Defence Science Journal, Vol. 60, No. 1, January 2010, pp. 55-60

Path Planning in the Presence of Dynamically Moving Obstacles with Uncertainty

G.K. Singh* and Ajith Gopal**

* *National Aerospace Laboratories, Bangalore*
***BAE Systems, Johannesburg, South Africa*
E-mail: gksnal@rediffmail.com;gksingh@nal.res.in

ABSTRACT

In this paper, the problem of path-planning with dynamically moving elliptical obstacles is addressed. A new analytical result for computing the axes aligned bounding box for the ellipses with bounded uncertainty in the position of the centre and the orientation is presented. Genetic algorithm is utilised for finding the shortest path from the initial to goal position avoiding the moving obstacles.

Keywords: Path-planning, genetic algorithm, elliptical obstacles

1. INTRODUCTION

The problem of robotic navigation can be approached in two different ways: (i) motion planning–which includes dynamical modelling, and (ii) path planning–which restricts itself to spatial and geometrical modelling. In the former, a feedback control law that deals with torques to manipulator joints and drive wheels is generated, in order to track a supplied reference trajectory. In the latter, a trajectory or path is generated through the associated space of possible configurations of the robot and the known obstacles in the working area, considering their positions and the desired final position of the robot. The motion planning approach is used mainly in real-time guidance applications. On the other hand, the path planning finds applications mainly in high level off-line navigation tasks.

Path-planning for robotic systems has been extensively studied. The main idea is to find a feasible path without hitting known obstacles from a given starting point to a specified destination. The path found should also be optimal in some sense, i.e., either minimum distance or time or fuel spent. These problems are normally posed as numerical optimisation problems. Various numerical methods have been utilized for this purpose. Genetic algorithms[1] are a powerful tool based on models of natural selection and evolution and allow an exhaustive search over large spaces. Castillo and Trujillo[2] used multiple-objective genetic algorithms (MOGA) for the problem of offline point-to-point path-planning on a flat 2-D terrain represented as a 2-D grid with static obstacles and dangerous ground that the robot must evade. The objectives minimised are the length of the path and the degree of difficulty. The motor commands are used as independent variables to formulate the path-planning problem for the 'Khepera' robot and is solved using GA's as done by Thomaz[3], *et al.* A monotonic trajectory in 2-D domain is used to represent a path as done by Shahidi[4], *et al.*; however no backward movement is allowed in sub-paths. All of the above works address the case of static obstacles in the area of interest. However, in many practical problems the obstacles could be moving dynamically, e.g., in the case of multiple-robotic manipulators on an assembly line, the other robots which act as obstacles could be transient or dynamic. The presence of a moving obstacle introduces the 'time' dimension into the path-planning problem which could otherwise be solved for in the configuration space only.

Tychonievich[5], *et al.* applied the maneuver-board approach for path-planning in the presence of moving circular obstacles. Fiorini and Shiller[6] provide a brief survey of the motion-planning in dynamic environment. They have proposed the concept of velocity obstacle, based on which local avoidance manoeuvres can be computed. A sequence of such avoidance manoeuvres computed at discrete time intervals defines the 'trajectory'. Vadakkepat[7], *et al.* considered that the obstacles and the goal could be moving and not necessarily stationary. The independent variables are used to define the individual potential fields for the obstacles and the goal and the robot motion is assumed along the resultant potential field. Katz and Delrieux[8] use the Lee's routing algorithm for a real-time path-planning in the presence of moving obstacles and target. The above referenced works deal with the motion-planning. Most of the strategies above are based on the 'sense-and-avoid' principle. Hence the paths generated by these are locally optimal and need not be globally optimal. For an autonomous robot

Received 22 September 2009

to navigate in an uncertain road traffic situation, such an approach is obviously essential.

However this may not be always true in industrial robotics, where there are specific entities performing their designated tasks dynamically. In such cases the path tracked by each entity is known *a priori*, albeit with some uncertainty while they are being executed. van den Berg and Overmars[9] started with the assumption that the only information about the obstacles is the maximum velocity (no direction information), hence the obstacles are represented as disks that grow in size over time. This, however, is quite a pessimistic approach. Guibas[10], *et al.* assumed polygonal obstacles with uncertainty on the position of the vertices. It employed a probabilistic approach to path-planning to reduce the collision-risk probability while navigating to the 'target'.

