An Efficient System for Nonholonomic Mobile Robot-Path Planning

Xuefu Xiang Jiye Zhang Chenlong He

Traction Power State Key Laboratory, Southwest Jiaotong University, Chengdu 610031, China

Abstract

This paper presents a novel simple, but efficient robot path planning algorithm, which plans the obstacle-avoidance path according to map information, makes robots reach the target. In the paper, the actual environment is discretized into a topologically organized map consisting of targets, obstacles and free spaces. We quantify the road roughness, which affects the motion of robots to the degree of pass convenience with a special mathematical model, and introduced into the model for path planning. Targets propagate the so-called "vector-distance" to whole workspace through free-space grid points in a special direction. Because the distance propagating in this way has direction and size, it is called "vector-distance", which is constrained by the motion of nonholonomic mobile robot. So the path obtained by this algorithm is the optimal path suitable to the motion of nonholonomic mobile robot. This model is characterized by high computational efficiency, small storage cost and perfectly simulating the actual map. This algorithm can be conveniently used in practice.

Keywords: Path-planning, Nonholonomic mobile robot, navigation, Obstacle avoidance

1. Introduction

The robot-path planning with obstacle avoidance is a fundamentally important issue in robotics. There are much research on robot-path planning using various approaches, such as the grid-based A^* algorithm, artificial potential field and fuzzy algorithm and so on [1-3]. However, some of those algorithms have some drawbacks such as computational complexity and the local minima. Stimulated by the dynamic properties of Hodgkin and Huxley's membrane model [4] and shunting model [5] developed by Grossberg, Yang and Meng [6-9] proposed a new neural network model for generating the path for point/nonholonomic mobile robot in static/dynamic environment, which had major advantages in generating the safety path for robots and computing path efficiently. In Yang and Meng's model, the target globally attracted the robot in the entire workspace through neural activity outward propagating, while the obstacles only locally had influence on the robot motion. The point/nonholonomic mobile robot-path generated by a lot of neuron was characterized by a shunting equation or an additive equation connected together, and the path was safety. However, the path was not shortest sometimes. A. R. Wilmas proposed distance propagating dynamic system [10], which had all advantages from Yang and Meng's model. In this model, the environment is topologically organized by grid. The target propagate outward the distance to entire environment, and each grid has an associated variable which records the distance between this grid point to nearest target. Because of the distance variable only recording the distance of path to the target, this robot path is shorter than the path generated by Yang and Meng's model occasionally. However, this algorithm cannot plan the path for the nonholonomic mobile robot. The robot environments of above two models were discretized into a lot of grid points. Each grid point was the target, obstacle or free space; they did not consider the diversity of the influence on the robot motion caused by the free-space face roughness.

Inspired by Yang and Meng's model and based on distance-propagating model we propose a simple robot-path planning model, which is called vector-distance propagating model. There are two main differences between our model and Wilmas's model. First, unlike the distance-propagating model in which the distance propagation is omni-directional, the vector-distance is directionally propagated through neighbour grid point in free-space. So the vector-distance recorded by each grid point is a function of both distance and direction subject to the nonholonomic constraint, and then the model can plan path for the nonholonomic mobile robot. Second, the vector-distance here not only records the length of path from target to free space grid point, but also records the measurement of the influence on the robot motion caused by the road surface roughness. Since the influence on the robot motion caused by the road surface roughness is considered, the path planed by our model is suitable for nonholonomic mobile robot stable motion, with little power and small mechanical wear. The numerical value of measurement of the influence on the robot motion caused by the road surface roughness is a real, positive constant number. This model has the same principle and mechanism as Wilmas's model. This algorithm of model is computationally simple and efficient. The robot path is planed without explicitly optimizing any global cost function, and any local obstacle-checking procedures at each step. Since the motion of robots only depends on current position and orientation without any information of motion history, the cost of storage is quite small. When the vector-distance dynamically propagates, the dynamic system converges in a small number of iterative to a state, that is, the vector-distance recorded by each grid point converges to a real constant number. This algorithm can always plan an optimal path, by which the nonholonomic mobile robot can safely and quickly reach the target. This model is well adapted to actual application.

