MECHANIKA TEORETYCZNA
I STOSOWANA
Journal of Theoretical
and Applied Mechanics
3, 31, 1993

NEURAL NETWORKS IN ROBOTICS

TERESA ZIELIŃSKA

Institute of Aircraft Engineering and Applied Mechanics
Warsaw University of Technology

The essence of the development of technical sciences is the search for new theoretical and practical solutions. Currently analitical solutions to technical problems are substitited by applications of neural networks. The paper describes the utilisation of neural networks in robotics. Neural networks are applied to robot control systems, especially to sensor data processing and solution of kinematics and dynamics problems. Different types of networks as well as the features of the learning process (e.g. types of patterns used, learning phases) are described. The paper is concluded by the discussion of difficulties in the selection of neural network, learning algorithms and network parametrs.

1. Introduction

The human brain contains over 10^{11} neurons. Each neuron possesses tree-like structures (called *dendrites*) specialised to receive incoming signals from other neurons or receptors. Neurons junctions are called *synapses*. A single output fiber called *axon* transmitts activation value of the neuron to many other neurons or effectors (with a speed of about $1 \div 100$ m/s). A neuron has more than 1000 synapses on its input and output. There are different kinds of synapses. Some of them are *stimulatory* (or *activatory*), and others are *inhibitory*. The latter produce in the dendrites the electrical signals of reverse polarity; when such signals arrive they raise the excitation threshold of a neuron.

Neurons and connections between them create a neural network. Due to specific signal changes in the neural network (and changes of the network's structure) the behaviour of the organism at a given instant depends not only on the received stimulus but also on the earlier acquired experience.

Models of neurons and results of investigations of brain functioning are a useful clue for the development of artificial neural networks.

Alexander (1991) gives the following definition of artificial neural networks (neural computing)

- "Neural computing is the study of cellular networks that have a natural propensity for storing experimental knowledge. Such systems bear a resemblance to the brain in the sense that knowledge is acquired through training rather than programming and is retained due the changes in node functions."

2. Models of neural networks

The McCulloch and Pitts (Mikhailov, 1990) model of the neuron (1943) is a basis for more neural node models.

This model specifies the state of a neuron i by a binary variable s_i which assumes the value 1 if this neuron is active within a given interval of time and the value 0 if it is passive. A synaptic connection is characterized by its weight G_{ij}. The absolute value of G_{ij} specifies the strength of a connection between neuron i and j. The positive value of G_{ij} is for activatory synapse and negative for inhibitory. If $G_{ij} = 0$, a connection between neurons i and j is absent.

The potential which is induced on a given neuron i by all other neurons j in the network is equal to

$$U_i = \sum_j G_{ij} s_j \qquad (i \neq j) \tag{2.1}$$

If a bias (threshold) B_i of a neuron i is introduced, neuron will become active when $U_i \geq B_i$ and passive when $U_i < B_i$

$$s_i = H(\sum_j G_{ij} s_j - B_i) \qquad (i \neq j) \tag{2.2}$$

where $H(\cdot)$ is the step function

$$H(x) = \begin{cases} 1 & \text{for} \quad x \geq 0 \\ 0 & \text{for} \quad x < 0 \end{cases}$$

By using the states of neurons at a given time step n, we find the new states of all neurons i at the next discrete step $n+1$

$$s_i^{n+1} = H(\sum_j G_{ij} s_j^n - B_i) \qquad (i \neq j) \tag{2.3}$$

Despite its simple definition this model of a neural network has extremely rich behavior.

Actual state of the art in the field of artificial neural networks has embraced a very broad scope of neural processing models (e.g. learning and correlation matrix linear models, cooperative-competitive neural networks modes, associative memory, delta learning for generalized information storage).

3. Neural networks in robotics

A computational study reveals that the central nervous system must solve the following three problems (Kawato et al., 1989)

- determination of a desired trajectory in the visual coordinates (coordinates of the image created in the eye),
- transformation of the visual coordinates of the desired trajectory into the body coordinates,
- generation of motor commands.