This paper considers the robotic path-planning problem in the presence of moving elliptical obstacles with a bounded uncertainty in the position and orientation of the obstacles. As in most of the above references, the robot is assumed to be a 'point', since the dimensions of the robot can always be incorporated as an appropriate increase in the obstacle size. It is further assumed that the nominal movement trajectory of the obstacles in time is known with some bounded uncertainty. A new analytical result for computing the axes aligned bounding box for the ellipses with bounded uncertainty is presented. Genetic algorithms have been employed to solve the path-planning problem.

2. PATH REPRESENTATION

The first step in solving the path-planning problem is to have a parametric representation of a path from designated 'start' to 'stop' positions within a specified domain. This study restricts itself to motion in a 2-Dl plane but the methods discussed here can be easily extended to higher dimensions. The domain over which a robot can move is typically restricted, and in this work it is restricted to [0,1] × [0,1], without loss of generality. Further, the starting point and destination represented as (x_0, y_0), and (x_d, y_d) are assumed to be (0,0) and (1,1) respectively. There are several ways of representing a path in a 2-D plane:

1. A vector of $(\Delta x, \Delta y)_k$ increments can be used, Here at the k[th] instance, the trajectory moves Δx_k, Δy_k from (x_k, y_k). Thus the trajectory is defined as $x_k = x_0 + \Sigma_k (\Delta x_k)$, and similarly for the y-axis. Some works (for e.g. Castillo and Trujillo[2]) also employ a cell representation of the domain, and operate over fixed grid points.
2. Candido[11] uses a vector of $(R, \theta)_k$ values. Here at the k^{th} instance, the robot moves a distance R_k at an angle θ_k from (x_k, y_k).
3. Alternatively, the *x* and *y* trajectories could be modelled using B-splines as done by Kostaras[1], *et al.* and Walther[12], *et al.* Here the path is represented as a composition of piece-wise continuous segments defined locally.

In the first two representations, it becomes difficult to ensure that a path generated by the respective representation will stay within the domain of interest and/or reach the specified goal location. Hence, Castillo and Trujillo[2] require a 'path-repair' mechanism to ensure that infeasible path can be converted into a feasible one. The spline representation can be easily bounded to ensure that trajectories stay within the domain and also reach the target location. Hence, the B-spline representation has been used in this work. A path is represented as a linear combination of a set of say *N* local basis functions defined over some experimental time interval $[0, t_{expt}]$. Since the control parameters have a local effect on the curve, it is easy to enforce the 'goal-reaching' condition by an appropriate choice of the last (N^{th}) control parameter, based on the previous $[(N-1)^{th}]$ parameter. It should be noted that the curve will arrive at the 'goal' some time, say t_{goal} before t_{expt}, depending on this selection. The requirement that the curve remain within the domain of interest [0,1] × [0,1], can be easily met by constraining the $[(N-1)]$ free parameters to lie within [0,1). Using the same set of basis functions for the *x* and *y* trajectories over time ensures that both of them reach at the goal (1,1) at the same time. Thus, the set of possible trajectories is easily parameterised using $2^*([(N-1)]$ free parameters constrained as above.

3. OBSTACLES

Various kinds of obstacles have been considered in the literature. These include the polygonal (e.g. Guibas[10], *et al.*) and circular obstacles among others. Most of the cited works use circular obstacles so that the rotation of the obstacle becomes irrelevant. Further, a polygonal obstacle may be equivalently represented as a collection of circular obstacles[6]. In present work, the class of elliptical obstacles is considered, since their rotation assumes significance. An ellipse in a 2-D plane can be easily represented by five parameters viz,

(a) Position of the centre, (x_C, y_C),
(b) Lengths of the semi-major and minor axis, (a,b) respectively, and
(c) Rotation from a reference axis (θ).

