function, impeding the use of subpaths deletion techniques (Smith, 1996), and provoking the appearance of a bigger number of local minima.

Another research line was devoted to the improvement of Hopfieldtype recurrent networks, and their application to diverse problems of optimization, in which some results proved to be better than those obtained by traditional Operations Research techniques (Smith & Krishnamoorthy, 1998). Takefuji's work (Takefuji, 1992) (Lee et al., 1992)(Takefuji & Wang, 1996), with a great number of publications in international media, must be highlighted. Their results have been overcome by the OCHOM model (GalánMarín & MuñozPérez, 2001).

MULTIVALUED DISCRETE RECURRENT MODEL. APPLICATION TO COMBINATORIAL OPTIMIZATION PROBLEMS

A new generalization of Hopfield's model arises in the works (MéridaCasermeiro, 2000) (MéridaCasermeiro et al., 2001), where the MREM (Multivalued REcurrent Model) model is presented.

The Neural MREM Model

This model presents two essential features that make it very versatile and that increase its applicability:

- The output of each neuron, s_i, is a value of the set $\mathcal{M} = \{m_1, m_2, \ldots, m_L\}$, which is not necessarily numeric.
- The concept of similarity function *f* between neuron outputs is introduced. *f*(*x*,*y*) represents the similarity between neuron states *x* and *y*.

This way, the energy function of this model is as follows:

$$E(\boldsymbol{S}) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{i,j} f(s_i, s_j) + \sum_{i=1}^{N} \theta_i(s_i)$$

where $\theta_i : \mathcal{M} \to \mathbb{R}$ is a generalization of the thresholds of each neuron.

The features mentioned above make that in this model certain optimization problems (as the travelling salesman problem), have a better representation than in the unipolar or bipolar Hopfield's models, and their successors.

It is clear that MREM includes Hopfield's models (with outputs in $\mathcal{M} = \{-1,1\}$ or in $\mathcal{M} = \{0,1\}$) if we consider the similarity function given by the product *f*(*a*,*b*) = *ab*. Other multivalued models, like MAREN or SOAR (Erdem & Ozturk, 1996) (Ozturk & Abut, 1997), are also generalized by MREM.

The dynamics for this network is chosen according to the problem to be tackled.

Application to Several Combinatorial Optimization Problems

This multivalued model has been successfully applied to diverse optimization problems, outperforming the best-established algorithms. Several of these applications can be found at (MéridaCasermeiro et al., 2003) (MéridaCasermeiro & LópezRodríguez, 2005) (López-Rodríguez et al., 2006).

These problems are typical representatives of the NPcomplete complexity class, indicating their degree of difficulty in resolution.

The Travelling Salesman Problem

Traveling Salesman Problem (TSP) is one of the most wellknown and studied combinatorial optimization problems due to its wide range of reallife applications and intrinsic complexity.

Reallife applications cover aspects such as automatic routing for robots and hole location in printed circuits design (Reinelt, 1994), as well as gas turbine checking, machine task scheduling or crystallographic analysis (Bland & Shallcross, 1987), among others.

This problem can be stated as follows: given N cities $X_1,...,X_N$ and distances $d_{i,j}$ between each pair of cities X_i and X_j, the objective is to find the shortest closed tour visiting each city once.

In order to get the TSP solved by this neural model, two identifications must be done:

- A network state must be identified to a solution to the TSP: Since a solution to the N cities TSP can be represented as a permutation in the set of numbers $\{1,...,N\}$, the net will be formed by N neurons, taking value in the set $\mathcal{M} = \{1,\ldots,N\}$, such that state vector $\boldsymbol{S} = (s_1,...,s_N)$ represents a permutation of $\{1,...,N\}$. With this representation, $s_i = k$ means that kth city will be visited in the ith place.
- The energy function must be identified to the total distance of a tour: If we let $f(x,y) = -2d_{x,y}$ and

$$w_{i,j} = \begin{cases} 1, & (j = i+1) \vee ((i = N) \wedge (j = 1)) \\ 0, & \text{otherwise} \end{cases}$$

the energy function obtained is

$$E(\boldsymbol{S}) = \sum_{i=1}^{N-1} d_{s_i,s_{i+1}} + d_{s_N,s_1},$$

the total distance of the tour represented by state vector $\boldsymbol{S}$.

