Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Evolutionary Univector Field-based Navigation with Collision Avoidance for Mobile Robot

Yusun Lim*, Seung-Hwan Choi*, Jong-Hwan Kim* and Dong-Han Kim**

** Department of Electrical Engineering and Computer Science, KAIST, Daejeon, Korea, (e-mail: {yslim, shchoi, johkim}@rit.kaist.ac.kr)}*
*** Kyung Hee University College of Electronics and Infomation, Yongin, Korea, (e-mail: donghani@khu.ac.kr)}*

Abstract: This paper proposes a novel vector field navigation method by using the velocity vector of a robot to avoid obstacles in dynamic environment. Shifting vector and virtual obstacle are proposed to derive the center of repulsive field which makes the robot move away from the obstacle: The shifting vector is derived from the velocity vector of the robot and the obstacle and the virtual obstacle is made by adding the shifting vector to the real obstacle. By considering the velocity vector of an obstacle with a shifting vector and a virtual obstacle, the robot can avoid not only stationary obstacles but also dynamic obstacles, where the fields are optimized by evolutionary programming. The performance of proposed method is demonstrated by simulations and experiments in robot soccer system.

1. INTRODUCTION

Navigation is the process that generates a path from environment information and guides a robot follow the path (P. J. McKerrow [1991]). The process involves three tasks: i) sensing and mapping the environment, ii) path planning and selection, and iii) path following. Although there have been numerous research achievements in the study of navigation, it is still difficult to realize the navigation in the dynamic and uncertain environment.

Navigation is generally classified into the separated navigation and the unified navigation. First of all, the separated navigation consists of two processes: path planning and path following. In the path planning, a path which connects the start point to the end point without collision can be derived. Then, the robot can follow this path in the path following step. Because of this separation of the process, the separated navigation method has problems when applied in the environment which has uncertainty or changes with respect to time. When the robot moves away from the planned path, it tries to return to the path without considering the change of the environment and this movement may cause collision with obstacles. It is also difficult to generate a complete path at every control time in a dynamic environment to solve this kind of problem due to the heavy computation.

In the unified navigation method the path planning process and the path following process are unified in one step. The unified navigation method only considers desired motion at the current position and does not generate a complete path. The most well-known unified navigation is the artificial potential field navigation (J. Borenstein and Y.Koren [1991]) in which the robot moves in a direction proportional to the force which consists of an attractive force from the destination and a repulsive force from the obstacles.

The unified navigation allows a robot to navigate through moving obstacles by changing only the field at the robot position instead of changing the whole path. So, it has an advantage that it can be easily applied to the dynamic environment with moving obstacles (Kai-Tai Song and Charles C. Chang [1999] and Chales C. Chang and Kai-Tai song [1997]). However, this pragmatic method does not necessarily guarantee the obstacle-avoidance ability.

Univector field navigation is one of the unified navigations. In this navigation, the normalized two-dimensional univector field is used as a navigation function. It determines a desired posture of a fast moving mobile robot with consideration of the target position and obstacle avoidance (Y.-J. Kim et al. [2001]). To express the repulsive force from obstacles, univector field navigation employs 'non-uniform rational B-spline (NURBS)' and defines a univector field matrix for NURBS. Evolutionary programming (EP) is applied to optimize constant parameters in the univector field matrix.

As expected, navigation in dynamic environment is one of the most important issues in robot researches, because robots should move around in the real world interacting with human. This paper proposes a novel navigation method in the dynamic environment using a univector field. Conventional navigation systems consider positions of real obstacles only. Proposed navigation, however, adopts the concept of virtual obstacles, which considers the velocities of the robot and obstacles. The virtual obstacle enables the robot to avoid obstacles efficiently with simple repulsive field. There are three constant parameters which should be optimized: a distance in which the repulsive field affects, and two constant parameters for velocities of the robot and obstacle. In this paper, EP is employed to optimize these constant parameters.

In Section II, collision avoidance path planning based on evolutionary univector field method is described. Section III reports the results of simulations and experiments, respectively. Concluding remarks follow in Section IV.

