Perspective of a neural optical solution of the Traveling Salesman optimization Problem

Koffi C. Yao (1) , P. Chavel (2) , P. Meyrueis (1)

(1) - Laboratoire des Systèmes Photoniques, ENSPS/ULP,
7 rue de l'Université, F-67000 STRASBOURG

(2) - Institut d'Optique (laboratoire associé au CNRS), Centre Universitaire d'Orsay
Bât. 503, BP 147, F-91403 ORSAY CEDEX

ABSTRACT : In this paper, we consider the use of optical neural networks to approach solutions of optimization problems. A computer neural simulation ot the Traveling Salesman optimization Problem is presented. An optical system designed to achieve this task, using a synaptic matrix composed of NxN computer-generated holograms is described. We discuss how these systems could be extended for use in future machines.

SUBJECT TERMS : neural networks, optical computing, optimization problems, computer-generated holograms, pattern recognition.

CONTENTS

1 - INTRODUCTION

The basic properties of highly interconnected neural networks such as collective signal processing and individual fault tolerance are clearly well-suited for use in the associative mode [6-9]. But neural networks also show promise in domains like solving optimization problems [3]. The present work is related to the optical neural network implementation of the Traveling Salesman Problem (T.S.P.) solution. The TSP is known to be an optimization difficult problem.

Based on the algorithm introduced by Hopfield and Tank [3], we have performed a computer neural simulation of the TSP. Some modifications on the dynamical equation of neurons have been done to make it adaptable to our optical implementation. N^2 neurons permit to depict the N cities and their position in a tour list. We propose to implement the space varying optical interconnects required by a set of NxN Computer-Generated Holograms (CGH), one for each neuron, with the point-spread function of each hologram being the set of synaptic coefficients of the corresponding neuron.

We present the operation of the system on some initial states, suggest solutions to achieve the nonlinear feedback using spatial light modulators and to tackle the negative quantities and then, show description of the complete process.

2 - HOPFIELD NEURAL NETWORK MODEL : A SUMMARY

A formal neural network could be designed as an assembly of highly-interconnected simple elements which can collectively provide solution to difficult problems by a judicious choice of interconnexion weights [1-3]

A neuron i is considered as a computing entity having an input potential Ui depending on all the other neurons of the system and an ouput state Vi representing the response to this potential. A crude simplification of biological neurons behaviour can be used to represent the mutual interactions between formal neurons by the function :

$$Vi = g(Ui) = 1/2 \left[1 + \tanh(\alpha\, Ui)\right] \tag{1}$$

where α is the gain of the input-output sigmoïd function g represented in figure 1.

Accounting for the full connection between neurons, the neural potential Ui of neuron i is described by the following relation :

$$Ui = \sum_{j=1}^{N} Tij\ Vj \tag{2}$$

Tij representing the synapse between neuron i and neuron j. The set of all synapses forms the synaptic matrix which contains all the interconnection weights. Generally, synapses are analog and are taken to be excitatory if $Tij > 0$ or inhibitory if $Tij < 0$. In the same way neurons are analog but regarding to equation (1), the sigmoïd of the I/O function tends to give them a binary character (i.e. $Vi = 0$ or $Vi = 1$)

The energy E of such a system is given by

$$E = -1/2 \sum_{i}^{N} \sum_{j}^{N} Tij\ Vi\ Vj \tag{3}$$

and Ui derives from E $\quad Ui = -(\partial E/\partial Ui) \quad (4)$

Equation (1) guarantees evolution to a minimum value of E. When the network is used as a Content-Addressable Memory (CAM), the local minima of E are the metastable states (or stored memories) of the network [1]. But when it is used in optimization problems, the purpose is to reach the best possible metastable state, i.e. that of smallest energy. The sigmoïd shape of figure 1 provides some possibility to escape a metastable state and fall into a better one.

3 - THE TSP COMPUTER SIMULATION

3.1. The Traveling Salesman Problem (TSP)

In this problem, a Salesman has to visit a set of N cities. The TSP consists of finding the shortest length of different tours (defined by an ordered list of the N cities). Although it seems to be easy, this problem is really difficult because when the number of cities increases the number of possible solutions augments as n ! The TSP therefore is a difficult optimization problem having combinatorial complexity.