Unlike van den Berg and Overmars[9], it is assumed that a nominal time-trajectory (both translation and rotation $[x_N(t), y_N(t), \theta_N(t)]$) of the obstacle is known. It is assumed that the physical dimension of the obstacle specified by (a,b) is known and does not vary over time. However, it is not a restrictive assumption, and can be easily included in analysis if so desired. Further, it is assumed that there is a bounded uncertainty associated with: (1) the position of the centre, and (2) the rotation. The uncertain terms are indicated by the subscript 'U'. In particular, the actual trajectory of the obstacle would be modified by uncertain terms as below:

$$x_{aC}(t) = x_N(t) \pm x_U(t)$$

$$y_{aC}(t) = y_N(t) \pm y_U(t)$$

$$\theta_{aR}(t) = \theta_N(t) \pm \theta_U(t)$$

where,

$$| x_U(t) | \leq h$$

$| y_U(t) | \leq k$

$| \theta_U(t) | \leq \Theta$.

For bounded values of $[h,k,\Theta]$. The subscript a is used to indicate the actual values including the perturbation from the nominal.

3.1 Axis-aligned Bounding-box for Obstacles

With the time trajectory of obstacles as defined above, it becomes essential to characterise a bounding-box for the set of possible ellipses, defined by the nominal elliptical obstacle and the permitted levels of uncertainty. Once that is identified, it becomes easy to check if a point in the path is likely to be in collision with an obstacle or not. The axis-aligned rectangular bounding-boxes are computed towards this end. The bounding rectangle is also parameterised by the same set of five parameters. The centre and rotation of the box is assumed to be the same as the nominal trajectory, i.e., $[x_N(t), y_N(t), q_N(t)]$. The major and minor dimensions of the bounding rectangle (a_{RB}, b_{RB}) need to be computed for the obstacle for each time instant. This can be easily computed as follows:

It should be noted that the translation of the ellipse-centre from the origin does not affect the dimension of the bounding-box; hence the following developments assume that the ellipse-centre is at the origin. At any time instant t, an elliptic obstacle centred at (0,0) can be easily represented in a parametric form as: (the time dependence has been omitted in the following for brevity).

$$x_e(p;\ [x_U, y_U, \theta_U]) = x_U + a*\cos(p)*\cos(phi) - b*\sin(p)*\sin(phi)$$

$$y_e(p;\ [x_U, y_U, \theta_U]) = y_U + b*\sin(p)*\cos(phi) + a*\cos(p)*\sin(phi)$$

where, p is a running variable from $[-\pi$ to $+\pi]$, and phi is the angle $\theta_{aR} = \theta_N + \theta_U$. The bounding box is axis aligned, i.e. its centre and rotation is specified as $[0,0,\theta_N]$. The distance from any point of the ellipse (defined above) to the rotated axes (x_{eR}, y_{eR}) can be shown to be:

$$x_{eR}(p;[x_U, y_U, \theta_U]) = x_e(p;[x_U, y_U, \theta_U]) * \cos(\theta_N) + y_e(p;[x_U, y_U, \theta_U]) * \sin(\theta_N)$$

$$y_{eR}(p;[x_U, y_U, \theta_U]) = -x_e(p;[x_U, y_U, \theta_U]) * \sin(\theta_N) + y_e(p;[x_U, y_U, \theta_U]) * \cos(\theta_N)$$

The dimensions of the bounding box can be computed as follows:

$$a_{RB} = \max\ \{x_{eR}(p;\ [\ x_U,\ y_U,\ \theta_U])\}\ \text{over}\ \{[-\pi \text{ to } +\pi];\ [h,\ k,\ \Theta]\};$$

$$b_{RB} = \max\ \{y_{eR}(p;\ [\ x_U,\ y_U,\ \theta_U])\}\ \text{over}\ \{[-\pi \text{ to } +\pi];\ [h,\ k,\ \Theta]\};$$

The $x_{eR}(.)$ and $y_{eR}(.)$ in the earlier equation can be simplified to get:

$$x_{eR}(p;[x_U, y_U, \theta_U]) = x_U*\cos(\theta_N) + y_U*\sin(\theta_N) + a*\cos(p)*\cos(\theta_U) + b*\sin(p)*\sin(\theta_U)$$

$$y_{eR}(p;[x_U, y_U, \theta_U]) = -x_U*\sin(\theta_N) + y_U*\cos(\theta_N) + a*\cos(p)*\sin(\theta_U) + b*\sin(p)*\cos(\theta_U)$$

