

The Design of An Opti-Electric Hybrid Computer For Expert System(OEHCES)

Shen Xuan-Jing

Li Zhong-Rong

Dept. of Computer Science and Engineering
Harbin Institute of Technology
Harbin 150006, P.R. China

Abstract

This paper, based on the analysis of various problems which are met in developing expert system and comparison of various methods to solve these problems, proposes a scheme of an opti-electric hybrid computer for expert system(OEHCES) which can be realized easily. The OEHCES is consisted of three parts: main control machine, knowledge base and optical matcher. In the OEHCES, the knowledge base is used to store "0-1" matrix form knowledges, and the optical matcher is consisted of a LCLV, two lenses and 1-D photo-detector, and it parallelly performs matching operation. The main control machine is a 16-bit microcomputer, it exchanges information with man, and controls the knowledge base and the optical matcher, it also performs other operations except matching in the OEHCES. The OEHCES uses a modified knowledge presentation and the optical matcher to partially solve the problem that the matching speed is too slow in conventional computer.

Keywords: Expert system, optical information processing, hybrid computer.

1. Introduction

At present, Expert System(ES) is one of the most active branches in the field of artificial intelligence. Because of remarkable economic benefit and social benefit, the development of the ES is very fast in these years. An ES is a program system which has a number of special knowledges and experiences, and most of them is implemented on the conventional Von.Neumann computer[1]. With the further developing of the ES and continuous expanding of the ES application, the number of knowledges to be stored is increasing and the response time required is decreasing, it is found that the conventional computer is not suitable for developing these artificial intelligent(AI) system. This is because the many AI problems need parallel computing, but the conventional computer can only performs series computing. For this reason, various schemes of the non-Von.Neumann computer were

proposed, such as Lisp machine, Database machine, Parallel processors etc.. These special computers can perform some parallel computing to some extent, and increase response speed. But because all these computers are based on 1-D electric information, problems difficult to overcome such as interconnection, time skew, long communication wire etc. are met when to further increase parallelism. Optical information is natural 2-D or 3-D information, so the computing system based on optics will be have the characteristics such as interconnection convenient, without time skew, full parallelism etc.[3]. This means that optical computer will be suitable for developing AI system and those systems which require a number of parallel computing. Based on the above idea, many research workers have proposed various schemes of optical ES and inference machines[4-7],but realized systems

are a few. Its main reason is that at present technology condition(specially the developing level of optical elements), it is difficult to develop a full optical computer system, and it is believed that the developing of opti-electric hybrid computing system is more real[8]. For this reason, we propose a scheme of an opti-electric hybrid computer for ES(OEHCES) which is easily realized at present technology condition.

2. The system overview

An ES is composed of knowledge base, inferencer, interpretation part and acquisition part. Among these, the part to consume the most of time when running is the operation of knowledge matching in the inferencer, it takes up 90 percent time of inferencing[9]. This means that matching operation is an operation bottleneck in an ES. To solve this bottleneck, various algorithms and approaches such as index structure etc. are proposed. But because the conventional computer is based on serial computing, there are some problems to basically solve this problem. We consider that the approach to solve this problem is to adopt parallel matching. To realize parallel matching needs hardware support. In the OEHCES, parallel matching is realized by the optical matcher.

