

Real Time Adaptive Obstacle Avoidance in Dynamic Environments with Different D-S

Mohammad Javad Mollakazemi, Farhad Asadi

Abstract— In this paper a real-time obstacle avoidance approach for both autonomous and non-autonomous dynamical systems (DS) is presented. In this approach the original dynamics of the controller which allow us to determine safety margin can be modulated. Different common types of DS increase the robot's reactivity in the face of uncertainty in the localization of the obstacle especially when robot moves very fast in changeable complex environments. The method is validated by simulation and influence of different autonomous and non-autonomous DS such as important characteristics of limit cycles and unstable DS. Furthermore, the position of different obstacles in complex environment is explained. Finally, the verification of avoidance trajectories is described through different parameters such as safety factor.

Keywords—Limit cycles, Nonlinear dynamical system, Real time obstacle avoidance.

I. INTRODUCTION

WHEN a robot wants react to arbitrary forms of disturbance fast, we seek methods by which it will be effortless and will gain control input from the control laws. These encompass many of disturbances dealt with by robotics such as, when an obstacle suddenly appears in the robots path or when the robot pushed away from its trajectory in duration of its motion. In these conditions, there is no enough time to re-plan and it is not depend to fast reaction of re-planning technique and hence alternative techniques must be sought.

Obstacle avoidance is a classical problem in robotics field and many different methods have been proposed to solve it and the portion of each conventional approach is shown in Fig. 1. These methods usually classify between local and global approach, depending on whether the obstacle affects the behavior only locally or everywhere. For example Vector Field Histogram [1] and the Curvature-Velocity method [2] offer fast response in the face of perturbations. Although these algorithms have locally optimal solution but are not ensured to always find a feasible path, but a global method, such as those dealt with by path planning algorithm [3] ensures to find a valid solution, if it exists.

Despite recent effort at reducing the computational time of such global searches for finding a feasible path [4], [5], they cannot offer the fast reaction for swiftly avoiding obstacles that appear suddenly.

Mohammad Javad Mollakazemi is master student of mechanical engineering in dynamic and control, K. N. Toosi, Tehran (phone: +989192762663, e-mail: molakazemi.javad@yahoo.com).

Farhad Asadi is master student of mechanical engineering in dynamic and control, K. N. Toosi, Tehran (phone: +989357430949, e-mail: f.asadi@mail.kntu.ac.ir).

The reshaping method like Elastic Band method [6] helps at real time trajectory adaption in complex and dynamic environments. In this method first, initial shape of the elastic band is a free path and in the presence of obstacles this shape is deformed by applying repulsive forces. In [7] the original path is deformed locally to show changes in the environment topology. Generally, in these approaches if the path is not feasible due to the obstacles coming into its way, the reshaping algorithm cannot be used any more. In [8] hybrid systems that switch between local and global approach offer a compromise. In [9] a task is decomposed into several sections locally. If the local approach is not usable, the global method is used. For example we can change algorithm between re-planning and deformation of the path during execution of a task. In this method, the planner first try to locally modify the trajectory in the presence of an obstacle and in this duration where deformation is no longer possible a new trajectory is re-planned.

In [10] an adaptive motion planner that considers the Simultaneous path and trajectory planning of high -DOF robots is proposed. This method provides diverse trajectories at all time to allow fast adaption of robot motion due to the change of environment. Or in another method of elastic roadmap [11] allow the modification of the vertices and edges during the execution of the task, so roadmap represent task suitable motions.

In [12] a dynamical based method for obstacle avoidance is proposed. This method is very similar to the Attractor Dynamics approach because it changes the original dynamics of motion by introducing a factor in the motion equation that makes the motion away from the obstacle. In [13] Harmonic Potential functions were first developed to overcome the limitation of potential fields. In contrast to potential field-based, harmonic potential- based approach are powerful as they do not have local minima. Harmonic potentials have been used for control in numerous ways in the past few years. Dynamical systems can generate motion trajectories that are robust against perturbations. In this method movement trajectory is not pre-defined or pre-planned, but generated during the motion from a differential equation. When controlled through a Dynamical System (DS), a robot motion generate in time with no need to re-plan, because the dynamical system represent a whole flow field instead of a single-trajectory for robot. Thus, suitable chosen dynamical systems can automatically correct for perturbations of the state and guarantee convergence to a purpose. In this paper, we discuss the advantage and limitation an obstacle avoidance method that can be integrated into existing DS-based motion

control approaches. Although, a large set of DS is selected including locally and globally asymptotically stable or unstable DS, but there is difference between them which should be noticed for choosing them in different dynamic environments. The other advantage of this algorithm is that it can instantly modify the robot's trajectory to avoid collisions with obstacles, because the attractors of the main DS are also the attractors of the modulated DS.

