H. Piech*, M. Spiewak,**
* Institute of Mathematics and Computer Science,
Czestochowa University of Technology
** Institute of Machines Technology and Production Automation,
Czestochowa University

The Neural Control of a Robot in the Conditions of Movable Obstacles

The proposed concept of robot control assisting uses a neural network, whose operation relies on the activation of neurons delimiting a path from the source to the target with evading movable obstacles The complexity of the control algorithm is O (n). The proposed adjustment of neuron sensitivity using a two-element pencils of planes passing over the shortest path of the robot makes it possible to obtain a set of solutions with simultaneous classification in terms of a very important path length criterion.

Предложена концепция сопровождения управления роботом с использованием нейронной сети, работа которой основана на активизации нейронов, определяющих путь от исходной точки до цели с уклонением от подвижных препятствий. Сложность алгоритма управления составляет O (n). Предложенная настройка нейронной чувствительности с использованием двухэлементных пучков плоскостей, пересекающих кратчайший путь робота, позволяет получить множество решений с одновременной классификацией по критерию длины пути.

K e y w o r d s: robot control, neural networks, neuron activation thresholds.

The utilization of the neural structure for control is a unified control system enabling the dynamic and easy determination of weights and thresholds. The effectiveness of using different methods of aggregation can be analyzed. The model for establishing the values of weights and activation thresholds of aggregating neurons relies on the position of the robot and the location of the line connecting the robot's starting and target points (Fig. 1).

The prediction obstacle position at the moment of robot approach enables this situation to be allowed for in the threshold forming planes (Fig.2). The purpose of the study is to examine the algorithmic possibilities of the realization of the control model with the use of a neural structure of adjustable neuron activation levels.

The mechanism of formation of neuron activation thresholds. The value of neuron activation can be defined using two planes, $S1$, $S2$, which form a pencil (have the common edge PK). The equations of the two planes (Fig. 3) can be defined in an intercept form: $\frac{x}{a}+\frac{y}{b}+\frac{z}{c}=1$. The calculation of the values of a, b, c

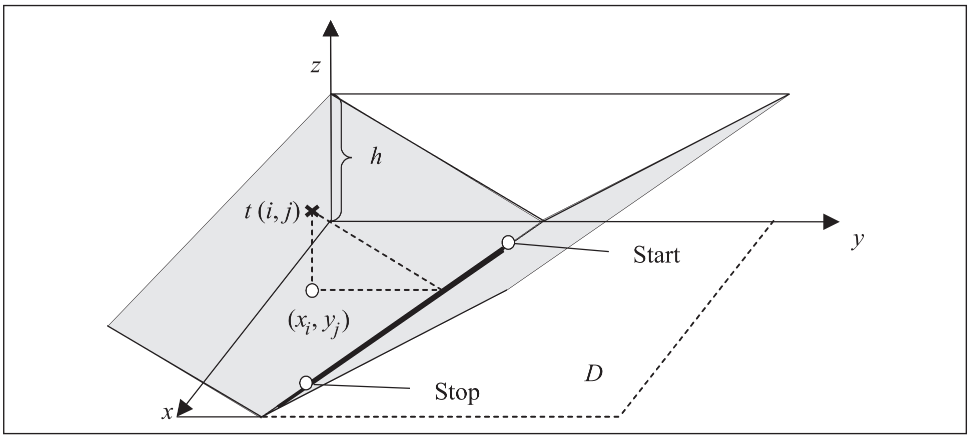

Fig. 1. Graphical illustration of the method of determining the neuron activation threshold $t(i,j)$: h—quantity controlling the activation threshold levels; D — area available for robot peregrination

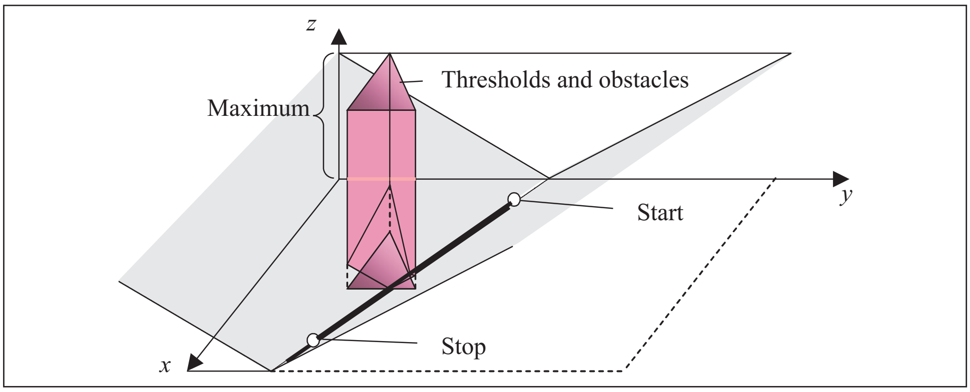

Fig. 2. Taking account of obstacle prediction in the formation of thresholds (maximum)

