Neurocomputing 4 (1992) 249–254
Elsevier

Neural network computing for knight's tour problems

Yoshiyasu Takefuji[a] **and Kuo Chun Lee**[b]

[a] *Keio University, Faculty of Environmental Information, 5322 Endoh, Fujisawa 252, Japan, and Department of Electrical Engineering, Case Western Reserve University, Cleveland, OH 44106, USA*
[b] *R&D Department, Cirrus Logic, Inc., Fremont, CA 94538, USA*

Received 25 June 1991
Revised 20 February 1992

Abstract

Takefuji, Y. and K.C. Lee, Neural network computing for knight's tour problems, Neurocomputing 4 (1992) 249-254.

This paper presents the first algorithm for finding a knight's tour on the general chessboard where the proposed algorithm is based on neural network computing. In the knight's tour problem a knight must traverse all of the squares on an $m \times n$ chessboard but visit every square once and only once and return to the originated square where the knight moves in an L-shape route.

Keywords. Knight's tour problems; neural networks; parallel algorithms; combinatorial optimization.

1. Introduction

The earliest attempt to find a knight's tour on the chessboard was made by Leonhard Euler in 1759 [2]. A knight must traverse all of the squares on an $m \times n$ chessboard but visit every square once and only once and return to the originated square where the knight moves in an L-shape route. Many great mathematicians including De Moivre, Vandermonde [9], Warnsdorff [10], Pratt [7], Roget, Legendre [5] and De Lavernede [3] attempted to solve the general problem. Their methods are based on either divide-and-conquer or backtracking dedicated for solving the 8×8 chessboard knight's tour problem. No general methods have been given to the problem in the last two centuries ([1]). W.W.R. Ball and H.S.M. Coxeter reintroduced the problem in their book in 1892 and 1974. A parallel algorithm for finding a knight's tour on the $m \times n$ chessboard is presented in this paper. The problem is to find a constrained Hamiltonian circuit which belongs to NP-complete problems. However, we are not sure whether the knight's tour problem is NP-compete or not. The artificial neural network computing makes it possible to provide the elegant algorithm in parallel which takes advantage of the simplified biological neural computation. Several examples are shown to demonstrate the capability of our parallel algorithm.

Correspondence to: Yoshiyasu Takefuji, Keio University, Faculty of Environmental Information, 5322 Endoh, Fujisawa 252, Japan.

2. Neural representation and motion equations

A two-dimensional triangular network representation is introduced. The system requires $p(p-1)/2$ processing elements or hysteresis McCulloch-Pitts neurons where p is the number of squares on an $m \times n$ chessboard. Each state of the neurons represents a path. In order to establish a knight's path on a square on the chessboard, two neurons must be fired because one is for coming into the square and the other is for going out of the square. In our algorithm the following motion equation for the ijth neuron is used to solve the problem:

if $\mathrm{d}_{ij} = 1$ then

$$\frac{\mathrm{d}U_{ij}}{\mathrm{d}t} = -\left(\sum_{k=1}^{p} V_{ik}\mathrm{d}_{ik} - 2\right) - \left(\sum_{k=1}^{p} V_{kj}\mathrm{d}_{kj} - 2\right)$$

if $\mathrm{d}_{ij} = 0$ then

$$\frac{\mathrm{d}U_{ij}}{\mathrm{d}t} = 0 ,$$

where the upper triangular elements in the two-dimensional array are only used: V_{ij} for $i < j$ and V_{ij} for $i > j$ is given by V_{ji}. The state of V_{ij} actually represents a knight's tour between the ith and the jth square. In other words, the upper triangular neural array represents the nondirected adjacency matrix in order to find a Hamiltonian circuit. In Eq. (1) d_{ij} is 1 if the move from the ith square to the jth square is legal or valid, and 0 otherwise. The first term and the second term are to determine a move between the ith and the jth square if it is valid. The first term forces the ijth neuron to have two valid moves from the ith square and the second term from the jth square respectively. The state of V_{ij} is updated by the hysteresis McCulloch-Pitts function: $V_{ij} = f(U_{ij}) = 1$ if $U_{ij} > 3$, 0 if $U_{ij} < 0$, and no changes otherwise.