3.2. The basic neural algorithm

In general, solving optimization problems by neural networks can be described by the following process :

i) Determination of the Energy-function (or Cost-function) E depending on problem specific constraints

ii) Minimizing this function E by an appropriate dynamics leads to sub-optimal solutions.

In the permutation matrix representation used in the previous work[3], the fact that the city X is in the i^{-th} position in a tour list is described by the state of neuron xi. Vxi is equal to "1" if the city X occupies the position i, and "O" otherwise. The energy-function E of this Euclidian TSP with distances dxy between cities x and y (dxy <1) is given by :

$$E = A/2 \sum_{x}^{N} \sum_{i}^{N} \sum_{j \neq i}^{N} Vxi\ Vxj + B/2 \sum_{i}^{N} \sum_{x}^{N} \sum_{x \neq y}^{N} Vxi\ Vxj + C/2 \left(\sum_{x}^{N} \sum_{i}^{N} Vxi - n' \right)^2$$
$$+ D/2 \sum_{x}^{N} \sum_{y \neq x}^{N} \sum_{i}^{N} dxy\ Vxi\ (Vy,i+1 + Vy,i-1) \qquad (5)$$

where A, B, C, D and n' are constants and N the number of cities. According to equations (1) and (4) we obtain :

$$Uxi = -A \sum_{j \neq i}^{N} Vxj - B \sum_{y \neq x}^{N} Vyi - C \left(\sum_{x}^{N} \sum_{j}^{N} Vxj - n' \right)$$
$$- D \sum_{y}^{N} dxy\ (Vy,i+1 + Vy,i-1) \qquad (6)$$

In equation (6), the dynamics of the neurons is not taken into account, but instead only the steady state is given. Therefore, the dynamics depends on signal propagation and on non-linear components.

3.3. The TSP simulation results

i) Data

The main data in this calculation are the set of N(N-1)/2 distances between cities. To have a comparison reference, the same data used by Hopfield [3] have been chosen for our simulations. Then extrapolating the graphic representation in figure 2, a symetrical [dxy] matrix has been generated. The constant data are set to be :

A = B = D = 500 ; C = 200 ; α = 50 ; n' = 15 for N = 10 cities.

ii) Simulation process

As in Hopfield's algorithm, elements Uoo of the potential matrix [Uxi] at the initial instant are chosen such that

$$\sum_{i}^{N} \sum_{x}^{N} V_{xi} = N \tag{7}$$

and by using the inverse function $g^{-1}(V_{xi})$. In our case all the Vxi have been fixed to 1/N and then, to favour randomly any tour, we add noise δU_{xi} to all Uxi with

$$U_{xi} = U_{oo} + \delta U_{xi} \tag{8}$$

and $-0.1/\alpha \leqslant \delta U_{xi} \leqslant 0.1/\alpha$

According to the asynchronous evolution of neurons, neural output states are randomly choosen to be up dated by equations (1) and (6). The evolution process (fig. 3a) can be described by the following steps :

a) a neuron is choosen at random ;
b) the corresponding potential is calculated by using equation (6) ;
c) the neural state is up dated by equation (1)

Operations a,b and c are repeated $10 \times N^2$ times and then an iteration is obtained.

iii) Convergence

Applying equation (5) to the resulting matrix $[V_{xi}]$ at the end of an iteration, the energy of the system is calculated. If this energy remains constant after few iterations (5 iterations in our case) convergence is assumed to be attained. The tour list represented by $[V_{xi}]$ is valid if all the constaints required by the problem are satisfied. Then a permutation matrix is obtained and the tour length is calculated. Figure 3b represents result of 199 tries with more than 50 % converging on valid tours. The highest frequency of appearance is attained for l around 3.50 (l is the tour length) although the shortest length l of the tours is about 2.716 (fig. 3c).