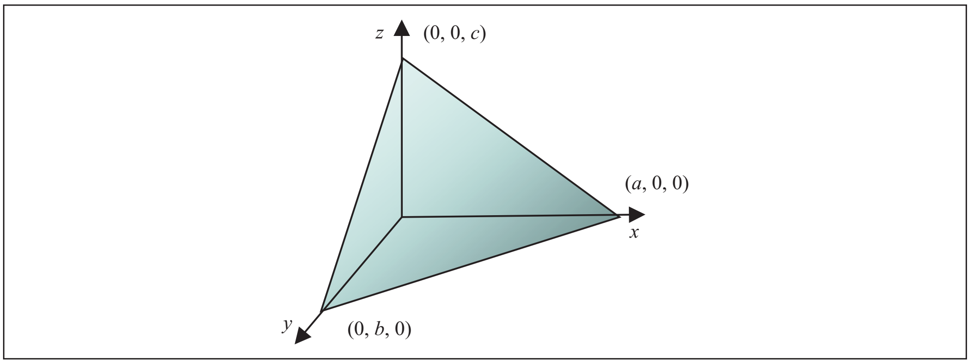

Fig. 3. Plane in an intercept form

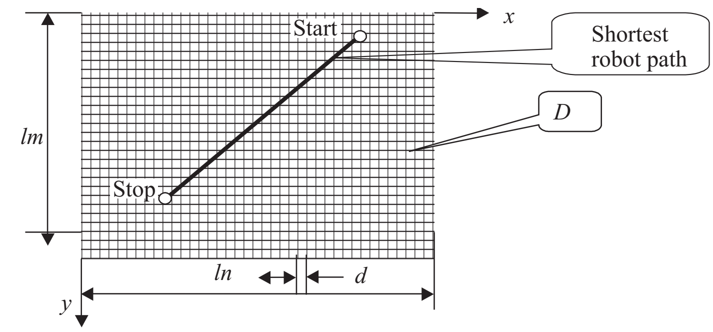

Fig. 4. Graphical illustration of the location of neurons for the determination of their activation thresholds: $x = i \cdot d, 1 \le i \le ln$; $y = j \cdot d; 1 \le i \le lm$; *ln* is a number of neurons in a layer; *lm* is a number of layers

will be carried out in the simplest possible way by using the equation of the straight line connecting the starting and the end points, *P*, *K*, of the robot's route. The equation of this straight line is written as follows:

$$y - y(P) = \frac{y(K) - y(P)}{x(K) - x(P)}(x - x(P)).$$

The determination of the value of the coefficient *a* only requires the substitution of the value of $y = 0$

$$a = x(P) - \frac{x(K) - x(P)}{y(K) - y(P)} y(P), \tag{1}$$

whereas for the calculation of *b*, we substitute $x = 0$ to obtain:

$$b = y(P) - \frac{y(K) - y(P)}{x(K) - x(P)} x(P),$$

while the value $c = h$. Ultimately, the intercept equation of the plane passing through two points, *P* and *K*, will have the form of

$$\frac{x}{x(P) - \frac{x(K) - x(P)}{y(K) - y(P)} y(P)} + \frac{y}{y(P) - \frac{y(K) - y(P)}{x(K) - x(P)} x(P)} + \frac{z}{h} = 1.$$

So, to calculate the threshold value, we can use the following equation:

$$z = h - \frac{xh}{x(P) - \frac{x(K) - x(P)}{y(K) - y(P)} y(P)} + \frac{yh}{y(P) - \frac{y(K) - y(P)}{x(K) - x(P)} x(P)},$$

where $x = i \cdot d$, $y = j \cdot d$, *d* is a grid size.