This paper is organized as following. In section 2 we will first introduce the kinematics model of nonholonomic mobile robot, and mathematical model of measurement of the influence on the robot motion caused by the road surface roughness, vector-distance propagating system, and robot motion model. The simulation studies including motion planning for maze environment and parallel parking are presented in section 3. A simple conclusion of this model is addressed in section 4.

2. Modeling

In this section, the kinematics model of nonholonomic mobile robot and mathematical model of the influence on the robot motion caused by the road surface roughness are briefly introduced. Finally, the vector-distance propagating system and motion model are presented.

2.1. Vehicle model

For a mobile robot with size and shape, its configuration in the 2-D Cartesian workspace can be uniquely determined by (x, y, θ), spatial position of the center of gravity and the orientation angle.

The so-called control variables v_L and v_R is the velocity of left and left drive wheel, respectively. L is the distance between two drive wheel; α is the angle that robot turns over during the time interval of one step Δt (See Fig.1(a)).The kinematics' constraint of nonholonomic mobile robot is described as [9], [11], [12]

$$\dot{x}\sin\alpha = \dot{y}\cos\alpha \quad (1)$$

In this robot model, given the robot velocity of two independent drive wheels, v_L and v_R in very short interval, the following equations can be obtained through geometric relationship.

$$\begin{aligned} v &= \frac{v_R + v_L}{2} \\ \Delta v &= v_R - v_L \\ \alpha &= \int_0^{\Delta t} \frac{\Delta v}{L} dt \end{aligned} \quad (2)$$

After the integration over the time interval of one step, based on (1), (2) the next robot position is given by

$$\begin{aligned} \theta(t+\Delta t) &= \theta(t) + \alpha \\ x(t+\Delta t) &= x(t) + v\Delta t\cos\alpha \\ y(t+\Delta t) &= y(t) + v\Delta t\sin\alpha \end{aligned} \quad (3)$$

The robot velocity v and orientation angle θ (Δv) depend on the two control variables. The v and Δv are limited to $v_{\max}$ and $\Delta v_{\max}$, respectively, i.e. $|v| \le v_{\max}$ and $|\Delta v| \le \Delta v_{\max}$. The Cartesian product of two intervals is given by

$$\{-v_0, v_0\} \times \{-\Delta v_{\max}, 0, \Delta v_{\max}\}$$

and gets one set with the six elements. The six elements represent the six next possible robot configurations of a given robot configuration (See Fig.1 (b)) [8] [9].

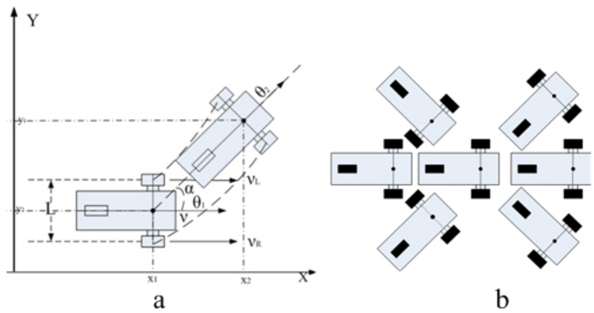

Fig.1: Diagram of a nonholonomic mobile robot. (a) Kinematics of a mobile robot, where (x_1, y_1) is robots' position at time t_k , and θ_1 is orientation. (x_2, y_2, θ_2) is robot configuration at time $t_k + \Delta t$. (b) Six possible next robot configurations of a given robot configuration depends on v and Δv .

2.2. Model of influence on robots motion caused by the road surface roughness

We assume that the surface of robot movement space is known. Divided the surface into a lot of sub-surfaces, whose projection on the level is a unit square. The subsurface becomes a unit square plane, which is parallel to ground level by a special mathematical technique. So the robot environment is discretized into M grid points, labeled by an index i.

