Spline Guided Path of a Mobile Robot with Obstacle Avoidance Characteristics

D. K. Biswas
TIC, CMERI, Durgapur, India

(Email: dbiswas@cmeri.res.in)

Abstract

Path planning of a mobile robot is a wide field of research. Researchers in different places of the world have developed various techniques/algorithms for planning the movement of a mobile robot - online or offline. On line path planning is difficult than off line but when there is any obstacle appear on the path during movement in off line planning that obstacle avoidance is difficult. The developed off line path planning algorithm based on spline CURVE PATH FOLLOWING method, can be useful for offline planning and obstacle avoidance. The obstacle is treated as a static obstacle in 2D environment. The obstacles are detected by the mobile robot and avoid it smoothly. The method of path generation is continuous between any two successive control points.

Key Words: Mobile robot, path planning, spline, control point, obstacle avoidance

1 Introduction

Automated guided vehicle follow the predefined path in a known environment. Path points are found by various guiding methodology. Basically the path points are defined prior to start of vehicle movement. The essential points as inputs have high density but if the point density is very low then the question of path tracking like a path planning program such as via point following is coming into play. To generate trajectory through via points various methods are available. Robot path planning is also scientific for tracking via points. But via points tracking and obstacle avoidance at the same times is really a difficult task.

Mobile robot path planning has a wide range of techniques. Some of the methods are offline and some are online [1, 2]. Offline techniques are useful for predefined or guided path and for implementation environmental characteristics are not required to feed to the path guiding algorithm. Whereas, in online path planning [3, 4, 5, 6, 7], information about the working environment, is not required. In most of the offline techniques obstacle avoidance characteristics are ignored but in this algorithm this characteristic is present and that's why this algorithm is useful in both the cases. As the obstacle avoidance characteristic present in this path following method, so the offline planning becomes online when an obstacle is detected.

In this paper, I have started with brief discussion of spline method [8]. Then I have chosen a typical spline which has the ability to target towards the goal avoiding simple obstacle as well as poses proper continuity, segment wise. At last a simulation result is given considering point mobile agent/ robot.

2 The Cardinal Spline

The algorithms used in CAD and other graphical representation are based on different curve fitting technique. Polynomial curves constructed piece wise keeping certain continuity across various curve section. The piece wise curve so constructed has certain order or degree of polynomial [8, 9, 10, 11].

For data interpolation and/or smoothing a wide class of one-dimensional or multi-dimensional spline functions are used as shown in Fig. (1). The splines of finite dimensional in nature have proper utility in computations and representation. Splines are bounded through control points. The closeness of the curve to the control points depends on the tension parameter.

Various types of splines are used in computer graphics. All the polynomial curves are constrained by certain continuity condition across each section and the entire polynomial is defined by a series of control points and boundaries.

All the curves may be categorized broadly in two groups: explicit and parametric curve. Explicit curves are very simple and in 2D it is defined as $y = f(x)$. For example, $y = ax^3 + bx^2 + cx + d$. Parametric curves are more general and can be defined as $p = a_0 + a_1u + a_2u^2 + ... + a_nu^n$ (n^{th} order parametric curve).

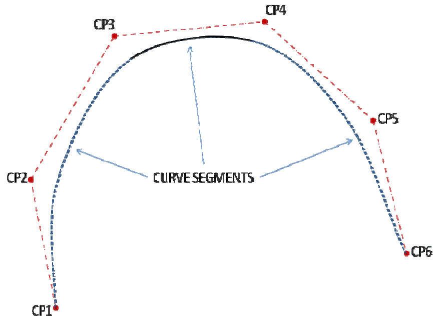

Fig. 1: Curve segments and the control points

A special spline called cardinal spline can form a large curve by joining each segment keeping proper continuity sequentially. An array of control points and a tension parameter specify this spline. This spline passes smoothly through each control point that means no sharp corners are present in the tightness of the curve. The following illustration shows a set of points and a cardinal spline that passes through each point in the set. It is to be noticed that the curve does not start from the first point and not reaches at the last points, but these points affect the shape of the curve. The tension parameter used is 0.1 as shown in Fig. (2).