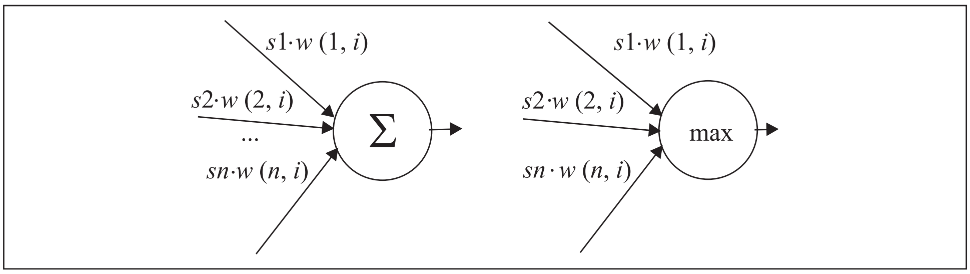

Fig. 5. Selected types of aggregation that can be employed to the «neural simulation» of the robot working space

Relating the neuron location to the coordinates x and y is not complicated, i.e. it is sufficient to perform some simple recalculations that are related to the situation shown in Fig. 4.

The simulation of neural structure operation requires the methods of aggregation of signals coming to each neuron to be specified more precisely. These can be the operations of summation of input signals or mini-max operations (Fig. 5).

Inter-layer communication. The number of neural network layers lm determines the number of iterations of inter-layer transfers, which will be equal to lm – 1. The algorithm for the operation of the neural structure can be represented in the form of a block diagram, as in Fig. 6.

Algorithm description:

0. — data input: number of layers (lm); number of neuron in a layer (ln); input signals (s(i)); beginning and end of the route (coordinates x, y), respectively; parameter of neuron response sensitivity (h);

1. 2. 3. — determination of the values of neuron activation thresholds (z (i,j), where i is the number of the layer, j is the number of the node. We use equation (2);

4. 5. 6. — determination of (or correction to) the values of weights, for example depending on the velocity of approach of obstacles (the proposal will be presented in the next section);

7. 8. 9. 10. 11. 12. 13. — simulation of neural network operation:

7. — transition to subsequent layers;

8. — location of the neuron sending signals (the k-th layer);

9. — location of the neuron receiving signals (the k + 1 layer);

10. — aggregation of signals (summation is chosen);,

11. — condition for the activation of the neuron of the coordinates j, k + 1;

12. 13. — result of neuron activation;

14. 15. — transfer of the signal to the subsequent layer.

The algorithm can be implemented in either a static (one-off transition: forward-propagation) or a dynamic (change in the robot and obstacles position with

