Path Space Approach for Planning 2D Shortest Path Based on Elliptic Workspace Geometry Mapping

Ihn Namgung*
Director, Intelligent Robot Center, Yusong-gu, Jang-Dong 48,
Daeduk college, Jungkok 3F, Daejon 305-715, Korea

A new algorithm for planning a collision-free path based on algebraic curve is developed and the concept of collision-free Path Space (PS) is introduced. This paper presents a Geometry Mapping (GM) based on two straight curves in which the intermediate connection point is organized in elliptic locus (δ, θ). The GM produces two-dimensional PS that is used to create the shortest collision-free path. The elliptic locus of intermediate connection point has a special property in that the total distance between the focus points through a point on ellipse is the same regardless of the location of the intermediate connection point on the ellipse. Since the radial distance, δ, represents the total length of the path, the collision-free path can be found as the GM proceeds from $\delta=0$ (the direct path) to $\delta=\delta_{\max}$(the longest path) resulting in the minimum time search. The GM of elliptic workspace (EWS) requires calculation of interference in circumferential direction only. The procedure for GM includes categorization of obstacles to reduce necessary calculation. A GM based on rectangular workspace (RWS) using Cartesian coordinate is also considered to show yet another possible GM. The transformations of PS among Circular Workspace Geometry Mapping (CWS GM), Elliptic Workspace Geometry Mapping (EWS GM), and Rectangular Workspace Geometry Mapping (RWS GM), are also considered. The simulations for the EWS GM on various computer systems are carried out to measure performance of algorithm and the results are presented.

Key Words : Robot Motion Planning, Path Space, Collision-free Path Space, Collision-free Path Planning, Geometry Mapping, Elliptic Workspace Geometry Mapping, Robot Task Planning

Nomenclature

S	: Start point of path
G	: Goal point of path
Q	: Intermediate connection point
PS, PS$_n$, CPS	: Path space, n-dimensional path space, and complete path space
FS	: Free space, where obstacle images are not mapped
GM	: Geometry mapping
CWS GM	: Circular workspace GM
EWS GM	: Elliptic workspace GM
RWS GM	: Rectangular workspace GM
AWS	: Actual workspace
P, P$_i$	: Vertex of an obstacle, i-th vertex of an obstacle
d	: Focus length of ellipse
μ	: Parameter describing the line-segment for an obstacle edge
θ, δ	: Parameter describing the location of Q, the angular distance and the radial distance

1. Introduction

Robot path planning and collision avoidance is an essential step toward high-level command/

* **E-mail** : inamgung@dif.or.kr
TEL : +82-42-867-8553; **FAX** : +82-42-862-8326
Director, Intelligent Robot Center, Yusong-gu, Jang-Dong 48, Daeduk college, Jungkok 3F, Daejon 305-715, Korea. (Manuscript **Received** February 28, 2003; **Revised** October 6, 2003)

control of robot. It is increasingly necessary to communicate robot in an advanced level, or task level. The path planning and collision avoidance would eliminate the tedious process of manual control/programming of a robot. The Robot operator would specify a start point and goal point, and the path planning and collision avoidance takes care of the rest. This will greatly reduce robot control/programming efforts, and reduces accidents in workplace, and also enhance man-machine interface.

A path planning here is concerned with a path connecting two points in a space without interfering with any obstacles. A number of algorithms have been introduced to solve this problem. These algorithms include the visibility graph search, gird search, potential field method, and others. In the visibility graph search, connections are made among vertices of obstacles and start and goal points. The connections that do not intersect any obstacle are organized in a tree structure. The shortest path is found from the tree structure. (Lozano-Pérez and Wesley, 1979) were the first to introduce the idea of visibility graph search method. Variations of the method were presented by (Jun and Shin, 1991 ; Fujimura, 1994 ; Xue et al., 1996 ; Mansor and Morris, 1999 ; Jiang et al., 1999).