We define B_i to be the point set of free spaces that are adjacent neighbor to grid point i, $d_{\max}$ and $d_{\min}$ to be the minimum and maximum distances between any two adjacent neighbors in the grid of free space and target respectively and d_{ij} to be the length of minimum path joining two free spaces i and j through free-space [10]. As illustrated in Fig.2, for regular unit square grid, the adjacent neighbor set for the grid point labeled zero and $d_{\max}$, $d_{\min}$ are

$$B_0 = \{1,2,3,4,5,6,7,8\}$$

$$d_{\max} = d_{03} = \sqrt{2}\ , d_{\min} = d_{02} = 1 .$$

16	17	18	19	9
15	7	8	1	10
14	6	0	2	11
13	5	4	3	12

Fig.2: Regular square grid and $d_{\max}$, $d_{\min}$

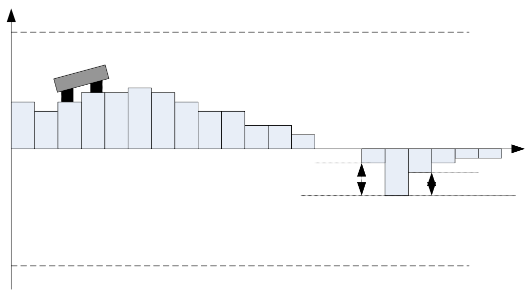

Fig.3: Model of Influence on Robots Motion caused by the Road Surface Roughness.

For simplicity but without losing generality, when the robot R arrives at i position at time t_k (the position of the center of gravity of robot R is grid point i), the influence on the robot motion caused by the grid point i surface roughness replaces the influence on the area which contacts with robot at time t_k. We assume that nonholonomic mobile robot motion heading up at β over grid point j. Fig.3 gives the demonstration. We define $d_{hl}{}^{j(\beta)}$ and $d_{hr}{}^{j(\beta)}$ to be the absolute value of the difference of altitude between grid point j and its right and left adjacent neighbor in the direction β. We obtain the value $f(j,\beta)$ by following equation

$$f(j,\beta) = k_l d_{hl}{}^{j(\beta)} + k_r d_{hr}{}^{j(\beta)} \tag{4}$$

where $\beta \in \{0, \pi/4, \pi/2, 3\pi/4, \pi, 5\pi/4, 3\pi/2, 7\pi/4\}$. $k_l + k_r = 1$ and $k_l / k_r = d_{hl}{}^{j(\beta)} / d_{hr}{}^{j(\beta)}$. $k_i = 0$, if and only if $d_{hi}{}^{j(\beta)} = 0$ or $d_{hi}{}^{j(\beta)} > 2H$. $k_i \in [0,1]$, $i = r, l$.

The influence on the robot motion caused by the road surface roughness should only locally affect robot-path planning. Assign $f(j,\beta)$ to a real number $I_j(\beta)$ ($I_j(\beta) \in [0, d_{\max}]$) using the following equation

$$I_j(\beta) = f(j,\beta)/(H \times d_{\max}) \tag{5}$$

where H is the maximum absolute value of the altitude difference between any grid point and ground level.

Each grid point has eight different I_i corresponding to β. The model defines the value of I_i of target and obstacle point to be zero and D, respectively. The value of I_i is given by

$$I_i(o) = \begin{cases} 0, & i \text{ is target} \\ D, & i \text{ is obstacle} \\ [0, d_{\max}], & \text{otherwise} \end{cases} \tag{6}$$

where the value of D ($D = 4Md_{\max}$) is sufficiently large.

2.3. Vector-distance propagating system

The nonholonomic mobile robot configuration in the 2-D Cartesian workspace can be uniquely determined by the spatial position (x, y) of the center of gravity and the orientation angle θ. We use eight orientations of robot configuration representing the robot yaw angle from 0° to 360° with a step of 45°.