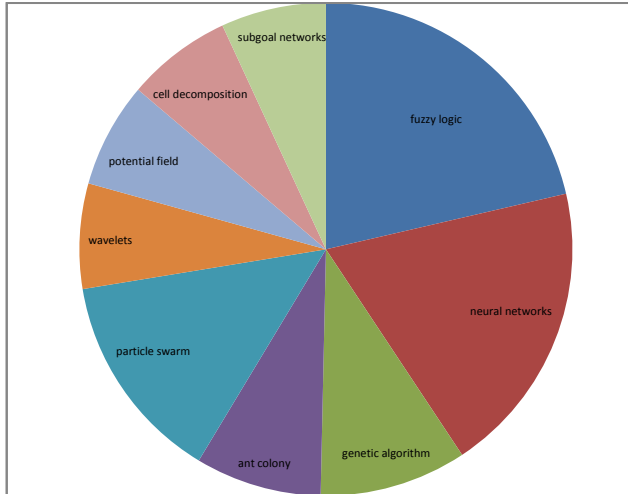


Fig. 1 The portion of each conventional approach

II. METHOD AND APPROACH

In this approach when we designed global or local stable DS, trajectories that generated from planner always reach the target and in dynamic environment local DS deforms the trajectory and both the path generation and deformation are done simultaneously at each time step. Generally speaking, in this method a state variable $\xi \in R^d$ is considered which defines the state of a robotic system and its temporal evolution may be governed by either time variant or time invariant DS according to (1) and (2):

$$\xi = f(\xi), \quad f: R^d \text{ to } R^d \quad \text{autonomous DS} \quad (1)$$

$$\xi = f(t, \xi), \quad f: R^+ \times R^d \text{ to } R^d \quad \text{non-auto. DS} \quad (2)$$

where $f(\cdot)$ is a continuous function. By giving an initial point ξ , the robot motion along time can be computed by integrating $f(\cdot)$ recursively as (3):

$$\xi_t = \xi_{t-1} + f(\cdot)\delta_t \quad (3)$$

where δ_t the integration time step and t is a positive integer. Different functions can be applied in this method and also the objects or obstacles in environment of robot create a modulation throughout the robots state space that those obstacles have recognized and defined with different nonlinear functions [11]. An iteration of this algorithm is shown in Fig. 2. According to this figure we gather trajectory data from multiple runs of motion planners and then this data is used for modifying of the algorithm.

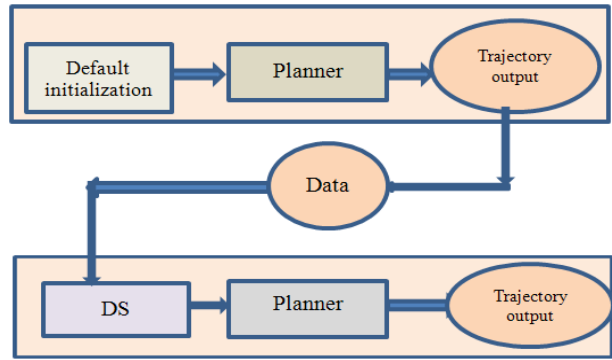


Fig. 2 A simple diagram of multiple runs of algorithm

A conceptual sketch explaining how the presented obstacle avoidance can be used in robots is shown in Fig. 3. According to this figure, initial information about object from vision or sensors section is sent to an object recognition module to identify the objects. After that, their related properties like analytical relation of the boundary of obstacle, safety factor and etc. are sent to obstacle avoidance module and according to this information the original dynamics will be changed [10], [13].

