

Development of a Sensor-Based Approach for Local Minima Recovery in Unknown Environments

Danial Nakhaeinia¹, Tang Sai Hong² and Pierre Payeur¹

¹School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON, Canada

²Department of Mechanical and Manufacturing Engineering, University Putra Malaysia, Selangor, Malaysia
[dnakhaei, ppayeur]@uottawa.ca, saihong@eng.upm.edu.my

Abstract—This paper introduces a new methodology for escaping from local minima using an actual-virtual target switching strategy. In particular, this approach proposes suitable steps to detect trap situations and guide the robot away from local minima even when the environment is completely unknown. In this work the navigation system consists of two layers. In the low-level layer, a Nearest Virtual Target (NVT) approach is adapted as a reactive collision avoidance method for mobile robot navigation to achieve collision free motion in cluttered, dense and troublesome scenarios. Where the robot is surrounded by obstacles and a trap situation is likely to occur, the high-level layer becomes responsible to plan a path to pull the robot out of the trap. Finally, the performance of the proposed approach is validated by simulation results.

Keywords— mobile robots; sensor-based navigation; obstacle avoidance; virtual target; fuzzy control; path planning.

I. INTRODUCTION

Autonomous navigation is a fundamental issue in the design and development of intelligent mobile robots. It is the process of generating a feasible and safe trajectory from the current robot location to a goal without collision and in different workspaces. The workspace can be constructed, known or unstructured, or even unknown and dynamic [1]. Various approaches have been developed for reactive navigation in unknown and dynamic environments. Although, a number of successful approaches have been proposed to solve navigation problems in the presence of uncertainty in dynamic environments, most of the approaches have shortcomings in dealing with local minimum situations which are also called dead end, limit-cycle or deadlock. In the literature, boundary-following methods [2, 3], actual-virtual target (virtual sub-goal) strategies [4-6] and behavior integration methods [7-10] are often used to address the local minima problems. Among these approaches, virtual sub-goal methods are more promising for dealing with local minima traps. However, they may overproduce virtual targets, get stuck in trap situations, regress into the old dead ends or set the virtual target in an unreachable place [11, 12]. In our previous work [13], a reactive collision avoidance method, called Nearest Virtual Target (NVT), was proposed to overcome navigation problem in unknown and dynamic environments. This approach works well in dealing with dynamic and troublesome scenarios but according to the sensory limitations, it is difficult to determine the size or location of obstacles, and

as a result, the robot is not able to escape from some invisible traps. To overcome the navigation problem in local minima scenarios, this work presents the design of a new approach which includes two layers: in the first layer (low-level), the NVT approach [13] is applied to compute collision free motion in unknown and dynamic environments. In the second layer (high-level), a local-minimum planner is designed using the actual-virtual target strategy to avoid trap situations. Where a trap situation is detected, the planner obtains a virtual target outside of the trap using a set of heuristic rules and creates a path for the robot to move away from the local minimum.

II. PROPOSED APPROACH DESIGN

In the proposed approach, the action selection and the interaction between the layers are based on the obstacles configuration. The action selection algorithm starts by constructing a local occupancy map. Initial locations of the robot and the global target are set arbitrarily by the user for each navigation task according to a base frame with origin at the lower left side of the simulation environment.

The information about obstacles position and the areas free of obstacles is provided by an on-board range sensor to identify navigable areas in the robot path towards the target. If a proper navigable area (safe region) exists in the robot path towards the global target, the NVT approach is applied to define an optimal collision free path. To do so, it generates temporary virtual targets to guide the robot towards the global target. However, if there is no navigable area, a trap situation is likely to occur and the proposed local minimum planner becomes responsible to obtain an alternative path.

A. Nearest Virtual Target Method

The approach is designed to achieve comprehensive obstacle avoidance for fast-moving mobile robots in unknown and dynamic environments [13]. In this approach a laser range finder provides a local model of the environment to identify obstacles and navigable areas around the robot in different situations. The model is updated when new information about the environment is collected by the sensor. As shown in Fig. 1(a), if there are obstacles over the straight-line path between the robot and the global target defined by the user, navigable areas (safe regions) are first identified in the neighbourhood of the robot. An obstacle free area is navigable when it is wide enough for the robot to safely traverse it while moving