ORC	1	2	3
yaw angle	22.5° - 67.5°	67.5° - 112.5°	112.5° - 157.5°
ORC	4	5	6
yaw angle	157.5° - 202.5°	202.5° - 247.5°	247.5° - 292.5°
ORC	7	0	
yaw angle	292.5° - 337.5°	337.5° - 22.5°	

Table1: The orientation of robot configuration corresponding to robot yaw angle.

In detailed, orientation of robot configuration (ORC) 0 corresponds to the robot yaw angle from 337.5° to 22.5°, and other knowledge given in Table1.

A set of orientation of robot configuration:

$$O = \{0,1,2,3,4,5,6,7\}$$

For example, robot configuration ($x_1, y_1, 3$) represents that the spatial position of center of gravity of robot is (x_1, y_1), and the robot yaw angle is between 112.5° and 157.5°.

Each grid point i recording the vector-distance is propagated from target outward to point i through the free space adjacent neighbor. Now, assume that the vector-distance is propagated from grid point j to its adjacent neighbor grid point i. We define the direction θ_i of vector $j \to i$ to be the direction of vector-distance propagation of grid point i. The direction of vector-distance of grid point i, $\vec{\theta}_j$ is given by following equation

$$\vec{\theta}_i = f(\vec{\theta}_j, \theta_i, \delta) = \begin{cases} -1 \times s \times \theta_i, & \delta > \dfrac{\pi}{2} \\ 1 \times s \times \theta_i, & \delta < \dfrac{\pi}{2} \end{cases}$$

$$s = \begin{cases} -1, & \vec{\theta}_j < 0 \\ 1, & \vec{\theta}_j > 0 \end{cases} \tag{7}$$

where $\theta_i \in O$, δ ($\delta \le \pi$) is included angle between $\vec{\theta}_j$ and θ_i.

When the environment of robot is determined, the needed robot configuration in target position is determined; we call this configuration as target-robot configuration.

The system defines the direction of the vector-distance of target $\vec{\theta}_t$ to be the reverse orientation of target-robot configuration, and the direction of the vector-distance propagation of target θ_t to be D.

As illustrated in Fig.4 (b), $\vec{\theta}_t = 7$, and $\theta_t = 0$. We assume that the direction of vector-distance of grid point j is 4 ($\vec{\theta}_j = 4$), and grid point j propagates the vector-distance to its adjacent neighbor grid point i. The direction of vector-distance of grid point i can be obtained by (7). Because of $\theta_i = 3$, $\vec{\theta}_j = 4$ and $\delta < \pi/2$, which δ is the included angle between $\vec{\theta}_j$ and θ_i, so $\vec{\theta}_i = 3$. Similarly, $\theta_k = 1$ and $\delta > \pi/2$, which δ is the included angle between $\vec{\theta}_j$ and θ_k, and then $\vec{\theta}_k = -1$.

We call the grid point i and k as the positive and negative vector-distance propagating point of grid point j, respectively.

Note that the model requires $\delta \ne \pi/2$ in the vector-distance propagating process.

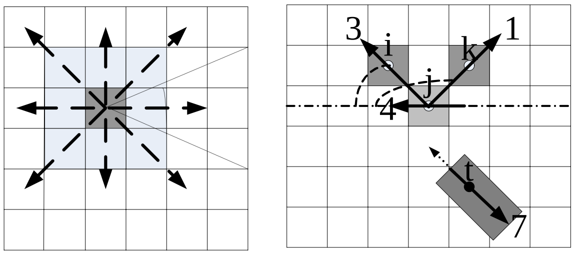

Fig.4: (a) Orientation of the nonholonomic mobile robot configuration. (b) The direction of vector-distance and its propagating

We define the function $L_{ti}(n+1)$ at time step $n+1$ given by

$$L_{ti}(n+1) = \begin{cases} 0, & \text{target} \\ \min(L^+_{ij}, L^-_{ij}) & \text{othrewise} \end{cases} \tag{8}$$

where

$L^+_{ij} = \min\limits_{j \in S_i^+}(d_{ij} + x_j(n,o))$ (i is positive vector-distance propagating point of point j);

$L^-_{ij} = \min\limits_{j \in S_i^-}(d_{ij} + x_j(n,o) + 2d_{\max})$ (i is negative vector-distance propagating point of point j).