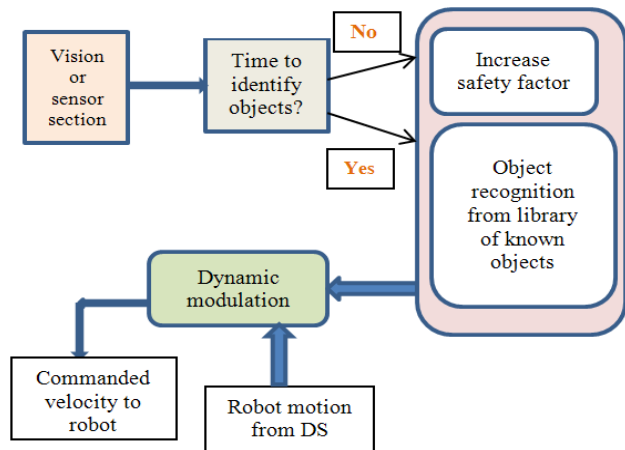


Fig. 3 A simple diagram of information processing

It should be said in this algorithm despite of continuous control input only a few control commands need to be specified, and the differential equations generate the corresponding trajectories [14], [15]. Therefore, it reduces the control load. In the above figure, when there is no time for object recognition the local planner is replaced and increased safety factor whenever the robot gets stuck in a local minimum. And in next we show real-time adaption of algorithm to different environments and limitation of this method is explained further.

III. SIMULATION RESULTS FOR DIFFERENT DS

The first dynamical system is plotted in Fig. 4. In this DS, the differential equation converges to the goal point and adapts their trajectory in presence of perturbation.

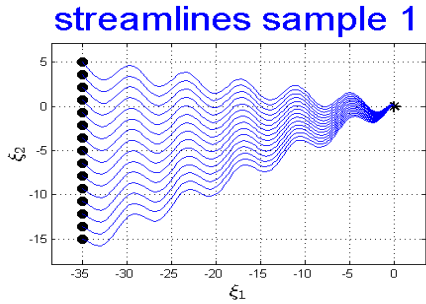


Fig. 4 Streamlines of sample 1

Now, this dynamical system is investigated in presence of different obstacles. However in many robot experiments, not only the robot should avoid the obstacle, but also it should reach a target, which we further denote ξ^* . In other words, we would like the modified motion to preserve the convergence property of the original dynamics while still ensuring that the motion does not penetrate the object. In this section, we discuss the stability of DS when they are modulated with the proposed obstacle avoidance method. However, in the presence of an obstacle, the target may not remain the unique equilibrium point of the system. Other possible equilibrium points may be created due to the modulation term $M(\xi^{\sim})$. In Fig. 5 the streamlines in face of obstacle in near of robot are plotted and ability of method for adaption and passing the obstacle is shown.

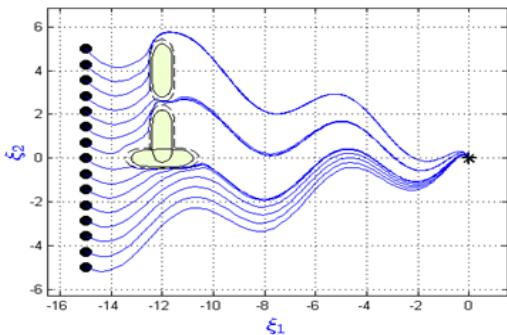


Fig. 5 Streamline in face of multiple obstacle

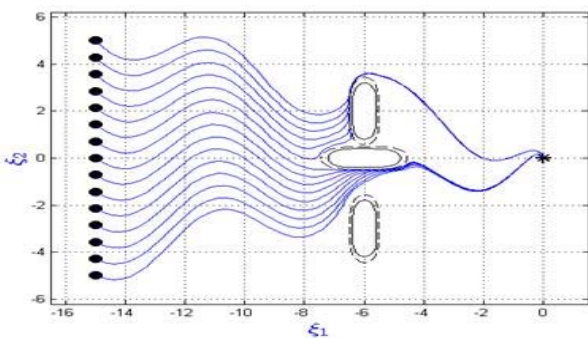


Fig. 6 Streamline in face of multiple obstacle

In Fig. 6 another situation and also variation for adaption of dynamical system is shown. It is obvious related to obstacle, the generated movement is reached to goal and the time for reaching is different by changing situation.

In next section when the obstacle is distributed in different positions when environment of robot is plotted in Figs. 5 and 6. In Fig. 7 the complex environment is evident which makes the bending of the trajectory line and effect to path planning and for showing this subject in a better way, the first obstacle is located near the initial condition. The robustness of this method to this perturbation is shown in Fig. 8.

