A Reactive Collision Avoidance Approach for Mobile Robot in Dynamic Environments

Tang S. H. and C. K. Ang
Universiti Putra Malaysia (UPM), Malaysia
Email: saihong@eng.upm.edu.my, ack_kit@hotmail.com

D. Nakhaeinia
University of Ottawa, Canada
Email: dania@uottawa.ca

B. Karasfi
UPM & Islamic Azad University, Qazvin Branch
Email: karasfi@qiau.ac.ir

O. Motlagh
Universiti Teknikal Malaysia
Email: motlagh7@gmail.com

***Abstract*—This paper describes a novel reactive obstacle avoidance approach for mobile robot navigation in unknown and dynamic environment. This approach is developed based on the "situated-activity paradigm" and a "divide and conquer" strategy which steers the robot to move among unknown obstacles and towards a target without collision. The proposed approach entitled the Virtual Semi-Circles (VSC). The VSC approach lies in integration of 4 modules: division, evaluation, decision and motion generation. The VSC proposes a comprehensive obstacle avoidance approach for robust and reliable mobile robot navigation in cluttered, dense and complex unknown environments. The simulation result shows the feasibility and effectiveness of the proposed approach.**

***Index Terms*—Dynamic environment, Mobile robots, Obstacle avoidance, Reactive navigation.**

I. Introduction

Obstacle avoidance task is one of the most important issues in the design and development of intelligent mobile robots [1]-[3]. It consists of the ability of a robot to generate a feasible and safe trajectory from the current robot location to a goal without collision. Dynamic (reactive) collision avoidance approaches in contrast to global (static) path planning do not need the global model of environment, the robot perceives its surrounding environment using different kinds of sensors to plan and executes local (reactive) navigation. These approaches generate control commands based on the current sensory information. Therefore, they have a quick response in reacting to unforeseen obstacles and uncertainties with changing the motion direction [4].

The first works on the local motion planning are Lumelsky's bug algorithm [5], the Khatib's Potential Fields method [6] and the Cox and Yap's method [7]. During last decades various approaches have been proposed which most of them are variations of some general approaches. Bornstein [8] proposed a real-time obstacle avoidance approach which entitled Virtual Force Field (VFF). This approach developed based on the two concepts of Certainty Grids and Potential Fields. The certainty grid concept used for representation of (inaccurate) sensory data about obstacles and Potential Fields hinges on the principle of repulsion and attraction forces where obstacles exert repulsion force and the target exerts an attractive force on the robot [9]. The Dynamic Window Approach [10] is a robust reactive obstacle avoidance approach which considers kinematic and dynamic constraints of the robot. It uses geometric operations and describes a search for commands controlling the velocities of the vehicle which is carried out directly to the velocity space. In another work developed by Minguez and Montano [11], they addressed a collision avoidance method which called Nearness Diagram (ND). In this approach the "divide and conquer strategy" used to simplify the navigation problems in troublesome scenarios. Shi et al [12], proposed a local obstacle avoidance that combines the prediction model of collision with a modified beam curvature method (BCM) to avoid moving obstacles in dynamic environments. The most challenging problems in the existing reactive navigation approaches include oscillatory motion, long path generation, moving among cluttered dynamic obstacles without collision or they require very large memory and computation [13]. In

Manuscript received July 3, 2012; revised December 17, 2012.

doi: 10.12720/joace.1.1.16-20

this work we address a new reactive approach for fast obstacle avoidance of mobile robots using *situated-activity paradigm* [14] and *divide and conquer strategy* [11]. First, the robot's status to the obstacle distribution is identified in a part of workspace within division and evaluation of the regions. Then based on the identified situations, a decision is made to choose a proper path toward the target and finally the robot conqueres the free spaces toward the target in the selected path.

II. Virtual Semi-Circles (VSC) Method

In the most reactive navigation approaches the challenge is to cope with cluttered, dense and complex scenarios which the robot should move among random obstacles. The proposed approach is called Virtual Semi Circles (VSC). The VSC method describes how the exiting navigation problem can be solved with simplified algorithms. The VSC path planning method is divided to 4 modules: division, evaluation, decision, motion generation.

A. Division

Fig. 1 shows geometric configuration of the robot in the X-Y plane. Where θ_r is the robot rotational angle from the horizontal axis, A_i is the difference angle between the robot's heading and obstacle, the range of A_i is [-90^0, 90°] and R_i is the distance between the sonar *i* and the obstacle. Assume the robot starts from position $x_i = x_1$, $y_i = y_1$ and $\theta_r = \theta_1$ then after a short time its position becomes $x_i = x_2$, $y_i = y_2$ and $\theta_r = \theta_2$.