Figure 1 shows how the knight moves in an L-shape route. In a 3×4 chessboard knight's tour problem, 12 squares are numbered from left to right and from top to bottom as shown in *Fig. 2a*. From #2 square there are three valid moves: V_{28}, V_{29} and $V_{2,11}$ where only two moves are needed to find the Hamiltonian circuit.

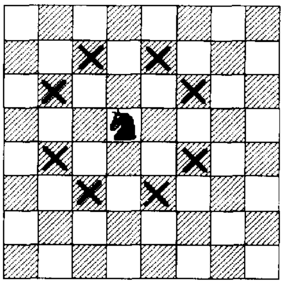


Fig. 1. The knight moves in an L-shape route.

Figure 2(b) shows the $p(p-1)/2 = 66$ neurons for this problem where the black squares indicate their outputs are ones. *Figure 2(b)* depicts the two closed loops as shown in *Fig. 2(c)* which is not the Hamiltonian circuit. Unfortunately there is no solution for the 3×4 knight's tour problem. In 1943 Schuh [8] stated that the number of squares has to be even which is necessary but not sufficient.

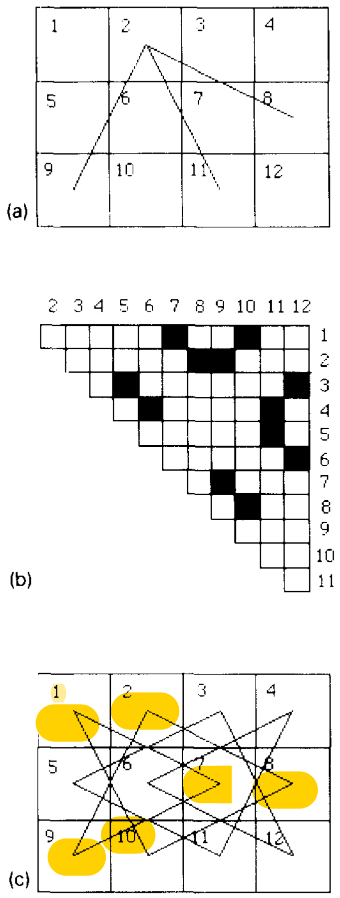


Fig. 2. (a) A 3 × 4 chessboard; (b) Neural representation; (c) Unsatisfactory solution.

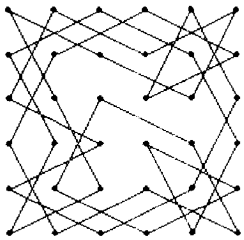


Fig. 3. 6 × 6 chessboard.

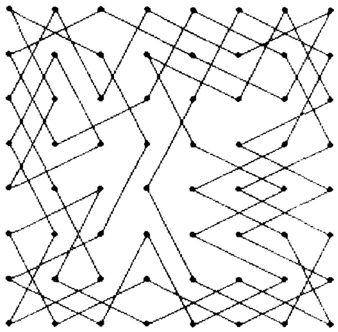


Fig. 4. 8 × 8 chessboard.

3. Results

We have developed the simulator on a Macintosh SE/30 based on Eq. (1) to verify our algorithm. *Figures 3* through *10* show solutions for 6×6, 8×8, 10×10, 12×12,

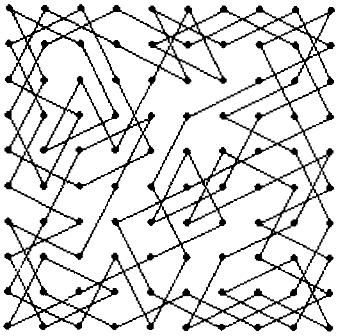


Fig. 5. 10×10 chessboard.