2. UNIVECTOR FIELD NAVIGATION

2.1 Univector field navigation

The vector in univector field has been normalized and the direction of the vector is the desired heading of a robot at a specific position and time. The univector field consists of 'basic univector field', 'move-to-goal univector field' and 'avoid-obstacle univector field.' The move-to-goal univector field leads the robot to move to the destination with desired posture and the avoid-obstacle univector field helps the robot avoid the obstacles. The move-to-goal univector field and the avoid-obstacle univector field are designed based on basic univector field.

All the univector fields used in this paper is represented in terms of angles, as follows:

$$\phi : F \to [-\pi, \pi]. \tag{1}$$

2.2 Basic univector field

Two basic univector fields are designed: the first one, a hyperbolic spiral vector field, is used in the move-to-goal univector field to make a robot move to the destination and the second one, a repulsive vector field, is employed in the avoid-obstacle univector field to make a robot move away from the obstacles.

1) Hyperbolic spiral univector field

The hyperbolic spiral univector field is defined as

$$\phi_h(p, d_e) = \begin{cases} \theta \pm \dfrac{\pi}{2}\left(2 - \dfrac{d_e + K_r}{\rho + K_r}\right), & if \quad \rho > d_e \\ \theta \pm \dfrac{\pi}{2}\sqrt{\dfrac{\rho}{d_e}}, & if \quad 0 \le \rho \le d_e \end{cases} \tag{2}$$

with

$$N_h(p, d_e) = [\cos\phi_h, \sin\phi_h]^T$$

where

θ	angle from x-axis at the position p
K_r	adjustable parameter
ρ	distance between the origin and position p
d_e	predefined radius that decides the size of the spiral.

Fig. 1 shows the hyperbolic spiral univector field. If K_r becomes larger, the spiral becomes smoother. The notation ± represents the direction of motion, where + means that the robot moves clockwise and − means that the robot moves counter clockwise (Y.-J. Kim [2003]).

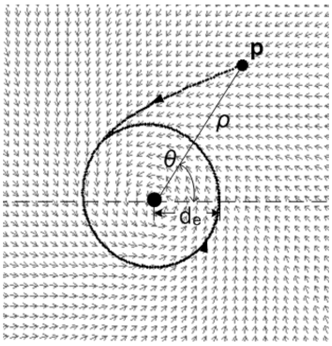


Fig. 1. Hyperbolic spiral univector field

2) *Repulsive univector field*

Fig. 2 shows the repulsive univector field, which is defined as

$$\phi_r(p) = \theta. \tag{3}$$

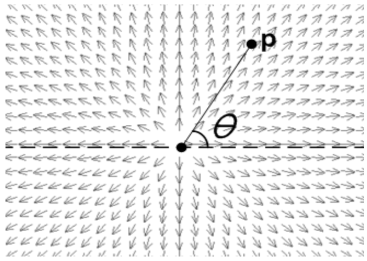


Fig. 2. Repulsive univector field

2.3 Move-to-goal univector field

The move-to-goal univector field makes the robot move to the destination. Obstacles are not considered in the move-to-goal univector field. A basic univector field is used directly as the move-to-goal univector field, or two or more basic univector fields can be composed to build one move-to-goal univector field. In this paper, to drive the robot to the destination with desired orientation, two hyperbolic spiral univector fields are combined. It is assumed that the robot kicks the ball to the right side of the field, as can be seen in Fig. 3. The move-to-goal univector field should be generated as follows:

$$\phi_{TUF} = \begin{cases} \angle \dfrac{y_l N_{h(CCW)}(p_l, d_e) + y_r N_{h(CW)}(p_r, d_e)}{2d_e} & if \;\; -d_e \le y < d_e \\ \phi_{h(CW)}(p_l, d_e) & if \quad y < -d_e \\ \phi_{h(CCW)}(p_r, d_e) & if \quad y \ge d_e \end{cases} \tag{4}$$

with

$$y_l = y + d_e,\ y_r = y - d_e$$

$$p_l = [x, y - d_e]^T,\ p_r = [x, y + d_e]^T.$$

where

cw	clock wise
ccw	counter clock wise.

