

On-line Obstacle Detection using Data Range for Reactive Obstacle Avoidance

José Miguel Vilca Ventura, Lounis Adouane, Y. Mezouar

▶ **To cite this version:**

José Miguel Vilca Ventura, Lounis Adouane, Y. Mezouar. On-line Obstacle Detection using Data Range for Reactive Obstacle Avoidance. Advances in Intelligent Systems and Computing, 2012. hal-01714855

HAL Id: hal-01714855
https://hal.science/hal-01714855

Submitted on 23 Feb 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

On-line Obstacle Detection using Data Range for Reactive Obstacle Avoidance

José Miguel Vilca , Lounis Adouane and Youcef Mezouar
Institut Pascal
UBP – UMR CNRS 6602
Clermont-Ferrand, France
Jose_Miguel.VILCA_VENTURA@univ-bpclermont.fr

Abstract This paper deals with the reactive navigation in clustered environment. It proposes an online and adaptive elliptic trajectory to perform smooth and safe mobile robot navigation. These trajectories use limit-cycle principle already applied in the literature [3]. The main contribution proposed here is to perform this navigation in a completely reactive way while using only range sensor data. At this aim, each obstacle to avoid is surrounded by an ellipse and its parameters are obtained online while using the sequential range data and appropriate method to identify the enclosed ellipse. Different methods to obtain these ellipse parameters are presented and implemented. A specific criterion is taken into account to obtain always smooth change in these parameters. A large number of simulations permit to show the efficiency of our proposal for the navigation in cluttered environment.

Key words: Mobile robot navigation; Obstacle detection and avoidance; Telemetry; Parameter identification; Least square.

1 Introduction

An important issue for successful mobile robot navigation is obstacle avoidance. In fact, this function permits to prevent robot collision and insure thus robot safety. One area of research in obstacle avoidance is focused on reactive methods, where only local sensors information is used rather than a prior knowledge of the environment [9], [13], [1]. In [11], the author proposes a real-time obstacle avoidance approach based on the principle of artificial potential fields. In this work, it is assumed that the robot actions are guided by the sum of attractive and repulsive fields.

Several other approaches can be found in the literature, such as the obstacle avoidance based on orbital trajectories. This approach is described by circular limit-cycle differential equations [12], [10] and [2]. Our work uses elliptical trajectories that was presented in [3]. Therefore, more generic and efficient obstacle avoidance is performed and this even with different obstacle shapes, for instance long walls. In fact, an ellipse fit better this kind of obstacles than a circle.

For this purpose, different techniques have been proposed in the literature to enclose the data with an ellipse [15], [14] and [7]. In [15], the author presents a summary of the methods to fit a set of data with an ellipse. The presented methods are the least square fitting based on algebraic and Euclidean distance, Kalman filtering method and robust

estimation. In [14], the author proposed a technique to obtain the smallest enclosing ellipse by a set of data improving thus the Welzl's algorithm [14] with linear increasing time with regards to data dimension. In [7] the authors constructed an ellipse using the mean and covariance of the data and the Mahalanobis distance to analyze the relationship among them. The maximum Mahalanobis distance is used for different purposes, namely for detection of outliers and for investigating the representativity between two data sets. This work proposes an on-line heuristic method based on the distance between the data to compute the parameters of ellipse.

The rest of the paper is organized as follows: in the next section, the task of navigation using elliptic trajectories is presented. In section 3, the details of the control architecture are introduced and obstacle avoidance algorithm is given. Section 4 presents the methods for enclosing the range data with an ellipse. Simulation results are given in section 5. Finally, conclusion and future works are given in section 6.

2 NAVIGATION IN CLUTTERED ENVIRONMENT

First, let us assume that the obstacle O can be surrounded by elliptical box (cf. Fig. 1). The elliptical shape is represented by its Cartesian form:

$$(x-h)^2 / a^2 + (y-k)^2 / b^2 = 1 \tag{1}$$

where $(h,k) \in \Re^2$ are the ellipse center coordinates and $a,b \in \Re^+$ are the semi-axes with $a \geq b$.