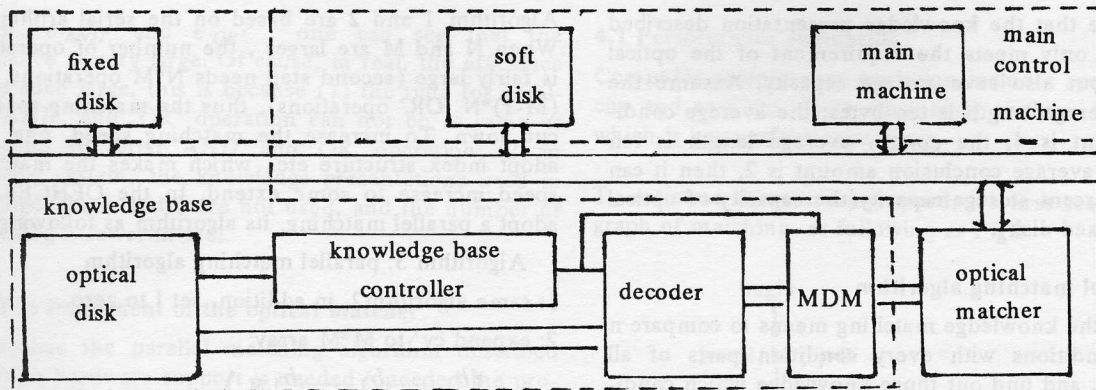


Fig.1. Diagram of the OEHCES

From Fig.1, which shows the overview of the OEHCES, one can see that it has three parts: main control machine, knowledge base and optical matcher. The knowledge base is consisted of optical disk, fixed disk, multidimension memory(MDM), decoder and controller. The optical disk is used to store the condition parts of all knowledges used ES, the fixed disk is used store the conclusion parts of all knowledges, the MDM is used to store the condition parts of knowledges that are used the running ES. The decoder performs the transformation of different knowledge presentations. The MDM is a two ports memory, one of ports is used to access data in word, another is used to access data in bit plane. The optical matcher is consisted of liquid crystal valve (LCLV) which can parallelly perform logic 'AND' operation, cylinder lenses, LED and photodetector array, decoder and coder, and it parallelly performs the knowledge matching operation. The main control machine is consisted of a 16-bit microcomputer(IBM PC/XT, 640k memory, 20M fixed disk, 2*360k softdisk), it exchanges information with man, and coordinates the knowledge base and the optical matcher work. When the main control machine is turn on and runs one ES, it first controls the optical disk and MDM to transform the condition parts of the ES knowledges and store into the MDM. Meantime, the conclusion parts of the ES knowledge are stored to main memory. After these initial work is completed, the ES may begin to work.

3. Optical matcher

3.1 Knowledge presentation

At present, there are many knowledge presentations(such as logic, semantic, network, frame and production etc.) used in ES. Among these, production presentation are often adopted, because it is direct and easy[1]. In the OEHCES, the modified production presentation is used for meeting the requirement of the optical matcher.

If there is a knowledge KS_i , then we can present it with production presentation as following:

$$KS_i: \text{IF } [cond_{i,1} \text{ AND } cond_{i,2} \text{ AND } \dots \text{ AND } cond_{i,j}] \\ \text{THEN } [conc_{i,1}, conc_{i,2}, \dots, conc_{i,j}, \dots, conc_{i,m_i}]$$

(here, the $cond_{i,j}$ indicates the j th condition of the i th knowledge, and the $conc_{i,j}$ indicates the j th conclusion of the i th knowledge)

Assume the number of knowledges in the ES is N , the number of different $cond_{i,j}$ is M , and the number of different $conc_{i,j}$ which are not in the $cond_{i,j}$ is M' , then we use integers $1 \sim M$ to encode the different $cond_{i,j}$ and use integers $M+1 \sim M+M'$ to encode the different $conc_{i,j}$. In addition, we define following several values:

Definition1:

Coded condition vector ccv_i : ccv_i is a vector with n_i elements i.e. $ccv_i = (I_1, I_2, \dots, I_j, \dots, I_{n_i})$ in which I_j is $cond_{i,j}$ code, ($i = 1, 2, \dots, N$).

Definition2:

Coded conclusion vector clv_i : clv_i is a vector with m_i elements i.e., $clv_i = (K_1, K_2, \dots, K_j, \dots, K_{m_i})$ in which K_j is $conc_{i,j}$ code, ($i = 1, 2, \dots, N$).

Definition3:

Condition vector cv_i : cv_i is a vector with M elements, i.e., $cv_i = (a_{i,1}, a_{i,2}, \dots, a_{i,j}, \dots, a_{i,M})$, in which

$$a_{i,j} = \begin{cases} 0 & \text{if there is not the } j\text{th condition in the } i\text{th } KS_i \\ 1 & \text{if there is the } j\text{th condition in the } i\text{th } KS_i. \end{cases}$$

$$(i = 1, 2, \dots, N, j = 1, 2, \dots, M)$$

Definition4:

Condition array CA_i : CA_i is an $M \times M$ array which is consisted of $cv_{i \times M+1} \sim cv_{i \times M+M}$, ($i = 0, 1, 2, \dots, N/M$).