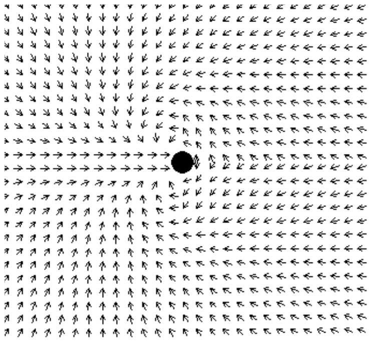


Fig. 3. Generated move-to-goal univector field. The black circle at the center is a ball.

2.4 Avoid-obstacle univector field

The avoid-obstacle univector field makes the robot avoid obstacles in the field. In this paper, a virtual obstacle is proposed by adding shifting vector $\vec{s}$ to obstacle's position vector. The avoid-obstacle univector field uses only repulsive univector field and its center is the virtual obstacle instead of the real obstacle. Fig. 4 shows virtual obstacle and related vectors. Proposed avoid-obstacle univector field is defined as

$$\phi_{AUF}(p) = \phi_r(p - P'_{obstacle}) \tag{5}$$

with

$$P'_{obstacle} = \begin{cases} P_{obstacle} + \vec{s}, & if \quad d \geq |\vec{s}| \\ P_{obstacle} + \dfrac{d}{|\vec{s}|}\vec{s}, & if \quad d < |\vec{s}| \end{cases},$$

$$\vec{s} = K_0(\vec{V}_{obstacle} - \vec{V}_{robot})$$

where

$P'_{obstacle}$ virtual obstacle's position
$P_{obstacle}$ real obstacle's position
P_{robot} robot's position
d distance between $P_{obstacle}$ and P_{robot}
$\vec{V}_{obstacle}$ obstacle's velocity vector
$\vec{V}_{robot}$ robot's velocity vector.

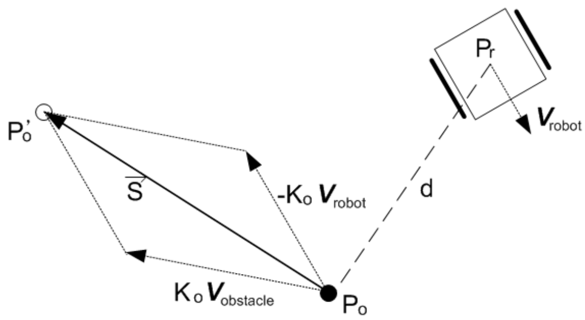


Fig. 4. Virtual obstacle in the avoid-obstacle univector field. $P'_{obstacle}$ is the position of the virtual obstacle.

Fig. 5 illustrates the generated avoid-obstacle univector field by the defined equation. With stationary obstacles, the shifting vector has the opposite direction and the size proportional to the velocity vector of the robot. When an obstacle locates in the way of the moving direction of the robot, the shifting vector allows the robot to perceive the obstacle's location closer than real as in Fig. 6(a). Therefore, the robot can avoid an obstacle earlier than without the shifting vector. If the obstacle is in back of the robot, the robot senses the obstacle farther than real distance as in Fig. 6(b). So, the obstacle, which does not hinder the robot's path, has much less influence on the robot's movement.

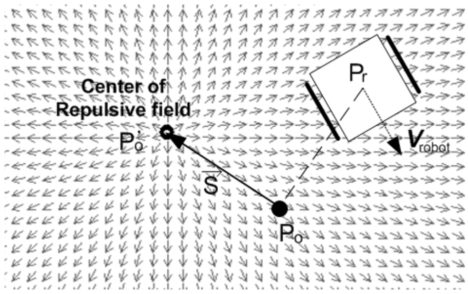
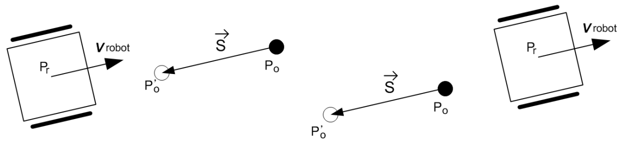


Fig. 5. Generated avoid-obstacle univector field. P_o is position of the real obstacle and P'_o is the position of the virtual obstacle. P_r is the center of the robot.



(a) (b)

Fig. 6 (a) The situation that obstacle is in front of the robot. (b) The situation that obstacle is in back of the robot.