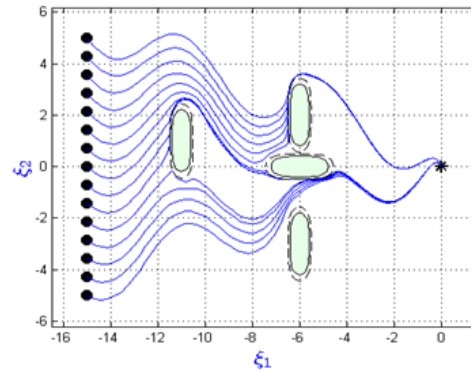


Fig. 7 Streamline in face of multiple obstacle

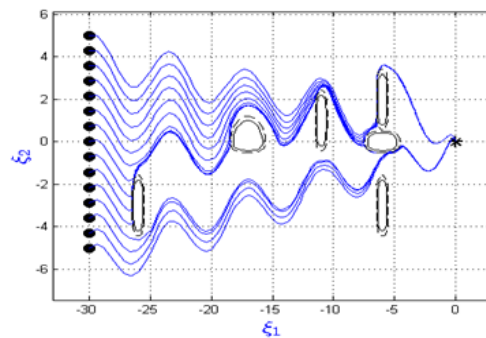


Fig. 8 Streamline in face of multiple obstacle

IV. EXPERIMENTAL RESULT WITH SECOND DS

The proposed method can handle multiple obstacles without modifying the equilibrium points of the original dynamic. The presented approach requires a global model of the environment and an analytical modeling of the obstacles boundary. When the analytical description of the obstacle is available, our method guarantees that all obstacles will be avoided safely. However, the analytical equation of the obstacle or the output of the vision system or its accurate status (i.e. position and orientation) may not be available all the time. In this condition we need more robustness and adaptable DS for designing our method and limit cycle is a good one because their structure is flexible for changing. The stability and function of this limit cycle is expressed below. In Fig. 9 a simple form of this DS is showed. It is important to say that the DS stability depends to initial point of limit cycle

and number of points influences the ability of DS for passing the obstacle in environment of robot. The second function of limit cycle with more points is plotted in Fig. 10.

streamlines sample 2

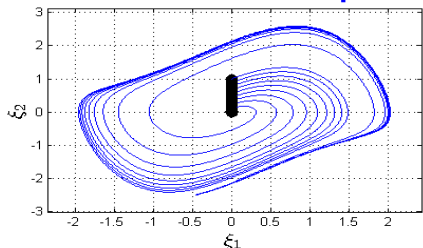


Fig. 9 Streamlines of sample 2

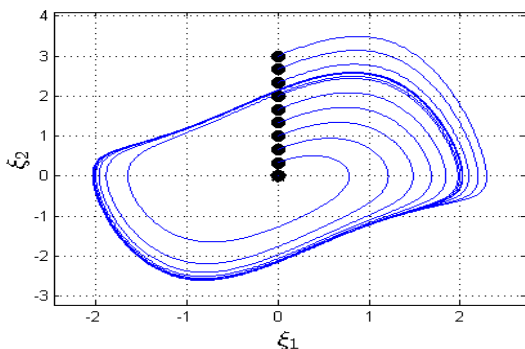


Fig. 10 Streamlines of sample 2 with different initial conditions

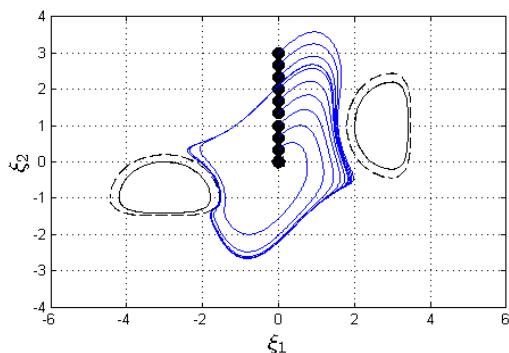


Fig. 11 Limit cycle with obstacle in out of it

Firstly, the obstacle in different locations of limit cycle and the ability of stable limit cycle for passing these obstacles are shown in Figs. 11-13. Respect to the location of these obstacles, the time for passing and the steps for solving the equation is increased. For example in Fig. 11 first we located two obstacles in out of region of limit cycle and then we see that the behavior of trajectory is smoother and the time for passing the obstacle is less than the two others. In the next figure one obstacle is located inside the region of limit cycle. In final figure two obstacles are located inside the limit cycle. The adaption of trajectory for passing the obstacle is not so smooth (Fig. 13) but it is possible by increasing the safety

factor by which we can smooth this trajectory behavior well and better that is explained further in article.