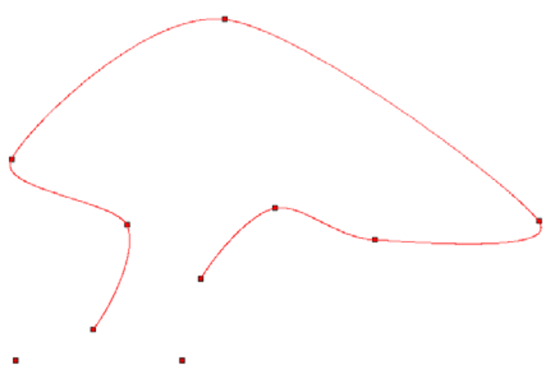

Fig. 2: Two Dimensional Cardinal spline

Fig. (3) shows four cardinal splines passing through the same set of control points. The value of tension parameter is displayed for each spline. This shows that the value of tension parameter 0 corresponds to infinite physical tension which forcing the curve to take the shortest way (straight lines) between points. For tension parameter 1 corresponds to no physical tension and the curve to takes least total bend. If the tension parameter is more than 1, the curve behaves like a compressed spring and takes a longer path (as shown, first and last points are not displayed).

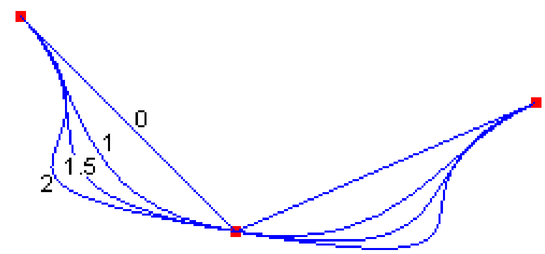

Fig. 3: Cardinal Spline with different tension parameter

The above discussion shows that a cardinal spline of tension parameter value between 0 & 1 can be chosen to get a path of required continuity at each joint.

3 Spline Method for Path Planning

At the starting a 2D cardinal spline is considered to get an obstacle free path. A general third order parametric spline curve [8, 9, 10, 11] can be represented as follows.

$$P(u) = au^3 + bu^2 + cu + d \text{ ----------(1)}$$

In x-y space this can be represented as

$$\left.\begin{aligned} x(u) &= a_x h^3 + b_x u^2 + c_x u + d_x \\ y(u) &= a_y h^3 + b_y u^2 + c_y u + d_y \end{aligned}\right\} \text{ ---------(2)}$$

Where u is bounded as; $0 \le u \le 1$.

In matrix form this can be written as

$$P(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} . \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \text{ ---------(3)}$$

So for each section of the Cardinal Spline, the coefficients a, b, c and d are to be determined, considering the boundary conditions P(0) and P(1), and considering tangent continuity (continuity depends on the required smoothness at each junction) at the extreme points.

Cardinal Spline uses four consecutive control points to set the boundary condition for each of the segments. The gradient at Pk is the gradient of the line connected Pk-1 and Pk+1. Similarly the gradient at Pk+1 is the gradient of the line passing through the control points Pk and Pk+2, as shown in Fig. (4). Thus the result of the curve segment so formed by the four control points is in the intermediate two points. Whereas at the starting and ending, control points give the continuity at the intermediate points.