The predictable problem of this new avoid-obstacle univector field is the situation such as in Fig. 7 that the virtual obstacle is located behind the robot, but the real obstacle is in front of the robot. In this situation, the robot can not avoid collision with the obstacle. To prevent this situation, the distance between the real obstacle and the virtual obstacle should not be longer than the distance between the real obstacle and the robot. To satisfy this condition, there is a condition $d \geq |\vec{s}|$ in Eq. (5). If $d < |\vec{s}|$, $\vec{s}$ is modified to a vector which has the same direction with $\vec{s}$ and magnitude d. Therefore, d is the maximum value of $\vec{s}$. If the obstacle is moving, its velocity vector is added to the shifting vector. Therefore, in this wsy the robot can predict the obstacle's movement.

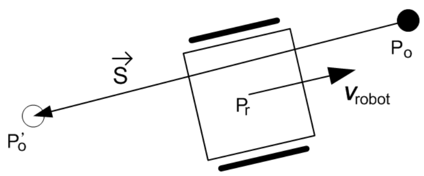


Fig. 7. The situation that the robot is between the real and virtual obstacle.

2.5 Composition of move-to-goal univector field and avoid-obstacle univector field

To apply the move-to-goal univector field and the avoid-obstacle univector field, the composition process is required. Gaussian function is used as a compound ratio in the composition process along with the distance between a robot and an obstacle. A constant $d_{\min}$ defines a minimum distance in which the move-to-goal univector field can be applied. The avoid-obstacle univector field is solely applied if the distance is less than $d_{\min}$ from the center of repulsive field, and composition of the move-to-goal univector field and the avoid-obstacle univector field is applied outside of distance $d_{\min}$. Detailed composition process is defined as follows:

$$\phi_{composed} = \begin{cases} \phi_{AUF} & if \quad R \le d_{\min} \\ \phi_{AUF} G(R-d_{\min},\delta) + \phi_{TUF}(1-G(R-d_{\min},\delta)) & if \quad R > d_{\min} \end{cases} \tag{6}$$

where

$$G(r,\delta) = e^{-\frac{r^2}{2\delta^2}} .$$

2.6 Evolutionary programming (EP)

In this method, there are some constants to be assigned such as $d_{e,} K_r$ in the move-to-goal univector field, K_o in the avoid-obstacle univector field, and $d_{\min}, \delta$ in composition of move-to-goal univector field and the avoid-obstacle univector field. To select proper values for these constants, Evolutionary Programming (EP) is applied. EP is one of evolutionary algorithm, which is a directed and stochastic algorithm. It has the population of individuals and each individual represents a potential solution at hand. Individual is assessed by the defined evaluation function. Selected individuals undergo the transformation such as mutation to generate the population of next generation. The best individual converges to a sub-optimal solution by repeating this procedure (D. B. Fogel [1994]).

To optimize the move-to-goal univector field and AUF, the conditions of the evaluation function in EP are as follow:

- The robot should not collide with obstacles.

- The robot should arrive at destination in finite time.

If either one of these conditions is not satisfied, the individual of population gets penalty in the evaluation stage.

First, the move-to-goal univector field for the navigation is optimized in a situation without obstacles and then the avoid-obstacle univector field with the previously optimized the move-to-goal univector field is optimized by EP. The evaluation function is defined as follows:

$$PI_i = tK_t + \theta_{error}^2 K_p + y_{error}^2 K_d \tag{7}$$

$$PI_{total} = \sum_{i=1}^{5} PI_i \tag{8}$$

where *t* is the elapsed time, θ_{error} is the orientation error and y_{error} is the position error. Fig. 8 illustrates the variables in the evaluation function. Every individual is tested at five positions and the final evaluation value is the summation of the five evaluation values by Eq. (8). The better individual has the lower value in the evaluation function.