Notes:

1) $\vec{\theta}_j$ is not perpendicular to $\vec{\theta}_i$,and $\delta < \pi/2$, which δ is the included angle between $\vec{\theta}_j$ and θ_i, if $j \in S_i^+$ and $S_i^+ \subset B_i$.

2) $\vec{\theta}_j$ is not perpendicular to $\vec{\theta}_i$,and $\delta > \pi/2$, which δ is the included angle between $\vec{\theta}_j$ and θ_i, if $j \in S_i^-$ and $S_i^- \subset B_i$.

Each grid point has eight different I_i corresponding to β. When the vector-distance along the direction of vector-distance propagation of grid point i is propagated to grid point i, the corresponding I_i will work. Each grid point has an associated variable x_i given by

$$x_i(n+1, o) = \min(D\ ,\ (L_{ti}(n+1) + I_i(o))) \tag{9}$$

where $o = \vec{\theta}_i$ represents the direction of vector-distance.

The vector-distance propagating system is characterized by equations (8), (9). The vector-distance is completely specified by a magnitude x_i and a direction θ_i. Here $\vec{\theta}_j$ is not perpendicular to $\vec{\theta}_i$ and $\delta \neq \pi/2$. Since the grid point j's vector-distance is a certain value at time step n (n grid step), and the vector-distance spreading speed is one grid step in each time step. The grid point i can record the variable x_i propagated from j at time $n+1$. The value of x_i recorded by each grid point will be $x_i \in [0, D]$.

The system is initialized by setting the variables $x_t(0)=0$ and $\vec{\theta}_t$ to the reverse orientation of target-robot configuration and all other locations $x_i(0)=D$ and $\vec{\theta}_i(0)=D$. The computational burden of this system depends on the total number of grid point M and the complexity of the environment.

The proposed model and the distance model [10] have the same principle and configuration. The distance-propagating model has proven that x_i converges to y_i and given the computational procedure of speed of convergence. We will not specify here again. However, unlike the distance model, the distance converges to y_i; our model's vector-distance converges to $y^*{}_i$, where $y^*{}_i$ is the sum of length of optimal path and measurement of the influence on the robot motion caused by the optimal path surface roughness. And the vector-distance x_i will converge to $y^*{}_i$ in N time steps; the number of grid step N is bounded by

$$\left\lceil \frac{y^*{}_i}{4d_{\max}} \right\rceil \leq N \leq \left\lfloor \frac{y^*{}_i}{d_{\min}} \right\rfloor$$

where $\lceil \zeta \rceil, \lfloor \zeta \rfloor$ represent ζ rounded upward or downward to the nearest integer, respectively.

2.4. Robot movement

Since the vector-distance propagation is subject to the kinematics' constraints of nonholonomic mobile robot, and once the vector-distance recorded by the grid point i is a certain value, the optimal path Γ_{ti} joining target and the grid point i through adjacent neighbors of free space for nonholonomic mobile robot is determined. Namely, the vector-distance along path Γ_{ti} is propagated from target to grid point i. The robot location $r(t)$ is specified as an index of one of the points on the grid, and is a function of real time $t \geq t_0$ [10]. Initially,

$$r(t_0) = i_0 \text{ and } \theta^*{}_{r(t_0)} = -\vec{\theta}_{i_0}$$

Where $\theta^*{}_{r(t_0)}$ is orientation of robot configuration at position $r(t_0)$, and $-\vec{\theta}_{i_0}$ is reverse direction of vector-distance of grid point $r(t_0)$.