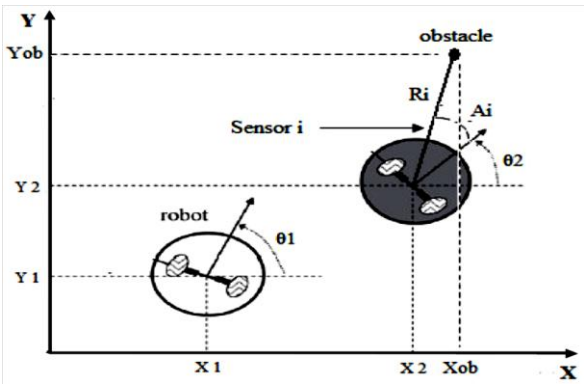


Figure1. Geometric configuration of robot in the *X-Y* plane

The obstacle positions detected by sonar *i* can be calculated by the following equation:

$$x_{ob} = x_i + R_i \cos(\theta_r + \mathbf{A_i})$$
$$y_{ob} = y_i + R_i \sin(\theta_r + \mathbf{A_i}) \qquad (1)$$

where, x_{ob} and y_{ob} are the obstacle position coordinates.

The robot is equipped with six sonar sensors with 35^0 radius of detection. The arrays of six sonar sensors (S0–S5) are shown on the robot Circumference (Fig. 2). For more accuracy in obstacle detection in the maximum range of sensors is set to *3m*. Corresponding to 6 sensors arrangement the robot work space is divided to six sub spaces (Fig. 2). The subspaces are *R* (Right), *RF* (Right-front), *FR* (Front-right), *FL* (Front-left), *LF* (Left-front) and *L* (Left) which represent target direction and obstacles position.

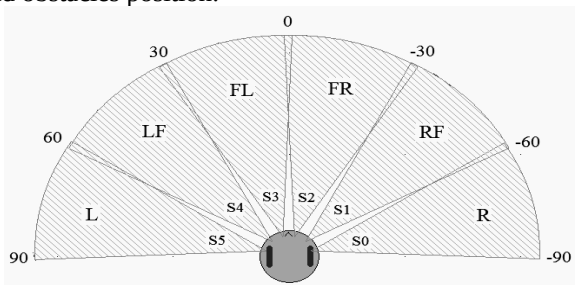


Figure2. Array of six sonar sensors on robot circumference and sub spaces

In addition, three semi-circles are assumed around the robot with radius of 1m, 2m, and 3m from its centre which divide the robot's work space to three regions. These subspaces represent the obstacles distance from the robot (Fig. 3):

1. *N (Near):* Inside of the semi-circle with radius of 1m;
2. *M (Middle):* The existing gap between the two semi-circles with radius of 1m and 2m; and
3. *F (Far):* The existing gap between two semi-circles with radius of 2m and 3m

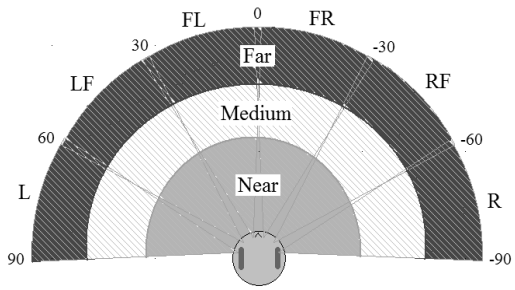


Figure3. The robot's subspaces

Therefore, the robot work space is divided to 18 regions totally (6×3=18). In each step of the robot movement the sensory information are updated and show the robot's status to the obstacles. Therefore, the accurate obstacle positions and free obstacle spaces are obtained corresponding to the sonar readings. For example if an obstacle detected by a sonar is inside the semi-circle with radius of *1m* and $-90 < A_i < -60$, the obstacle region is *NR (Near-Right)*. The *Near* region is considered as security zone for the robot. Since the algorithm needs to fulfil real time requirement, less number of regions is better for online computation. Therefore, the robot work space is divided to six angular division and three radius measurement. To increase the accuracy of the robot manoeuvrability, the subspaces can be changed (increased/decreased) according to the robot size and environment characteristics with changing radius of the semi-circles or angular divisions.

