NEURAL NETWORK BASED PARALLEL EXPERT SYSTEM

Saša Prešeren, Ludvik Gyergyek,
Anton P. Železnikar, and Sonja Jerman,
Iskra Delta and
Jožef Stefan Institute Ljubljana

Keywords: neural network, parallel processing, expert system

ABSTRACT This paper describes a proposal for an expert system which is based on a neural network principles. Properties of a neural network based expert system are pointed out, the structure of neural network is formulated, processing elements are proposed and the formulation of an expert system with the neural network properties is suggested. The model is supposed to use a parallel computer with distributed processing capability, distributed memory and distributed knowledge. The "forgetting" and the "learning" functions are proposed for adaptability of the neural network based parallel expert system.

PARALELNI EKSPERTNI SISTEM, KI TEMELJI NA NEVRALNI MREŽI Članek podaja predlog ekspertnega sistema, ki temelji na principih nevralne mreže. Poudarjene so značilnosti ekspertnega sistema. ki temelji na nevralni mreži, formulirana je struktura nevralne mreže, predlagani procesni elementi in sugerirana formulacija ekspertnega sistema, ki ima lastnosti nevralne mreže. Model naj bi uporabljal paralelni računalnik s porazdeljenimi procesnimi sposobnostmi, porazdeljenim pomnilnikom ter porazdeljenim znanjem. Predlagani sta funkciji učenja in pozabljanja, ki omogočata prilagodljivost paralelnega ekspertnega sistema.

1. INTRODUCTION

Nowadays we are faced with a great interest in a neural network modeling. Neurocomputing is considered as a nonalgorithmic approach to information processing (2). This surprising fact of nonalgorithmic approach is possible because all procedural capabilities of an artificial neural network are incorporated and hidden in a special purpose hardware. Such special purpose hardware for implementing neural network algorithms is already available for character recognition (1), for analog VLSI networks for vision systems (3) and others. The learning laws, the scheduling functions and the transfer functions are in those systems realized in hardware and incorporate equations which modify the adaptive coefficients, called weights in a processing element's local memory. In this way the neural network gains adaptability and has a so called self organizing capability.

The most interesting property of neural networks is the interconnection of a large number of processing elements, even if these elements are represented by a simplified function. Different approaches for a network representation have been considered as for example a multilevel neural network with selforganizing capability and forward/backward signal flow routing as well as distributed network of individual processing elements with good message passing capabilities (5).

In this paper we are not interested in biological details of living neural structures. Rather we use the idea of a neural network structure, organization and functioning and implement a computer system with similar properties but using powerful processors. Very important is to identify those properties which make a multiprocessor based expert system to act like a neural network.

2. PROPERTIES OF NEURAL NETWORK BASED EXPERT SYSTEM

Most research work in neuralcomputing is focused to the implementation of a fine grain special purpose hardware which is performing some intelligent subfunction. Processing elements have to be "adaptive" as well as capable of communication with the neighbors. In order to incorporate the neural network properties into an expert system we first have to focus our attention to the following properties of a neural network:

- Patterns (representational features) are memorized over the whole system in a sense that properties of an object are stored in different memory modules. Therefore we say that we have a distributed information storing.

- Recognition (for example object recognition) is performed through feature extraction, that means symbolic representation rather than by memorizing the whole pattern (image).

- A neural network is fault tolerant so that if a part of a system is damaged or fails the system still works, although the probability for wrong identification is higher and/or time required for identification is longer.

- Recognition and learning are performed by local adaptation in the sense that a set of knowledge atoms is updated, a set of representational features is updated and a set

of distribution probabilities is locally updated. A new strategy is selected in real time.

- The dynamic of the knowledge network is parallel and asynchronous.

- The knowledge network contains elements which have high percentage of fuzziness. The neural network system can function with incomplete or inaccurate input data.

Many neural network based systems are problem oriented and therefore limited to a specific task. A classical neurocomputing incorporates algorithmic properties in a special purpose hardware, therefore it is limited to some problems as for example pattern recognition, handwritten character recognition, etc. Our goal is to apply neural network properties in a complex knowledge base system which may be from any expert-system domain. The rules used in solution would be generated by training data and may stay hidden to the system programmer.

A natural property of most expert systems is parallelism. The expert system is composed of a number of modules which are executed on different processors. Calling of a module depends on a data pattern. Knowledge network, together with all accompanying steps of symbolic representation, normalization and optimization gives the framework for parallelism.

3. OPERATORS AND OPERANDS

Neurocomputing can not copy the human brain structure, but much more interesting – it successfully applies some concepts of human brain. Let us formulate the elements and the structure of a neural network.

In a neural network we have:

- operators (network elements) and
- operands (stimulus, output).