Computational dynamics is based on starting with a random feasible initial state vector and updating neuron outputs to keep the current state vector inside the feasible states set. To this end, at each iteration, a 2opt update will be made on current state vector, that is, every pair of neurons, p,q with $p > q + 1$, is studied and checked in parallel whether there exists a cross between segments (s_p, s_{p+1}) and (s_q, s_{q+1}). In this case, the next relation holds:

$$d_{s_p,s_{p+1}} + d_{s_q,s_{q+1}} > d_{s_p,s_q} + d_{s_{p+1},s_{q+1}}$$

Then, the trajectory between cities s_{p+1} and s_q is inverted, that is, if $\boldsymbol{S}$ is the current state, the new state vector $\boldsymbol{S'}$ will be defined by:

$$s_i^{'} = \begin{cases} s_{q+p+1-i}, & p+1 \le i \le q \\ s_i, & \text{otherwise} \end{cases}$$

As an additional technique for improvement, it has also been considered 3opt updates: the tour is decomposed into three consecutive arcs, A, B and C, which are then recombined in all possible ways: {ABC,AC B,AB'C,ABC',AB'C',AC'B, ACB', AC'B'}, where A',B',C' are the reversed arcs corresponding to A, B, and C, respectively. Note that {ABC,AB'C,ABC',AC'B'} are 2opt updates, so there is no need to check them again.

The next state of the net will be the combination that decreases most the energy function. Further details in (MéridaCasermeiro et al., 2003).

In (MéridaCasermeiro et al., 2003), some experimental results are provided, for problems from the TSPLIB repository (see Table 1). This model is compared against KNIES (Aras et al., 1999), a model based on Kohonen's self organizing map. MREM proved to outperform KNIES, obtaining in many cases almost optimal solutions.

Figure 1. Best solution found by MREM (left, error=1.3%) and optimal solution (right)

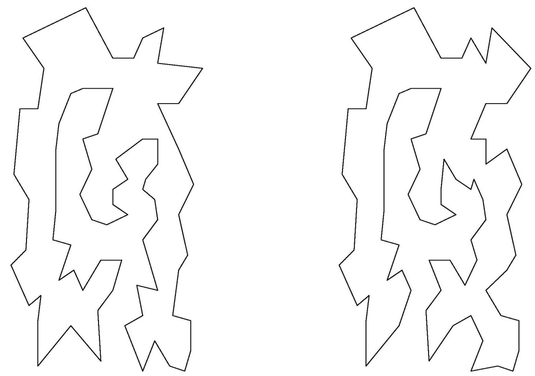


Table 1. Results of KNIES and MREM for the TSP for some instances from TSPLIB

Instance	Optimum	KNIES	MREM		
		Best (%)	Best (%)	Av. (%)	t (sec)
eil51	426	2.86	0.23	2.43	3.12
st70	675	1.51	0.00	1.89	9.01
eil76	538	4.98	1.30	3.43	10.80
rd100	7910	2.09	0.00	3.02	61.70
eil101	629	4.66	1.43	3.51	27.76
lin105	14379	1.29	0.00	1.71	28.83
pr107	44303	0.42	0.15	0.82	49.79
pr124	59030	0.08	0.00	1.23	59.51
bier127	118282	2.76	0.42	2.06	66.29
kroA200	28568	5.71	3.49	6.70	318.44

The Graph Partition Problem

Let $\mathcal{G} = (\mathcal{V}, \varepsilon)$ be an undirected graph without self-connections. $\mathcal{V} = \{v_i\}$ is the set of vertices and ε is the set of n_e edges. For each edge $(v_i, v_j) \in \varepsilon$ there is a weight $c_{i,j} \in \mathbb{R}^+$. All weights can be expressed by a symmetric real matrix C, with $c_{i,j} = 0$ when it does not exist the arc (v_i, v_j).