In section 4, it will be shown how the ellipse parameters can be efficiently computed from range data. Let us also represents the robot and the target by circles C_R and C_T of radius R_R and R_T respectively (cf. Fig. 1). One can define:

1. D_{RO} as the minimal distance between the robot and the obstacle "O" [8].
2. *Ellipse of influence* (E_f) as an ellipse that has the same center (h,k) and tilt angle Ω as the ellipse which surround the obstacle (1) while its major (minor) semi-axis is $a_{lc}(b_{lc})$ are defined as follows
$$\begin{cases} a_{lc} = a + R_R + Margin \\ b_{lc} = b + R_R + Margin \end{cases}$$
where *Margin* represents safety tolerances encapsuling: perception uncertainties, control reliability and accuracy.
3. "l" as the line passing through the center of C_R and C_T. As we will see in the sequel, our method only needs to know if it exists intersection points between l and E_f (cf. Fig. 1).

The objective of this navigation task in cluttered environment is to lead a mobile robot towards a specific target in an unstructured environment. This task must be achieved while avoiding static and dynamic obstacles O which can have different shapes. In this work, we consider that the cluttered environment has only static obstacles. Furthermore, the presence of dynamic obstacles will be developed in future work.

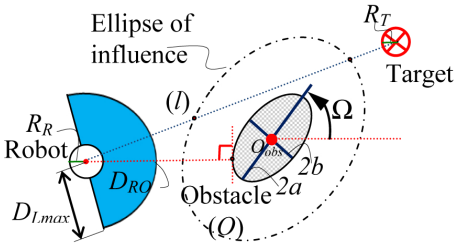


Fig. 1 The used perceptions for mobile robot navigation.

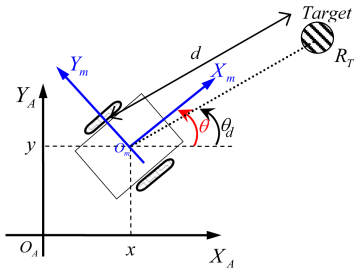


Fig. 2 Robot configuration in a Cartesian reference frame.

3 CONTROL ARCHITECTURE

The used control structure is based on [3] (cf. Fig. 3). It aims to manage the interactions between elementary controllers while guaranteeing the stability of the overall control as proposed in [1]. Its objective is also to insure safe, smooth and fast robot navigation. The specific blocks composing the global controller are detailed below.

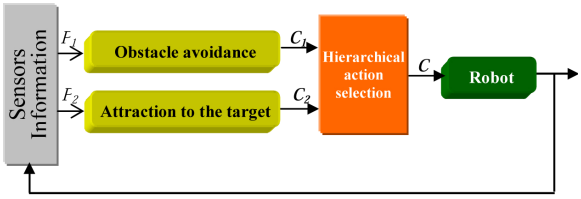


Fig. 3 Control architecture for mobile robot navigation.

This control architecture uses a hierarchical action selection mechanism to manage the switches between the controllers, according to environment perception. It has a mechanism that activates the obstacle avoidance controller as soon as it exists at least one obstacle which can obstruct the future robot movement toward its target [2]. This permits to anticipate the activation of obstacle avoidance controller unlike what is proposed in [6] or [4], which wait until the robot is in the immediate vicinity of the obstacle (i.e. $D_{RO} \leq R$ "a certain radius value").

Before describing each elementary controller, let us briefly recall the kinematic model of an unicycle robot (cf. Fig. 2)

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{2}$$

where x, y, θ are configuration state of the unicycle at the point O_m, v and ω are respectively, the linear and angular velocity of the robot at the point O_m.

The attraction to target controller guides the robot toward the target. This controller is based on the control position of the robot to the target. In this paper, we focus on the obstacle avoidance controller, more details about attraction to target controller are given in [5].

3.1 Obstacle avoidance controller

To perform the obstacle avoidance behavior, the robot needs to follow accurately limit-cycle trajectories as what is given in [12] and [2]. In these works authors use a circular limit-cycle characterized by a circle of influence of R_I radius. In [3], it is proposed to extend this methodology for more flexible limit-cycle shape (an ellipse). The main ideas of this controller are detailed below.