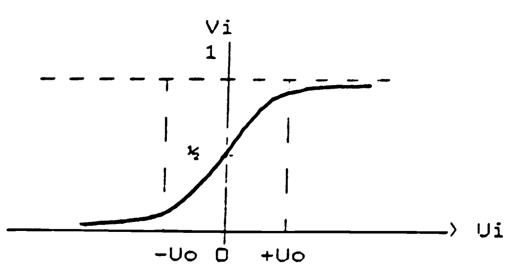

Fig. 1 :

The sigmoïd I/O function of a neuron Vi = g(Ui)

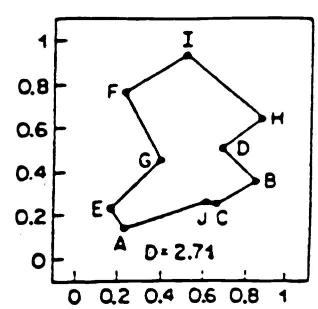

Fig. 2 :

The shortest path found by the analog convergence on 10 random cities (after Hopfield an Tank)

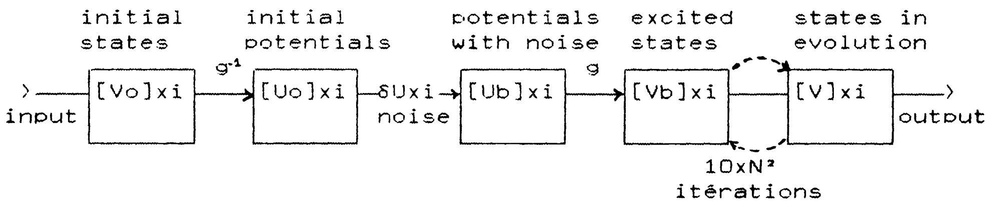

Fig.3a : Schemtic diagram of dynamical evolution of neurons in our simulations

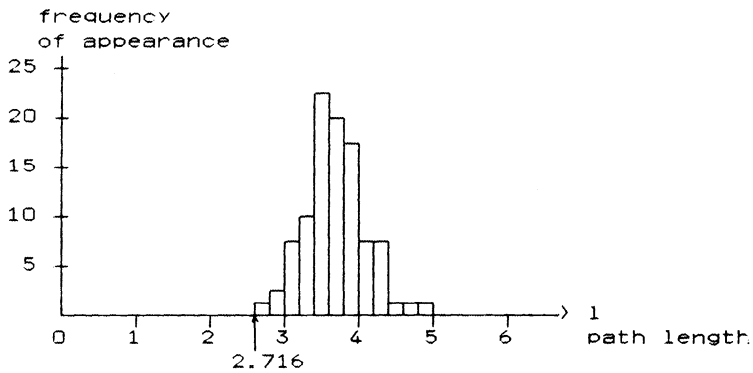

Fig.3b : An histogram of different path lengths obtained after 100 convergences with our algorithm

```
Iter. n°1   energy = 4705.54

0 0 0 0 0 0 1 0 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0
0 0 0 1 0 0 0 0 0 0

Iter. n°2   energy = 4856.04

0 0 0 0 0 0 1 0 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 1 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0

Iter. n°3   energy = 4704.71

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 1 1 0 0 0 0 0

Iter. n°4   energy = 4638.09

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 1 1 0 0 0
0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0

Iter. n°5   energy = 4334.39

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0

Iter. n°6   energy = 4290.72

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0

Iter. n°7   energy = 4295.03

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0

Iter. n°8   energy = 3858.08

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0

Iter. n°9   energy = 3858.08

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0

Iter. n°10   energy = 3858.08

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0

Iter. n°11   energy = 3858.08

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0

Iter. n°12   energy = 3858.08

0 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0

STOP AT ITERATION  n°12

OBTAINED TOUR LIST = ( 2  4  8  9  6  7  5  1  10  3  2 )

PATH LENGTH = 2.716
ENERGY OF CONVERGENCE  Ev = 3858.08
```

Fig. 3c : One neural evolution converging on the shortest path length in a 10 cities TSP