MaxCut Problem: to find a partition of $\mathcal{V}$ into K disjoint sets A_i such that the sum of the weights of the edges from ε, that have their endpoints in different elements of the partition, is maximum. Therefore, the function to maximize is

$$\sum_{i|v_i \in A_m} \sum_{\substack{j|v_j \in A_n \\ m>n}} c_{i,j}$$

To solve the MaxCut problem with MREM, we need N neurons, one per node in $\mathcal{V}$. The output of neuron i, $s_i \in \mathcal{M} = \{1, 2, \ldots, K\}$, will denote that ith node is assigned to A_{si}.

Since it is equivalent to maximize the cost of edges cut by the partition and to minimize the cost of edges with endpoints in the same set of the partition, the objective function can be modelled as an energy function by taking $w_{i,j} = -2c_{i,j}$ and $f(x,y) = \delta_{x,y}$ (that is, $f(x,y) = 1$ if, and only if, $x = y$, otherwise it is 0), considering $\theta_i = 0$.

The dynamics used in (MéridaCasermeiro & López-Rodríguez, 2005) was named best2.

best2 consists in getting the greatest decrease of the energy function by changing the state of only two neurons at each time. If neurons p and q are to be updated, energy increments $\Delta E(i, j)$ when $s_p = i$ and $s_q = j$, for $i, j \in \{1,...,K\}$, are computed. Then, the state of minimum increase is chosen as the new network state.

By using this dynamics, in (MéridaCasermeiro & LópezRodríguez, 2005), the MREM model is compared against some other networks, like OCHOM (Galán-Marín & MuñozPérez, 2001), obtaining the best results in authors' experiments (see Table 2).

Table 2. Results for MaxCut comparing MREM and OCHOM

N	dens	MREM			OCHOM		
		Best	Av.	t	Best	Av.	t
50	0,05	276,8	256,28	0,05	276,8	242,15	0,0023
	0,25	1013,2	970,84	0,06	999,6	926,26	0,0026
	0,5	1778,8	1724,08	0,06	1778,8	1694,44	0,0033
	0,75	2663,6	2475,48	0,05	2646	2432,47	0,0036
	0,9	2941,8	2876,18	0,06	2940,4	2865,83	0,0031
100	0,05	990,2	917,72	0,15	958,8	867,64	0,0064
	0,25	3719,2	3620,9	0,14	3725,5	3571,24	0,0086
	0,5	6711,6	6637,08	0,13	6695,8	6585,54	0,0126
	0,75	9816,2	9524,1	0,14	9816,2	9444,33	0,0118
	0,9	11348,8	11215,06	0,14	11391,3	11148,4	0,0109
150	0,05	2009,8	1933,6	0,26	1929,6	1837,43	0,0147
	0,25	7990	7807,16	0,26	7940,2	7690,35	0,0258
	0,5	14701,4	14531,06	0,24	14658,4	14489,5	0,0209
	0,75	21126,2	20899,94	0,22	21124	20907,6	0,0252
	0,9	24926	24589,62	0,22	24859,7	24533,1	0,0256
200	0,05	3411,4	3321,84	0,38	3409,5	3316,28	0,0276
	0,25	13741	13533,9	0,35	13617,9	13439,7	0,0468
	0,5	25750,8	25500,18	0,34	25770,8	25526,8	0,0451
	0,75	37038,6	36789,2	0,32	36932	36683,4	0,0486
	0,9	43584,8	43296,26	0,33	43420,6	43104,6	0,0462

The 2Pages Graph Layout Problem

In the last years, several graph representation problems have been studied in the literature. Most of them are related to the linear graph layout problem, in which the vertices of a graph are placed along a horizontal ``node line'', or ``spine'' (dividing the plane into two halfplanes or ``pages'') and then edges are added to this representation as specified by the adjacency matrix. The objective of this problem is to minimize the number of crossings produced by such a layout.

Some examples of problems associated to this linear graph layout problem (or 2 pages crossing number problem, 2PCNP) are the bandwidth problem (Chinn et al., 1982), the book thickness problem (Kainen, 1990), the pagenumber problem (Malitz, 1994), the boundary VLSI layout problem (Ullman, 1984) and the singlerow routing problem (Raghavan & Sahni, 1983), or printed circuit board layout (Sinden, 1966) and automated graph drawing (Tamassia et al., 1988).