The differential equations giving elliptic limit-cycles are:

$$\dot{x}_s = \ my_s + x_s(1 - x_s^2/{a_{lc}}^2 - y_s^2/{b_{lc}}^2 - cx_sy_s) \tag{3}$$

$$\dot{y}_s = -mx_s + y_s(1 - x_s^2/{a_{lc}}^2 - y_s^2/{b_{lc}}^2 - cx_sy_s) \tag{4}$$

with $m = \pm 1$ according to the direction of avoidance (clockwise or counter-clockwise, cf. Fig. 4). (x_s, y_s) corresponds to the position of the robot according to the center of the ellipse; a_{lc} and b_{lc} characterize respectively major and minor elliptic semi-axis (cf. Fig. 1); c if $\neq 0$ gives the Ω ellipse angle.

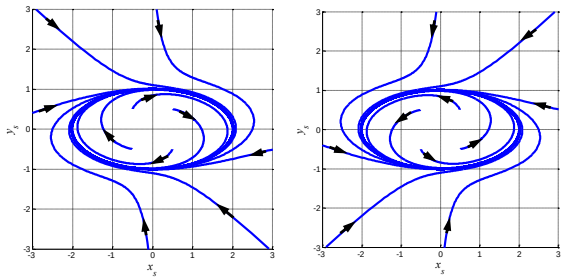


Fig. 4 Clockwise and counter-clockwise shape for the elliptic limit-cycles.

The algorithm for obstacle avoidance is summarized in the following main steps:

- The nearest hindering obstacle is detected,
- The direction of avoidance is chosen according to the position of the robot with regards to the position of obstacle and the target,
- The robot avoids the obstacle while following an elliptic limit cycle which has the semi-axes a_{lc} and b_{lc}, with $a_{lc} \geq b_{lc}$.

More details of the used obstacle avoidance algorithm is given in [3].

The desired robot orientation is thus given by the differential equation of the limit-cycle (3) and (4) as:

$$\theta_d = arctg\,(\dot{y}_s/\dot{x}_s) \tag{5}$$

and the error by:

$$\tilde{\theta} = \theta_d - \theta \tag{6}$$

3.2 Control law

The used control law is expressed as follows [5]:

$$v = v_{max} e^{-1/d} cos(\tilde{\theta}) \tag{7}$$

$$\omega = \quad \omega_r + K_p \tilde{\theta} \tag{8}$$

where v_{max} is the maximum linear velocity, K_p is a constant such that $K_p > 0$ and if the *obstacle avoidance* controller is activated, d is equal to D_{RO} (cf. Fig. 1) else d is the distance between the robot and the target when the *attraction to the target* controller is activated. The robot reaches the target when $0 < d \leq R_T$, where R_T is the radius of the target (cf. Fig. 1).

It is interesting to notice that only one control law is applied to the robot even if its architecture of control contains two different controllers. Only the set points change according to the applied controller [5].

4 ENCLOSING DATA RANGE WITH AN ELLIPSE

During the robot movement, it is important for the robot to detect on-line and to avoid the hinder obstacle. The robot will try to surround the observed range data with the closest ellipse to apply elliptic limit-cycle approach.

For this purpose, let us consider a set of n points in $\Re^2$ with coordinates $P_i(x_i, y_i)$ (cf. Fig. 5). These points are computed from the data range of the robot, and it is considered that it was previously filtered, i.e, without noisy and outliers. In this section, it will be shown how to compute the ellipse that encloses all points. An important condition in this work is that the methods need to start at least with three different points.

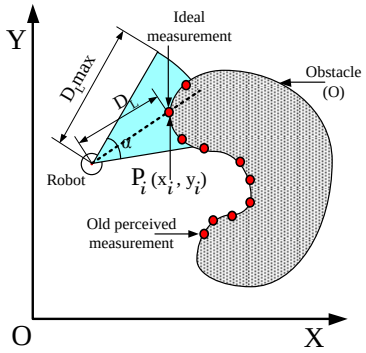


Fig. 5 Data set of n points.

