Mobile Robot Path Planing Using Gauss Potential Functions and Neural Network

Kasac, J.; Brezak, D.; Majetic, D. & Novakovic, B.

Faculty of Mechanical Engineering and Naval Architecture
University of Zagreb, Zagreb, Croatia
Email: {josip.kasac, danko.brezak, dubravko.majetic, branko.novakovic}@fsb.hr

Abstract:

This work deals with the problem of potential field based mobile robot motion planning in unorganised environment. The new approach, using a combination of negative gradient and vortex field based on Gauss potential functions is proposed. Radial Basis Function Neural Network (RBF Neural Network) learns the dependence between Gauss function parameters and velocity of mobile robot (or relative velocity between robot and obstacle in dynamical environment) ensuring passage between two closely spaced obstacles and smooth path condition for different mobile robot initial conditions. This approach overcomes some standard problems in classical potential field methods like local minima avoidance, problems of no passage between closely spaced obstacles, strong variation of the repulsive force near the minimum distance, and avoidance of moving obstacles. The method is illustrated on the example of mobile robot navigation between several closely spaced obstacles and example of avoidance of moving obstacle.

Keywords:

Mobile Robot, Motion Planning, Potential Field, Radial Basis Function Neural Network, Real-Time Control.

1. Introduction

Potential field method is one of the most promising methods, widely used in applications for the mobile robot motion planning. The method can be used both for off-line global planning and for real time local planning in unorganised environment when sensors measure the presence of obstacles.

This method is conceptually simple, it can be easily implemented in a feedback form and also extended to moving obstacles. The basic concept of the potential field method is a creation of an artificial potential field in the robot workspace in which the robot is repulsed from obstacles and is attracted to its goal position (Khatib, 1986). The robot follows the gradient of this potential toward its minimum.

However, this method has some inherent limitations (Koren, 1991) which include the following: 1) possible existence of local minima for certain composition of target and obstacles; 2) no passage between closely spaced obstacles (e.g., passing through a doorframe); 3) oscillations in the presence of obstacles; and 4) oscillations in the narrow passages. There is also an additional problem concerning non-reachable goals with obstacles nearby (Ge & Cui, 2000).

Various extensions of classical potential field method aimed at overcoming the above problems, like generalised potential field method (Krogh, 1984), vector field histogram method (Borenstein & Koren, 1990), navigation function method (Rimon & Koditschek, 1992), etc. are proposed.

An elegant way for classical potential field method extension is the introduction of vortex field concept (De Medio & Oriolo, 1991). The basic idea of vortex field method is to replace the negative gradient of the given repulsive field with a circulatory field, which reorient the velocity of the robot guiding around the obstacle. The main advantage of this method is local minima avoidance and stable mobile robot movement without oscillations. However, this method has a serious drawback because it cannot guarantee the collision avoidance, especially when distance between the two obstacles becomes smaller.

In order to avoid this problem a combination of classical potential field and circulatory field is proposed (Khatib & Chatila, 1995), which guarantee a minimal distance between robot and obstacle. Another possibility is the circulatory field approach (Singh at al., 1996) motivated by a charged particle in a magnetic field generated by a current flowing around the obstacle.

Almost all previously mentioned methods use hyperboloidic repulsive potential functions (e.g., the inverse square law) for gradient and circulatory field. This choice of repulsive potential functions would ensure that a robot never penetrates the boundary at which the field goes to infinity. However, in practice the choice of the gain parameter of the repulsive potential function is very delicate. A small value of this parameter induces a strong variation of the repulsive force near the minimum distance and exceeds constraint on the velocity of the robot. Furthermore, it is possible that for greater velocity of the robot, controller produce extremely large force in very short time interval like a Dirac delta function, and response on this impulse excitation cannot guarantee the collision avoidance. On another side, a great value of the gain may over constrain the movement in place where the robot may

pass.
In order to overcome mentioned problems, in this work, Gauss function is proposed for a repulsive potential function instead of hyperboloidic function. In additional, Gauss function parameters depend on velocity of the mobile robot ensuring passage between closely spaced obstacles and minimal or close to minimal obstacle detouring path. Radial Basis Function (RBF) Neural Network approximates this dependence on the base of simulation results with the best avoidance properties. On the other side, bounded value of Gauss function near the minimum distance provide acceptable variation of the repulsive forces satisfying control constraints. Additional consequence of velocity dependent Gauss potential function is possibility of avoidance of moving obstacles.
Mobile robot model and a new potential field method are presented in section 2. In section 3, a model of Radial Basis Function Neural Network is shown, and in section 4 the results of simulation experiment are analysed. Finally, section 5 summarises the conclusions and indicates the future research directions.