B. Evaluation

This module describes the robot and obstacles relation within evaluation of the regions. Fig. 4 represents that how each region gets a value of 1, 2 or 3 corresponding to the obstacles position (gray regions) from the robot. For

example if $2<S0<3$ then the *MR* value is 2. Therefore, the regions with higher value are considered as safe region which the regions should have the value of more than 1. In Fig. 4, the *FR* and the *L* regions have the higher values so they are the safest navigable regions toward the target. In each control period, the regions are updated based on the sensory information to identify the robot's situation. At the same time, the regions value are extracted from the obstacles position and free obstacle areas to aid the robot in decision making for next action. Therefore, the next step is to decide which safe region optimizes the navigation path and reaches the robot to the target without any collision in the shortest time.

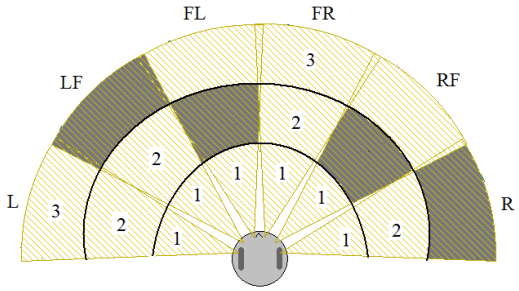


Figure4. Evaluation of the robot's work space region.

C. Decision

This module explains how the robot makes a decision to choose proper path toward the target among existing safe regions. First the situations are categorized as following:

a) *Target in safe region:* this happens when safe region and target have the same direction (Fig. 5(a)).

b) *Target in different region:* this happens when safe regions and target have different direction (Fig. 5(b)).

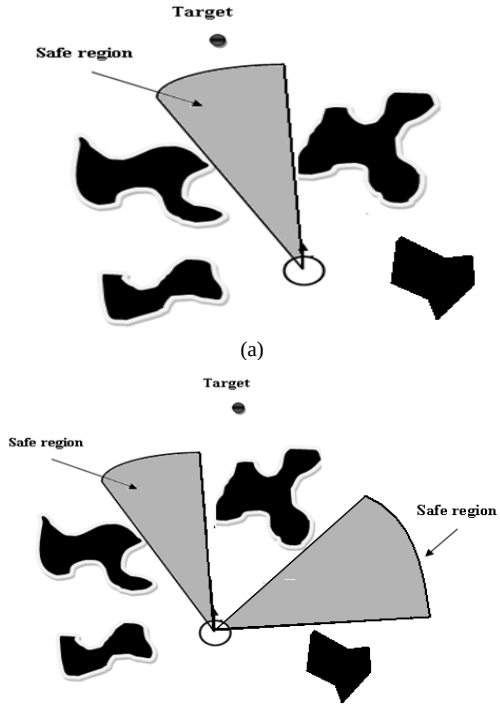


Figure5. (a) Target in a safe region, (b) Target in different region.

According to each situation, a proper action steers the robot toward the target without collision. The closest safe region to the target have the highest priority in choosing a path. Therefore, the priority is with the safe region which has the same direction with the target. However, If there are more than one safe region when target is in different region, the nearest safe region to the target direction is the best path.

D. Motion Generation

Once a proper path selected by the decision module, the robot conqueres the free spaces toward the target in the path. The motion commands are based on the safe region direction and value. Direction of motion (θ) is equal to the angle between the safe region direction and the robot heading. For example if the selected safe region is at *RF (Right Front)*, then $\theta= [(-30-60)/2] =-45$. The translation velocity (V_t) is calculated by the safe region value. Where the safe region value is 3 then V_t is maximum ($V_t=V_{max}$) otherwise, the robot velocity reduces to normal velocity ($V_t=Vn$). The integration of these modules adresses a versatile and comprehensive ractive obstacle avoidane approach for mobile robot. Fig. 6 depicts the robot's perfomance in a dense scenario which successfully chooses a safe region of motion and conqures the free obstacle area among the obstacles.