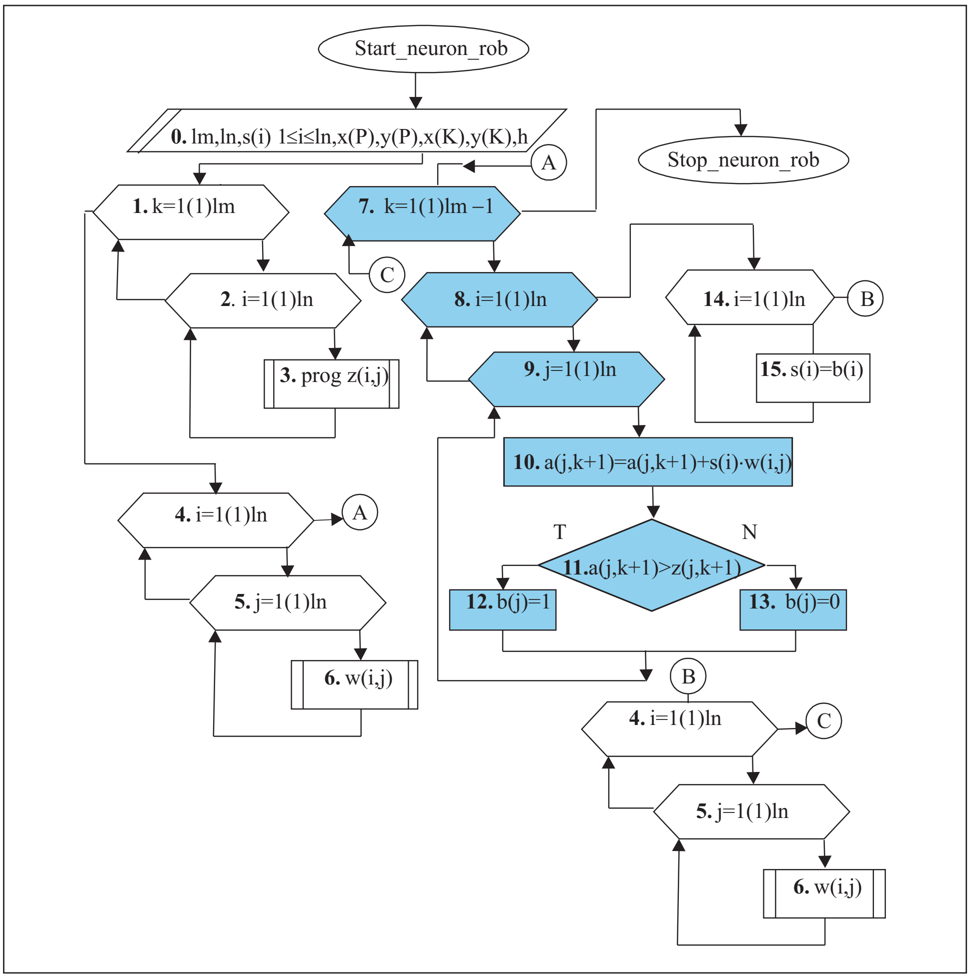

Fig. 6. Algorithm of the inter-layer communication of the neural network indicating the robot path: k is a number of the layer from which the transferred signal originates; i is a number of the neuron transferring signals; j is a number of the neuron receiving signals; a (j, k) is a size of the aggregated signal in the neuron of the number j of the k-th layer

each «turnaround»: back-propagation) variant. The variant with turnarounds will also require correction to the weights prior to each return. In the static variant, we will use the algorithm of the prediction of obstacle position at the time of the robot approaching to the obstacle.

Determination of the values of weights in respect to the relative position and velocity of motion of the obstacle in relation to the robot. The basis for the determination of the threat resulting from the possibility of a collision is the

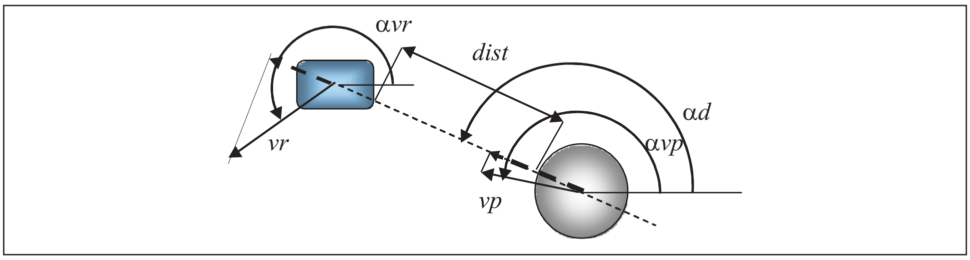

Fig. 7. Relative position of the robot and the obstacle

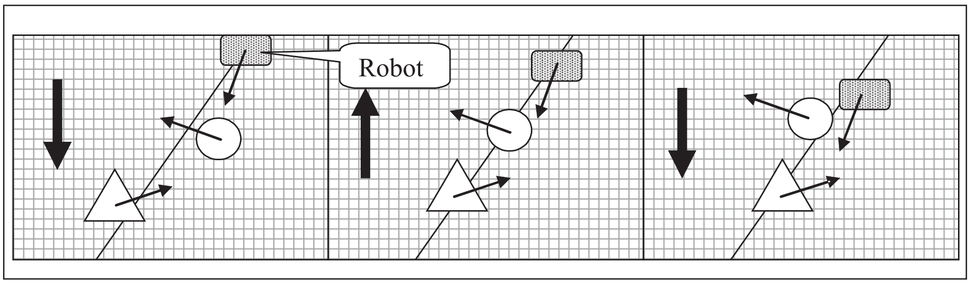

Fig. 8. Corrections to robot and obstacles positions associated with the change in the direction of propagation

relativization of the situation existing between the robot and the obstacle concerning their relative position and the rate of its change (Fig. 7).

The safe distance between the robot and the obstacle results from the satisfying of the following condition:

$$dist = (Or, Op) - Rr - Rp > 0,$$

where $(Or,\ Op)$ is a distance between the robot and obstacle centres $(\sqrt{(xOr - xOp)^2 + (yOr - yOp)^2})$; Rr — maximum size of the robot, as counted from its center; Rp — maximum size of the obstacle, as counted from its centre.

The velocity of obstacle and robot approach can be estimated as follows:

$$vpr = vr \cdot \cos(\alpha vr - \alpha d) - vp \cdot \cos(\alpha vp - \alpha d);$$

$$tg\,(\alpha d) = (yOr - yOp)/(xOr - xOp)\,.$$

The reduction of the weights of signals coming to the neurons lying within the hazardous zone (between the robot and the obstacle) can be accomplished by presetting the number of reduction thresholds (on condition of satisfying the relationship $vrp \cdot vp < 0$: the robot and the obstacle approach each other);

$$w(i,j) = dist - \sqrt{(x - xOr)^2 + (y - yOr)^2})/dist, \text{ provide that } vrp \cdot vp < 0,$$

$$w'(i,j) = [dist - \sqrt{(x - xOr)^2 + (y - yOr)^2})/dist \cdot wsk] \cdot sck,$$

$$\text{provided that } vrp \cdot vp < 0,$$

where $w'(i,j)$ is a stepwise reduced weight of signals coming to the neuron (i, j); wsk is a number of weight reduction thresholds; sck is a weigh correction scale, vrp/wsk.