In (LópezRodríguez et al., 2007), a neural model, derived from MREM, is designed to solve this problem. One of the differences of this model with the algorithms developed in literature is that there is no need of assigning a good ordering of the vertices at a preprocessing step. The model, as well as the relative position of the arcs, computes this optimal node order.

To solve the 2PCNP problem, authors have considered two MREM neural models:

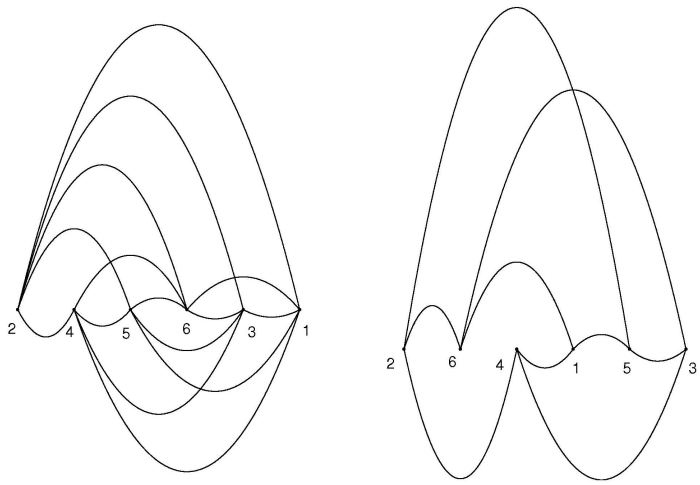
- The first network will be formed by N neurons, being N the number of nodes in the graph. Neurons output (the state vector) indicate the node ordering in the line. Thus, $s_i = k$ will be interpreted as the kth node being placed in the ith position in the node line. Hence, the output of each neuron can take value in the set $\mathcal{M}_1 = \{1, 2, \ldots, N\}$.
- The second network will be formed by as many neurons as edges in the graph, M. The output of each neuron will belong to the set $\mathcal{M}_2 = \{-1, 1\}$. For the arc (v_i, v_j), $S_{(v_i, v_j)} = -1$ will indicate that the edge will be drawn in the lower halfplane, and $S_{(v_i, v_j)} = +1$, in the upper one.

Initially, the state of the net of vertices is randomly selected as a permutation of $\{1,2,...,N\}$. At any time,

Table 3. Comparison between MREM and the heuristics mentioned in Cimikowski's work

Graph	N	M	MREM	CN	e-len	1-page	greedy
K_6	6	21	3	3	3	4	5
K_7	7	28	9	9	11	9	13
K_8	8	36	18	18	18	30	27
K_9	9	45	36	36	42	50	50
K_{10}	10	55	60	60	80	92	80
$C_{20}(1, 2)$	20	40	0	2	0	0	0
$C_{20}(1, 2, 3)$	20	60	19	24	36	48	40
$C_{20}(1, 2, 3, 4)$	20	80	74	74	90	118	108
$C_{22}(1, 2, 3)$	22	66	22	26	40	54	44
$C_{22}(1, 3, 5, 7)$	22	88	198	200	306	294	286
$C_{24}(1, 3)$	24	48	11	14	22	16	22
$C_{26}(1, 3)$	26	52	11	16	24	16	24
$C_{28}(1, 3, 5)$	28	84	80	86	138	138	130
$C_{30}(1, 3, 5)$	30	90	92	96	148	150	140

Figure 2. Optimal layouts for graphs K_6 (left) and $K_{3,3}$ (right)



the net is looking for a better solution than the current one, in terms of minimizing the energy function. This is achieved by permuting the output of two neurons (node positions) and changing the location of an edge (from the upper halfplane to the lower one, and viceversa).

In (LópezRodríguez et al., 2007), this new model is compared against some heuristics (Cimikowski, 2002) specially designed for this problem. MREM obtained the best solutions in the experiments, improving the best known solution in some cases (Table 3).

FUTURE TRENDS

Recurrent neural networks can be used to solve many optimization problems. Researchers and practitioners can benefit from the application of the neural model MREM to diverse optimization problems.

Other problems where these models can be applied cover aspects such as data classification, image compression by vector quantization, etc. It must be noted that many graph-based problems can be easily formulated in terms of minimizing the energy function of this model: degreeconstrained minimum spanning tree, maximum clique, etc.