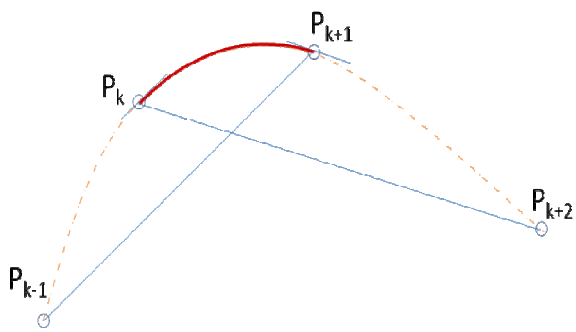

Fig. 4: Control points and the Gradients

Thus the boundary conditions can be defined as:

$$\left.\begin{aligned} P(0) &= P_k \\ P(1) &= P_{k+1} \\ \dot{P}(0) &= \frac{1}{2}(1-t)(P_{k+1}-P_{k-1}) \\ \dot{P}(1) &= \frac{1}{2}(1-t)(P_{k+2}-P_k) \end{aligned}\right\} --------(4)$$

Substituting the above into Eq. (3) and evaluating it, the cardinal spline equation becomes

$$P(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} . M_c . \begin{bmatrix} P_{k-1} \\ P_k \\ P_{k+1} \\ P_{k+2} \end{bmatrix} ----(5)$$

Where, Mc is the cardinal matrix and obtained as

$$M_c = \begin{bmatrix} -s & 2-s & s-2 & s \\ 2s & s-3 & 3-2s & -s \\ -s & 0 & s & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} ------(6)$$

and s is defined as

$$s = \frac{1}{2}(1-t) \qquad ----------(7)$$

The value of t i.e. the tension parameter conform the looseness or tightness of the curve passing through the control points.

Expanding Eq. (5) we will get

$$\begin{aligned} P(u) = {} & P_{k-1}C_0(u) + P_kC_1(u) + \\ & P_{k+1}C_2(u) + P_{k+2}C_3(u) \qquad -------(8) \end{aligned}$$

Where, C_n (n: 0-3) are termed as the Cardinal blending function. To obtain each of the coordinate points along the constrained curve so formed blending function is essential to estimate indexed continual and uniformly spaced curve points with required continuity.

The cardinal blending functions are:

$$\left.\begin{aligned} C_0 &= -su^3 + 2su^2 - su \\ C_1 &= (2-s)u^3 + (s-3)u^2 + 1 \\ C_2 &= (s-2)u^3 + (3-2s)u^2 + su \\ C_3 &= su^3 - su^2 \end{aligned}\right\} ------(9)$$

4 Implementation

From the previous section it is clear that to get a smooth continuity curve minimum four control points are necessary. But this results that the curve only passes through the intermediate control points. To get the curve passing through all the control points the initial and final control points are repeated only. This is solving the problem of passing the curve through all the control points. So if there are m number of control points then the total number increases in 2, i.e. total number of control points becomes (m+2) only. This does affect the basic relation of the Cardinal Spline generation as shown in Fig. (5).

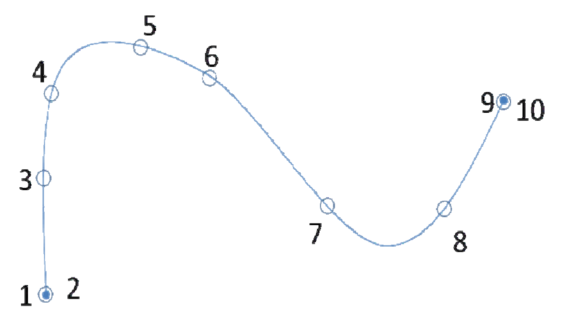

Fig. 5: Formation of Path Curve

As initial and final two points are repeated so they are superimposed and they are the similar points. Control points 2 and 9 are starting and ending of the formed curve. Initial orientation of the path is defined through control points 1 and 3, whereas 8 and 10 defines the final orientation. The path curve passes through the all other control points.

The formed cardinal spline formed takes four consecutive control points at a time. Consider the first segment generated taking control points 1, 2, 3 and 4. The curve so formed connects control points 2 & 3. To get the next segment connecting control points 3 & 4, the control points considered are 2, 3, 4 and 5. Thus the total curve formed as 2->3, 3->4, 4.... etc and so on considering four consecutive control points for each of the segments. The continuity is also maintained as per the discussion in section 3.