towards the target. Then the obstacle avoidance planner generates a virtual target corresponding to each safe region. The virtual target which generates the shortest path towards the target is selected and guides the robot. As shown in the example of Fig. 1(b), virtual target 1 generates the shortest path in this scenario, the corresponding path is therefore selected. The path keeps being updated using the new sensory information at each step. Finally, motion generation is handled by a fuzzy logic controller (FLC) and a suitable action is taken in response to each situation. The proposed fuzzy controller [13] has two inputs and two outputs. The FLC inputs are obstacle position (OP) and obstacle distance (OD). For 3-set partitioning of the OP and 5-set partitioning of the OD the fuzzy rule bases contains 15 rules (Table 1). After fuzzification of inputs, the fuzzy interference converts fuzzy input sets to outputs. These fuzzy outputs are the Rotational Velocity (R_v) and the Translational Velocity (T_v). The rotational and translational velocities change according to the obstacles distribution. Where the robot is not surrounded with obstacles and the workspace is not very dense and cluttered, the robot can move with a higher speed towards the target in areas free of obstacles. However, the robot speed reduces in the presence of obstacles to prevent collision with them over the robot path towards the target.

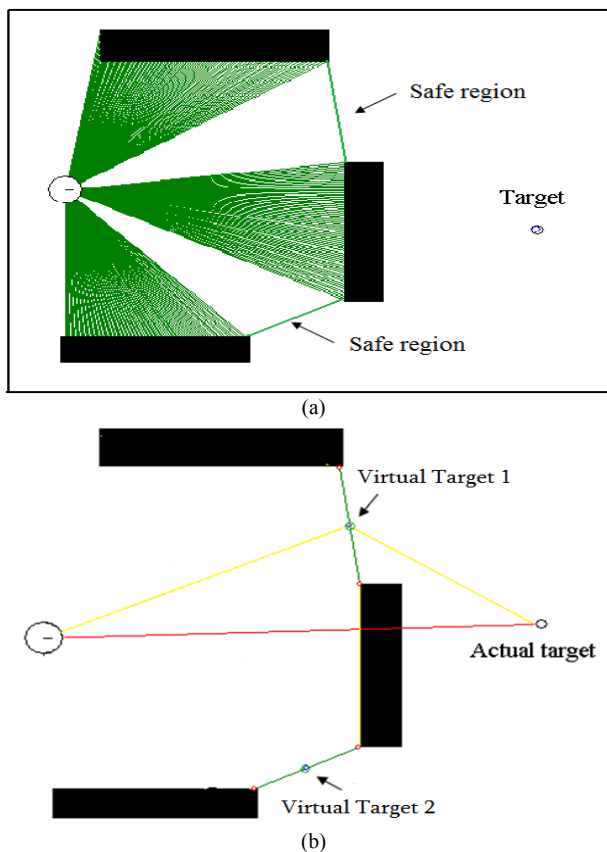


Fig. 1. (a) Definition of safe regions, and (b) virtual target 1 guides the robot towards the actual target over the shortest collision-free path.

The advantage of this approach is that it can drive the robot successfully towards the target in a priori unknown or dynamic environments (Fig. 2). Furthermore, it enables the robot to avoid trap situations where the local minimum is identified (visible) using the local model of the environment. A local minimum is visible when the robot can detect the local trap situation completely, that is

when the problematic configuration of space lies within the range sensor field of view and depth of field, as exemplified in Fig. 3(a). When the local minimum is visible, the NVT creates a path using the existing safe region outside of the local minimum for the robot to move away from the trap (Fig. 3(b)). However, the shortcoming of the NVT approach described above is that the robot cannot find its way out of a local minimum when that minimum is invisible, that is when the local minimum cannot be detected using the local model of environment.

TABLE 1. The Fuzzy rule base.

If		Then				
Rule no.	OP		OD	R_v		T_v
1	D	&	VN	Large	&	Slow
2	U	&	N	Medium	&	Medium
3	S	&	M	Smooth	&	Fast
4	D	&	F	Large	&	Medium
5	U	&	VF	Medium	&	Fast
6	S	&	VN	Medium	&	Medium
7	D	&	N	Large	&	Slow
8	U	&	M	Medium	&	Medium
9	S	&	F	Smooth	&	Fast
10	D	&	VF	Large	&	Medium
11	U	&	VN	Medium	&	Slow
12	S	&	N	Smooth	&	Medium
13	D	&	M	Large	&	Slow
14	U	&	F	Medium	&	Medium
15	S	&	VF	Smooth	&	Fast