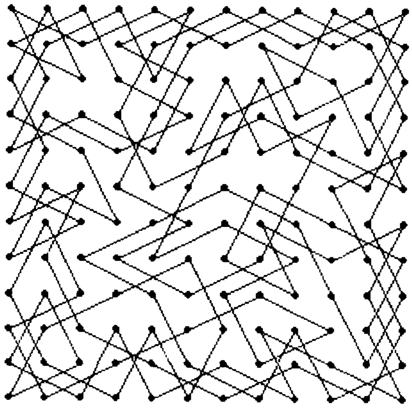


Fig. 6. 12×12 chessboard.

14×14, 16×16, 18×18 and 20×20 chessboard knight's tour problems respectively.

The average number of iteration steps takes less than 100 steps. We have observed that the larger the problem, the more often the state of the system converged to the unsatisfactory solution with several closed loops. Without the hysteresis property, the state of the system tends to oscillate and hardly converges to the solution.

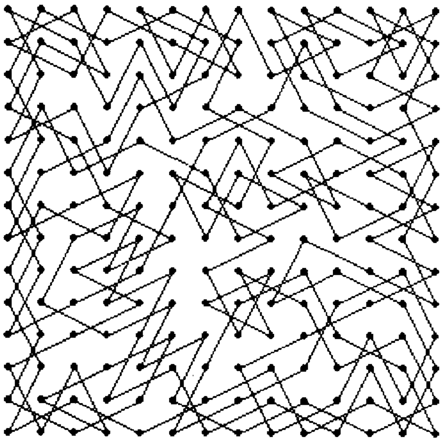


Fig. 7. 14×14 chessboard.

References

[1] R.W.W. Ball and H.S.M. Coxeter, *Mathematical Recreations & Essays*, 1892 (1st ed.) and 1974 (12th ed.) (University of Toronto Press).

[2] Euler, L., *Memoire de Berlin for 1759* (Berlin, 1766) 310–337.

[3] De Lavernede, *Memoires de l'Academie Royale du Gard* (Nimes, 1839) 151–179.

[4] K.C. Lee and Y. Takefuji, Finding a knight's tour on an $N \times M$ chessboard with $O(NM)$ hysteresis McCulloch-Pitts neurons, submitted.

[5] Legendre, *Theorie des Nombres*, 2nd ed. (Paris, 1830) 165.

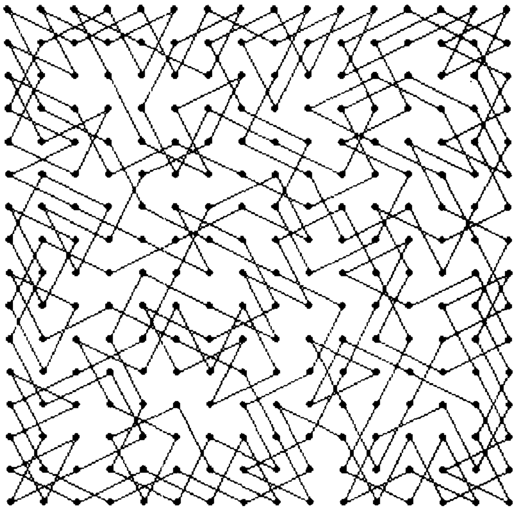


Fig. 8. 16 × 16 chessboard.

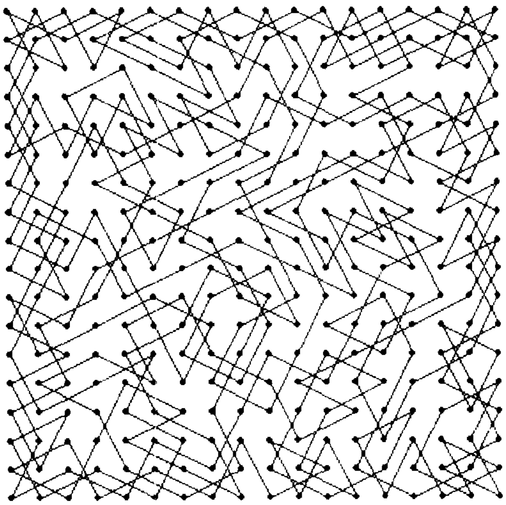


Fig. 9. 18 × 18 chessboard.