In the grid search method, the workspace is divided into several cells and obstacle occupancy of the cells is determined as empty, mixed and full. Mixed cells are recursively further divided into smaller cells and occupancy is determined again. Empty cells are organized in a tree structure and a connection of adjacent empty cells from start to goal point results in a path. The free space search method or the occupancy map method uses similar technique to find a path as grid search method. Some of the notable works that belong to this method includes those by (Singh and Wagh, 1987 ; Wu et al., 1987 ; Kambhampati and Davis, 1986 ; Barbehenn and Hutchinson, 1995 ; Jung and Gupta, 1997 ; Szczerba et al., 1998).

In the potential field method, obstacles are represented by repulsive pole and the goal point is represented by attractive pole. A collision-free path from start to goal point is obtained by optimization methods. The problem of this method is the existence of local minima created by multiple objects in the robot workspace. When the search is directed to the local minima, the method fails to find the goal point. The potential field search method was proposed by (Khatib, 1986). Others who contributed to the method include (Khosla and Volpe, 1988 ; Barraquand and Latombe, 1991 ; Hwang and Ahuja, 1992 ; Koditschek and Rimon, 1992 ; Wang and Hamam, 1992 ; Zelinsky, 1994 ; Alsultan and Aliyu, 1996 ; Li and Bui, 1998 ; Ong and Gilbert, 1998 ; Ge and Cui, 2000, 2002) to name a few. It should also note that penalty function method is similar to the potential field method. When the proposed path intersects with obstacles, a penalty is imposed on the cost function and an iterative optimization technique is used to find a path.

All of the above method used some form of iterative method and hence it is desirable to find a method that does not employ iterative method to find collision-free path. As far as the author is aware of, all path-planning methods so far presented or published are concerned with producing only one path. However, theoretically, collision-free path connecting two points is not just one but infinite. All collision-free paths belong to a class of space defined as a collision-free path space, which is a subset of Path Space (PS).

2. Theory

2.1 Concepts of path space

Let define **W** be the robot workspace in Euclidean space, and **S**, **G** be the start and goal point (i.e. **S**, **G**∈**W**). Define **L** be a path based on a class of algebraic curve that connects points **S** and **G**. Define $\mathbf{L}_{\text{free}}$ be a collision-free path and $\mathbf{L}_{\text{int}}$ be an obstacle interfering path. Now define an n-dimensional path space (PS_{n}) that contains all possible paths based on the same class of algebraic curve as follows :

$$PS_n=\sum_{i=0}^{\infty}L_i^n=\sum_{i=0}^{\infty}L_{free\_i}^n+\sum_{i=0}^{\infty}L_{int\_i}^n,\ n=1,\cdots,\infty \quad (1)$$

where PS_{n} is n-dimensional path space and n is the number of parameters that are used to define path L_i^n. For example, $PS_1 = L_{free\_0} + L_{int\_0}$ is a special case that consists of only one path, the direct path from S to G. PS_2 is a 2-dimensional path space based on two independent parameters, which may be based on two linear curves or a single quadratic curve in 2-dimensions and this is the case treated in this paper. PS_3 is a 3-dimensional path space based on three independent parameters, which may be a combination of a linear and a quadratic algebraic curve in 2-dimensions, etc.

Clearly, an n-dimensional path space (PS_n) consists of collision-free path space (PS_{n-free}) and obstacle interfering path space (PS_{n-int}).

$$PS_n = PS_{n-free} + PS_{n-\mathrm{int}} \tag{2}$$

The complete PS, denoted as CPS, is the whole path space that consists of all class of path space.

$$CPS = \sum_{n=1}^{\infty} PS_n \tag{3}$$

The relationships are graphically shown in Fig. 1.

The methods for object modeling, (Bernabeu and Tornero, 2000), or configuration space, (Moutarlier et al., 2000 ; Lynch et al., 1998 ; Lozano-Perez, 1979, 1981), are not treated in this paper, and it is assumed that those are already done. Hence, robot is assumed to be a point.