2. Potential field based motion planning

A wheeled mobile robot in a two-dimensional flat plane is considered. The main goal is to lead the mobile robot from the initial configuration to the final one by avoiding obstacles.

2.1 Kinematical model of mobile robot
The kinematical model of wheeled mobile robot is

$$\begin{aligned} \dot{x} &= u_1 \cos q \\ \dot{y} &= u_1 \sin q \quad , \\ \dot{q} &= u_2 \end{aligned} \tag{1}$$

where (x, y) is position of the robots centre of the mass, q is orientation of the robot, u_1 is translational velocity and u_2 is rotational velocity. This model applies to a large class of mobile robots. Although the control inputs are at the velocity level, this is not restrictive for real mobile robot control because the modelling can be easily extended to include system dynamic. The main difficulties in dealing with the system (1) are getting from the fact that it is essentially underactuated, having less independent inputs then motion planning variables.
A two-stage approach is convenient in solving the planning problem: (i) potential field approach is used to generate a holonomic collision-free reference position, and (ii) tracking this using nonholonomic tracking (Kyriakopoulos at al., 1996), or local incremental motion planning (De Luca & Oriolo, 1994). The resulting feedback control law uses only local information limited to the range of distance sensors.

2.2 Potential field approach
The potential function is given as follows

$$U(q) = U_a(q) + U_r(q), \tag{2}$$

where $q = [x\ y]^T$ is the configuration vector of the robot, $U_a(q)$ is the attractive potential, and $U_r(q)$ is the repulsive potential. The most commonly used form of attractive potential is

$$U_a(q) = \frac{1}{2} h_a (q - q_g)^T (q - q_g), \tag{3}$$

where q_g is goal position, and h_a is the gain factor that specifies the strength of the attractive potential. The standard form of the repulsive potential is

$$U_r(q) = \begin{cases} \frac{1}{2} h_r \left(\frac{1}{r(q,q_0)} - \frac{1}{r_0} \right) & if \quad r(q,q_0) \leq r_0 \\ 0 & if \quad r(q,q_0) > r_0 \end{cases}, \tag{4}$$

where $r(q,q_0)$ is minimum distance of q from the obstacle, q_0 denotes the point on the obstacle observed by sensors, r_0 is the influence rang of the repulsive potential field, and h_r is the gain factor that specifies the strength of repulsive potential. Then the desired configuration vector $\dot{q}_d = [\dot{x}_d\ \dot{y}_d]^T$ can be obtained using gradient descent scheme

$$\dot{q}_d = -\nabla_q (U_a(q) + U_r(q)), \tag{5}$$

or a less conventional approach, vortex field method

$$\dot{q}_d = -\nabla_q U_a(q) \pm J \nabla_q U_r(q), \tag{6}$$

where J is 2-dimensional skew symmetric matrix

$$J = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \tag{7}$$

and + sign corresponding to counterclockwise rotation of the vortex field.

To complete the planning method, it is convenient to use

$$\dot{q}_d = tan^{-1} \frac{\dot{y}_d}{\dot{x}_d} - q. \tag{8}$$

Taking into account previously mentioned problems with choice of repulsive potential, in this work a different type of repulsive potential is introduced

$$U_r(q)=\begin{cases} m\,e^{-\frac{1}{2s}r^2(q,q_0)} & if \quad r(q,q_0)\le r_0 \\ 0 & if \quad r(q,q_0)> r_0 \end{cases} \tag{9}$$

and combination of gradient descent field and vortex field is considered

$$\dot{q}_d = -\nabla_q U_a(q) - \nabla_q U_{r,g}(q) + sign(q,q_0)\,J\,\nabla_q U_{r,v}(q)\ , \tag{10}$$

where $U_{r,g}(q)$ is the repulsive potential for negative gradient field, $U_{r,v}(q)$ is the repulsive potential for the vortex field, and

$$sign(q,q_0)=\begin{cases} +1 & if \quad q^T(q_0-q)>0 \ \& \ q^T J(q_0-q)\ge 0 \\ -1 & if \quad q^T(q_0-q)>0 \ \& \ q^T J(q_0-q)<0 \\ 0 & if \quad q^T(q_0-q)\le 0 \end{cases}\ . \tag{11}$$

It can be seen from expression (11) that vortex field vanish when robot go away from obstacle. Repulsive potential functions $U_{r,g}(q)$ and $U_{r,v}(q)$ contains four parameters: $m_g,\ s_g,\ m_v,\ s_v$ which can be adjusted to satisfy appropriate conditions like passage between closely spaced obstacles. This approach can be additionally improved by including Gauss function parameter dependence on velocity of robot. This dependence can be determined in simulation process and then estimated using the RBF Neural Networks in real time working environment.