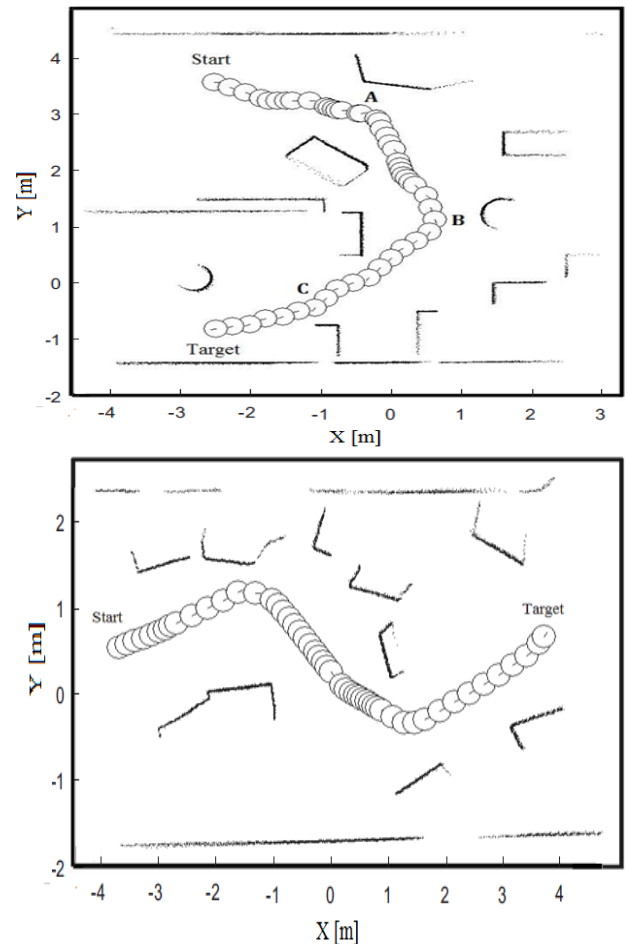


Fig. 2. Robot navigation using the NVT approach in unknown environments.

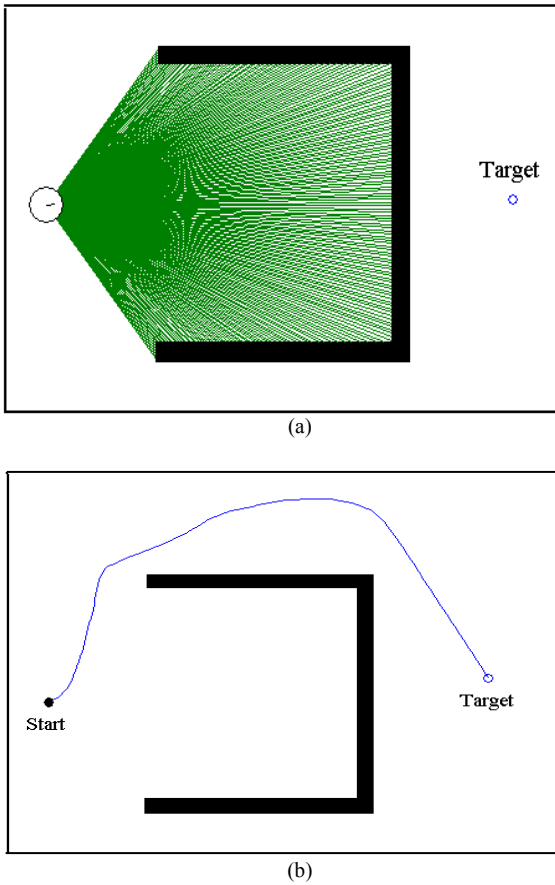


Fig. 3. (a) Example of a visible local minimum; (b) The NVT steers the robot to move away from visible local minimum.

As shown in Fig. 4(a), according to the sensor's limitations, the robot is not able to entirely map the region that creates the local minimum on parts of it that lie beyond the sensor's depth of field. As a result, there is navigable area in the direction of the target that would apparently allow it to move towards the target. In such situation, the robot driven by the NVT approach alone inherently moves towards the inside of the bounded contour and eventually reaches a local minima where it gets trapped (Fig. 4(b)). To overcome the problem, the present work adopts and implements a local minimum planner (LMP) using the "actual-virtual target switching" strategy to avoid the trap situations and to find alternative reliable and traversal paths towards the target.

B. Local Minimum Planner Method

A local minimum situation typically occurs when the target is located aside a long wall, concave obstacles, or maze-like and u-shaped environments. This section introduces a local minimum planner (LMP) method to avoid invisible local minima situations. The LMP is a set of heuristic rules that requires no memorizing. A local minimum detection criterion is defined that corresponds to situations where the robot is surrounded by the obstacles and there is no safe region (navigable area) in the possible robot paths that connect to the target (Fig. 4).