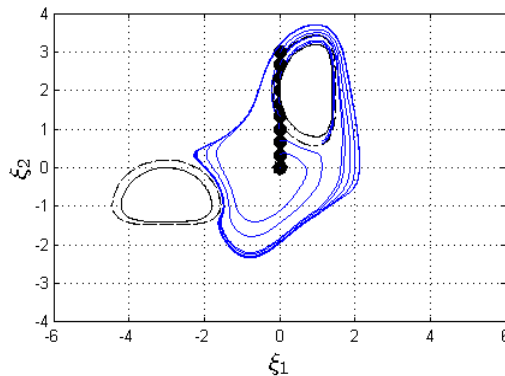


Fig. 12 Limit cycle with one obstacle in it

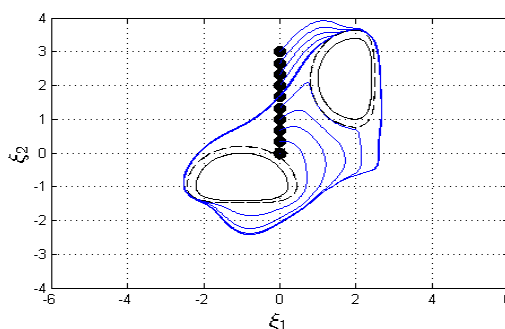


Fig. 13 Limit cycle with two obstacles inside it

It should be mentioned that in some situations we seek minimizing energy and time and maximizing manipulability and depended to this target we can adjust our parameters of DS and design or change the DS.

V. EXPERIMENTAL RESULT WITH THIRD DS

The third DS is plotted in Fig 14. We can design arbitrary equilibrium point and can develop and increase this point in our region which is valuable advantage of this method.

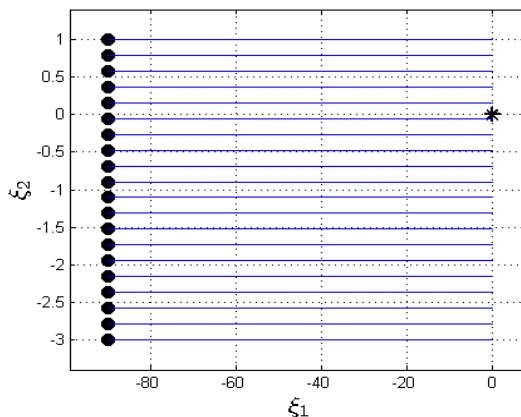


Fig. 14 DS sample 3

Next, another DS in face of one obstacle is shown in Fig 15. The effect of safety parameter that influence on the trajectory shape of path is explained further.

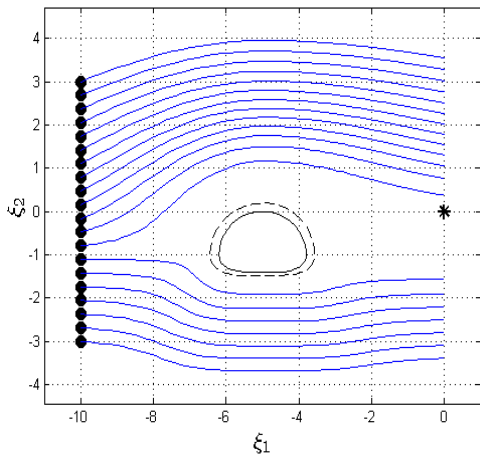


Fig. 15 Streamlines in face of one obstacle

When doing obstacle avoidance, sometimes it is more practical to customize the path to avoid an obstacle based on the object's property. For example, fragile or sharp objects may require a large safety margin while soft and round objects may not. Furthermore, it is essential to react and deflect the robot trajectory earlier when it goes toward an obstacle. In this section, we extend the proposed obstacle avoidance approach to incorporate user's preference during obstacle avoidance. In the proposed obstacle avoidance formulation, the modulation due to the obstacle continues affecting the motion even when the robot is moving away from the obstacle. This effect of the obstacle on trajectories is called tail-effect [11]. By changing the safety number in around the obstacle we delete this phenomenon that is evident and is shown in Fig. 16. In case of uncertainty in sensing, such a behavior may be beneficial as it would mitigate imprecise detection of the real volume of the obstacle.