3. The basic characteristics of Radial Basis Function Neural Networks

Radial Basis Function Neural Networks (RBF Neural Networks) shown by Figure 1. are well-known static feedforward one step learning artificial neural networks, which can be used as tools for solving various regression and classification problems.

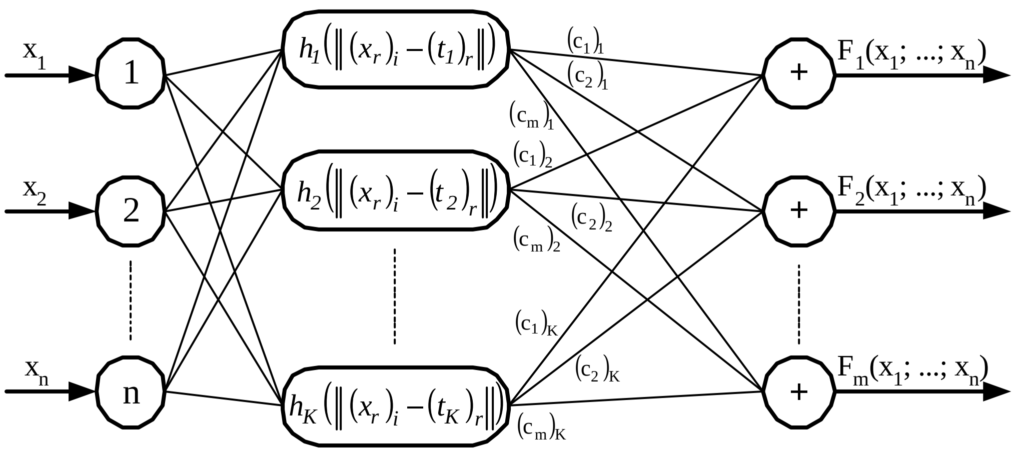

Fig. 1. RBF Neural Network model

Their structure is based on three mutually connected neurone layers: input, hidden and output layer. Input layer consists of n number of input neurones, where n is equivalent to the number of input vectors x_r (r = 1, …, n), which distributes input values toward hidden layer neurones. Each hidden layer neurone is parameterised with n centres (dimension of a vector t_j) and every centre is related to the belonging input vector. This means that input vector elements defines the amounts of the centres. Number of hidden layer neurones and the centre amounts has to be established during the learning phase. Learning an input-output mapping ($R^n \rightarrow R^m$) from an example is a hypersurface reconstruction problem (Poggio & Girosi, 1989), and is related to the classical approximation techniques. In a classical approximation theory, the approximation (interpolation) of a certain function f can be done by using another function F which can successfully approximate (interpolate) function f. In the case of RBF NN, function F belongs to the *radial basis functions* such as:

$$h(d) = e^{-\left(\frac{d}{c}\right)^2}, \tag{12}$$

$$h(d) = \frac{1}{\sqrt[a]{c^2 + d^2}}, a > 0, \tag{13}$$

$$h(d) = \sqrt[b]{c^2 + d^2}, 0 < b < 1, \tag{14}$$

$$h(d) = |d|, \tag{15}$$

where d is an Euclidean distance between the elements in R^n, and c, a ?and?b are radial basis function parameters. In another words, every neurone of a hidden layer evaluate radial basis function $h\left(\left\| (x_r)_i - (t_j)_r \right\|\right)$, where $\left\| (x_r)_i - (t_j)_r \right\|$ is previously mentioned Euclidean distance between the i-th element of all input vectors x_r (r = 1, …, n) and the input vectors related centres $(t_j)_r$ of a j-th hidden layer neurone. Hidden layer is connected with the output layer through the weight coefficients (vectors c_w, w = 1, …, m, where m is the number of output vectors). The amounts of these coefficients are unknown and, as a main objective, they have to be determined in the learning phase.