In the functional structure of the robot control system a hierarchy of levels executing similar to the above mentioned actions can be distinguished

- environments recognition, task planning and actions leading to its execution; in this case information gathered by sensors (e.g. vision system) are necessary,
- determination of motion trajectories; during the determination of consecutive trajectory points information from sensors (e.g. tactile sensors) can be taken into account,
- generation of motion commands; in typical robot feedback from current position (or velocity, acceleration) of each degree of freedom, this information is supplied by position sensors (e.g. encoders).

Neural networks have been used on each of the above mentioned levels of control structure replacing traditional analytical methods.

Taking into account the type of the utilized information, the application of neural networks in robotics should be considered in two aspects

- utilization of networks in finding solutions to robot kinematics and dynamics problems,
- utilization of networks in processing data obtained by sensors.

In the first case an analytical kinematics/dynamics model of the robot is substituted by an adequately trained neural network. The "profit" here arises from an increase of computation speed. Computations usually executed by neural networks are only simple additions and multiplications.

Another advantage of applying networks is that construction of the model (kinematical and/or dynamical) is avoided. When the neural network learns it utilizes the real system, so usually the results obtained from properly trained networks are more accurate than the results obtained by utilizing models.

Of great interest are the problems of application of neural networks to processing information by sensors.

Sensors typically provide signals that are both incomplete and ambiguous. To solve this problem we have two options. First, to try to make perfect sensors. Second, to make effective use of several sensors, so allowing individual sensors to complement each other in a graceful and integrated fashion.

In both cases the problem of interpretation of information supplied by sensors remains to be solved. This issue is of great importance in robotics. Brady (Grant, 1991) defined robotics as the "*intelligent connection of perception to action*".

In the simplest case networks are used in the "recognition" of information supplied by sensors, e.g. pattern recognition and object identification by vision systems. In such an application networks execute actions assigned to environment recognition layer of the control structure. In a complex neural controller, the network can take over all of the functions of the controllers, that is on the basis of information obtained from sensors it can directly generate commands controlling the motions of all the degrees of freedom of the manipulator.

The subsequent part of the paper will be dedicated to a more detailed discussion of neural network applications.

4. Solution to kinematics/dynamics problems; co-ordinate transformations

In a robot controller, the spatial coordinates of the desired trajectory must be interpreted in terms of a corresponding set of body coordinates, such as joint angles. Motor command (e.g. torque) must be generated to coordinate the activity of motors so that the desired trajectory is realized.

Examples of application of neural networks to solving inverse kinematic problem (transformation of external coordinates into generalized coordinates) usually refer to redundant robots, for which it is difficult to find analytical solutions. Topographically mapped terminal attractors are used here to define a neural network whose synaptic elements can encapture the inverse kinematic information using a priori generated examples and, subsequently generalize to compute the joint-space coordinates required to achieve arbitrary end-effector configuration (cf Barhen and Gulati, 1991; Macukow, 1992).

In this case a neural network learns the functional mapping from the N_x-dimensional end-effector space to the N_q-dimensional joint space of the manipulator. The network is presented with k training pairs of end-effector and joint-space variables obtained from the forward kinematic relations. The goal of network learning is to determine the synaptic interconnection strengths that can correctly encapture the inverse kinematic mapping, imbedded within the training samples.

The algorithm of learning enforces the convergence of every neural state in two sets of attractors corresponding to the presented end-effector task coordinates and joint coordinates respectively. The minimized energy function is the measure of accurate coordinate transformation.

Several neuromorphic algorithms have been proposed, e.g. Hopfield network Hecht-Nielsen (1991), Pao (1989), Platt and Barr's basic differential multiplier method by Barhen and Gulati (1991). These methods differ by relationships describing the iterative process of estimation of synaptic coupling between neurons so that the energy function is minimized.

Such application of neural networks entails the generation of joint angles with hight degree of precision, error tolerances can be less than 0.05% (Barhen and Gulati, 1991).

Neural network substituting the dynamic model learns to generate the motor control signals (torques, forces) so that the robot may be driven to follow the trajectory (cf Kung and Hwang, 1989; Kawato et al., 1989).

In this case neural networks are usually utilized in elastic manipulators, where it is difficult to define a model fully describing the dynamic properties of the system.

The goal of neural network learning is to minimize the errors between the desired and actual motor command. In the first stage of learning neural controller receives as input the set of resulting trajectories and yields a set of reference motor commands.