From Definition1-4 we can know that the ccv_i , clv_i and cv_i are the deformed knowledge presentations. In the OEHCES, the conclusion parts stored in the fixed disk are clv_i form, and the condition parts stored in the optical disk are ccv_i form. N cv_i are stored the MDM, and fetched in CA_i form to send the optical matcher.

We believe that the knowledge presentation described above not only meets the requirement of the optical matcher, but also save storage capacity. Assume the $cond_{i,j}$ average length is ten bytes; the average condition amount is 4; the $conc_{i,j}$ average length is ten bytes; the average conclusion amount is 2, then it can save 90 percent storage capacity (the capacity of optical disk and fixed disk).

3.2 Parallel matching algorithm

In an ES, the knowledge matching means to compare n known conditions with every condition parts of all knowledge, and find out those knowledge which condition parts are as same as the n known conditions or in that.

Definition5:

Complete matching: the procedure that finds out those knowledge which condition parts is fully as same as the n known conditions.

Definition6:

Redundant matching: the procedure that finds out those knowledges which condition parts is in the n known condition.

According definition5-6, we design following matching algorithms. Here we assume that before entering the matching algorithms, the n known conditions have been presented into the coded condition vector form (i.e., $v' = (J_1, J_2, \dots, J_n)$).

Algorithms 1: complete matching algorithm

1: change the v' into condition vector form cv'; first set cv' to zero, then set those elements which are corresponding J_i ($i=1, 2, \dots, n$) to one.

2: according following formula(1), calculate the degree of matching s_i :

$$s_i = \bigcup_{j=1}^M v_{i,j} \oplus v'_j \quad (i=1, 2, \dots, M) \quad \dots(1)$$

(here $v_{i,j}$ is the jth element of cv_i , v'_j is the jth element of cv')

3: find out those knowledges which code is i and $s_i = 0$.

Algorithm 2: redundant matching algorithm

cv' to full one, then set those elements which are corresponding J_i ($i=1, 2, \dots, n$) to zero.

2: according following formula(2), calculate the degree of matching s_i :

$$s_i = \bigcup_{j=1}^M v_{i,j} \cdot v_j \quad (i=1, 2, \dots, N, j=1, 2, \dots, M) \quad \dots(2)$$

3: same the algorithm 1.

Algorithm 1 and 2 are based on the serial arithmetic. When N and M are larger, the number of operations is fairly large (second step needs $N \cdot M$ operations, and $(M-1) \cdot N$ 'OR' operations), thus the matching speed is cut down. To increase the matching speed, one may adopt index structure etc., which makes the matching speed increase to some extent. In the OEHCES, we adopt a parallel matching, its algorithm as following:

Algorithm 3: parallel matching algorithm

1: same algorithm2, in addition, set I to zero.

2: expand cv' to $M \cdot M$ array:

$$CA' = \begin{bmatrix} b_{1,1}, \dots, b_{1,j}, \dots, b_{1,M} \\ \vdots \\ b_{i,1}, \dots, b_{i,j}, \dots, b_{i,M} \\ \vdots \\ b_{M,1}, \dots, b_{M,j}, \dots, b_{M,M} \end{bmatrix} \quad (i, j=1, 2, \dots, M)$$

(here $b_{i,j} = v'_j$ ($i=1, 2, \dots, M$))

3: calculate the AND of CA' and CA_I , A:

$$A = CA' \cdot CA_I = \begin{bmatrix} a_{1,1}, \dots, a_{1,j}, \dots, a_{1,M} \\ \vdots \\ a_{i,1}, \dots, a_{i,j}, \dots, a_{i,M} \\ \vdots \\ a_{M,1}, \dots, a_{M,j}, \dots, a_{M,M} \end{bmatrix}$$

(here $a_{i,j} = b_{i,j} \cdot c_{i,j}$, $c_{i,j}$ is the element of CA_I)

4: calculate the degree of matching s_i :

$$s_i = \bigcup_{j=1}^M a_{i,j} \quad (i, j=1, 2, \dots, M)$$

5: find out those i which $s_i = 0$ (the i is the knowledge code matched), in addition, set $I = I + 1$.

6: if $I > \lfloor N/M \rfloor$, then stop, otherwise goto 3.