Radial Basis Function Neural Network maps n-dimensional inputs to m-dimensional outputs in the form:

$$F_w(x_r) = \sum_{j=1}^{M} h_j\left(\left\| (x_r)_i - (t_j)_r \right\|\right)(c_w)_j, \quad w = 1,\ldots,m, \; i = 1,\ldots,N, \; r = 1,\ldots,n,$$

$$M = \begin{cases} N, & \text{in the case of interpolation} \\ K, & \text{in the case of approximation} \end{cases}, \tag{16}$$

where N is a total number of data points and K represents the number of hidden layer neurones. In the case of interpolation, all elements of every input vector in the learning phase should be used for centres. This means that the number of hidden layer neurone (K) is equal to the number of input (output) vector elements (N). On the other hand, in the case of approximation, K is less than N. Using the equation:

$$F_w\left((x_r)_i\right) = (y_w)_i \ , \quad i = 1,\ldots,N \ , \tag{17}$$

and the expression

$$H_{ij} = h\left(\left\| (x_r)_i - (t_j)_r \right\|\right) \tag{18}$$

for $N \times N$ (interpolation) or $N \times K$ (approximation) matrix of radial basis functions (H_{ij}), the Equation 16 can be written in the matrix form

$$Y = H\, C \ , \tag{19}$$

where Y is an output elements matrix.
The matrix of weight coefficients C (with dimension $K \times m$) can be calculated from previous expression as

$$C = H^{-1} Y \ , \tag{20}$$

for interpolation process of learning, or from the equation

$$C = H^{+} Y \ , \tag{21}$$

for approximation learning process, where H^{+} is Moore – Penrose pseudoinverse of a rectangular matrix H that can be computed as

$$H^{+} = \left(H^T H\right)^{-1} H^T . \tag{22}$$

According to the Equations (16) to (22) it is obvious that the weight calculations are performed in only one step. Once established in the learning procedure, the amounts of weights and centres are fixed for a learned problem.

4. Description and the results of an experiment

As it was mentioned in Section 1., the main idea of this work was implementation of a new repulsive potential fields (gradient descent field and vortex field) through RBF Neural Network, which was then used as a tool for mobile robot obstacle detouring in unorganised environment. Attraction of robot to its goal position was performed by standard type of attractive potential field (Section 2.).
Avoidance of obstacles in unorganised robot workspace was accomplished by defining a set of Gauss repulsive function parameters of both repulsive potential

fields for different amounts of robot velocities. The parameter calculations were performed on behalf of the condition that for every velocity amount the parameters should be defined in a way to ensure minimal or close to minimal obstacle detouring path. Minimal distance between robot and obstacle was limited on 0.1 m. In order to define the desire parameter amounts, simulation processes of mobile robot model path defining (from point A to point B) in the workspace with two obstacles were carried out for robot velocities from 0.1 to 2 m/s with step 0.1m. In that way, 20 sets of parameters were calculated (every set is connected to one of the velocity amount) and all of them ensured approximate equal path lengths (from 6.9154 to 6.93 m) and robot distances from an obstacle (from 0.1004 to 0.1068 m). The results of simulations are presented by Figure 2. (relationships between Gauss function parameters and robot velocities) and Figure 3. (one of the obstacle detouring path for velocity v=0.6 m/s).