This paper considers that the robot detects one obstacle at a time. The segmentation method of the set of points will be used in future works to the detection of more than one obstacle at a time and for big obstacles.

The following sub-sections will present different methods to enclose the data points with an ellipse. Sub-sections 4.1 and 4.2 are applied and (more or less) modified approach with regards to what is proposed in the literature and the method given in subsection 4.3 is a completely new approach to deal with this problem.

4.1 Fitting using least square approach

This approach uses the general conic equation which is given by:

$$f(x,y) = Ax^2 + 2Bxy + Cy^2 + 2Dx + 2Ey + F = 0 \tag{9}$$

According to the real constants A, B, C, D, E anf F, we obtain the analytic equation of the different kind of conics (parabola, ellipse and hyperbole). An ellipse is defined if the conic parameters (9) satisfy the following condition $B^2 - AC < 0$.

The problem is to fit a conic section (9) with a set of n points $\{p_i\} = \{(x_i, y_i)\} \mid i = 1, \ldots, n$. As the data are noisy, it is unlikely to find a set of parameters (A, B, C, D, E, F) (except for the trivial solution $A = B = C = D = E = F = 0$) such that $f(x_i, y_i) = 0$. Instead, we will try to estimate them by minimizing some objective function J.

A common practice is to minimize the algebraic distance $f(x_i, y_i)$, we use the least-square fitting based on algebraic distance [15] to minimize the following function:

$$J = \sum_{i=1}^{n} f^2(x_i, y_i) \tag{10}$$

Note that there is no justification for using algebraic distance apart from easy implementation. To avoid the trivial solution, $f(x_i, y_i)$ should be normalized. We use the normalization $A + C = 1$ because it is one of the most popular methods proposed in the literature [15], and its implementation is easy.

Using the normalization in (9), the set of equation $f(x_i, y_i)$ can be written as:

$$f_i \equiv \mathbf{a}_i \mathbf{s} - \mathbf{b}_i = 0 \tag{11}$$

where $\mathbf{s} = [A, B, D, E, F]^T$, $\mathbf{a}_i = [x_i^2 - y_i^2, 2x_i y_i, 2x_i, 2y_i, 1]$ and $\mathbf{b}_i = -y_i^2$. The solution that minimize the functional J is given by

$$\mathbf{s} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \tag{12}$$

where $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]^T$ and $\mathbf{b} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n]^T$. This method is known as the pseudo inverse technique.

4.2 Covariance approach

This method is based on the analysis of the relationship between the n points. We use the covariance of the data and the Mahalanobis Distance (*MD*) [7]. In the field of multivariate calibration, the *MD* is used for different purposes, namely: for the detection of outliers, the selection of calibration samples from a large set of measurements and for investigating the representativity between two data sets. In the original variable space, the *MD* takes into account the correlation in the data, since it is calculated using the inverse of the covariance matrix of the data set of interest.

The principle of covariance approach is given below: First the covariance matrix $\mathbf{C}_x$ is constructed:

$$\mathbf{C}_x = \frac{1}{(n-1)} (\mathbf{X}_c)^T (\mathbf{X}_c), \tag{13}$$

where $\mathbf{X}_c$ is the column-centered data matrix $(\mathbf{X} - \bar{\mathbf{X}})$. $\mathbf{X}$ is the data matrix containing n objects in the rows measured for p variables and $\bar{\mathbf{X}}$ is the data mean.