From the algorithm3, one can see that it carries on $\lfloor N/M \rfloor$ 'AND' operations and $\lfloor N/M \rfloor$ 'OR' operations. Suppose the time for parallel 'AND' and 'OR' operation is as same as serial 'AND' and 'OR' operation, then the efficiency increased is:

$$e'_{AND} = \frac{\text{the number of serial 'AND' } N \cdot M}{\text{the number of parallel 'AND' } N/M} = \frac{N \cdot M}{N/M} = M^2$$

$$e'_{OR} = \frac{\text{the number of serial 'OR' } N \cdot (M-1)}{\text{the number of parallel 'OR' } N/M} = \frac{N \cdot (M-1)}{N/M} = M^2$$

From e_{AND} and e_{OR} , one can see that the efficiency is very large. Of course, in real, the efficiency is no such large, this is because (1) the time for parallel 'AND' and 'OR' operation can not be as same as the time for serial 'AND' and 'OR' operation; (2) in the real system, the knowledge presentations used in the algorithm 1 and 2 are not used, and the number of matching is fewer in real.

3.3 The implement of the optical matcher

To realize the parallel matching algorithm described above, a hardware support is needed (needed the processor which can simultaneously perform $M \times M$ bits 'AND' operations and other auxiliary circuit). Such hardware support can be constructed by electric elements in theory, but which have some shortcoming such as the number of wire is too much, M can not be very big etc.. For this reason, we propose the scheme to realize parallel matching by optical matcher.

The diagram of the optical matcher proposed is shown as Fig.2. It is consisted of LED array(D), two cylinder lenses(L1,L2), 'AND' operation part LCLV(binary liquid crystal light valve), photodetector(P), coder and decoder. The decoder performs the function of algorithm 3 step 1, i.e., it transforms the coded condition vector into condition vector cv' . The LED array D changes the cv' into light dot vector, and the cylinder lens L1 expands the light vector into $M \times M$ light dot array (performs the function of algorithm 3 step 2) and sends to LCLV. The LCLV another input is the CA_i light dot array sent by MDM, and the LCLV 'AND's the two inputs (performs the function algorithm 3 step 3). The lens L2 changes the 'AND' results into 1-D light dot vector (performs the function of the algorithm 3 step 4). The photodetector P performs the detection of the 1-D light dot vector and changes it into '0-1' vector to send the coder. The coder encodes the vector and sends it to the main control machine.

4. Performance evaluation and possibility analysis

4.1 Performance evaluation

Computer performance evaluation includes speed, capacity and cost etc.. Here we mainly analyze the speed which is key performance.

To estimate the matching speed, we first see the flow graph of matching, as following as Fig.3.