5 Path planning

The spline algorithm is implemented in a point robot which has the ability to pass through via points, considered as control points, as well as has the capability to avoid obstacle. Presently robotic research aims at building of autonomous mobile robots, which can plan its collision free, time optimal path through the structured environment in the presence of static obstacles.

Adaptive approach is applicable for a long-range navigation problem, depending on the working terrain condition. The robot behavior adapts, by selecting the appropriate perception, trajectory generation and execution of control functionalities. Basically, two kinds of navigation modes are used for solving most of the navigational problems, such as reflex mode and planned mode [1, 2, 3, 5, 6]. In reflex mode the environment is easy, i.e., essentially flat and lightly cluttered and the robot can efficiently move just on the basis of information provided for navigation. Whereas, in the planned mode the environment is complex and the robot may be trapped in dead-ends for instance: in such cases, trajectory planners need a model of the environment.

This spline method is used to get collision-free path for a point robot working in lightly cluttered 2D space. Simulation is carried out, in a grid of 500x500 m2 and the maximum velocity of the robot is assumed to be 2m/s. A simulation result is shown in Fig. (6). The simulation is carried out in MatLab on windows vista OS.

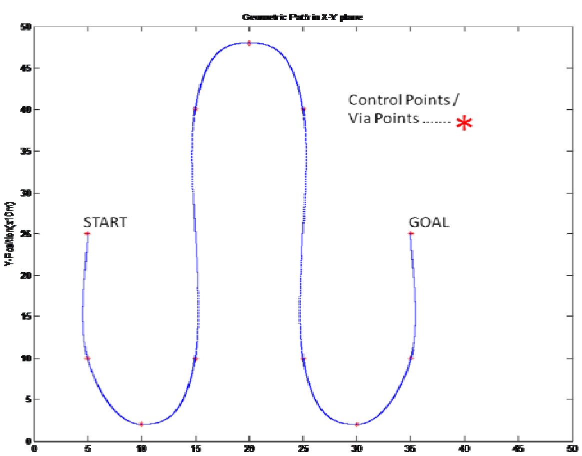

Fig. 6: Tracing the path through via points

The robot is considered as a point robot. It starts from the initial position (50, 250) and tracked all the intermediate via points (as shown in the figure above) and finally reached the target location (350, 250). It follows a zigzag path which resembles that it can survey a certain space passing through some intermediate target and can reach at the goal safely and silently.

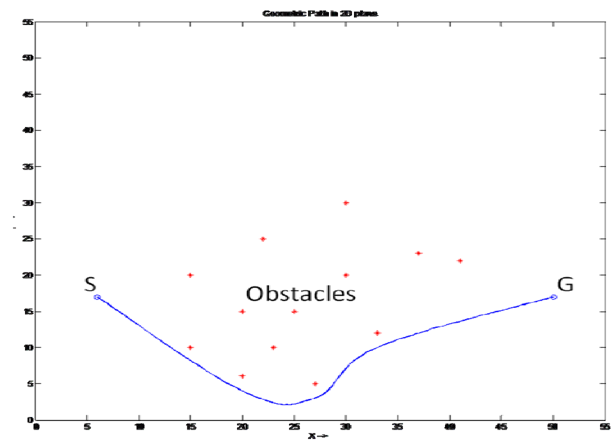

Fig. 7: Obstacle avoidance by spline method

Subsequently a question may arise in the robotic researcher's mind, whether this spline algorithm possess any obstacle avoidance ability? I have tried to find out the answer and got it. The following result shows clearly that the algorithm has the ability to bring the fruit shown in Fig. (7). The obstacles are small in number and all the obstacles are grouped together to avoid the complexity of the algorithm.