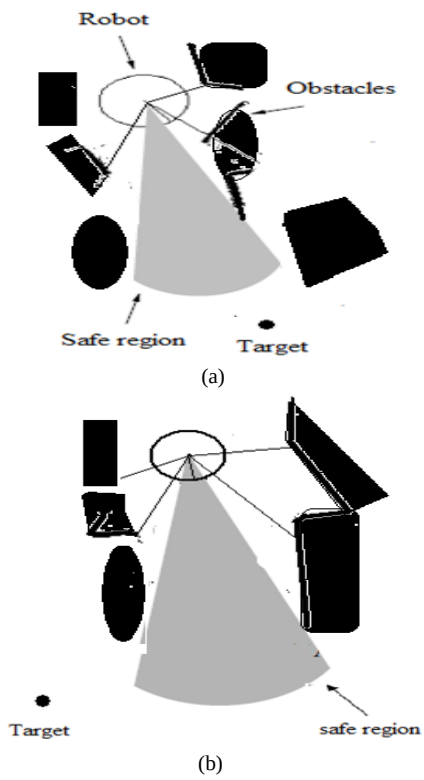


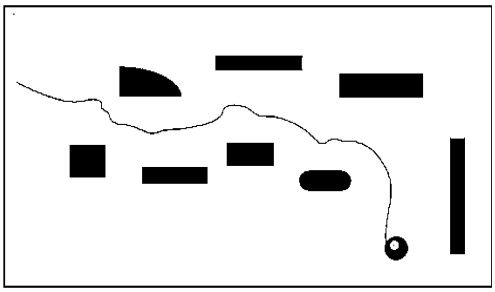
Figure6. Robot performance in dense scenarios with narrow places, (a) Target in safe region, (b) Target in different region.

III. Simulation Results

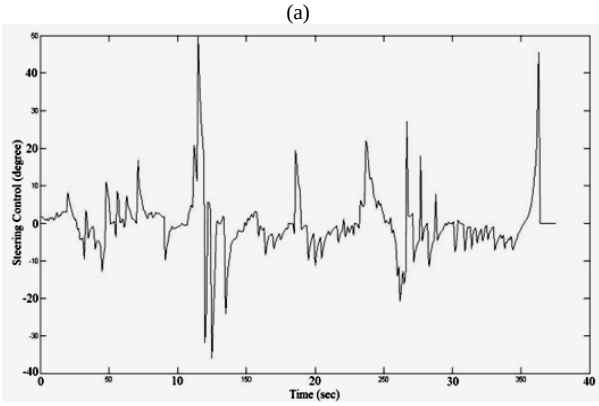
The simulation result proves the effectiveness and robustness of the proposed approach. In the simulation investigation, the robot has been modelled as a circle which is equipped with six sonar sensors for distance

measurement. The start and the final points are given and the sample environments are completely unknown. The maximum translation velocity is set to V_{max}=0.4 *m/s* and normal velocity is set to *Vn*=0.2 *m/s*.

In example 1 (Fig. 7(a)), the start point is at (x_r, y_r) = (6m, 11m) and the target point is at (x_t,y_t)= (17m, 10m). The robot reached the target without collision with obstacles. Fig. 7(b) shows the robot's steering control.

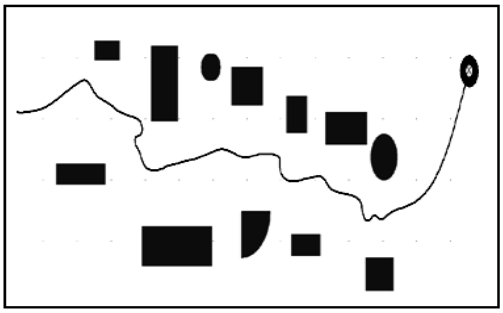


(a)

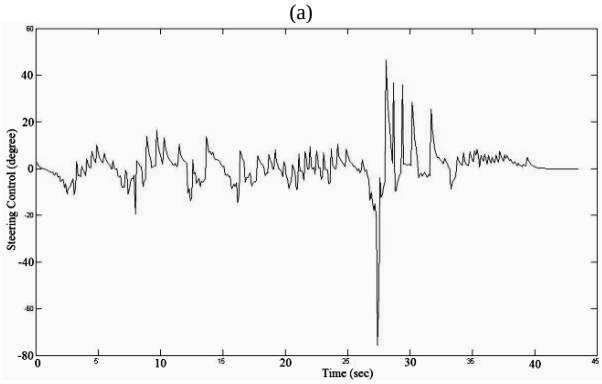


(b)

Figure 7. Trajectory executed in, (a) Example 1, (b) Steering control profile



(a)



(b)

Figure 8. Trajectory executed in, (a) Example 2, (b) Steering control profile.

In example 2 (Fig. 8(a)), the start point is at (x_r, y_r) = (5m, 12m) and the target point is at (x_t,y_t)= (14m, 15m). The robot navigated in a dense, complex, and cluttered environment among narrow passages while avoiding obstacles. Fig. 8(b) represents the robot's changing direction toward the target.

IV. Conclusion

This paper proposed novel reactive obstacle avoidance approach for mobile robot navigation in unknown environment. The VSC method can be applied to implement reactive navigation methods adapted to the obstacle avoidance context. The perception-action process and cooperation of the modules reduced the tasks difficulty and increased the reactivity. The VSC approach differs from the existing methods in the use of simple algorithm with high efficiency, integrating different modules which do not require very large memory and computation. However, the approach cannot obtain the shortest path.The simulation result showed that the robot autonomously avoids collision with the obstacles in very dense, cluttered and complex unknown environments.