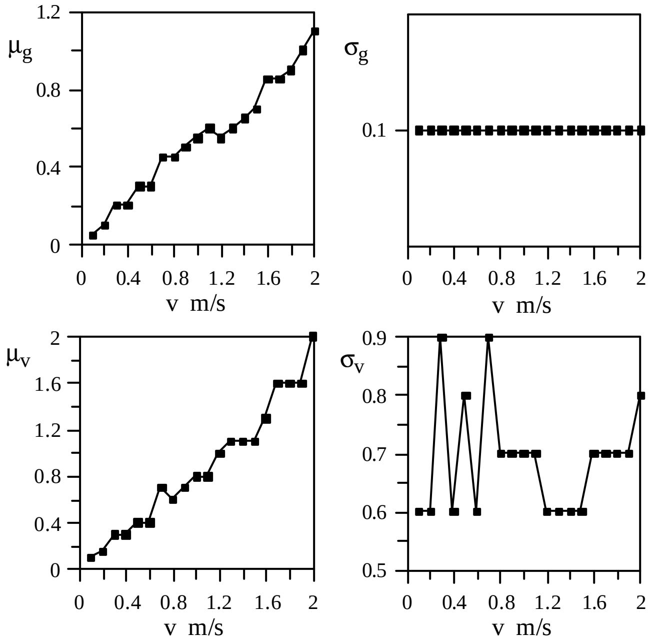

Fig. 2. Relationships between Gauss functions parameters and robot velocities

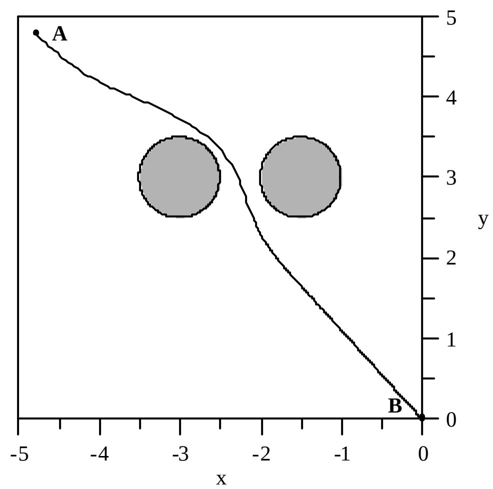

Fig. 3. Simulated obstacle detour path for v=0.6 m/s

On behalf of the parameters simulated in the robot workspace with two obstacles, which ensure minimal (or close to minimal) detouring paths for the specific mobile robot velocities, it is possible to achieve smooth and short obstacle detouring paths in the cases when there are many obstacles in the robot workspace.
In order to determine the Gauss repulsive function parameters for all velocities between 0.1 and 2 m/s the RBF Neural Network was used for parameter approximations. Simulated sets of parameters and their velocities were used in the learning phase in a way that the network input vector was defined by the velocity amounts and output vectors were defined by 3 of 4 Gauss function parameters. Gauss function width parameter of the gradient descent field σ_g was not used because its values turnout to be constant ($\sigma_g = 0.1$) for all the velocities. Also, several sets of parameters for new velocities were calculated in simulation process and approximated by RBF Neural Network. Measure of neural network performance was specified by using a non-dimensional error index NRMS, *Normalized Root Mean Square error* (Majetic, 1995), given by:

$$NRMS = \frac{\sqrt{\frac{\sum_{n=1}^{N}(d_n - O_n)^2}{N}}}{s_{d_n}}, \quad s_{d_n} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(d_n - \bar{d}\right)^2}, \quad \bar{d} = \frac{1}{N}\sum_{n=1}^{N} d_n \quad , \tag{23}$$

where d_n represents desired n-th network output, O_n represents calculated network outputs, $\bar{d}$ represents mean value of all desired outputs, s_{d_n} is standard deviation of the target data and N is the number of all input data. Results for all tested velocities are presented in Table 1.

v m/s	NRMS
0.16	0.00960712
0.33	0.00192603
0.47	0.00598637
0.65	0.00854716
0.84	0.01314951
1.14	0.00526180
1.32	0.00497401
1.43	0.00823509
1.65	0.00572662
1.83	0.00583949
1.97	0.00678056

Table 1. Test results achieved by RBF Neural Network

The NRMS error was calculated on behalf of the co-ordinates of the paths achieved

by calculated Gauss function parameters.
One of the detouring path comparisons between the path achieved by simulation process and RBF Neural Network with v=1.14 m/s is shown by Figure 4.