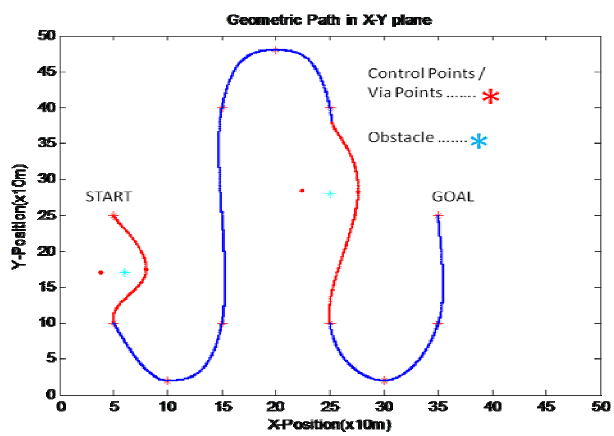

Fig. 8: Path tracking through the via points as well as obstacle avoidance

In Fig. (8) it is shown that the algorithm is useful to track vehicle through via points and at the same time it can avoid any obstacle around its path. It is to be noted that the continuity and gradient is as per the requirement of the tracked path and can be set through the tension parameter. It is also to be noted that the path is obtained by defining the appropriate via points and the target.

6 The Algorithm

The algorithm developed for path planning in the above fashion by using a simple cardinal spline. Appropriate use of certain technique gives a fruitful result. Some typical methodologies to be followed are given below.

- Set a 2D environment with lightly cluttered obstacles.
- Select required starting and target set of the robot.
- Repeat the starting and target locations.
- Set via position to be followed by the robot serially.
- Place the robot at starting position.
- Let the robot to move towards via positions to track the target.
- If there is any obstacle nearest the path of the robot then allow the robot to follow the obstacle avoidance method to pass through a new and obstacle free path.
- Allow the robot to continue till target is reached.

The obstacle avoidance method to be followed by the cardinal spline algorithm is very easy and computationally fast. Obstacles are detected before reaching the robot within the obstacle region. If any obstacle is detected then the avoidance algorithm, based on cardinal spline, generate via points on the either side of the obstacle. Just before reaching within the obstacle region, searching of generated via point is completed and the algorithm consider the newly accepted point as an obstacle via point. The path then regenerated through the newly formed via point maintain the conformity and continuity of the path. Once the obstacle is avoided the robot again starts to follow the subsequent via point left till to reach to the goal.

7 Software architecture & Discussion of result

Broadly the software architecture of the path planning algorithm based on cardinal spline is shown in Fig. (9).

Fig. (6) shows the path tracing the via points control points. The curvature continuity at each of via points is maintained (discussed earlier). If there is any requirement of station keeping any via points then the specified via points may be treated as temporary goal. The robot stops moving for certain period and again starts for the next via point till to reach to goal. Simulation result shows that the robot passes through via points, treated as control points of the cardinal spline.

Fig. (7) shows the obstacle avoidance capability of the cardinal spline. It considered all the obstacle points together and chose the shortest route to follow. The robot passes through the control points so generated by the forth coming obstacle points maintaining continuity characteristic.

The obstacle avoidance characteristic as well as via position passing phenomena is illustrated by the simulation result in Fig. (8). The confined path so generated by the cardinal spline methodology fulfills the basic requirement of smoothness of the path and target tracking.

So from the above discussion it is seen that the cardinal spline method may be useful in real life problem of path planning. This is the initiation. It may have some problems that can be resolved in future works. But it true that as this algorithm is capable of reaching the target, it may not have any trapping of local mini problem. The path so formed is a smooth curve so oscillation problem is not present in this method.