The proposed local minimum planner developed under the actual-virtual target switching strategy involves that each time the local minimum trap criterion is satisfied, a new virtual target is generated and this virtual target is appointed to replace the actual global target temporarily, until the robot gets out of the trap and reaches to the

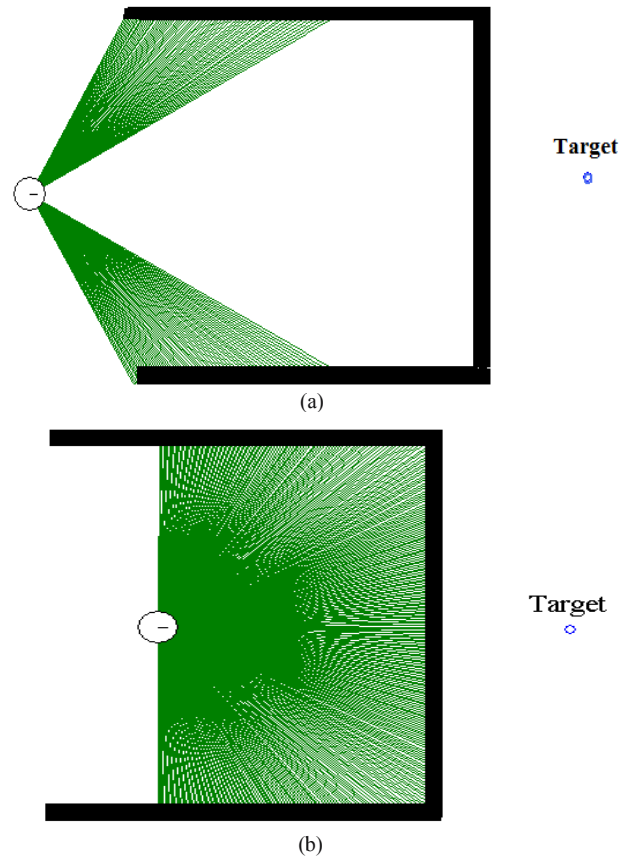


Fig. 4. Example of an invisible local minimum: (a) the robot cannot detect the local minimum, and (b) the robot gets trapped in the local minimum.

virtual target. However, since the environment is assumed not globally known, the virtual target might be placed on an obstacle, or not in a reachable location. Therefore, it is not required that the robot reaches exactly the virtual target, but it should move in its direction. Once the robot gets close to the virtual target, then the current target switches back to its previous location (either that of the actual global target, or that of a previous virtual target if there was one defined) and the robot reorients its navigation accordingly. Another condition which is applied to make sure that the robot would not get trapped when dealing with long walls and local minima bounded by large obstacles, is that if the robot experiences a virtual target at the same location more than once, this situation is considered as a trap situation by default, and navigation toward that virtual target location is not pursued.

The virtual target location is computed as a function of the distance between the robot and the current actual/virtual target (RTD), the obstacles position, and the difference angle between the robot heading orientation and the relative target direction (RTA), as shown in Fig. 5.

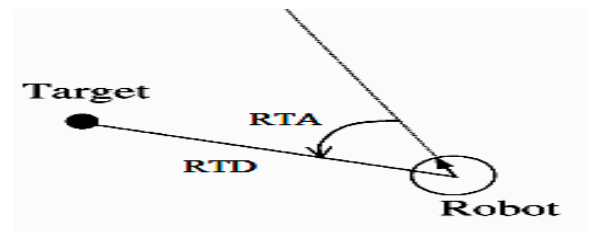


Fig. 5. Definition of RTA and RTD [13].

To guide the robot out of a local minimum the virtual target should be located outside of the trap. Therefore, whereas a local minimum is likely to occur, the actual target translates and rotates around the robot centre according to the obstacles configuration and the RTA. The virtual target translation (VTD) is defined using the first or last obstacles position, within its possible scanning directions (Fig. 6), detected by the sensor on the left side (ROL) or the right side (ROR) of the robot (Fig. 7). Then, if the RTA > 0, the target rotates counter clockwise, and if RTA < 0, then the target rotates clockwise about the robot centre. Therefore, the new virtual target location is calculated as follows:

$$\text{If } RTA > 0 \text{ then} \\ \theta = RTA \text{ and } VTDx = ROL_x + \alpha_x, \quad VTDy = ROL_y + \alpha_y \quad (1)$$

$$\text{If } RTA < 0 \text{ then} \\ \theta = -RTA \text{ and } VTDx = ROR_x + \alpha_x, \quad VTDy = ROR_y + \alpha_y \quad (2)$$

$$\alpha_x = \alpha \cos(\theta_0) \\ \alpha_y = \alpha \sin(\theta_0) \quad (3)$$

where the α parameter is an experimentally determined distance to locate the virtual target out of the trap with a safe distance from the obstacles (in this work $\alpha = 60$ cm), θ_0 is the ROR or ROL angle with respect to the base frame and θ is the rotation angle about the robot centre.