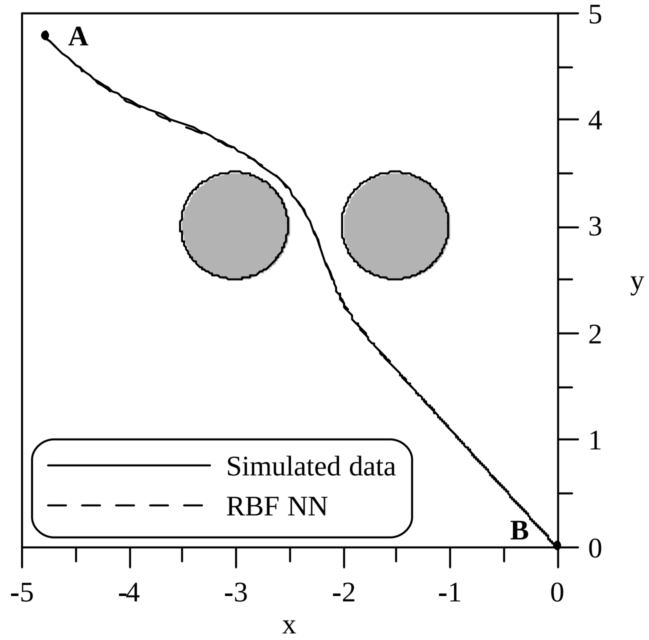

Fig. 4. Test results of simulated and RBF NN obstacle detour path for v=1.14 m/s

As it can be noticed from the presented results, it is obvious that the RBF Neural Network managed to achieve satisfactory approximation characteristics. This concept was also very successfully tested on several examples with more than two obstacles (Figure 5.) and in the problems with moving obstacles (Figure 6.).

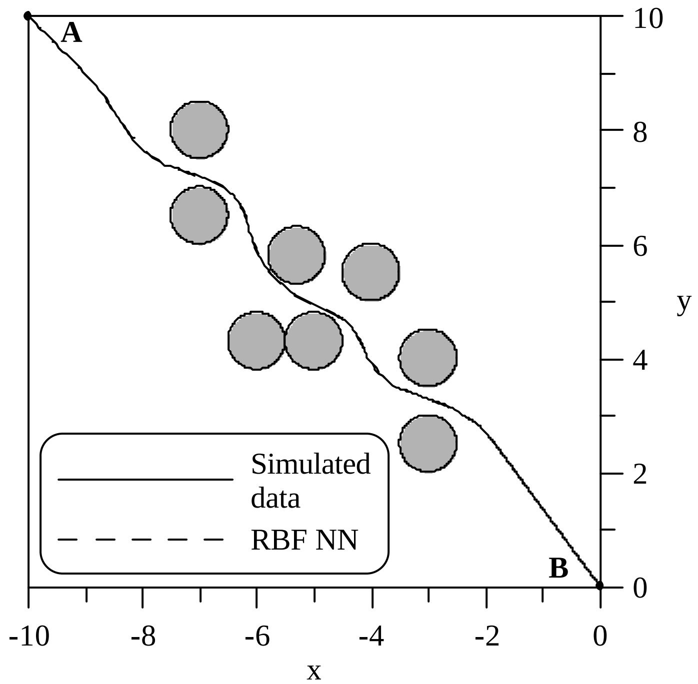

Fig. 5. Test results of simulated and RBF NN obstacle detour path for v=1.43 m/s in the case of 8 obstacles in the robot workspace

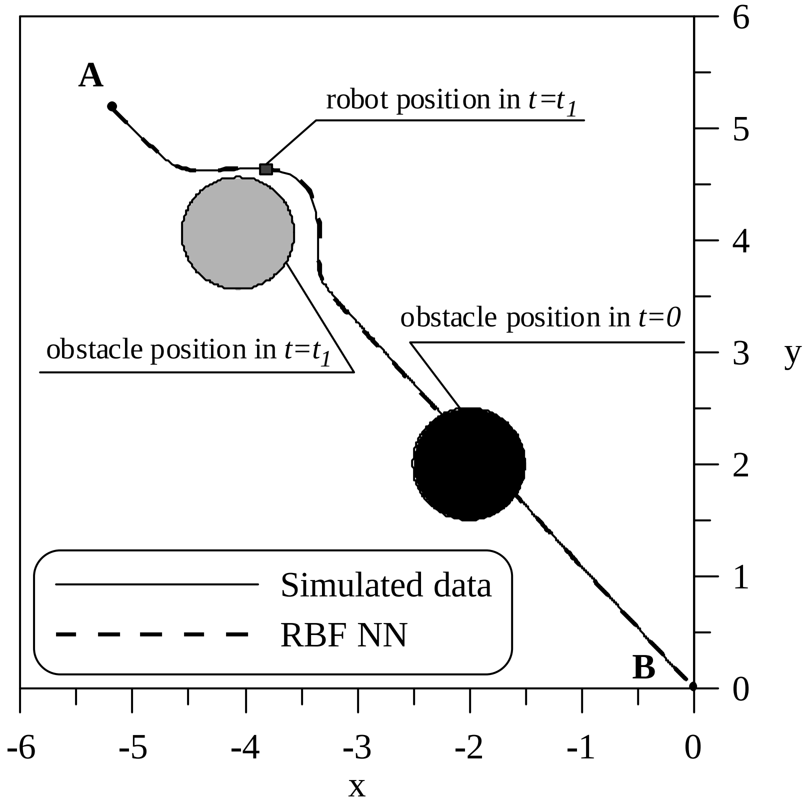

Fig. 6. Test results of simulated and RBF NN obstacle detour path for v(robot)=0.65 m/s and v(obstacle)=1 m/s in the case of moving obstacle in the robot workspace