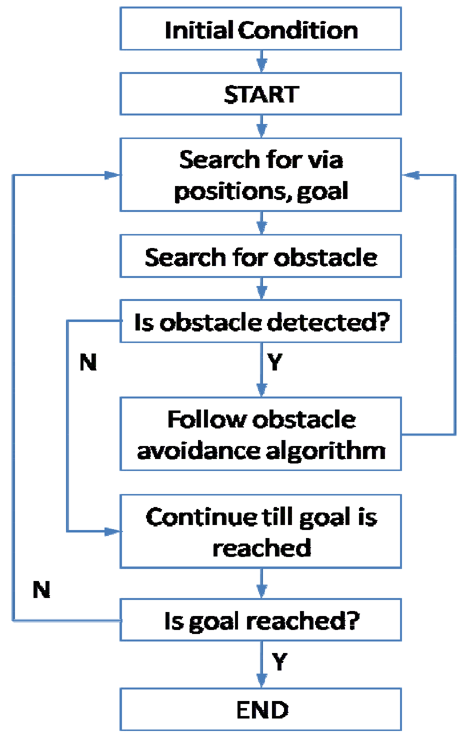

Fig. 9: General flow chart of the algorithm

8 Scope for Future work

The result of this algorithm brought is to a satisfactory level. The control points i.e. via points are treated as static. This algorithm is not tested in a dynamic environment. So there is a scope of further development of this algorithm where all the obstacles are moving.

The 2D space as considered for simulation is lightly cluttered and few obstacles are present. This algorithm has enough scope to simulate as real life problem in densely cluttered obstacles.

It is to be informed that only single robot is considered to justify the feasibility of the algorithm. So there is scope for developing the algorithm to use in a multi-agent-system.

Only 2-D workspace is considered in the present work but in future, an attempt will be made to consider a 3-D workspace.

9 Conclusion

This paper represents a novel obstacle avoidance technique as well as another technique to follow via points for various surveillance and inspection purpose. The ability of this methodology is the secret of the formation of Cardinal Spline. This methodology will be useful in real life application. This is the study of the feasibility of a parametric curve to use in robotic application.

References

[1] Jean Claude Latombe, Robot Motion Planning: Kluwer Academic Press, 1991.

[2] D. K. Biswas, D. K. Pratihar & S K Ghosh, "Path Planning of Multiple Robots Working in the Common Workspace – Using Potential Field Approach", National Conference on Advance Manufacturing and Robotics; 10-11 Jan, 2004; at CMERI, Durgapur.

[3] Simon Lacroix, Sara Fleury, et al, "Reactive Navigation in Outdoor Environments", IJRR special issue on 'Field and Service Robotics', Spring 1998.

[4] D. K. Pratihar, W. Bibel, "Path Planning for Cooperating Robots Using a GA-Fuzzy Approach", Proc. Of Dagstuhl-Seminar (01431), 2001, pp. 193-210.

[5] O. Khatib, "Real-time Obstacle Avoidance for Manipulators and Mobile Robots" International Journal of Robotics Research, 1986, pp. 5(1):90-98.

[6] J. Borenstein and Y. Koren, "Real-time Obstacle Avoidance for Fast Mobile Robots in Cluttered Environments", Proceedings of the 1990 IEEE International Conference on Robotics and Automation, Cincinnati, Ohio, May 13-18, 1990, pp. 572-577.

[7] R.W. Beard, T.W. McLain, "Motion Planning Using Potential Fields", Brigham Young University, Provo, Utah 84602, January 31, 2003.

[8] Halit Eren, Chun Che et all, "Implementation of the Spline Method for Mobile Robot Path Control," IEEE, 1999, pp: 739-744.

[9] de Figueiredo, R. J. P.; Markandey, V., "Spline-based algorithms for shape from shading (for robot vision application)," Proceedings of the SPIE, The International Society for Optical Engineering, 1988, vol. 850; pp. 152-5.

[10] Boehm, W., "Multivariate spline algorithms," Mathematics of Surfaces. Proceedings of a Conference. Clarendon Press, Oxford, UK, 1986, pp. 197-215.

[11] Spath, H., "One Dimensional Spline Interpolation Algorithms," A. K. Peters, Wellesley, Massachusetts, 1999.