We assume that the robot's travel path is updated at a set of real-time values $t_0 <...< t_k < t_{k+1} <...$ [10]. At time t_k, the robot's location is the grid points $r(t_k)$, and $\theta^*{}_{r(t_k)}$ must be equal to $-\vec{\theta}_{r(t_k)}$. The next updating time t_{k+1} and location $r(t_{k+1})$ are determined (the robot motion heads up at the direction $-\theta_{r(t_k)}$ from $r(t_k)$ to $r(t_{k+1})$ by the speed of one grid step in each time step). The $r(t_{k+1})$ is defined as:

$$r(t_{k+1}) = \begin{cases} r(t_k), & r(t_k) \text{is target or obstacle} \\ Ind(r(t_k), o), & \text{otherwise} \end{cases} \tag{10}$$

Defined as [10], $Ind(i,o)$ is the index of the closest neighbor through which the value of x_i was calculated and propagated to $r(t_k)$.

If the grid point $r(t_k)$ is target or obstacle, the robot does not move, otherwise at next update time t_{k+1}, the robot will be at location $r(t_{k+1})$ and

1) $\theta^*{}_{r(t_{k+1})} = -\vec{\theta}_{r(t_{k+1})}$, and $\vec{\theta}_{r(t_k)}$ is not perpendicular to $\vec{\theta}_{r(t_{k+1})}$;

2) $r(t_{k+1}) \in S_{r(k)}$, $S_{r(t_k)} = S^-{}_{r(t_k)} \bigcup S^+{}_{r(t_k)}$; and

3) $x_{r(t_k)} = I_{r(t_k)} + L^-{}_{r(t_k)r(t_{k+1})}$ or $x_{r(t_k)} = I_{r(t_k)} + L^+{}_{r(t_k)r(t_{k+1})}$.

For the workspace of robots, the distance $d_{r(t_{k+1})r(t_k)}$ between $r(t_k)$ and $r(t_{k+1})$ is one grid step, and the time interval $\Delta t = t_{k+1} - t_k$ depends on $d_{r(t_{k+1})r(t_k)}/v$. The robot can simply follow the optimal path to the target using equation (10).

3. Simuliation studies

3.1. Motion planning for maze-type environment

In this simulation, we show the solution to a maze-type problem and the influence on the robot motion caused by the road surface roughness. The robot can find its way through a complicated environment, which is topologically organized by 20×20. The position of entrance is (20, 1) and the position of exit is (1, 20). Obstacles are shown as black solid shaded area, as illustrated by Fig.5. Note in the circle, the broken line represents the path planed by other model [4] [5], which do not considered the influence on the robot motion caused by the road surface roughness. The measurement of the influence on the robot motion caused by the road surface roughness of grid point (16, 10) and (16, 11) is 1 and $\sqrt{2}$, by contraries, both grid point (15, 10) and (15, 11) is zero. The vector-distance spreads along the optimal path, rather than suboptimal path, which is represented by broken line, and the robot move along this optimal path.

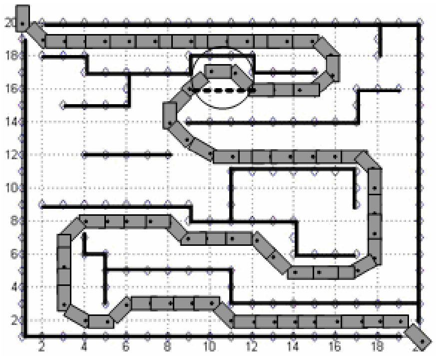

Fig.5: Maze through which the robot finds the shortest path and the path planed for the nonholonomic mobile robot is subject to the influence on the robot motion caused by the road surface roughness.

3.2. Motion planning for parallel parking and U-shaped environment

The path planning for parallel parking and U-shaped environment is shown in this simulation (see Fig.6).