[6] W.S. McCulloch and W.H. Pitts, *Bull. Mathemat. Biophys.* 5 (1943) 115.
[7] Pratt, *Studies of Chess*, sixth ed. (London, 1825).
[8] F. Schuh, *The Master Book of Mathematical Recreations* (Dover, New York, 1968).
[9] Vandermonde, *L'Historie de l'Academie des Sciences for 1771* (Paris, 1774) 566–574.
[10] H.C. Warnsdorff, *Des Rosselsprunges einfachste und allgemeinste Losung* (Schmalkalden, 1823).

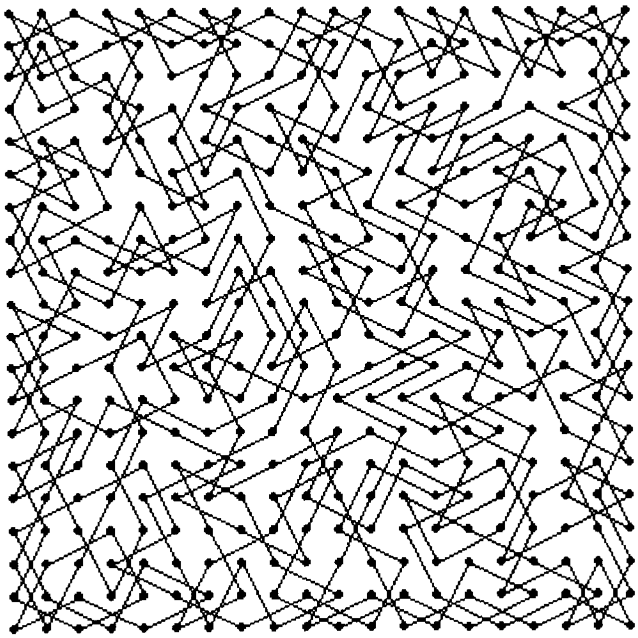
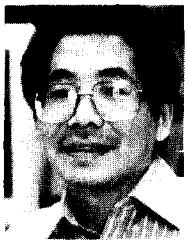


Fig. 10. 20 × 20 chessboard.



Dr. Yoshiyasu Takefuji is an associate professor on faculty of environmental information at Keio University since 1992 and also on faculty of Electrical Engineering at Case Western Reserve University since 1988. Before joining Case, he taught at the University of South Florida and the University of South Carolina. He received his BS (1978), MS (1980), and Ph.D. (1983) from Electrical Engineering from Keio University (Japan). His research interests focus on neural network parallel computing for solving real-world problems. He is interested in VLSI applications and silicon architecture. He received the National Science Foundation/Research Initiation Award in 1989 and is an NSF advisory panelist. A member of the IEEE Computer Society, ACM, International Neural Network Society, and American Association for the Advancement of Science, he received the Information Processing Society of Japan's best paper award in 1980. He authored the book entitled *Neural Network Parallel Computing,* from Kluwer Publishers in January 1992 and coauthored two books *Digital Circuits* (Ohm-Sha Publishers) in 1984 and *Neural Network Computing* (Baifukan Publishers) in 1992. He was an Editor of the *Journal of Neural Network Computing,* is an Associate Editor of *IEEE Trans. on Neural Networks and Neurocomputing,* and is a Guest Editor of *Journal Analog Integrated Circuits and Signal Processing* in the special issue on Analog VLSI Neural Networks, and also Guest Editor of *Neurocomputing* in the special issue on Neural Network Based Optimization (in preparation). He has published more than 90 papers.

Dr. Kuo-chun Lee received his B.S. from National Tsing-Hua University in Taiwan (1984), M.S. from National Chiao-Tung University in Taiwan (1986), and Ph.D. in Electrical Engineering from Case Western Reserve University (1991). Before joining Cirrus Logic, Inc. in 1991 he taught at Chinese Naval Academy Taiwan (1986–1988). He has published 20 journal papers and conference papers with Dr. Takefuji in neural networks. His current interest include placement and routing in CAD, and static timing analysis.