The new i^{th} virtual target location can be computed as follows:

$$\begin{bmatrix} T_{xi} \\ T_{yi} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & X_R \\ 0 & 1 & Y_R \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -X_R \\ 0 & 1 & -Y_R \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} VTDx \\ VTDy \\ 1 \end{bmatrix} \quad (4)$$

where $i = \{1, 2, \dots\}$ shows the number of virtual targets created each time a new trap is detected; T_{xi} and T_{yi} are the i -th virtual target coordinates, X_R and Y_R are the robot coordinates, T_{x0} and T_{y0} refer to the actual global target coordinates which are defined by the user, and θ is the rotation angle (Fig. 8).

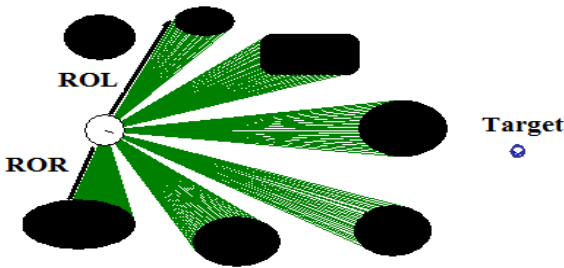


Fig. 6. ROL and ROR definition.

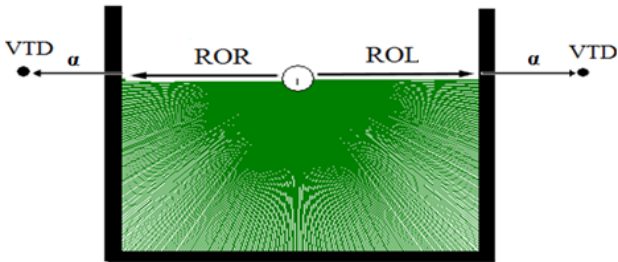


Fig. 7. Definition of VTD in a local minimum situation.

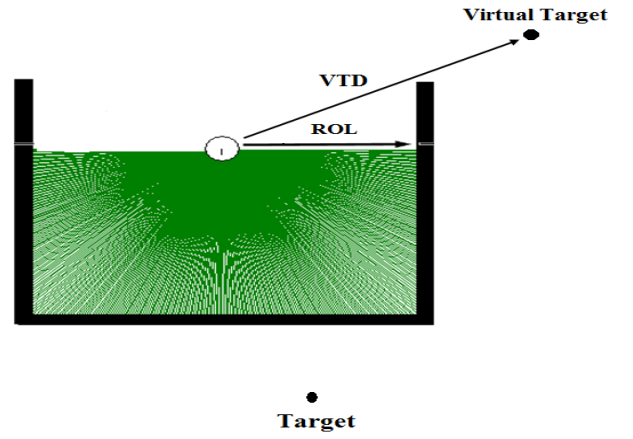


Fig. 8. Virtual target position in a local minimum situation.

As represented in Fig. 9, point (A) is a virtual target generated by the NVT. When the robot reaches to point (A), it moves toward the global target but a local minimum trap is detected. Therefore, the LMP produces a virtual target outside the wall at point (B) to avoid the local minimum. Point B is created by the LMP because a trap is detected and the virtual target at point (B) temporarily replaces the global target. The generated path by LMP guides the robot to move away from the trap while the NVT steers the robot towards the virtual target (which is temporarily defined as the current target to reach) to deal with the obstacles in the environment. When the robot is close to the virtual target (point B), the current target switches back to the global target location and the path is completed.

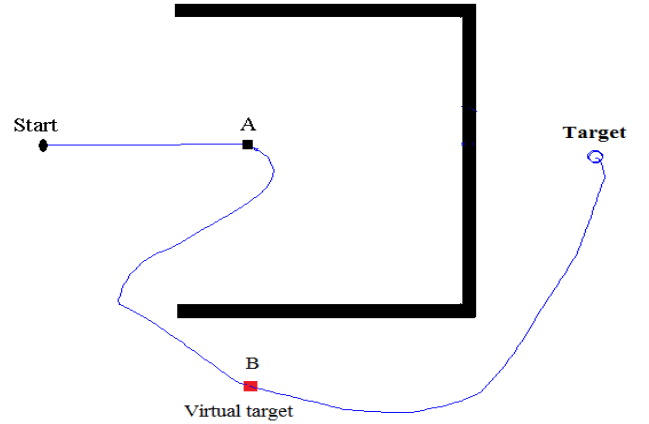


Fig. 9. Robot behavior in a local minimum situation.