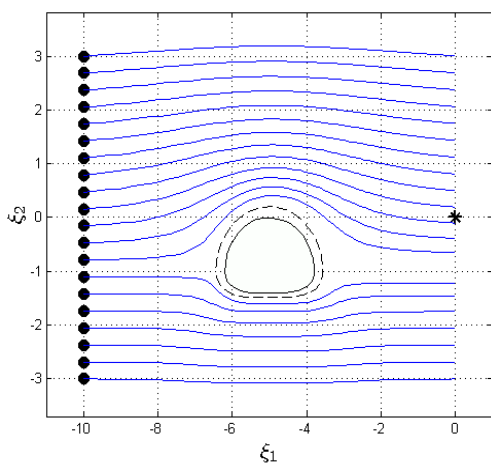


Fig. 16 Streamlines in face of one obstacle

The safety parameter is shown for two sample obstacles in Fig. 17 and this parameter is important parameter of our modeling.

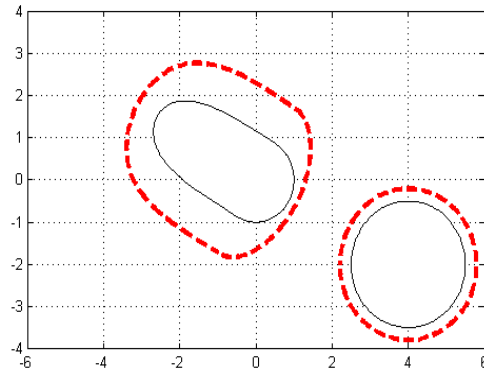


Fig. 17 Demonstration of the safety parameter

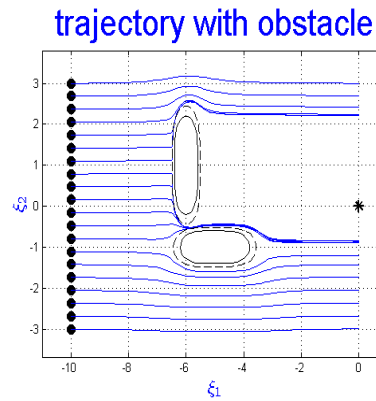


Fig. 18 Streamline in face of multiple obstacle

The ability of this approach in face of more obstacles in environment of robot is plotted in Figs. 18 and 19. It should be said that by defining this sample data in comparison of two other methods is better for reconstruction of trajectory after passing the obstacle and also the required time for sampling is shorter in comparison to other methods. Also, in this DS it is not important to locate the obstacle near or far from the robot as is shown in these figures.

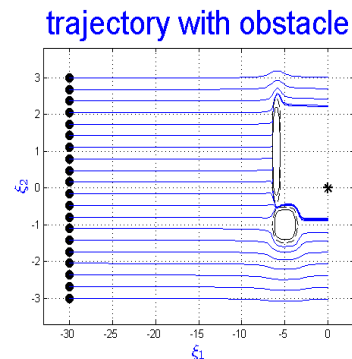


Fig. 19 Streamline in face of multiple obstacle

trajectory with obstacle

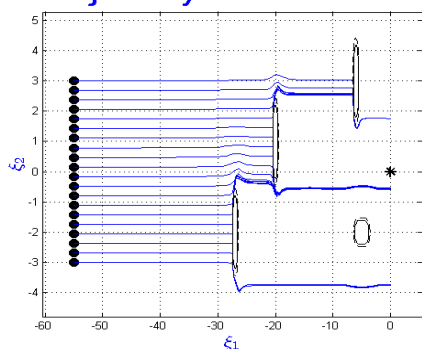


Fig. 20 Streamline in face of multiple obstacle

trajectory with obstacle

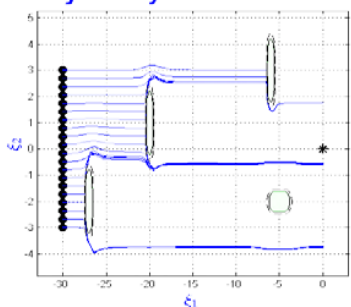


Fig. 21 Streamline in face of multiple obstacle

While doing obstacle avoidance in a dynamic environment, it is hardly possible to generate the BVs from the output of the vision system in real time. Thus, it is necessary to generate a library that stores the analytical formulations of different objects. In our implementation, we rely on a library of objects with known analytical convex envelopes. We use this analytical descriptor of the envelop both to detect the object and for our obstacle avoidance module. If the object moves very rapidly in environment, it is recommended to set a large value for the safety margin and adding the value of reactivity parameter increased the robustness to uncertainties. Furthermore, more complex environments in presence with more obstacles are plotted in Figs. 20 and 21 with modification of this number.