It is well known that for two functions f_1 and f_2,

$$\max(f_1+f_2) \leq \max(f_1) + \max(f_2)$$

Thus the maximum values of $x_{eR}(.)$ and $y_{eR}(.)$ can be computed using the maximum values of their component functions. The computations for $\max\{x_{eR}(.)\}$ proceed as shown below. In this case the component functions are taken as $f_{X1} = x_U*\cos(\theta_N) + y_U*\sin(\theta_N)$, and $f_{X2} = a*\cos(p)*\cos(\theta_U) + b*\sin(p)*\sin(\theta_U)$. It is easy to show that

$$\max(f_{X1}) = d_{hk}*\max(|\cos(\Gamma - \theta_N)|)$$

where,

$$d_{hk} = (h^2 + k^2)^{1/2}, \text{ and}$$

$$\Gamma = [\tan^{-1}(k/h),\ \tan^{-1}(-k/h)]$$

For a specified value of θ_U, the $\max(f_{X2})$ will occur at

$$p_{\text{XCrit}} = \tan^{-1}(b*\tan(\theta_U)/a)$$

Substituting $p = p_{\text{XCrit}}$, f_{X2} can be simplified to obtain,

$$f_{X2} = \text{sqrt}[a^2*\cos^2(\theta_U) + b^2 * \sin^2(\theta_U)]$$

The critical points for this function are at $\theta_U = [0, \pm k\pi/2]$. If the maximum rotational uncertainty $\Theta < \pi/2$, then

$$\max\{f_{X2}\} = \max\{a,\ \text{sqrt}[a^2*\cos^2(\Theta) + b^2*\sin^2(\Theta)\}$$

The computations for $\max\{y_{eE}(.)\}$ proceed along similar lines and is briefly sketched below. The component functions are taken as $f_{Y1} = -x_U*\sin(\theta_N) + y_U*\cos(\theta_N)$, and $f_{Y2} = a*\cos(p)*\sin(\theta_U) + b*\sin(p)*\cos(\theta_U)$. It is easy to show that

$$\max(f_{Y1}) = d_{hk}*\ \max(|\sin(\Gamma - \theta_{UN})|)$$

For a specified value of θ_U, the $\max(f_{Y2})$ will occur at

$$P_{\text{YCrit}} = \tan^{-1}(b/a/\tan(\theta_U)).$$

Substituting $p = p_{\text{YCrit}}$, f_{Y2} can be simplified to obtain,

$$f_{Y2} = \text{sqrt}[a^2*\sin^2(\theta_U) + b^2*\cos^2(\theta_U)]$$

As earlier, it can be shown that

$$\max\{f_{Y2}\} = \max\{b, \text{sqrt}\ [a^2*\sin^2(\Theta) + b^2*\cos^2(\Theta)\ \}.$$

Thus the dimensions of the bounding box can be easily obtained as

$$a_{RB} = d_{hk}*\max(|\cos(\Gamma - \theta_N)|) + \max\{a, \text{sqrt}\ [a^2*\cos^2(\Theta) + b^2*\sin^2(\Theta)\}$$

$$b_{RB} = d_{hk}*\max(|\sin(\Gamma - \theta_N)|) + \max\{b,\ \text{sqrt}\ [a^2*\sin^2(\Theta) + b^2*\cos^2(\Theta)\}$$

An example for the bounding box computation is shown in Figure 1. The nominal ellipse is centred at (0,0) with $(a,b) = (5,2)$ and rotated counter-clockwise by 135°. The uncertainty levels $[h,\ k,\ \Theta]$ are taken as [0.5, 1. 5, 20] respectively. The figure shows the nominal ellipse and a set of perturbed ellipses. For these values, the $[a_{RB},\ b_{RB}]$ can be computed to be [6.4142, 3.9552]. The bounding rectangle is also shown in Fig. 1. The bounding rectangles