References

[1] Y. C. Kim, S. B. Cho, and S. R. Oh, "Map-building of a real mobile robot with GA-fuzzy controller," *International Journal of Fuzzy Systems*, vol. 4, pp. 696-703, 2002.

[2] S. M. Homayouni, S. H. Tang, and N. Ismail, "Development of genetic fuzzy logic controllers for complex production systems," *Computers and Industrial Engineering*, vol. 57, no. 4, pp. 1247-1257, 2009.

[3] O. Motlagh, S. H. Tang, N. Ismail, and A. R. Ramli, "A review on positioning techniques and technologies: A novel AI approach," *Journal of Applied Sciences*, vol. 9, no. 9, pp. 1601-1614, 2009.

[4] S. H. Tang, D. Nakhaeinia, and B. Karasfi, "Application of fuzzy logic in mobile robot navigation," in *Fuzzy Logic-Controls, Concepts, Theories and Applications*, March 2012.

[5] V. J. Lumelsky and A. A. Stepanov, "Dynamic path planning for a mobile automation with limited information on the environment," *IEEE Transaction*, vol. 31, pp. 1058-1063, 1986.

[6] M. Khatib and J. J. Saade, "An efficient data-driven fuzzy approach to the motion planning problem of a mobile robot," *Fuzzy Sets and Systems*, vol. 134, pp. 65-82, 2003.

[7] J. Cox and C. K. Yap, "On-line motion planning: moving a planar arm by probing an unknown environment," Courant Institute of Mathematical Sciences, New York University, New York, 1988.

[8] J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robots," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, pp. 1179-1187, 1989.

[9] D. Nakhaeinia, S. H. Tang, B. Karasfi, O. Motlagh, and A. C. Kit, "Virtual force field algorithm for a behaviour-based autonomous robot in unknown environments," in *Proc. Inst. Mech. Eng. Part I-J Syst Control Eng*, vol. 225, 2011, pp. 56-62.

[10] D. Fox, W. Burgard, and S. Thrun, "Controlling synchro-drive robots with the dynamic window approach to collision avoidance," in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, 1996, pp. 1280 - 1287.

[11] J. Minguez and L. Montano, "Nearness Diagram (ND) Navigation: Collision Avoidance in Troublesome Scenarios," *IEEE Trans. Robot. Autom*, vol. 20, pp. 45-59, 2004.

[12] C. Shi, Y. Wang, and J. Yang, "A local obstacle avoidance method for mobile robots in partially known environment," *Robotics and Autonomous Systems*, vol. 58, pp. 425-434, 2010.

[13] D. Nakhaeinia, S. H. Tang, S. B. Mohd Noor, and O. Motlagh, "A review of control architectures for autonomous navigation of mobile robots," *Int. J. Phys. Sci.*, vol. 6, pp. 169-174, 2011.

[14] R. C. Arkin, "Behavior-Based robotics," Cambridge, MA: The MIT press, 1998.

BIOGRAPHIES

Tang Sai Hong received his PhD and BEng from Dublin City University and Universiti Pertanian Malaysia, respectively. He is an Associate Professor and attaches with the Department of Mechanical & Manufacturing Engineering, Universiti Putra Malaysia since 1997. Currently, he works in the fields of robotics, operations research and artificial intelligence.

Danial Nakhaeinia received his Bachelor and Master degrees from Iranian University and Universiti Putra Malaysia, respectively. Currently, he is pursuing his PhD in University of Ottawa. His research focus includes robotic motion planning and artificial intelligence.

Babak Karasfi received his Bachelor and Master degrees from Iranian Universiti+es. Currently, he is pursuing his PhD in Universiti Putra Malaysia. He is also an academic in the Islamic Azad University, Qazvin Branch. At this moment, he is focusing in robotic motion planning and artificial intelligence.

Omid Reza Esmaeili Motlagh received his Bachelor degree from Iranian University, while obtaining his MSc and PhD from Universiti Putra Malaysia. After working as a Fellow Post-Doc for two years in Universiti Putra Malaysia, he became a Senior Lecturer in Universiti Teknikal Malaysia. His research focus includes robotic motion planning and artificial intelligence.

Ang Chun Kit received his Bachelor degree from UCSI University, Malaysia. Currently, he is a PhD candidate of Universiti Putra Malaysia. His research focus includes robotic motion planning and artificial intelligence.