A neural network is a composition of operators. Operators are influenced by a learning process, aging process, memorizing process etc. In a neural network system a processor and a knowledge system can not be separated, because a processor itself is changed or influenced, and acquires knowledge by processing information. A processing system and a knowledge system are nonseparable homogeneous units.

An operation which is performed by an operator remains unchanged by time. Therefore, a type of each operator is fixed but its power varies in intensity. The variability of a power intensity is a consequence of constant and repeating interactions of the network with the environment. A self organization, a so called learning without a teacher, is explained by varying the power intensity of a particular operator. A weight of communication determines the power of operator (intensity of connection) and depends on a design of the network and on information that a network system has learned.

Information which enters a neural network is actually processed by a set of operators. Self organization of the neural network means that after some time the same information is processed by modified operators. Operator modification is a consequence of self learning process.

In a neural based parallel expert system operators are individual processors performing a sequence of feature extractions. By time the same processors would perform a different sequence or even a different set of feature extractions, because the system has "learned" how to do the work better. Operands in the expert system are symbolic descriptions of objects which are entering the expert system.

4. PROCESSING ELEMENTS

Human brain consists of interconnected neurons which are simple processing units. Each neuron is connected to as many as 10.000 other neurons forming therefore a highly parallel processing system. Connections are sometimes even bidirectional (input and feedback) and differ in power (strong or weak). No computer can imitate such complexity of the human brain.

An artificial neuron is a classical processing element in neurocomputing. The key to success of neurocomputing is parallel processing which opened new perspectives, approaches, reasoning and problem formulation in information processing. Our approach suggests much more powerful processing elements, equivalent to several thousand neuron-like processing elements. Therefore our system does not need a great number of processing elements, since they are more powerful. Obviously the approach with powerful processing elements has lower degree of parallelism than the special purpose neuron-like network.

The hardware for proposed neural network based parallel expert system is a highly capable set of interconnected processors and memory modules. The array of transputers is available technology which enables a configuration of a distributed parallel computer system with neural network properties and enables learning using a sample set of learning situations.

The T800 Transputer is proposed as a processing element which incorporate a processor and a local memory. It has 4 links to the neighboring elements. A local memory is used for storing a value of adaptive coefficient (operator), important for neural network learning and fast processing. The introduction of layer-groups of processing elements which all have the same transfer function is usually not necessary because of the powerful processor. If the processing speed should be increased, a multilayer network of transputers would have to be used. The hardware frame would stay unchanged, only the software distribution to different processors would have to be modified.

5. EXPERT SYSTEM

The task of the expert system is to extract stored information or knowledge. The representation of knowledge in a neural network system was studied by Kohonen (4), using a distributed associative memory model.

Many expert systems have to perform a set of feature extractions in order to reach the goal which might be object recognition, medical diagnostics or others. A framework for those expert systems is a knowledge network. Each node in a knowledge network is reached by a feature value V(i,j) which is obtained by a feature extraction. The leave node of the network is reached by the some set of feature values. This set of feature values represents a symbolic description of an object.

KNOWLEDGE NETWORK   LEAVES

V(k) is a set of possible feature values at
   node k in a knowledge network
V(k,j) is the feature value which connects
   node k to node j
V(k) = (V(k,j))   j=1,...,m

Fig.1.: The nowledge network is organized in a
tree   structure   with   feature   values   as
connections.

Each node of a knowledge network consists
of a set of knowledge atoms. The initial node
n(0) contains all knowledge atoms which
correspond to completely unidentified object.
All leave nodes consist of only one knowledge
atom O(i) which corresponds to a completely
identified object.

Nodes   are   processed   in   different
processors. In a large knowledge network it is
very likely that the number of processors is
going to be smaller than the number of knowledge
atoms. Therefore we perform a so called layer
distribution which means that only a set of
knowledge atoms branching from a father
knowledge atom are stored and processed in a
single processor. An example of a knowledge
network from fig 1. would be processed in the
following processors:

PROCESSOR	KNOWLEDGE ATOMS
1st	n(0), n(1), n(i+1),..., n(i+1+1),...
2nd	n(2), n(i+2),..., n(i+1+2),...
. .	
:	

other processors

The next step is required when the number
of son knowledge atoms from a particular father
knowledge atom is bigger than the available
number of processors in a neural network. This
is also very likely if we work with a standard
parallel machine having from 20 to 500 powerful
processors. Much work in parallel processing is
devoted to acquire available free processors for
parallel work. Different strategies have been
proposed which all require some central
supervisor or schedular.

6. LEARNING

The weights of interconnections are
variables and have to (self)adapt to such values
which determine the most probable path for fast
and correct identification. Huge expert systems
would be slow at the initial phase of learning
because of nonoptimal search path, but would
gradually became very efficient because of
learning.