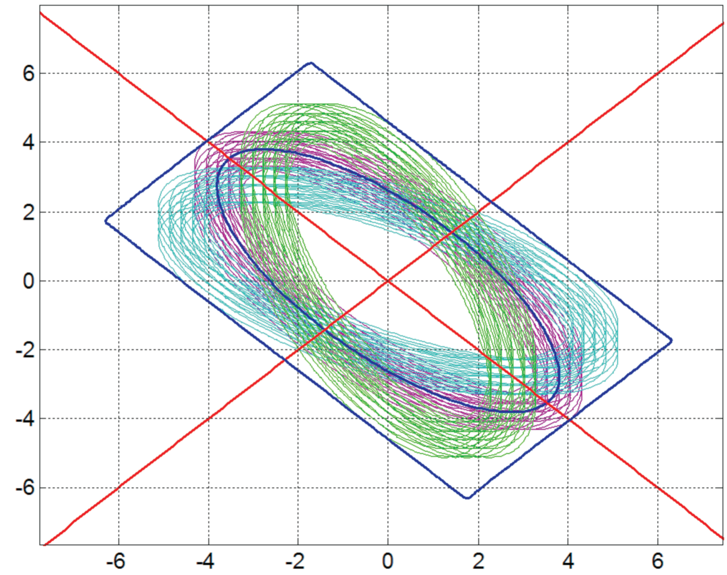

Figure 1. Axis-alilgned bounding box calculation.

over time for each obstacle can be similarly computed. The path planning problem then can be solved treating these bounding boxes as the obstacles, instead of the elliptic obstacles.

4. IMPLEMENTATION OF THE GENETIC ALGORITHM-BASED PATH PLANNING

As discussed earlier, the B-splines have been used for parametric path-representation. The knot locations are selected at time instants [0, 2, 4, … 10]. This fixes the time-horizon for the problem and has a bearing on the time taken to reach the goal. This should in principle be selected based on the velocity constraints on the robot. The spline basis functions are selected to be of polynomial order 2, i.e., linear. This guarantees C° continuous trajectories. One can select a higher polynomial order to get smoother trajectories. Thus, there are $N = 4$ basis spline functions and require the unknown vector to be of size 4. As disussed earlier, the N^{th} parameter is fixed to enforce the 'goal-reaching' condition based on the $(N–1)^{\text{th}}$ parameter as

$$x_N = 2 - x_{N\text{-}1}$$

It can be shown that with the above selection, the trajectory will reach the goal '1' at 7 s. Thus for the problem we are left with $(N–1) = 3$ independent parameters, which should be constrained to lie in [0,1). The same spline configuration is used for both the x and y trajectories with independent parameters $\{\boldsymbol{X}, \boldsymbol{Y} \in R^3\}$, and hence the path-planning problem requires a search over a 6 dimensional space. A possible x-y path is thus represented as $\text{P}(\boldsymbol{X},\boldsymbol{Y})$.

A set of obstacles is assumed given for the problem. The 'obstacle avoidance' is checked for each (x,y) point of the trajectory. If a particular point lies within any of the bounding rectangles, it implies an intersection with the obstacle. An obstacle function has been defined which gives a positive real number in case of a collision, and negative otherwise. The collision function (CF) is defined as the number of times the obstacle function is 'positive' along a path. It is also desired to have a minimum path length (PL). Additionally a minimum norm $|[\boldsymbol{X},\boldsymbol{Y}]|$ solution is being searched for. Thus the minimisation problem can be stated as below:

Find $([\boldsymbol{X},\boldsymbol{Y}])$ that minimises

$$\text{PL}(\text{P}(\boldsymbol{X},\boldsymbol{Y})) + \text{CF}(\text{P}(\boldsymbol{X},\boldsymbol{Y})) + |[\boldsymbol{X},\boldsymbol{Y}]|$$

Subject to

$$\boldsymbol{X},\boldsymbol{Y} \in [0,1)$$

The GAs offer an effective and powerful method of solving such optimization problems over large search spaces. It employs a population-based search method, and hence has a good chance of overcoming the local minima problem faced in conventional gradient-based search techniques. Each individual in a population is represented by a chromosome of independent variables. The search is implemented using genetic operators like cross-over, selection, and mutation. The current work uses the 'Genetic Algorithm and Direct Search Toolbox' available with MATLAB.

4.1 Example

As discussed earlier, an elliptic obstacle (E) can be described by the set of five parameters: position of the centre, the semi major and minor axes, and the rotation from a reference axis as $E := \{x_C, y_C, a, b, \theta\}$.