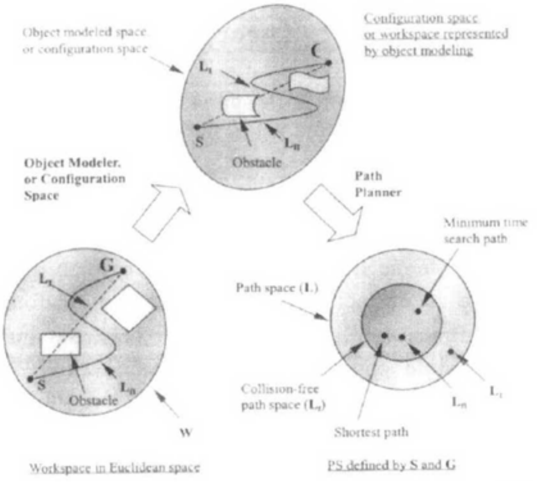


Fig. 1 Concept of path space and relationship with object modeling and configuration space

With a general path planner, additional constraints such as the shortest path, the minimum time search path or the minimum energy path, etc., can be imposed on the path planner. These specific cases are a subset of collision-free path in Fig. 1.

In this paper, a 2-dimensional path space, PS_2 is investigated. It is note that the collision-free path algorithm can be applied either for the mobile robots or for the robot manipulators. In order to apply the algorithm to the robot manipulators, configuration space has to be obtained first, and then the collision-free path algorithm should be applied to the configuration space. For mobile robot, it is not required to construct the configuration space and may practically proceed with the appropriate object modeling technique, (Bernabeu and Tornero, 2000). A mobile robot path planning can be done in the robot workspace after the appropriate object modeling of robot workspace environment.

2.2 Algebraic curve for path planning

The problem of finding collision-free path is here viewed as a geometric one requiring a solution to be a co-ordinated curve. The solution could be a connection of straight paths, a combination of straight and curved paths, a combination of curved paths, or a single high-order polynomial curve. A collision-free path, therefore, can be expressed as a connection of algebraic curve in the Euclidean space. An algebraic curve in Euclidean space can be represented in implicit form ($F(x, y) = 0$), or in explicit form ($y = f(x)$), or in parametric form ($x = f(s)$, $y = g(t)$). For some curves, for example closed curve or multi-valued curve, it is not possible to express the curve in explicit form. Explicit or implicit form of curve may also suffer from having infinite slop. On the other hand parametric representation has many advantages over other forms, notably it has directional property inherent in it. The directional property simplifies computation, and this is one reason that parametric curves/surfaces are widely used for Computer Graphics and CAD/CAM, see (Bezier, 1972 ; Rogers and Adams, 1990) for further information. (Namgung, 1989, 1997 ; Namgung and Duffy, 1996) introduced the use of parametric curve for robot path planning.

A collision-free path can be a series of connected line segment shown as path (a) of Fig. 2, or a high order polynomial curve shown as path (b) of Fig. 2. Whether the path is a connected line segments or a single polynomial curve, the shape of the curve has to be controllable and that the interference with any of the obstacles should be determined easily. It is clear that a collision-free path can be constructed by using algebraic curves. Moreover computation of interference check is deterministic for 1st or 2nd order algebraic curves, see (Namgung, 1989, 1997 ; Namgung and Duffy, 1996).

For example, in Fig. 2 path (a), the intermediate connection point controls the shape of path (a) by changing the location of them. In addition to controlling the shape of path, the defining parameters of connection points are used to create images of obstacle by plotting the obstacle interference. The region of parameter space, that causes obstacle interference, is the mapped image of the obstacle, and the empty space where obstacle image is not mapped corresponds to the collision-free paths. The collision-free path defined here is a subset of Path Space (PS) that is defined by the specific base curves. The process of creating obstacle image in parameter space is described in the following sections.

Another attractive feature of using algebraic curve for path planning is that the complexity of path can be controlled with limiting the number of control points or the order of base curve. In other word, PS can be constructed with base curves of increasing complexity. Once a class of PS is filled with object images, it indicates that the refinement of PS or a search with additional connection points or higher order base curve is required.