In the case of ambiguities in the solution to the inverse dynamical problem a given point on the trajectory of gripper motion can be attained with different values of generalized coordinates (i.e. with different motor control signals).

The selection of improper values of generalized coordinates renders impossible the realization of continuous motion trajectory. To avoid that a second stage of learning takes place. In this stage the neural network learns during robot motion – consecutive motor commands have to be determined in such way that the trajectory will be realized. Error-feedback informing about the difference between the current and goal positions, is utilized in this case. Methods used in network learning belong to a group of methods called supervised learning or back-propagation of error.

The learning phase performs the iterative updating of the synaptic weights for all the network connections. The weights are updated according to the responses generated in the outputs of the network. The weights updating, which adapts the mechanism of back-propagated corrective signals from the network outputs, is the essence of learning algorithms of the supervised learning type.

The most frequently used algorithm of this type is the generalized delta rule elaborated by Rumelhart (cf Pao, 1989; Zeidenberg, 1990; Hecht-Nielsen, 1991).

5. Sensor processing

When a robot is recognizing its environment it is essential for it to identify the obstacle locations in the workspace.

Traditionally, the sonar sensors have been very popular for this purpose. The stereo matching techniques are more reliable, but a computationally costly alternative.

Pattern recognition is chronologically the oldest example of neural network application to processing data obtained by sensors (Miyake and Fukushima, 1989). This domain is closer to pattern recognition theory than robotics. The problem of automatically recognizing characters in printed or hand-written material has been studied for decades. In robotics mainly mapping networks for compression of analog vector data (Hecht-Nielsen, 1991) are utilized.

The stereo vision recovers 3-D depth information from two images taken from two cameras. The main task in stereo vision is the correspondence problem – the determination of the valid matches of the feature primitives of the two images (Kung and Hwang, 1989). The Hopfield optimization network or Bolzman machine are used to solve the above problem (Hecht-Nielsen, 1991). In this case individual pixels from two images are compared. The aim of this comparison is the identification of pixels which represent the same 3-D point on the object. The difference of position of these pixels in the images is used to compute the distance of the point from the cameras (Kung and Hwang, 1989). When neural networks are applied each gray level is treated as an activation value of one neuron. The selection of pixel pairs which represent the same point in space consists in minimization of adequately defined function of network energy (cf Kung and Hwang, 1989; Zhongquan Wu, 1990).

Application of neural networks to the above described tasks (pattern recognition, stereo vision) substitutes traditional analytical methods used by environment recognition level in the control structure.

Neural networks are used to scaling sensors. In the previous paper (1992) application of a neural network to measurements of shear forces by a sensor, which can be mounted in a robot gripper, was described. Force measurements are done on the basis of recording displacements of the so called sensing elements. These displacements result from the action of shear forces.

To determine the relationship between the displacement of each sensing element and the applied shear force requires precise measuring instruments or needs the identification of complex analytical relationships. It was much easier to measure the resulting shear force exerted in an object placed on the sensor surface using a neural network. In the measurement of shear force a multi-input neural network was utilized. Each input was associated with the value of displacement of each sensing element. The network had two outputs – one related to the F_x component and other to the F_y component of the shear force. The generalized delta rule algorithm with back propagation of error (Rumelhart's algorithm, cf Hecht-Nielsen (1991), Pao (1989)) was utilized in teaching the neural network.

The neural networks can learn to determine the assembly actions on the basis of the information received from the sensors. Works on such applications of neural networks are in the initial stage.

Kuperstein (1991) describes a robot neural controller, which learns sensory-motor coordination from its own experience. It learns to accurately grasp an elongated object with almost no information about the geometry of the physical sensory-motor system. The controller called INFANT was implemented in a robot with stereo vision. The controller operates in two phases.

In the first phase, sensory-motor relations are learned via correlations between object sensation and self produced movement signals. In the second phase, the system uses learned correlations to transform the sensation of an object into a movement that reaches the object. During learning, a random movement generator first produces random postures of the multijoint arm in space, while the gripper holds an object such as a cylinder. Then the stereo cameras snap an image of the arm holding the object. These images are transformed into neural input maps (left and right visual map). A third input map (stereo map) is produced as a disparity between the left and right visual map. The signals coming from the input maps are modulated by a set of weight signals to produce target signals. The weights constitute the global association between all possible images of an object and the arm motor signals.