Now, we analyze the covariance matrix $\mathbf{C}_x$ using the eigenvalues and eigenvectors to obtain the parameters of the ellipse $f(x', y')$ (1). Furthermore, we use the Mahalanobis distance to select the dimension of the semi-axes (percentage the data inside the ellipse) [7]. The eigenvalues λ_1, λ_2 $(\lambda_1 > \lambda_2)$ are related to the semi-axes a and b as follows:

$$\begin{aligned} a &= MD_{max}\sqrt{\lambda_1} \\ b &= MD_{max}\sqrt{\lambda_2} \end{aligned} \tag{14}$$

where MD_{max} is the maximum Mahalanobis distance to ensure that all the set of data is enclosed by the ellipse. The eigenvectors $\mathbf{v}_1$, $\mathbf{v}_2$ are related with the orientation Ω.

$$\Omega = \arctan(\mathbf{v}_2/\mathbf{v}_1) \tag{15}$$

The center of the ellipse is the mean of the data $\bar{\mathbf{X}}$ and the maximum semi-axes a is the direction where there are the most important amounts of data.

4.3 Heuristic approach

The proposed heuristic approach uses the distance between the points to obtain one of the axes.

This method computes the distance between all the points $d_{ij} = \|p_i - p_j\|$ with $i, j = 1, \ldots, n$; and select the maximum distance d_{max}. We have thus, $d_{ij} \leq d_{max}$. With this manner, this maximum distance is not decreasing. The line connecting the corresponding two points (the points with the maximum distance) is one of the ellipses axes, where the ellipse center C_O is the middle point between the points with maximum distances and the first semi-axes is $a_1 = d_{max}/2$.

Now, we work in the new coordinates system $X'Y'$ to obtain the second ellipse semi-axes a_2. We transform the n points to new coordinates system using (16).

$$\mathbf{P}'_i = \begin{bmatrix} cos(\Omega) & sin(\Omega) \\ -sin(\Omega) & cos(\Omega) \end{bmatrix} (\mathbf{P}_i - \mathbf{C}_O) \tag{16}$$

Where Ω is the orientation of the line between the two points that have the maximum distance. $\mathbf{P}'_i(x'_i, y'_i)$ is the coordinate in the new system, $\mathbf{P}_i(x_i, y_i)$ is the coordinate in the initial system and $\mathbf{C}_O$ is the coordinate of the ellipse center in the initial system.

If the coordinate y'_i of the points is greater than the threshold $\varepsilon > 0$, we compute the distance of $\mathbf{P}'_i$ to the origin O', i.e., $|y'_i| > \varepsilon \Rightarrow d'_i = \|\mathbf{P}'_i\|$. The threshold is used to eliminate the points that are collinear with the two points that have the maximum distance (axes).

We know that all points inside the ellipse have the distance $d'_i < max\{a_1, a_2\}$. As we do not know the spatial distribution of the set of points and to ensure that all the points are inside the ellipse, we choose that $a_2 = max\{b_i\}$, where b_i is the computed semi-axes using $\mathbf{P}'_i$ in (1). In other words, $b_i \leq a_2 \leq max\{a_1, a_2\} \Rightarrow \mathbf{P}_i \in Ellipse$. Therefore, the ellipse will enclose all points without regard of the obstacle shape.

Finally, we obtain the semi-axes of the ellipse (1) such as:

$$\begin{aligned} a &= max\{a_1, a_2\} \\ b &= min\{a_1, a_2\} \end{aligned} \tag{17}$$

5 SIMULATIONS RESULTS

To demonstrate the efficiency of the proposed approach to enclosing the obstacle with the ellipse, a two statistical survey was made. We consider a mobile robot with a radius

of $R_R = 0.5\ m$ and a $\alpha = 180°$ laser range sensor with the maximum detected range is equal to $D_{Lmax} = 3.0\ m$ (cf. Fig. 5). The maximum linear velocity of the robot is $1.5\ m/s$ and the sample time is $0.01\ s$. For each simulation the robot starts at the same configuration and reaches the same final configuration. We do not start to use any method until we have the enough data range $n_{data} \geq 3$. The first survey is used to compare the methods presented above to identify the closer ellipse shape which surround the detected obstacle (cf. section 4).

Fig. 6 shows the identified ellipse parameters using the different methods. We observe that the heuristic parameters change more smoothly if they are compared with the others.