The nominal parameters for five elliptic obstacles are defined as:

$$E1{:}\{0.55+\sin(t)/20,\quad 0.45+\sin(t)/20,\quad 0.12, 0.06,\quad 1.5\sin(1.5t)\}$$
$$E2{:}\{0.20+\sin(2t)/20,\quad 0.15+\sin(t)/20,\quad 0.04, 0.12,\quad 1.5\sin(2t)\}$$
$$E3{:}\{0.80+\sin(t)/20,\quad 0.80+\sin(t)/20,\quad 0.04, 0.12,\quad \pi/4+1.5\sin(2.5t)\}$$
$$E4{:}\{0.20+\sin(2t)/20,\quad 0.80+\sin(t)/20,\quad 0.12, 0.04,\quad \pi/2+1.5\sin(1.25t)\}$$
$$E5{:}\{0.80+\sin(2t)/20,\quad 0.20+\sin(t)/20,\quad 0.12, 0.04,\quad \pi/3+\sin(3t)\}$$

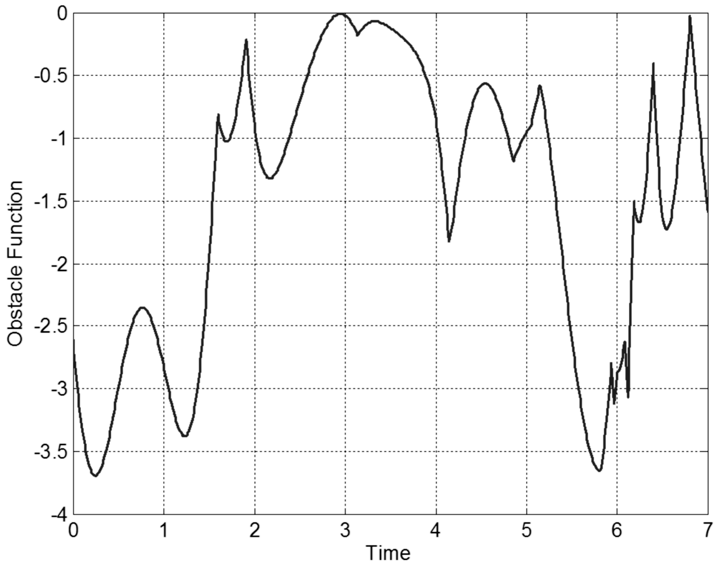

Figure 2. Plot of the obstacle-function over the optimal trajectory.

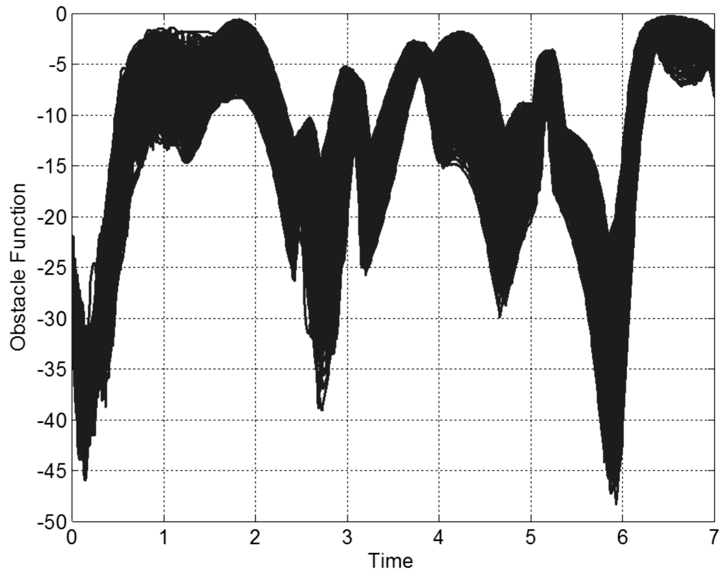

Figure 3. Plot of the Obstacle function over 500 simulation runs.

A GA optimisation call with 40 as the population size is used to obtain an optimal path. The fact that the trajectory avoids all the obstacles is established by evaluating the obstacle-function over time and is shown in Fig. 2. It is seen that the maximum of the obstacle function is –0.01 at t = 2.95 s. The fact that the trajectory indeed avoids the obstacles with uncertainty in the position and orientation, over and above the nominal dynamics, is assured by the non-positive values of the obstacle function as can be seen in Figure 3 over 500 simulation runs.