The above approach to robot control appears to be most promising for addressing the problem of adaptive control in uncertain environment. The neural network assumes all the functions of a traditional robot control system.

6. Other applications of neural networks

The problem of robot calibration is the identification of the non-geometrical errors (gear transmission errors, link and joint flexibility etc.). Modelling of these errors is a very complex and difficult process. There are only few papers dealing with this problem using traditional methods. Traditional methods, however, do not give satisfactory results as the quality of error identification strongly depends on the presumed structure of the model.

The Rumelhart's neural network learned to identify residual error function containing all non-geometrical effects (Renders et al., 1992). After teaching of the network the positioning error was reduced up to 20% of the initial error.

Neural networks are frequently used as elements of control systems, which solve optimal control problems. In planning optimal control a cost function must be defined. It is a function of several important variables, e.g., the distance between the current location and the starting position of the robot end-effector or the distance between the current location and the goal, etc. The network will find an optimal path (e.g. time-optimal) to the target position (Kung and Hwang, 1989).

A minimum torque-change model was proposed by Kawato et al. (1989) as a model, which predicts a wide class of trajectories in human multi-joint arm movements.

The same model was utilized for generation of robot motor commands by a neural network (for PUMA robot).

7. Conclusion

The basic computations in the control of robot manipulators are associated with kinematics, dynamics and interpretation of sensory information (which includes optimal path-planning).

Due to its robustness and adaptiveness neural networks can be very useful to all levels of these robotic applications.

When a neural network is to be applied to the execution of a certain task, three basic problems have to be solved

- appropriate structure of the network has to be found,
- effective learning algorithm has to be selected,
- learning process must to be correctly carried out.

Up till now no general guidelines satisfactorily solving the above problems have been elaborated. However there are publications trying to tackle the subject (e.g. description of neural network architecture utilized in robotics, Kung and Hwang (1989)). The number of applications of the suggested rules is to small to forejudge if the rules are always valid.

In the choice of architecture and learning algorithm and during network learning process scientific guess and intuition are the only guidelines. Regardless of these problems, recently a rapid increase in neural network research has been noted.

A survey of recent robotics research shows that neural networks are mainly used in adaptive neural controllers utilizing information obtained from sensors to generate motion commands.

Todays VLSI (Very Large Scale Integration circuits) and CAD (Computer Aided Design) technologies facilitate practical and cost-effective implementation of large-scale computing networks.

Acknowledgments

This work was partially supported by Polish Committy of Scientific Research grant no. 330019203 and Warsaw University of Technology grant no. 503/721/2. The author gratefully acknowledges both sources of funding.