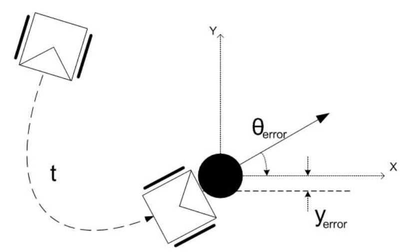


Fig. 8. Variables in the evaluation function of EP

3. EXPERIMENT

Robot soccer system for MiroSot, which is one of game categories for FIRA Cup (www.FIRA.net), has been used to experiment proposed univector field navigation (J.-H. Kim [1997] and J.-H. Kim et al. [2004]). The robot soccer system for experiments consists of a host computer, a vision system, a communication system and five soccer robots. The host computer is a Pentium 4 IBM PC. Uniqvision UC-685 10-bit color digital CCD camera is used in the vision system. Soccer robot is 7.5cm×7.5cm× 7.5cm in size and has a DSP TMS320F2811 PBK as a micro controller and two DC motors. The maximum speed of robot is 4 m/s.

A simulation program was used for parameter optimization and algorithm verification. The simulated robot was assumed that it did not slip and had limit acceleration speed.

3.1 Parameter Optimization

If a robot touched left side of the ball, it achieved the goal and stopped. (μ,λ) -strategy with $\mu = 10, \lambda = 20$ was applied (H.-P. Schwefel [1981]). The mutation was carried out by a self-adaptive Gaussian operator which is commonly used in EP (T. Back and H.-P. Schwefel [1993]). K_t, K_p, K_d in Eq. (7) were selected heuristically as $K_t = 10, K_p = 5, K_d = 2$. After 500 generations, each parameter became d_e =5.37(cm), K_r =4.15(cm), K_o =0.12, $d_{\min}$=3.48(cm), δ =4.57(cm).

3.2 Without Obstacles

Fig. 9 shows the simulation result which was the robot's movement without obstacles. Robots departed at four points toward the ball. The robot kicked the ball to the right side of field.

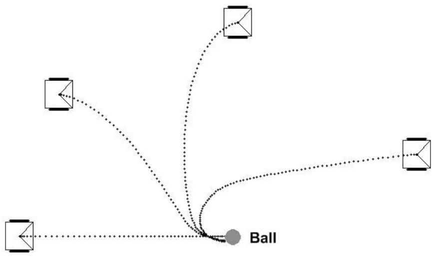


Fig. 9. Simulation without obstacles

3.3 Obstacle avoidance

Fig. 10 shows the robot's movement with one stationary obstacle. The robot should kick the ball toward the right without any collision with the obstacle. In the case of conventional navigation method using potential field, the center of repulsive vector field was the real obstacle. Therefore, the robot could not avoid the obstacle earlier, and it changed its direction hastily in the vicinity of the obstacle. However, the robot using proposed method moved more efficiently. There was no sudden direction change, and the movement was more stable.

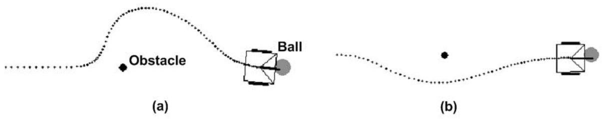


Fig. 10. Simulation with an obstacle. (a) Conventional potential field method (b) Proposed method.

3.4 Multiple obstacles

Fig. 11 shows the robot's movement with more than one obstacle. The robot using proposed method avoided obstacles quite well, but the robot using conventional potential field navigation moved inefficiently and it collided with an obstacle. The reason is the same as the situation avoiding one obstacle mentioned above.

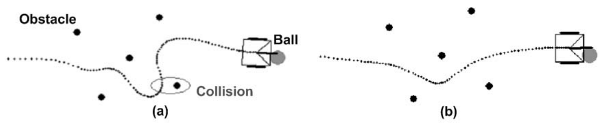


Fig. 11. Simulation with multiple obstacles.
(a) Conventional potential field method (b) Proposed method.

3.5 The movement around the obstacle

Fig. 12 depicts the robot's movement with obstacles. In this simulation, the obstacle did not interrupt the robot's path. However, in conventional navigation method using potential field, the path of the robot with the obstacle was different from the path without the obstacle. It means that the robot is influenced by the obstacle unnecessarily. On the other hand, in the proposed navigation, the path of the robot with the obstacle was almost the same as the path without obstacle. It indicates that unnecessary effect of the repulsive field around obstacles can be removed.