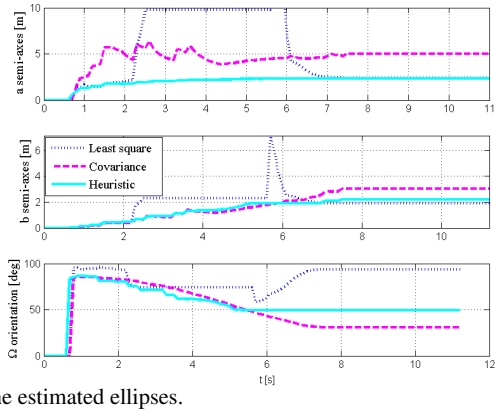


Fig. 6 Parameter of the estimated ellipses.

Fig. 7, 8, 9 shows the evolution of the identified ellipse for the least square, covariance and heuristic methods, respectively. We observe that the least square method and covariance change abruptly.

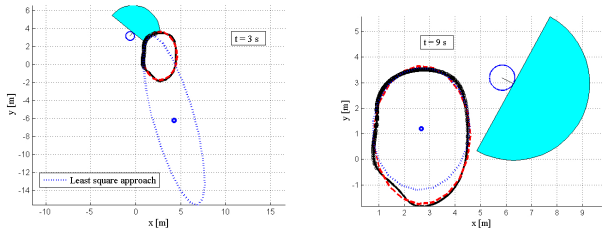


Fig. 7 Evolution of the obtained ellipse using least square approach.

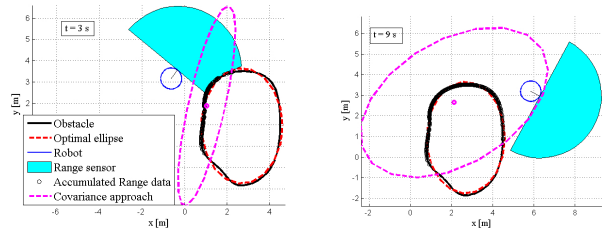


Fig. 8 Evolution of the ellipse using covariance approach.

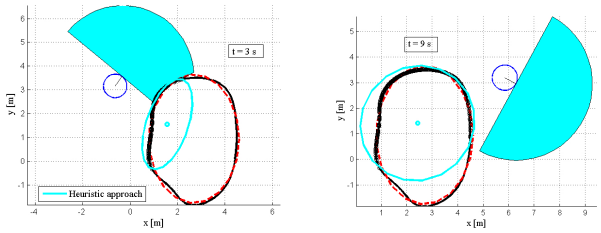


Fig. 9 Evolution of the ellipse using heuristic approach.

The second survey is used to make a focus only around the proposed heuristic method which gives satisfactory results and this while making an on-line navigation in cluttered environment (cf. Fig 10).

Fig. 10 shows the trajectory of the robot in the environment with three obstacles. We observe that the robot avoid the obstacles with a smooth trajectory. This trajectory was obtained while using the on-line obstacle avoidance algorithm [3] which takes its parameters (elliptical limit-cycle to follows) from the proposed heuristic approach.

Fig. 11 shows the distance between the optimal ellipse (obtained while knowing all the range data which surround the obstacle) and the position of the robot (red dotted line), and the distance between the ellipse using the heuristic method and the robot. This figure shows that the robot does never collide with any obstacles, thus the proposed on-line approaches is efficient to deal with cluttered unstructured environment. Future work will consist to implement this approach in real robots.

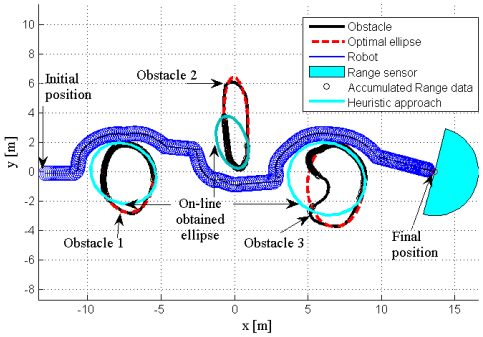


Fig. 10 Robot trajectory using the heuristic approach to enclosing the obstacle.