III. SETTING AND SIMULATION RESULTS

In simulation investigations, the mobile robot has been modeled by a circular object with radius of 14 cm operating in a two-dimensional workspace. The workspace dimension is 600x600 cm. A straight line marked over the robot shows the robot's heading direction. The specific shape of the robot, and its kinematic and dynamic constraints, are not taken into account. The robot task is to move from a start point to a global target point defined by the user. The represented mobile robot is equipped with a 2-D laser range finder (LRF) to detect obstacles and to measure distances on the periphery of the robot. The maximal range of the LRF (d_{\max}) is limited to 2 m. The time interval for each laser

scan takes 0.2 ms. The maximal range of the LRF can be increased or decreased according to the workspace configuration and dimension. For example in a vast environment with large obstacles a higher detection range is preferable and more efficient.

To verify that the proposed method complies with the objectives of this work, some representative scenarios are carried out and presented here in different environments.

Example 1: Recursive U-shaped environment

In this example, the robot must reach the target location avoiding a recursive U-shaped obstacle. The robot enters the local minimum region located within the U-shape obstacle, being initially attracted by the global target, but avoids getting trapped by means of the virtual targets generated by the LMP out of the U-shaped obstacle. Directions toward the obstacle free areas were selected by the NVT approach during the navigation. As shown in Fig. 10, first the robot moves toward the global target before the trap situation is detected at point “a”. The first virtual target is then generated by the LMP in point “A”. Next, the robot follows the new virtual target direction, toward point “b”, until a new trap is detected. Thus, another virtual target is generated in point “B” which guides the robot to move outside of the U-shape obstacle area. The robot keeps following this second virtual target direction until it reaches to point “B”. Then, the virtual target switches back to its previous location in point “A”. Finally, when the robot reaches the virtual target in point “A”, it moves toward the actual global target which is now reachable. During the entire navigation process, the NVT is responsible for obstacle avoidance while following the current actual or virtual target, as selected by the LMP method. The empty circles (a, b, c, d, e, f and g) show the virtual targets generated by the NVT to avoid obstacles.

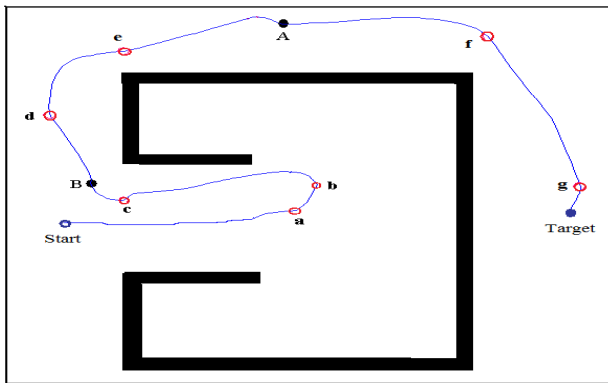


Fig. 10. Example 1: Trajectory executed in a recursive U-shaped environment.

Example 2: Motion in a corridor with several local minima

The effectiveness of the LMP is illustrated in Fig. 11 in an environment with a different topology made of a discontinued wall. At the start point, the actual target is located on the left side of robot. The robot can perceive a navigable area with its embedded sensors located on its left. Therefore, it favors that direction to initiate the navigation toward the actual global target. Within this detected navigable area, the first virtual target is generated

by the NVT (point “a”). However, once the robot reaches at point “a”, it cannot find a navigable area towards the global target as the range sensor now perceives the obstacle separating the robot’s position to the global target. At this point, a trap situation is detected and the LMP becomes responsible to generate a new virtual target out of the trap. The black circle (point “A”) shows the location of the resulting virtual target. The robot then moves toward the LMP generated virtual target while intermediate NVT generated virtual targets (denoted by red empty circles) steer the robot to avoid obstacles. The navigation in this environment shows that the robot succeeds to escape from the local minimum and the proposed algorithm properly supports navigation in such a complex environments. However, since the environment is unknown and the robot makes a decision according to the local model of environment, in some cases it might not find the shortest path towards the target.