VI. CONCLUSION

The presented algorithm is able to cope with uncertainty in the obstacle's position by allowing certain safety margins around the obstacle. The larger the safety margin, the more robust the system is to uncertainty in the obstacle position. Note that in the presence of an unforeseen object or uncertainty in the obstacle's position, our algorithm no longer guarantees the safe avoidance of the obstacle. However, it is also possible to integrate other algorithms to perform collision avoidance for unforeseen environments.

REFERENCES

- [1] V. Lumelsky and T. Skewis. *Incorporating range sensing in the robot navigation function*. IEEE Trans. on Systems, Man, and Cybernetics, 20:1058 – 1069, 1990.
- [2] R. Simmons. *The curvature-velocity method for local obstacle avoidance*. In Proc. of the IEEE Int. Conf. on Robotics and Automation, volume 4, pages 3375 – 3382, 1996.
- [3] J.J. Kuffner and S.M. LaValle. *RRT-connect: An efficient approach to single-query path planning*. In Proc. of IEEE Int. Conf. on Robotics and Automation, volume 2, pages 995 – 1001, 2000.
- [4] R. Diankov and J. Kuffner. *Randomized statistical path planning*. In Proc. of IEEE/RSJ Int. Conf. on Robots and Systems (IROS), pages 1 – 6, 2007.
- [5] M. Toussaint. *Robot trajectory optimization using approximate inference*. In 25th Int. Conf. on Machine Learning (ICML), pages 1049–1056, 2009.
- [6] O. Brock and O. Khatib. *Elastic strips: A framework for motion generation in human environments*. Int. J. of Robotics Research, 21(12):1031 – 1052, 2002 S.
- [7] T. Fraichard, M. Hassoun, and C. Laugier. *Reactive motion planning in a dynamic world*. In Proc. of the IEEE Int. Conf. on Advanced Robotics, pages 1028 – 1032, 1991..
- [8] E. Yoshida and F. Kanehiro. *Reactive robot motion using path replanning and deformation*. In Proc. IEEE Int. Conf. on Robotics and Automation, pages 5457 – 5462, 2011.
- [9] M. Barbehenn, P.C. Chen, and S. Hutchinson. *An efficient hybrid planner in changing environments*. In IEEE Int. Conf. on Robotics and Automation, volume 4, pages 2755 – 2760, 1994.
- [10] J. Vannoy and J. Xiao. *Real-time adaptive motion planning (ramp) of mobile manipulators in dynamic environments with unforeseen changes*. IEEE Trans. on Robotics, 24:1199 – 1212, 2008
- [11] Y. Yang and O. Brock. *Elastic roadmaps: Globally task-consistent motion for autonomous mobile manipulation in dynamic environments*. In Proc. Robotics: Science and Systems, 2007
- [12] H. Hoffmann, P. Pastor, D.-H. Park, and S. Schaal. *Biologically inspired dynamical systems for movement generation: automatic real-time goal adaptation and obstacle avoidance*. In Proc. of Int.Conf. on Robotics and Automation, pages 2587 – 2592, 2009.
- [13] H.J.S. Feder and J.-J.E. Slotine. *Real-time path planning using harmonic potentials in dynamic environments*. In Proc. of IEEE Int.Conf. on Robotics and Automation (ICRA), volume 1, pages 874 – 881, 1997.
- [14] Farhad Asadi, Mohammadjavad Mollakazemi, "Investigation of fluctuation locations and effect of data distribution in time series dynamical regimes".. Accepted and oral presentation in ICBCBBE 2014: XII International Conference on Bioinformatics, Computational Biology and Biomedical Engineering, October, 27-28, 2014, Istanbul, Turkey.
- [15] Farhad Asadi, Mohammadjavad Mollakazemi, "The influence of parameters of modeling and data distribution for optimal condition on locally weighted projection regression method".. Accepted and oral presentation in ICMSE 2014: XII International Conference on Mathematics and Statistical Engineering, October, 27-28, 2014, Istanbul, Turkey.