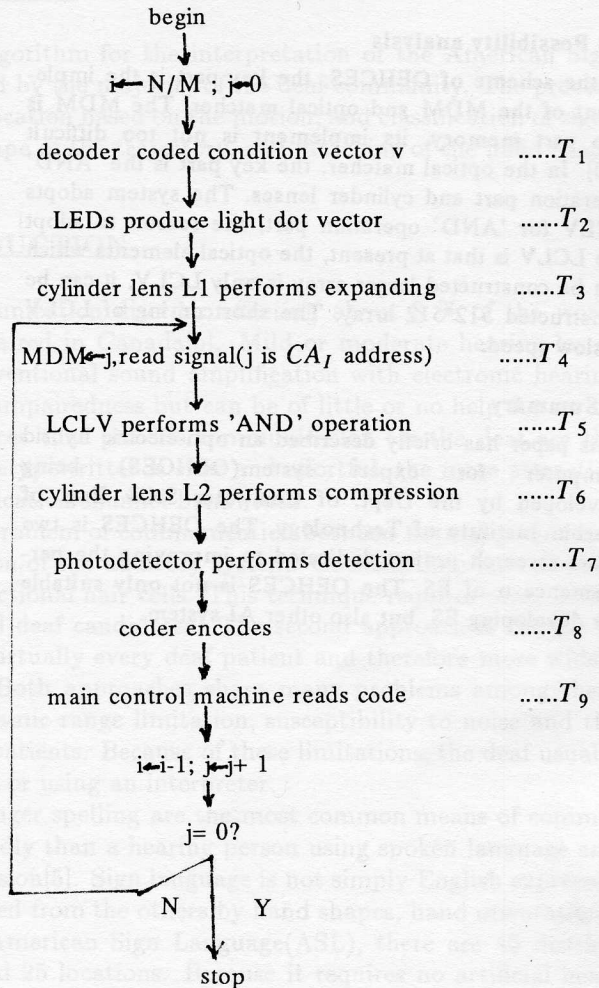


Fig.3 the flow graph of optical matcher

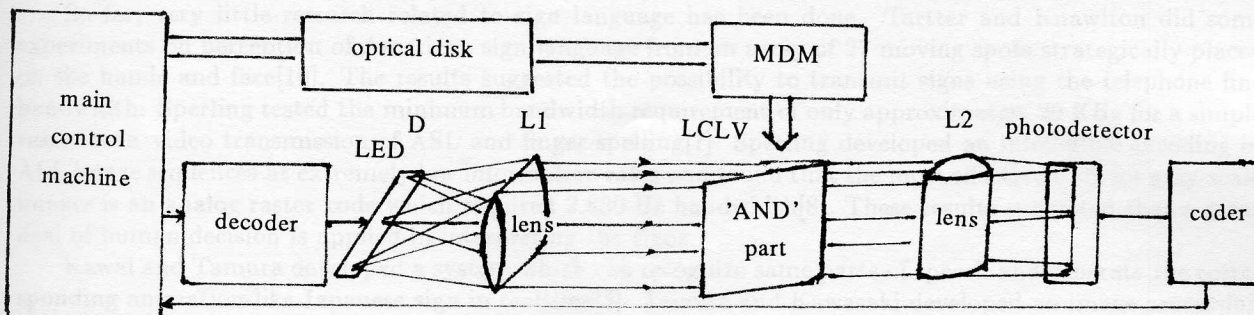


Fig.2. Diagram of the optical matcher

In Fig.3, the T_i indicates time to perform that step. Such, the time(T) to match once in N Knowledges is:

$$T = \sum_{i=1}^3 T_i + N/M \sum_{i=4}^9 T_i$$

and the matching speed v is:
 $v = N/T.$

4.2 Possibility analysis

In the scheme of OEHCES, the key part is the implement of the MDM and optical matcher. The MDM is two part memory, its implement is not too difficult [10]. In the optical matcher, the key part is the 'AND' operation part and cylinder lenses. The system adopts LCLV for 'AND' operation part, the reason to adopt the LCLV is that at present, the optical elements which can be constructed larger array is only LCLV, it can be constructed 512*512 array. The shortcoming of LCLV is slow speed.

5. Summary

This paper has briefly described an opti-electric hybrid computer for expert system(OEHCES) being developed by the Dept. of Computer Sci.& Eng. of Harbin Institute of Technology. The OEHCES is two years research project dedicated to improving the performance o of ES. The OEHCES is not only suitable for developing ES, but also other AI system.

6. Reference

- [1] Zhao Rui-Qing, 'The principal of expert system,' Meteorology publishing house, 1987.
- [2] Alastair D.Mcaulay, 'Real-time optical expert systems,' Applied Optics, Vol.26, No.10 1927(1987).
- [3] Li Ti-Ran, Zhang Ya-Ping, 'Optical computer,' Oct. 1985.
- [4] Cardinal Warde, James Kottas, 'Hybrid optical inference machines: architectural considerations,' Applied Optics, Vol.25, No.6, 940(1986).
- [5] Hareld H.Szu, H.John Caulfied, 'Optical expert systems,' Applied Optics, Vol.26, 1943(1987). Vol.26, No.10, 1943(1987).
- [6] G.Eichmann, H.J.Caulfied, 'Optical learning(inference) machine,' Applied Optics, Vol. 24, 2051(1985). Optics, Vol.24, 2051(1985).
- [7] X.J.Shen, 'The design of a real-time intelligent decision system,' 89' IFAC-MMS.
- [8] X.J.Shen, 'A dynamic target recognition system,' 9th Inter. Recognition Confer., Oct. 1988.
- [9] Forgy C.L., 'Rete: a fast algorithm for the many pattern/ many object pattern match problem,' AI(19)17-37, 1982.
- [10] Kenneth E. Batcher, 'The multidimensional memory in STARAN,' IEEE, Trans. on computer, Feb. 1977.