4 - OPTICAL IMPLEMENTATION OF SYNAPSES

In our approach of the optical implemenation of the TSP solving neural networks, we use an NxN array of computer-generated holograms (CGH's). According to the matrix representation defined by the algorithm, N^2 neurons are required to depict the N cities and their position in a tour list. One CGH represents one neuron and its point-spread function represents the set of synaptic coefficients of the corresponding neuron. An N^4 interconnexion is then required to build up this system. Regarding to the previous work presented by Caulfield [5], the N^4 interconnection could be realized by the matrix [Txiyi]

4.1. Implementation of the CGH's used

In the algorithm described earlier, the connexion matrix can be written as

$$\begin{aligned} Txi,yj &= - A\delta xy\ (1-\delta ij) - B\,\delta ij\ (1-\delta\ xy) - C \\ &\quad - D\ dxy\ (\delta j,i+1 + \delta j,i-1) \end{aligned} \tag{9}$$

The NxN sub-holograms comprised in the connexion matrix are deduced from equation (9). These arrays are coded in multigray level images and used to generate the CGHs by the classical way [4]. Then, these NxN CGHs are meticulously arranged to form the connexion matrix required to perform the N^4 interconnexion. An example of such an image is presented in figure 4 which shows an NxN array of constants used to generate the CGH Txi11.

4.2. Description of the system (figure n°5)

An NxN image with a constant gray level is used as an initial state. This matrix image [Vyj] is in plane π_1 at distance d = 2f, from the lens L1 (f is the focal of L1). In plane π , at the same distance d from L1, behind the back focal of L1, there is the NxN matrix [T] of CGHs. In the same plane π very close to the matrix [T], we put a lens L2 with focal F = 2f. Then image plane πo is placed in the back focal of L2. Collimated laser light is used to illuminate the object [Vyj] ; and from the plane π which contains the image of [Vyj] through lens L1, diffracted light forms the synaptic product (SP) in plane πo :

$$SP = \sum_{yj}^{N} Txiyj{*}Vyj \tag{10}$$

The neural potential Uxi is then obtained by adding the positive constant Cn' to the all negative synaptic products SP :

$$Uxi = Cn' + \sum_{yj}^{N} Txiyj{*}Vyj \tag{11}$$

The quantity Uxi is produced at many diffraction orders. The first order quantity is just considered and, an appropriate detector yields a multi-gray level image ready for feed back the input of the system. The latter operation can be accomplished in two ways :

i) Detecting and measuring light intensities at the center of all the NxN squares of the image. The input image [Vyj] is then considered as an amplitude and the detected amplitude is $(Uxi)^2$. Applying equation (1) to the quantities Uxi a multi gray image is then generated.

ii) Not just the center of each pixel is taken into account but integration of interference fringes is realized over the whole surface of each pixel. The input image [Vyj] in this case is considered as an energy flow and then, the detected quantity is equal to $\sqrt{-Txiyj}$. An appropriate transducer is then used to yield the multi-gray level image. Due to this operation, the detected image (a multi-gray level image) is the up dated image fed back at the input.

This operation is then repeated until the first order output image obtained at the plan π o, remains constant after few iterations, then a convergence of the system is obtained. A dynamical system could be implemented by using a suitable spatial light modulator to accomplish the feed back.

i								
8	A+C	C + D.d12	C + D.d13	C + D.d14	C + D.d15	C + D.d16	C + D.d17	C + D.d18
7	A+C	C	C	C	C	C	C	C
6	A+C	C	C	C	C	C	C	C
5	A+C	C	C	C	C	C	C	C
4	A+C	C	C	C	C	C	C	C
3	A+C	C	C	C	C	C	C	C
2	A+C	C + D.d12	C + D.d13	C + D.d14	C + D.d15	C + D.d16	C + D.d17	C + D.d18
1	C	B+C	B+C	B+C	B+C	B+C	B+C	B+C
	1	2	3	4	5	6	7	8	x