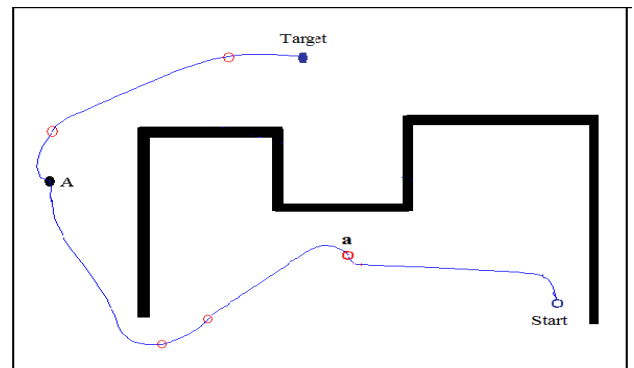


Fig. 11. Example 2: Trajectory executed in a corridor with several local minima.

Example 3: Robot surrounded by obstacles

This example demonstrates the performance of the proposed algorithm when the robot is surrounded by several obstacles and there is not enough space for the robot to pass in between the obstacles while moving towards the global target (Fig. 12). This situation is considered as a trap situation, and the local minimum planner (LMP) is involved to generate a virtual target outside of the trap. After the LMP generates the virtual target (point “A”), the NVT steers the robot towards this new target via intermediate virtual targets (red empty circles) which are located in safe regions according to the updated sensory information.

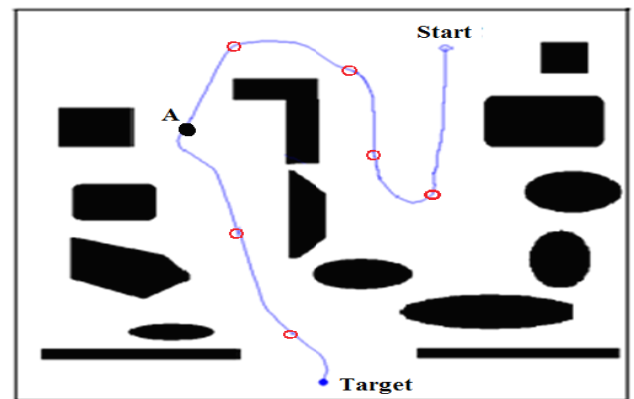


Fig. 12. Example 3: Trajectory executed in an environment where the robot is surrounded by several obstacles.

More examples of complicated environments with cluttered obstacles and trap situations are shown in Fig. 13 to demonstrate the effectiveness and robustness of the proposed approach.

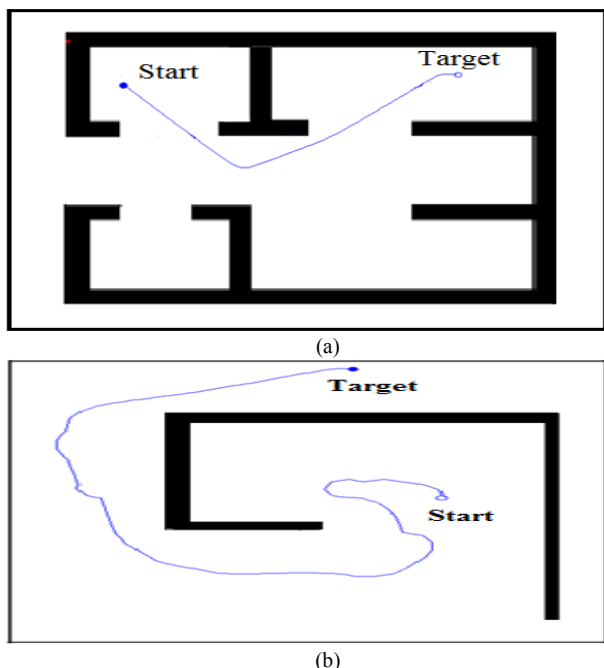


Fig. 13. Escape from trap situations in complicated environments with cluttered obstacles.

Comparison of performance of the proposed approach with some related works [2, 6, and 9] in a recursive U-shaped environment (Fig. 9) shows the effectiveness of the work. In [9] the robot motion is based on trial and error navigation through which the robot has to explore the whole dead end by making an extra go and return motion to find the exit. This approach can successfully reach the global target but the procedure is exhaustive, lengthy in time, produces long paths, and wastes robot's energy. The virtual target method [6] proposes a good approach to solve the limit-cycle problem of a fuzzy behavior-based mobile robot. However, it fails to reach the target in recursive U-shaped environments when it detects a new dead end. Krishna and Kalra [2] propose a landmark recognition approach which improves over the virtual target method of Xu *et al.* [6]. A dead end is detected by recognizing previously encountered landmarks. Then the robot follows the wall boundary to exit the dead end. This approach is not suitable for navigation in complex environments since it is difficult to choose a wall following direction and the trap detection highly depends on landmarks availability and their recognition.