5. Conclusion

In this work, a new approach to mobile robot navigation using a combination of negative gradient and vortex fields based on Gauss potential function is proposed. Parameters of Gauss potential functions depend on the mobile robot velocity and RBF Neural Network was trained to approximate this dependence on the base of simulation results which ensure passage between closely spaced obstacles and minimal or close to minimal obstacle detouring path.

This approach provides real-time mobile robot navigation in unorganised environment with closely spaced obstacles without local minima problems and trajectory oscillations. In additional, velocity dependent Gauss potential function provides effective avoidance of moving obstacles and other mobile robots in working environment.

These results are valuable at the level of holonomic motion planner. Next step is incremental motion planning which employ a feasible projection strategy to modify the output of the holonomic motion planner. Future research directions will include implementation of the proposed approach and local incremental planning on real mobile robot system (*Pioneer 2*) including complete robot and actuator dynamics. Also, an additional improvement of presented results will be achieved by introducing dependence between potential function parameters, velocity of mobile robot and distance from obstacles.

6. References

Borenstein, J. & Koren, Y. (1990). Real-time Obstacle Avoidance for Fast Mobile Robots in Cluttered Environments, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 572-577, Cincinnati, Ohio, May 1990.

De Luca A. & Oriolo, G. (1994). Local incremental planning for nonholonomic mobile robot, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 104-110, San Diego, California, May 1994.

De Medio C. & Oriolo, G. (1991). Robot obstacle avoidance using vortex fields," In: *Advances in Robot Kinematics*, Stifter S. & Lenarcic J. (Ed.), pp. 227–235, Springer-Verlag, Wien.

Ge S. S. & Cui Y. J. (2000). New Potential Functions for Mobile Robot Path Planning. *IEEE Transaction on Robotic and Automation*, vol. 16, no. 5, pp. 615-620, ISSN 1042-296X.

Khatib, M & Chatila R. (1995). An extended potential field approach for mobile robot sensor-based motions. Intelligent Autonomous System (IAS'4), Karlsruhe (Germany), pp. 490-496. *Available from:* http://www.laas.fr/~maher/pub.html *Accessed:* 2001-10-11.

Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots, *International Journal of Robotics Research*, vol. 5, no. 1, pp. 90-98, ISSN 0278-3649.

Koren Y. & Borenstein, J. (1991). Potential field methods and their inherent limitations for mobile robot navigation, *Proceedings of IEEE Conference on Robotics and Automation*, pp. 1398-1404, Sacramento, CA, April 1991.

Krogh, B.H. (1984). A generalised potential field approach to obstacle avoidance control, *Proceedings of SME Conference on Robotics Research*, Bethlehem, PA, August 1984. SME Paper No. MS84-484.

Kyriakopoulos, K.J.; Kakambouras, P. & Krikelis, N.J. (1996). Navigation of Nonholonomic Vehicles in Complex Environments with Potential Fields and Tracking, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 3389-3394, ISBN 0-7803-2988-4/96, Minneapolis, Minnesota, April 1996.

Majetic, D. (1995). Dynamic neural network for prediction and identification of non-linear dynamic systems, *Journal of Computing and Information Technology - CIT*, Vol. 3, No. 2, pp. 99-106, ISSN 1330-1136.

Poggio, T. & Girosi, F. (1989). A theory of networks for approximation and learning, *Available from:* ftp://publications.ai.mit.edu/ai-publications/1000-1499/AIM-1140.ps.Z *Accessed:* 2001-10-19.

Rimon, E. & Koditschek, D.E. (1992). Exact robot navigation using artificial potential functions. *IEEE Transaction on Robotic and Automation*, vol. 8, no. 5, pp. 501-518, ISSN 1042-296X.

Singh, L.; Stephanou, H & Wen, J. (1996). Real-time Robot Motion Control with Circulatory Fields, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2737-2742, ISBN 0-7803-2988-4/96, Minneapolis, Minnesota, April 1996.