3.6 Moving obstacles

Fig. 13 shows the robot's movement with a moving obstacle. The robot in Fig. 13(a) did not consider the velocity of an obstacle, but the robot in (b) considered it. The robot in Fig. 13 (a) failed to avoid the obstacle, and it moved inefficiently. However, the robot in Fig. 13 (b) predicted the obstacle's movement and it could avoid the collision.

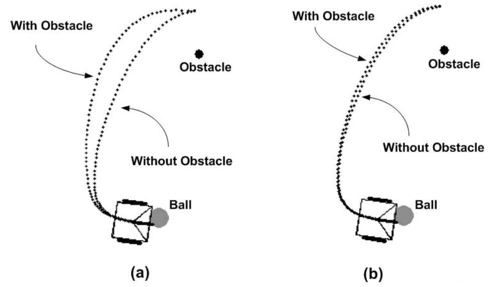
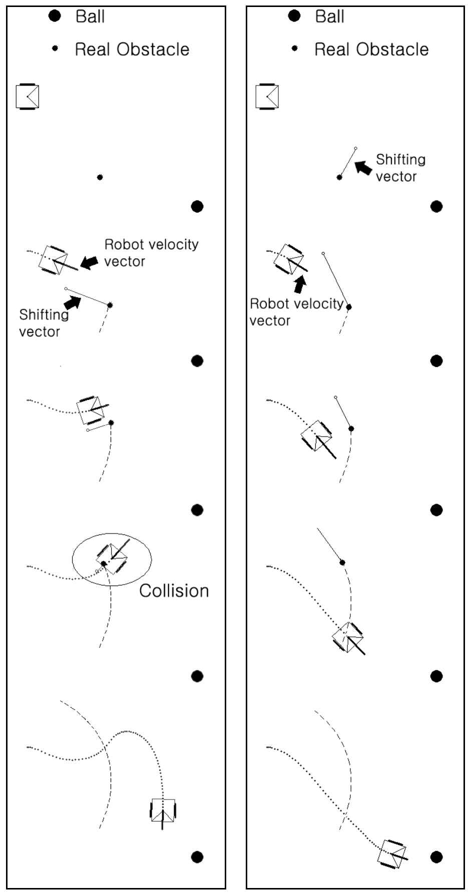


Fig. 12. Simulation that shows robot's movement around the obstacle. (a) Conventional potential field method (b) Proposed method.



(a) (b)

Fig. 13. (a) Simulation that did not consider the velocity of an obstacle. (b) Simulation that considered the velocity of an obstacle.

3.7 Results in MiroSot Environment

Fig. 14 is the results of the experiment with static obstacles. The white objects in Fig. 14 were static obstacles. The black line illustrated the trajectory of the robot. To show its movement, its postures are indicated at 0.3 second intervals. In Fig. 14 (a) the robot moved from the start point to the end point without collision. The robot in Fig. 14(b), (c), (d) also could avoid obstacles.

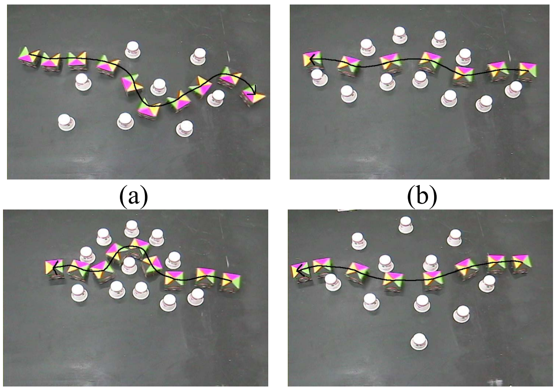


Fig. 14. Results of experiment with static obstacles

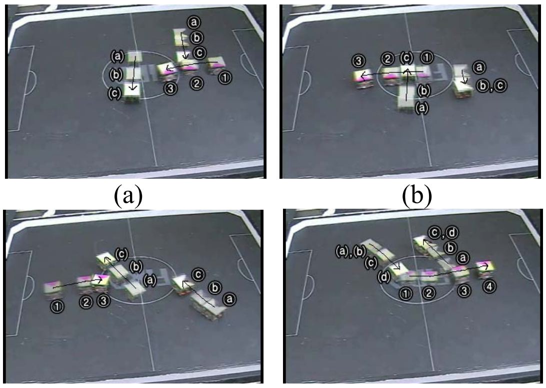


Fig. 15. Results of experiment with moving obstacles. The robot marked with ①→②→③ was a navigating robot, and other robots marked with alphabet were moving obstacles.