IV. CONCLUSION

This paper proposes a heuristic approach for mobile robot navigation and local minima recovery in a priori unknown environments. The aim is to dynamically plan a path when a trap situation is detected to guide the robot out of the trap while avoiding obstacles. The approach that is introduced is formed of two layers. In the low-level layer, a nearest virtual target (NVT) method is presented to compute an optimal obstacle free path toward the current target, whether it be the global target or an intermediate virtual target. In the high-level layer,

activated upon the detection of a local minimum trap situation, a local minimum planner (LMP) presents a simple and efficient algorithm to plan an intermediate trajectory for steering the mobile robot outside of the local minimum trap situation. The two methods are combined using an actual-virtual target switching strategy. The target switching occurs when an obstacle or a trap situation is detected in the robot path towards the current target and an intermediate virtual target temporarily replaces the target to free up the robot. The switching is alternatively controlled by the NVT or LMP methods according to the obstacles configurations. Simulation results verified the effectiveness the proposed approach in dealing with local minima and trap situations under numerous 2D workspace configurations.

REFERENCES

- [1] D. Xu, L. Han, M. Tan, and Y. F. Li, "Ceiling-Based Visual Positioning for an Indoor Mobile Robot with Monocular Vision," *Industrial Electronics. IEEE Transactions*, vol. 56, pp. 1617-1628, 2009.
- [2] K. M. Krishna and P. K. Kalra, "Perception and Remembrance of the Environment During Real-Time Navigation of a Mobile Robot," *Rob. Autom. Syst.*, vol. 37, pp. 25-51, 2001.
- [3] D. Nakhaeinia, S. H. Tang, B. Karasfi, O. Motlagh, and A.C. Kit, "Virtual Force Field Algorithm for a Behavior-based Autonomous Robot in Unknown Environments," *Proc. Inst. Mech. Eng. Part I-J Syst Control Eng.*, vol. 225 (1), pp. 51-62, 2011.
- [4] O. Motlagh, S. H. Tang, and N. Ismail, "Development of a New Minimum Avoidance System for a Behaviour-based Mobile Robot," *Fuzzy Sets Syst.*, vol. 160(13), pp. 1929-1946, 2009.
- [5] C. Ordóñez, E. G. Collins Jr, M. F. Selekwa, and D. D. Dunlap, "The Virtual Wall Approach to Limit Cycle Avoidance for Unmanned Ground Vehicles," *Rob. Autom. Syst.*, vol. 56 (8), pp. 645-657, 2008.
- [6] W. L. Xu, S. K. Tso, and Z. K. Lu, "A Virtual Target Approach for Resolving the Limit Cycle Problem in Navigation of a Fuzzy Behaviour-based Mobile Robot," *Conf. on Intelligent Robots and Systems*, Victoria, B.C., 1998.
- [7] O. Motlagh, D. Nakhaeinia, S. H. Tang, B. Karasfi, and W. Khaksar, "Automatic Navigation of Mobile Robots in Unknown Environments," *Neural Computing and Applications*, Springer (ed.), 2013.
- [8] W. Khaksar, S.H. Tang, M. Khaksar and O. Motlagh, "Sampling-Based Tabu Search Approach for Online Path Planning," *Advanced Robotics*, vol. 26(8-9), pp. 1013-1034, 2012.
- [9] M. Wang and J.N.K Liu, "Fuzzy Logic-based Real-time Robot Navigation in Unknown Environment with Dead Ends," *Rob. Autom. Sys.*, vol. 56, pp. 625-643, 2008.
- [10] H. Seraji and A. Howard, "Behavior-based Robot Navigation on Challenging Terrain: A Fuzzy Logic Approach," *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 308-321, 2002.
- [11] 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(2), pp. 169-174, 2011.
- [12] J. Minguez and L. Montano, "Nearness Diagram (ND) Navigation: Collision Avoidance in Troublesome Scenarios" *IEEE Trans. Robot. Autom.*, vol. 20 (1), pp. 45-59, 2004.
- [13] D. Nakhaeinia and B. Karasfi, "A Behavior-Based Approach for Collision Avoidance of Mobile Robot in Unknown and Dynamic Environments," *Journal of Intelligent and Fuzzy Systems*, vol. 24 (2), pp. 299-311, 2012.