Fig.4 : A matrix image for y=j=1 used to form the CGH Txi11

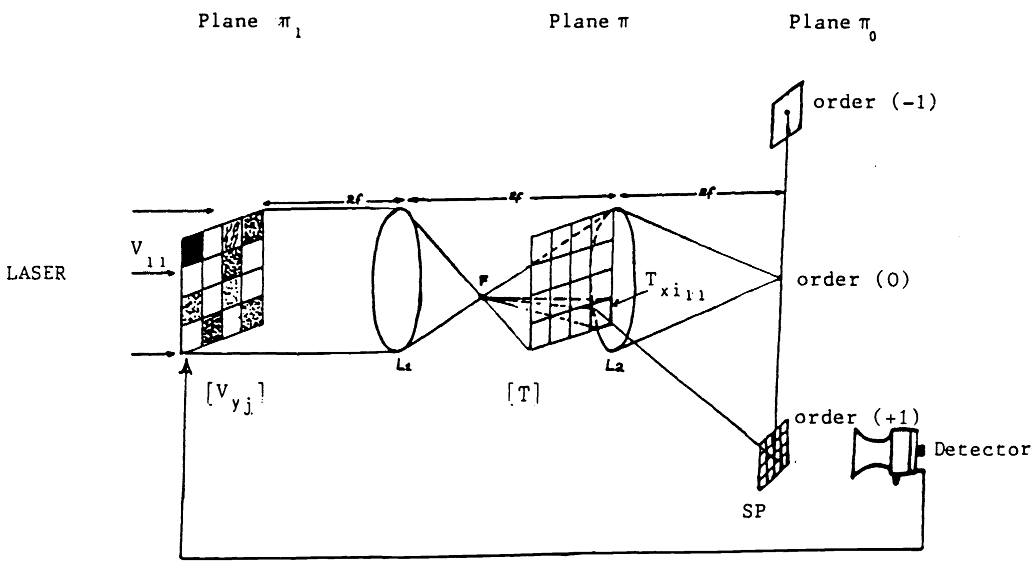

Fig.5: Schematic diagram of the TSP solving optical neural network .

5 - CONCLUSION

Considered earlier as well-suited for use in the associative mode, neural networks could be interesting for solving optimization problems. Using both optics and neural networks properties, we have suggested the feasability of optical neural systems for solving intrinsically difficult optimization problems. Adaptation of the optical setup to other optimization problems basically only requires changing the synaptic matrix i.e. the CGH array.

6 - ACKNOLEDGMENTS

Thanks are due to E.H. Soubari for useful discussions. This work has been supported by the dawning collaboration between Laboratoire des Systèmes Photoniques (ENSPS, University of Strasbourg) and Institut d'Optique Théorique et Appliquée (Orsay, University of Paris-Sud).

7 - REFERENCES

[1]- J.J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, Proc. Natl. Acad. Sci. USA, 79 (1982) 2554.

[2]- J.J. Hopfield, neurons with graded response have collective computationnal properties like those of two-stat neurons, Proc. Natl. Acad. Sci. USA, 81 (1984) 3088.

[3]- J.J. Hopfield, and D.W. Tank, Neural computation of decisions in optimisation problems, Biol. Cybern., 52 (1985) 141

[4]- A.W. Lohmann and D.P. Paris, binary Fraunhoffer Holograms generated by computer, App. opt. vol. 6(10), (1967) 1739

[5]- H.J. Caulfield, parallel N^4 weighted optical interconnexions, Appl. opt. vol. 26 (19), (1987) 4039

[6]- E.Gi Paeck and D. Psaltis, optical associative memory using Fourier transform holograms, opt. Eng. vol 26 (5), (1987) 428

[7]- H.J. White, N.B. Aldrige and I. Linsay, Digital and analogue holographic associative memories, opt. Eng. vol. 27 (1), (1988) 030

[8]- D. Psaltis, optical realizations of neural network models, SPIE vol. 700 IOCC, (1986) 278

[9]- R.C. Chevallier, G.Y. Sirat and A.D. Maruani, proc of neuro'88, Paris june 6-9, (1988) 789

[10]- M. Sakarovitch, optimisation combinatoire, Ed. Hermmann (Paris, 1984)