Moreover, the robots could avoid moving obstacles as Fig. 15 shows, where they navigated considering the position and the velocity of the moving obstacles.

The moving obstacles were realized by the same robots as the navigating robot. In Fig. 15, the trajectory of the navigating robot is marked with ①→②→③, while the trajectories of moving obstacles are marked with ⓐ→ⓑ→ⓒ and (a)→(b)→(c), as time passes. To illustrate the trajectories obviously, the number of via-points is limited to less than 4 in one picture.

In Fig. 15 (a), the obstacle moving along (a)→(b)→(c) interrupted the path of the robot. However, the robot decelerated its velocity considering the obstacle's velocity and avoided the collision. Fig. 15 (b), (c), (d) also show the trajectories of the robot using the same algorithm. These robots moved to the destination without any inefficient detour or any collision with obstacles.

4. CONCLUSION

This paper proposed a navigation scheme with collision avoidance using a univector field method for mobile robot with moving obstacles. Considering the symbiosis between human and robot in real life environment, the natural navigation is a crucial for a mobile robot. Human can easily avoid objects by considering their own movement as well as obstacle's movement. Similarly, proposed method planed the path by using the robot's own movement information along with the information of the obstacle's movement. This algorithm utilized a shifting vector which predicted expected movements of obstacles from the robot-oriented view. The shifting vector was calculated from the velocity information of the robot and obstacle with several constant parameters. The location of virtual obstacle was generated by adding the shifting vector to current obstacle's position. The robot could avoid obstacles by generating the univector field considering virtual obstacles. The constants used in navigation were optimized by evolutionary programming. The effectiveness of the proposed navigation algorithm was proved in both simulations and experiments using a soccer robot system.

REFERENCES

Chales C. Chang and Kai-Tai song (1997). Environment Prediction for a Mobile Robot in a Dynamic Environment. *IEEE Trans. On Robotics and Automation*, Vol. 13, No 6, pp. 862-872, Dec.

D. B. Fogel (1994). An Introduction to Simulated Evolutionary Optimization. *IEEE Trans. Neural Networks*, Vol. 5, pp. 3-14, Jan.

H.-P. Schwefel (1981). Numerical Optimization of Computer Models. Chichester, UK:John Wiley.

J. Borenstein and Y.Koren (1991). Real-time obstacle Avoidance for Fast Mobile Robots. *IEEE Trans. Syst. Man. Cybern.*, Vol. 7, No.3, pp. 327-328.

J.-H. Kim (1997). *Robotics and Autonomous Systems*, Special Issue: First Micro-Robot World Cup Soccer Tournament, MiroSot, vol. 21, no. 2, Sept.

J.-H. Kim, D.-H. Kim, Y.-J. Kim, K.-T. Seow (2004). *Soccer Robotics*, Springer Tracts in Advanced Robotics, Springer Verlag, 3540218599, 326, 326, Sep.

Kai-Tai Song and Charles C. Chang (1999). Reactive navigation in dynamic environment using a multisensor predictor. *IEEE Trans. Syst. Man. Cybern*, Part B, Vol. 29, No 6, pp. 870-880, dec.

P. J. McKerrow (1993). "Introduction to Robotics," Addison Wesley Publishing Company, pp. 431~435, 1991.

T. Back and H.-P. Schwefel, An overview of evolutionary algorithms for parameter optimization. *Evol. Comput.*, Vol.1, pp.1-23.

Y.-J. Kim, J.-H. Kim, and D.-S. Kwon (2001). Evolutionary Programming-Based Univector Field Navigation Method for Fast Mobile Robots. *Trans. Syst. Man. Cybern.*, Part B Vol. 31, No 3, pp. 450-458.

Y.-J. Kim (2003). Univector Field Navigation Method for Fast Mobile Robots, Korea Advanced Institute of Science and Technology, Ph.D. Thesis.