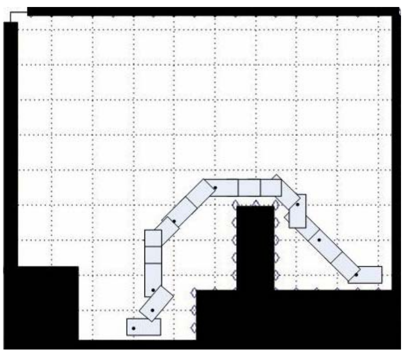

Fig.6: Path planning for parallel parking and U-shaped environment.

The system plans the optimal path from initial position (16, 3) (the robot configuration is (16, 3, 4)) to the position of target (19, 14) rounding the obstacle represented by black solid bar. When the robot is close to the target, the robot adjusts its configuration along the path, finally, the robot parks parallel with initial configuration. The proposed model is able to plan the safe and optimal path for nonholonomic mobile robot.

4. Conclusions

Being different from the model in [10] where the distance propagation is omni-directional, the vector-distance of proposed model propagates in some special direction and is vector. The novel concept, which is the influence on the robot motion caused by the road surface roughness, is proposed in our model by mathematical method. This model can comply with the property of real environment better, and the path planed by the model has more practicality. The model plans the path for nonholonomic mobile robot without explicitly searching over the obstacles and optimising any cost function, has advantages of low cost of storage and high computing efficiency. Our model is of advantages to distance-propagation model and can plan path for nonholonomic mobile robot. The robots can quickly and safely move along the path planed by our model and reach target.

Acknowledgement

This work is supported by National Program for New Century Excellent Talents in University (No.NCET-04-0889), Youth Science Foundation of Sichuan (No. 05ZQ026-015).

References

[1] C. W. Warren, Fast path planning using modified a^* method, *Proceedings of IEEE International Conference on Robotics and Automation*, pp.662-667, 1993.

[2] J. Chuang and N. Ahuja, An analytically tractable potential field model of free space and its application in obstacle avoidance, *IEEE Transactions on System, Man, and Cybernetic*, 28: 729-736, 1998.

[3] W. L. Xu and S. K. Tso, Real-Time self-reaction of a mobile robot in unstructured environment using fuzzy reasoning, *Engineering Application Artificial Intelligent*, 6:475-485, 1996.

[4] A. L. Hodgkin and A. F. Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve, *J. PhYSIOL. Londen*, 117:500-544, 1952.

[5] S. Grossberg, Nonlinear neural networks: principles, mechanisms, and architecture, *Neural Networks*, 1:17-61, 1988.

[6] S.X. Yang and M. Meng, An efficient neural network approach to dynamic robot motion planning, *Neural Network*, 13: 143-148, 2000.

[7] S.X. Yang and M.Meng, An efficient neural network method for real-time motion planning with safety consideration, *Robotics Autonomous System*, 32: 115-128, 2000.

[8] S.X. Yang and M. Meng, Real-time collision-free motion planning of a mobile robot using a neural dynamics-based approach, *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, pp. 3087-3092, 2003.

[9] S.X. Yang, *Neural network approaches to real-time motion planning and control of robotic systems*, Ph.D. dissertation, Dept. Electrical Computer Engineering of University Alberta, Canada, June 1999.

[10] A.R. Willms and S. X. Yang, An efficient dynamic system for real-time robot-path planning, *IEEE Transactions on Systems, Man, and Cybernetics*, 36: 755-766, 2006.

[11] L. Beji and Y. Bestaoui, Motion generation and adaptive control method of automated guided vehicles in road following, *IEEE Transactions on Intelligent Transportation Systems*, 6: 113-123, 2005.

[12] D.Y. Yu and H. Xue, Application of fuzzy control method to AGV, *Proceedings of the 2003 IEEE International Conference on Robotics, Intelligent Systems and Signal Processing,* pp.768-772, October 2003.