CONCLUSION

The first works in optimization by neural networks were inspired in Hopfield's models. These models did not obtain good results when compared to the wellknown Operations Research techniques.

Many researchers focused on developing new neural models to improve the performance of Hopfieldtype networks in this kind of tasks.

The problem of these binary models is that all the information given by the problem has to be specified by means of only two values ($\{0,1\}$ or $\{-1,1\}$), so some information is lost.

Multivalued neural models are designed to represent the information of the problem by means of more than two values, achieving a better representation of the problem.

With this improvement, computational dynamics of multivalued models can be easily designed to solve a given optimization problem. These advantages make this kind of networks a very powerful ally in tackling combinatorial problems.

The MREM model is a multivalued model that generalizes many other models, so it can be easily used to solve optimization problems, as shown in the text.

Some applications of the model are wellknown NPcomplete optimization problems, like the Traveling Salesman Problem, the Graph Partition Problem, and the 2 Pages Crossing Number Problem. As shown in the references, this model is able to outperform the bestalgorithmuptodate in each of the mentioned problems.

REFERENCES

Aarts, E. & Korst, J. (1988). Simulated Annealing with Boltzman Machines. Wiley.

Aras, N., Oomen B.J., & Altinel, I.K. (1999). The Kohonen Network Incorporating Explicit Statistics and its Application to the Travelling Salesman Problem. Neural Networks, 12:1273-1284.

Bland, R. & Shallcross, D. F. (1987). Large Traveling Salesman Problem Arising from Experiments in Xray Crystallography: a Preliminary Report on Computation. Technical Report No. 730, School of OR/IE, Cornell University, New York.

Chinn, P. Z., Chvátalová, L., Dewdney, A. K., & Gibbs, N. E. (1982). The bandwidth problem for graphs and matrices-a survey. J. Graph Theory, 6, 223-253.

Cimikowski, R. (2002). Algorithms for the fixed linear crossing number problem. Discrete Applied Mathematics, 122, 93 – 115.

Erdem, M. H. & Ozturk, Y. (1996). A new family of multivalued networks. Neural Networks, 9(6), 979-989.

GalánMarín, G. & MuñozPérez, J. (2001). Design and analysis of maximum hopfield networks. IEEE Trans. on Neural Networks, 12(2), 329-339.

GalánMarín, G. (2000). Redes Neuronales Recurrentes para Optimización Combinatoria. PhD thesis, Universidad de Málaga.

Hopfield, J. (1982). Neural networks and physical systems with emergent collective computational abilities, 79, 2254-2558.

Hopfield, J. & Tank, D. (1985). Neural computation of decisions in optimization problems. Biological Cybernetics, 52, 141-152.

Kainen, P. C. (1990). The book thickness of a graph, ii. Congr. Numer., 71, 127-132.

Kirkpatrick, S., Gellat, C. D., & Vecchi, M. (1983). Optimization by simulated annealing. Science, 220, 671-680.

Lai, W. K. & Coghill, G. G. (1988). Genetic breeding of control parameters for the hopfieldtank neural network. In Proc. Int. Conf. Neural Networks (pp. 618-623).

Lee, K. C., Funabiki, N., & Takefuji, Y. (1992). A parallel improvement algorithm for the bipartite subgraph problem. IEEE Transactions on Neural Networks, 3(1), 139-145.

LópezRodríguez, D., MéridaCasermeiro, E., Ortiz de LazcanoLobato, J. M., & GalánMarín, G. (2007). Two pages graph layout via recurrent multivalued neural networks. Lecture Notes in Computer Science, 4507, 192-199.

LópezRodríguez, D., MéridaCasermeiro, E., Ortiz de LazcanoLobato, J. M., & LópezRubio, E. (2006). Image compression by vector quantization with recurrent discrete networks. Lecture Notes in Computer Science, 4132, 595-605.

Malitz, S. M. (1994). On the page number of graphs. J. Algorithms, 17(1), 71-84.

McCulloch, W. & Pitts, W. (1943). A logical calculus of the ideas inmanent in nervous activity. Bulletin Mathematical Biophysic, 5, 115-133.