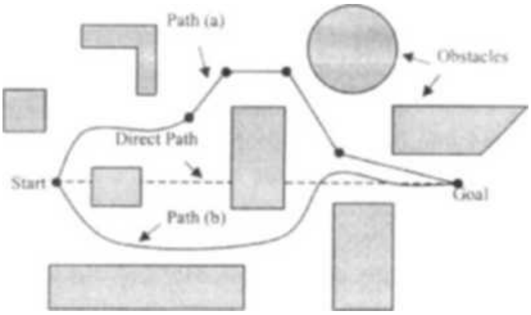


Fig. 2 Collision-free path in a 2D robot workspace

3. Geometry Mapping Based on Elliptic Locus of the Intermediate Connection Point

Figure 3 shows a situation where the direct path **SG** interferes with an obstacle OB1, and a straight path connecting the start point, S, to intermediate connection point, **Q**, and to the goal point, G, is formed to avoid the obstacle interference. The connection point, **Q**, can be structured in a coordinate that it can sweep the AWS and maps the possible interference with obstacle in a space defined by the defining parameters of intermediate connection point. This process is defined as GM and the one developed in this paper is based on the elliptic coordinate of the intermediate connection point. The transformed robot workspace is defined as PS. An ellipse can be defined by two parameters such as δ and θ in Fig. 3 where θ defines the angle and δ defines the distance from focus to the nearest major axis point. It can be defined in implicit form by the axis distance from a (major axis point) to b (minor axis point). The standard form is given by

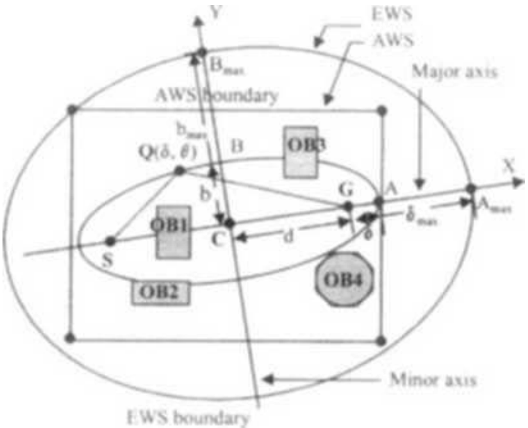


Fig. 3 Elliptic workspace and the locus of the intermediate connection point

$$\frac{x^2}{a^2}+\frac{y^2}{b^2}=1 \tag{4}$$

and the parametric form is expressed as

$$\begin{aligned} x&=a\cos\theta \\ y&=b\sin\theta \end{aligned} \tag{5}$$

An ellipse should also satisfy the following geometric relationships.

$$a=d+\delta=\frac{SQ+QG}{2}=BS=BG \tag{6}$$

The minor axis point b can be calculated using eq. (6) and the right triangle **CGB**.

$$b=\{(d+\delta)^2-d^2\}^{\frac{1}{2}}=\{\delta^2+2d\delta\}^{\frac{1}{2}} \tag{7}$$

$$\frac{x^2}{(d+\delta)^2}+\frac{y^2}{\delta^2+2d\delta}=1 \tag{8}$$

$$\begin{aligned} x&=R_x(\delta,\ \theta)=(d+\delta)\cos\theta \\ y&=R_y(\delta,\ \theta)=(\delta^2+2d\delta)^{\frac{1}{2}}\sin\theta \end{aligned} \tag{9}$$

where d is the distance from **C** to **G**, i.e. d=(**G**-**S**)/2. Now substituting eqs (6) and (7) into (4) and (5), an ellipse can be defined completely in terms of the radial distance, δ, from the focus point to the major axis point and the angle, θ, measured from the positive major axis. Fig. 4 shows constant angle locus that is obtained from eq. (9) with constant value of θ.