The effects of individual obstacles superimpose multiplicatively:

$$cw(i,j) = \prod_{k=1}^{lp} w_k(i,j),$$

where k— obstacle number; lp — number of obstacles. In the back-propagation algorithm, after reaching the end of the robot route, a turnaround takes place (the output signals from the last layer $lk \geq (y(K)/d)$ change the transfer direction and become input signals, and the layer numbers change in the reverse order, $k = lk(-1)lp,\ lp \leq (y(P)/d)$). Simultaneously with the velocity set for a single robot pass (through all important layers) we change the positions of the robot and the obstacles after each propagation (Fig. 8).

Conclusions. The neural control of a robot allows any changes in the robot movement environment to be taken into account flexibly and in a unified and easy-to-accomplish manner.

The proposals concerning threshold levels and weight values should be tested for each neural structure and the configuration of robot and obstacle movement parameters.

The neural solutions enable the implementation of a large number of concepts related to establishing the threshold and weight parameters. They reduce, at the same time, the complexity of geometrical-kinetic analysis.

The neural algorithm in the proposed form can be combined with the ant algorithm by setting the thresholds at levels allowing several solutions to be obtained.

Запропоновано концепцію супроводу управління роботом з використанням нейронної мережі, робота якої базується на активізації нейронів, що визначають шлях від вихідної точки до цілі з відхиленням від рухомих перешкод. Складність алгоритму управління складає $O(n)$. Запропоноване настроювання нейронної чутливості з використанням двоелементних пучків площин, що перетинають найкоротший шлях робота, дозволяє отримати велику кількість рішень з одночасною класифікацією за критерієм довжини шляху.

1. *Cormen T. H.* Wprowadzenie do algorytmow.— Warsaw: WNT, 1997.— 624 s.
2. *Dudeba I.* Metody i algorytmy planowania ruchu robotow mobilnych i manipulacyjnych. — Warsaw: Akademicka Oficyna Wydawnicza, EXIT, 2001.—311 s.
3. *Galicki M.* Wybrane metody planowania optymalnych trajektorii robotow manipulacyjnych. — Warsaw : WNT, 2000.—185 s.
4. *Goldberg D.* Algorytmy genetyczne. — Warsaw : WNT, 1995. — 342 s.
5. *Gwiazda T. D.* Algorytmy genetyczne. Zastosowania w finansach. — Warsaw: Wydawnictwo Wyzszej Szkoly Predsiebiorczosci i Zarzadzania im. L. Kozminski, 1998.—126 s.
6. *Jog P., Van Gucht D*. Parallelisation of probabilistic sequential search algorithms//Genetic algorithms and their applications. — 1987.— N1. — P.170 —176.
7. *Koziel S., Michalewicz Z.* Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization// Evolutionary computation. — 1999. —№.7 — P. 19—44.
8. *Osowski St.* Sieci neuronowe do przetwarzania informacji. — Warsaw: OWPW, 2000.— 365 s.
9. *Osowski St.* Sieci neuronowe w ujeciu neuronowym. — Warsaw: WNT, 1996. — 287 s.
10. *Siemiatkowska B.* Rastrowa reprezentacja otoczenia dla celow nawigacji autonomicznego robota mobilnego. — Warsaw : IPPT PAN, 1997. — 69 s.
11. *Syslo M., Deo N., Kowalik J.* Algorytmy optymalizacji dyskretnej. — Warsaw : PWN, 1995. — 426 s.
12. *Tadeusiewicz R.* Sieci neuronowe. — Warsaw: Akademicka Oficyna Wydawnicza, 1993. — 356 s.
13. *Wasserman P. D.* Neural Computing. Theory and Practice. — N. Y. : Von Nonstrand Reinhold, 1989. — 266 p.
14. *Zadeh L. A.* Rachunek ograniczen rozmytych. Projektowanie i systemy: Zagadnienia metodologiczne. — Warsaw: Ossolineum, 1980. — 345 s.
15. *Handbook* of genetic algorithms. Red. Davis Lawrence. — N. Y. : Van Nostrand Reinhold, 1991.— 412 s.

Поступила 12.03.07