MéridaCasermeiro, E. (2000). Red Neuronal recurrente multivaluada para el reconocimiento de patrones y la optimización combinatoria. PhD thesis, Universidad de Málaga.

MéridaCasermeiro, E., GalánMarín, G., & Muñoz Pérez, J. (2001). An efficient multivalued Hopfield network for the travelling salesman problem. Neural Processing Letters, 14, 203-216.

MéridaCasermeiro, E. & López Rodríguez, D. (2005). Graph partitioning via recurrent multivalued neural networks. Lecture Notes in Computer Science, 3512, 1149-1156.

MéridaCasermeiro, E., Muñoz Pérez, J., & Domínguez-Merino, E. (2003). An nparallel multivalued network: Applications to the travelling salesman problem.

Computational Methods in Neural Modelling, Lecture Notes in Computer Science, 2686, 406-413.

Ozturk, Y. & Abut, H. (1997). System of associative relationships (SOAR). Proceedings of ASILOMAR.

Raghavan, R. & Sahni, S. (1983). Single row routing. IEEE Trans. Comput, C32(3), 209-220.

Reinelt, G. (1994). The Travelling Salesman. Computational Solutions for TSP Applications. Springer.

Sinden, F. W. (1966). Topology of thin films circuit. Bell Syst. Tech. Jour., XLV, 1639-1666.

Smith, K. (1996). An argument for abandoning the traveling salesman problem as a neural network benchmark. IEEE Transactions on Neural Networks, 7(6), 1542-1544.

Smith, K., P. M. & Krishnamoorthy, M. (1998). Neural techniques for combinatorial optimization with aplications. IEEE Transactions on Neural Networks, 9(6), 1301-1318.

Sun, Y., Li, J. G., & Yu, S. Y. (1995). Improvement on performance of modified Hopfield network for image restoration. IEEE Transactions on Image Processing, 5, 688-692.

Takahashi, Y. (1997). Mathematical improvement of the Hopfield model for TSP feasible solutions by synapse dynamical systems. Neurocomputing, 15, 15-43.

Takefuji, Y. (1992). Neural Network Parallel Computing. Kluwer Academic.

Takefuji, Y. & Wang, J. (1996). Neural Computing for Optimization and Combinatorics, volume 3. World Scientific.

Tamassia, R., Di Battista, G., & Batini, C. (1988). Automatic graph drawing and readability of diagrams. IEEE Trans. Syst. Man. Cybern., SMC18, 61-79.

Tank, D. & Hopfield, J. (1986). Simple neural optimization networks: An a/d converter, signal decision circuit and linear programming circuit. IEEE Transactions on Circuits and Systems, 33(5), 533-541.

Ullman, J. D. (1984). Computational Aspects of VLSI. Computer Science Press.

Wilson, V. & Pawley, G. (1988). On the stability of the TSP problem algorithm of Hopfield and Tank. Biological Cybernetics, 58, 63-70.

Xu, X. & Tsai, W. T. (1991). Effective neural algorithms for the traveling salesman problem. Neural Networks, 4, 193-205.

KEY TERMS

2 Pages Graph Layout Problem: Problem of finding an ordering of the nodes of a graph on a straight line, and assigning, to each edge, a location in any of the two halfplanes induced by that line, such that the number of crossings between edges is minimum.

Artificial Neural Network: Structure for distributed and parallel processing of information, formed by a series of units (which may possess a local memory and make local information processing operations), interconnected via one-way communication channels, called connections.

Computational Dynamics: Updating scheme of the neuron outputs in a neural model.

Energy Function: Objective function of the optimization problem solved by a neural model.

MaxCut Problem: Problem of finding a partition of the set of nodes of a weighted graph, such that the sum of the costs corresponding to edges, with end-points in different sets of the partition, is maximum.

Multivalued Discrete Neural Model: A model of neural networks in which neuron outputs may take value in the set $\mathcal{M} = \{m_1, \ldots, m_L\}$, instead of $\mathcal{M} = \{-1,1\}$ or $\mathcal{M} = \{0,1\}$.

Travelling Salesman Problem: Problem of finding the shortest closed tour that visits a series of N cities. Each city must be visited exactly one time.