In a learning process we distinguish the
following:

- learn to perform a job faster
- learn to enlarge a knowledge base or a problem
  domain.

These two types of learning require different
strategies.

6.1 Strategy learning

Learning in a neural network system is
performed by a set of learning examples. The
resulting output is compared with the desired
output. A self learning in the neural networks
is achieved by using a number of methods which
are recommended for different applications (5).
For self learning of an expert system we are
proposing another method which is based on
experience of the neural network system.

Learning in order to perform a task faster
requires only modification of probability
coefficients c(i) in the local memories of
individual processors. The coefficient c(i) is a
pointer to different feature extractors in the
network. At each knowledge atom only those
coefficients c(i) exist which represent some
object. For instance if a knowledge atom n(i)
has connections to knowledge atoms n(k) and
n(k+1), than only two coefficients c(k) and
c(k+1) will exist in the knowledge atom n(i).
Each processor performs only that feature
extractor for which the coefficient c(i) has the
highest value.

In the begining the probability
coefficients are equivalent to c(i) = 1. After a
feature V(i) is identified by a processor i, the
new probability coefficient c(i) equals

$$c(i) = c(i) \begin{cases} c(i)/2 & \text{if } c(i) > 1 \quad \text{(forgetting func.)} \\ c(i) & \text{if } c(i) = 1 \\ & \text{if } (c(i) = c(max) \text{ and} \\ & \quad V(i) \text{ was selected)} \\ c(i)/2+1 & \text{if } V(i) \text{ was selected} \quad \text{(learning func.)} \end{cases}$$
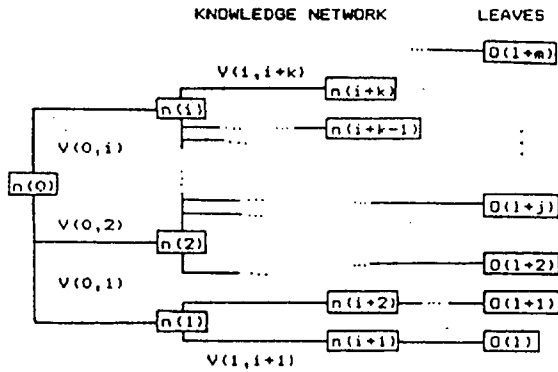
By this algorithm the changing of a
strategy is performed very fast, coefficients
are stored in local memory and a change of the
strategy is not influenced by stohastic events.
The forgetting function and the learning
function are important for selfadaptability of
the network.

6.2. Knowledge learning

Learning of a new knowledge or enlarging a
problem domain involves much more complex
procedure. This is:

- formulating the new knowledge,
- formulating the new feature values,
- establishing connections in the recognition
  tree,
- distributing the process to processors for
  parallel processing.

Several different applications as for
example simulation of complex socio-economic
systems, simulation of ecologic processes in
natural environment, computer vision, speech
recognition and some others are easier to
formulate as a neural network based parallel
expert system. The nature of all these
applications is parallelism and complexity.
Neural network approach enables easy
distribution of the problem to a parallel
computer system, enables step by step approach
in defining and adding details of a model,
gaining in that way the exactness as well as

accuracy of the model and not loosing the applicability and a speed of processing. We are considering simulation of other complex biological systems with special interest in learning, self organization, memorizing and fault tolerance.

7. CONCLUSION

Implementation of the neural network with standard microcomputer components has good price performance ratio. The capability of reformulating or upgrading the functions is much higher because neural network principles are implemented in software rather than in hardware. We have proposed a neural network based parallel expert system on transputer processors and described a problem parallelization and self learning function, based on the local coefficient adaptability.

The limitation of our proposal is problem decomposition which might be complex for some sophisticated expert systems. However, we think that a computer supported interactive decomposition have to be developed in order to facilite knowledge network decomposition.

Having in mind the future trends in parallel processing

8. REFERENCES

(1) H. P. Graf et al., "VLSI Implementation of a Neural Network Model", Computer, IEEE Computer Society, March 1988, pp 41-49.

(2) R. Hecht-Nielsen, "Neurocomputing: Picking the Human Brain", IEEE Spectrum, March 1988, pp 36-41.

(3) J. Hutchinson et al., "Computer Motion Using Analog and Binary Resistive Networks", Computer, IEEE Computer Society, March 1988, pp 52-63.

(4) T. Kohonen, "Associative Memories and Representations of Knowledge as Internal States in Distributed Systems", European Seminar on Neural Computing, London, 1988.

(5) P. Treleaven, "Parallel architectures for neurocomputers", European Seminar on Neurocomputing, London, 1988.