5. CONCLUSION

A novel analytical result for computing the axes-aligned bounding-box for the ellipses with uncertainty is presented. This has been applied to robotic path-planning problem in the presence of dynamically moving elliptic obstacles. The GA is used to obtain the shortest path from the initial to goal position avoiding the moving obstacles. The validity of the proposed method is illustrated with a simulation result.

ACKNOWLEDGEMENT

Dr G.K. Singh is grateful to the Mobile Intelligent Autonomous Systems Group at CSIR, South Africa, for providing the financial support and the opportunity to do the post-doctoral research.

REFERENCES

1. Kostaras, A.N.; Nikolos, I.K.; Tsourveloudis, N.C. & Valvanis, K.P. Evolutionary algorithm based on-line path planner for UAV navigation. *In* Proceedings of the 10th Mediterranean Conf. on Control and Automation-MED2002, 9-12 July 2002, Lisbon, Portugal.

2. Castillo, O. & Trujillo, L. Multiple objective optimisation genetic algorithms for path-planning in autonomous mobile robots. *Int. J. Comp. Sys, Signals,* 2005, **6**(1).
3. Thomaz, C.E.; Pacheco, M.A.C. & Vellasco, M.M.B.R. Mobile Robot Path Planning using Genetic Algorithms. *In* Proceedings of the Intl Work-conference on Artificial and Natural Neural Networks: Foundations and Tools for Neural Modelling. *Lecture Notes Comp. Sci.*, 1999, **1606,** 671-79.
4. Shahidi, N.; Esmaeilzadeh, H.; Abdollahi, M. & Lucas, C. Memetic algorithm based path-planning for a mobile robot. *Proc. World Academy Sci. Eng. Technol.*, January 2005, **1**.
5. Tychonievich, L.; Zaret, D.; Mantegna, J.; Evans, R.; Muehle, E. & Martin, S. (1989). A Maneuvering-board approach to path-planning with moving obstacles. *In* Proceedings of the 11th International Joint Conference on Artificial Intelligence, **2**, 1017-021.
6. Fiorini, P. & Shiller, Z. Motion planning in dynamic environments using velocity obstacles. *Int. J. Robotics Res.,* 1998, **17**(7), 760-772.
7. Vadakkepat, P.; Tan, K.C. & Ming-Liang, W. Evolutionary artificial potential fields and their application in real-time robot path-planning. *In* Proceedings of the 2000 Congress on Evolutionary Computation, **1**, 256-63.
8. Katz, R. & Delrieux, C. Logical architecture for robot path planning. *In* Argentian Symposium on Artificial Intelligence ASAI 2003. http://www.frcu.utn.edu.ar/deptos/depto_3/32JAIIO/asai/asai.html.
9. van den Berg, J. & Overmars, M. Planning time-minimal safe paths amidst unpredictably moving obstacles. *Int. J. Robotics Res.,* December 2008, **27**(11-12), 1274-294.
10. Guibas, L.J.; Hsu, D.; Kurniawati, H. & Rehman, E. Bounded uncertainty roadmaps for path planning. *In* International Workshop on the Algorithmic Foundations of Robotics, 2008. [downloaded from http://www.wafr.org/papers/wafr08-guibas.pdf].
11. Candido, S. Autonomous robot path planning using a genetic algorithm. Project Report. http://www-cvr.ai.uiuc.edu/~scandido/pdf/ GApathplan.pdf.
12. Walther M.; Steinhaus, P. & Dillmann R. Using B-Splines, for mobile path representation and motion control. *In* Proceedings of the 2nd European Conference on Mobile Robots, ECMR 2005.

Contributors

Dr G.K. Singh received his BTech (Aeronautical Engineering), MTech, and PhD (Electrical Engineering) from Indian Institute of Technology, Kanpur, in 1989, 1991, and 1999 respectively. Presently, he is a Scientist at the National Aerospace Laboratories, Bangalore since 1998. He is working on application of sliding mode controller designs to flight control. His main research interests are flight dynamics, control, and optimisation.

Dr A.K. Gopal received MSc(Engineering) and PhD from the University of Natal in 2000 and 2003 respectively. He is currently a Senior Design Engineer at BAE Systems, in South Africa. He is responsible for establishing the Mobile, Intelligent, Autonomous Systems (MIAS) Research Group at the CSIR in South Africa, in 2007, where he served as the Research Leader for 2007 and 2008.