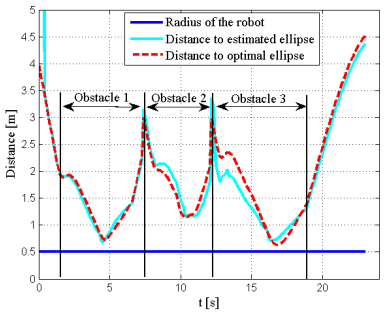


Fig. 11 Distance from the robot to the optimal ellipse and to on-line estimated ellipse using the heuristic approach.

6 CONCLUSION

This paper proposes an on-line elliptic limit-cycle trajectory to perform smooth and safe mobile robot navigation. This elliptic trajectory is obtained while using only local range data. The later, permits to obtain the parameters of the ellipse which surround the obstacle to avoid. After giving a review of different methods to enclosing an ellipse, an appropriate heuristic method was proposed and compared to the existing approaches

(based on least square or covariance). The proposed heuristic permits to obtain a smooth changes of the ellipse parameters. The overall obstacle avoidance strategy permits us to obtain generic and flexible navigation in very cluttered environments. The proposed reactive navigation was embedded in multi-controller architecture and permits for a mobile robot to efficiently navigate in environments with different obstacles shapes. In future works, the problem of dynamic obstacles, outliers detection and uncertainty of the range data will be considered. Furthermore, we will use the Kalman filter to deal with the uncertainty of the range data.

References

1. Adouane, L.: Hybrid and safe control architecture for mobile robot navigation. In: 9th Conference on Autonomous Robot Systems and Competitions. Portugal (2009)
2. Adouane, L.: Orbital obstacle avoidance algorithm for reliable and on-line mobile robot navigation. In: 9th Conference on Autonomous Robot Systems and Competitions. Portugal (2009)
3. Adouane, L., Benzerrouk, A., Martinet, P.: Mobile robot navigation in cluttered environment using reactive elliptic trajectories. In: 18th IFAC World Congress (2011)
4. Adouane, L., Le Fort-Piat, N.: Behavioral and distributed control architecture of control for minimalist mobile robots. Journal Europen des Systmes Automatiss **40**(2), pp.177–196 (2006)
5. Benzerrouk, A., Adouane, L., Martinet, P.: Lyapunov global stability for a reactive mobile robot navigation in presence of obstacles. In: ICRA'10 International Workshop on Robotics and Intelligent Transportation System (2010)
6. Brooks, R.A.: A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation **RA-2**, pp.14–23 (1986)
7. De Maesschalck, R., Jouan-Rimbaud, D., Massart, D.: The mahalanobis distance. Chemometrics and Intelligent Laboratory Systems **50**(1), 1 – 18 (2000)
8. Eberly, D.: Distance from a point to an ellipse in 2d. In: Geometric Tools, LLC. http://www.geometrictools.com/ (2008)
9. Egerstedt, M., Hu, X.: A hybrid control approach to action coordination for mobile robots. Automatica **38(1)**, 125–130 (2002)
10. Jie, M.S., Baek, J.H., Hong, Y.S., Lee, K.W.: Real time obstacle avoidance for mobile robot using limit-cycle and vector field method. Knowledge-Based Intelligent Information and Engineering Systems pp. 866–873 (2006)
11. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. The International Journal of Robotics Research **5**, pp.90–99 (1986)
12. Kim, D.H., Kim, J.H.: A real-time limit-cycle navigation method for fast mobile robots and its application to robot soccer. Robotics and Autonomous Systems **42(1)**, 17–30 (2003)
13. Toibero, J., Carelli, R., Kuchen, B.: Switching control of mobile robots for autonomous navigation in unknown environments. In: IEEE International Conference on Robotics and Automation, pp. 1974–1979 (2007)
14. Welzl, E.: Smallest enclosing disks (balls and ellipsoids). In: Results and New Trends in Computer Science, pp. 359–370. Springer-Verlag (1991)
15. Zhang, Z.: Parameter estimation techniques: A tutorial with application to conic fitting. Image and Vision Computing **15**, 59 – 76 (1997)