8. References

1. Aleksander I., 1991, *Practical Neural Computing for Robots: Prospects for Real-Time Operation,* Sensor Based Robots: Algorithm and Architectures, Ed. by C.S.George Lee, NATO ASI Series, 66
2. Barhen J., Gulati S., 1991, *Self-Organizing Neuromorphic Architecture for Manipulator Inverse Kinematics,* Sensor Based Robots: Algorithm and Architectures, Ed. by C.S.George Lee, NATO ASI Series, 66
3. Grant E., 1991, *Uncertainty in Robot Sensing,* Sensor Based Robots: Algorithm and Architectures, Ed. by C.S.George Lee, NATO ASI Series, 66
4. Hecht-Nielsen R., 1991, *Neurocomputing,* Addison-Wesley
5. Kawato M., Isobe M., Suzuki R., 1989, *Hierarchical Learning of Voluntary Movement by Cerebellum and Sensory Association Cortex,* Dynamic Iteractions in Neural Networks: Models and Data, Ed. by Michael A.Arbib, Shun-ichi Amari, Springer-Verlag
6. Kung S.-Y., Hwang J.-N., 1989, *Neural Network Architectures for Robotic Applications,* IEEE Transactions on Robotics and Automation 5, 5, 641-657
7. Kuperstein M., 1991, *INFANT Neural Controller for Adaptive Sensory-Motor Coordination,* Neural Networks, 4, 131-145
8. Macukow B., 1992, *Robot Control using Neural Networks,* Robotics Research and Applications, Ed. by Morecki A., Muszynski W., Tchon K., Warszawa
9. Mikhailov A.S., 1990, *Foundations of Synergetics. Distributed Active Systems,* Springer-Verlag
10. Miyake S., Fukushima K., 1989, *Self-Organizing Neural Network with the Mechanism of Feedback Information Processing,* Dynamic Interactions in Neural Networks: Models and Data, Ed. by Michael A.Arbib, Shun-ichi Amari, Springer-Verlag
11. Pao Y.-H., 1989, *Adaptive Pattern Recognition and Neural Networks,* Addison-Wesley
12. Renders J.M., Millan J.R., Becquet M., 1992, *Non-Geometrical Parameters Identification for Robot Kinematic Calibration by Use of Neural Network Techniques,* Robotic Systems: Advanced Techniques and Applications, Ed. by Spyros G.Tzafestas, Kluwer Academic Publishers
13. Zeidenberg M., 1990, *Neural Network Models In Artificial Intelligence,* Ellis Horwood Limited
14. Zhongquan Wu, 1990, *Optical Flow Estimation by Using Artificial Neural Network under Multi-Layers,* 2nd International Conference of Artificial Networks, 76-80
15. Zielińska T., 1992, *Tactile Sensing for Flexible Manufacturing,* GR/G64084 report, Loughborough Univ. of Technology

Sieci neuronowe w robotyce

Streszczenie

Ostatnie lata charakteryzują się gwałtownym wzrostem zainteresowania sztucznymi sieciami neuronowymi. Idea tych sieci wywodzi się z rezultatów badań nad mózgiem i budową oraz funkcjonowaniem pojedyńczego neuronu.

Zafascynowanie sztucznymi sieciami neuronowymi widoczne jest w robotyce. Stosuje się tutaj sieci neuronowe wykorzystując ich zdolności adaptacyjne oraz możliwości odwzorowywania, nie dających się analitycznie opisać prawidłowości.

Na wstępie przedstawiono *prosty model neuronu*, będący aktualnie podstawą do budowy wielu sztucznych sieci neuronowych.

Zastosowanie sieci w robotyce rozpatrzono w dwu aspektach:

- wykorzystania sieci do rozwiązywania zadań kinematyki/dynamiki robotów,
- wykorzystania sieci do przetwarzania informacji pochodzącej z czujników.

W pierwszym przypadku analityczny model kinematyki/dynamiki robota zastępowany jest odpowiednio nauczoną siecią neuronową.

W drugim zastosowaniu sieci służą do "rozpoznawania" informacji pochodzącej z czujników, np. do rozpoznawania kształtów oraz identyfikowania przedmiotów zaobserwowanych przez system wizyjny. W skomplikowanym sterowniku neuronowym sieć może przejąć wszystkie funkcje układu sterującego tzn. na podstawie informacji odebranej z czujników może bezpośrednio generować rozkazy sterujące ruchem poszczególnych stopni swobody robota.

Omówiono podstawowe zagadnienia związane z powyższymi zastosowaniami sieci neuronowych jak np. typ stosowanych sieci, strategie uczenia, zakres wykonywanych przez sieci działań.

Zwrócono uwagę na brak jednoznacznych wskazówek dotyczących wyboru struktur oraz algorytmów uczenia sieci neuronowych co jest istotnym utrudnieniem przy ich stosowaniu.

Dobrze nauczona sieć neuronowa może realizować tak skomplikowane funkcje, że opłaca się ponosić trud wyboru właściwej sieci oraz jej trenowania.

Przegląd ostatnich prac z zakresu robotyki wykazuje ukierunkowanie badań na zastosowanie sieci w adaptacyjnych sterownikach neuronowych wykorzystujących informacje pochodzące z czujników do generowania rozkazów ruchu. Takie zastosowanie sieci, bliskie istocie funkcjonowania organizmów żywych, otwiera pole do interesujących badań i zdaniem autora ma ono największą przyszłość.

Manuscript received March 4, 1993; accepted for print March 15, 1993