The EWS should enclose the whole AWS (Actual Workspace). Note that the eq. (8) can be modified to determine the relative position of a point whether it is inside or outside of the ellipse. Let assume that δ is arbitrarily set to d, and substitute AWS boundary vertices (V_x, V_y) into eq. (8) and rearrange to get eq. (10).

$$F=\frac{V_x^2}{4d^2}+\frac{V_y^2}{3d^2}-1 \tag{10}$$

Depending on the location of vertex, the value of F can be divided into three cases, i.e.

i) $F<0$: (V_x, V_y) is inside the ellipse
ii) $F=0$: (V_x, V_y) is on the ellipse locus
iii) $F>0$: (V_x, V_y) is outside the ellipse.

The eq. (9) is used to find the vertex of AWS boundary point that is farthest away from the center point, **C**.

After identifying the vertex, (V_x, V_y), that produce the biggest value of F, the maximum value of δ_{max} can be calculated from the following eq. (11).

$$\delta_{max}=\frac{1}{2}\{|V-S|+|V-G|\}-d \tag{11}$$

Where $|\mathbf{V}-\mathbf{S}|$ denotes distance between **V** and S, and $|\mathbf{V}-\mathbf{G}|$ denotes distance between **V** and **G**.

The extreme locus of the intermediate connection point, Q_{max}, is defined by δ_{max} of eq. (11), and the parametric value δ_{max} sets the EWS boundary. With this value of δ_{max}, the valid range of θ and δ for EWS are $0° < \theta < 360°$ and $0 < \delta < \delta_{max}$, respectively. Now the next step is to calculate the obstacle interference for the selected value of δ. A typical case of obstacle interference with an ellipse is given in Fig. 4. For a prescribed value of parameter δ, the interference range of the angular parameter are given by $\theta_1 < \theta < \theta_2$ (interference of **OB**1) and $\theta_a < \theta < \theta_b$ (interference of AWS boundary). The existence of these obstacles prohibits the intermediate connection point to be in the range of $\theta_1 < \theta < \theta_2$ and $\theta_a < \theta < \theta_b$ and the calculation of these angles are necessary. For example, in Fig. 4, the angle, θ_1, is obtained from the intersection between the ellipse and the line $\mathbf{SP}_1$, and the angle, θ_a, is obtained from the intersection between the ellipse and the edge $\mathbf{W}_3\ \mathbf{W}_4$.

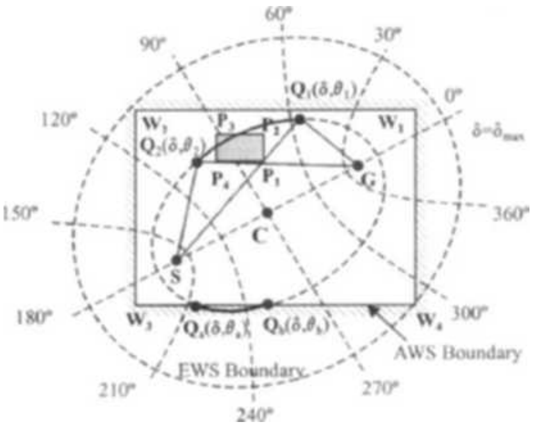


Fig. 4 A Typical case of obstacle interference and the constant angle locus of the connection point

Basically, there are two kinds of interference between an obstacle and the path that results in a blocked range of angle. One is the contact of an obstacle edge with the ellipse, for example the contact of the edge **P$_1$P$_4$** of **OB2** and the ellipse in Fig. 5. The other is the intersection of the extended line **SP$_1$** of **OB1** with the ellipse, in Fig. 5.

The condition for the first case requires that the point on the ellipse should also be on the line for the obstacle edge. The line equation expressing the edge by the vertices **P$_1$** and **P$_2$** of **OB2** in Fig. 5 is substituted into eq. (8) for an ellipse as follows.

$$\frac{\{(P_{2x}-P_{1x})\mu+P_{1x}\}^2}{(d+\delta)^2}+\frac{\{(P_{2y}-P_{1y})\mu+P_{1y}\}^2}{\delta^2+2d\delta}=1 \quad (12)$$

In eq. (12), parameters δ and μ are unknown variables and defines the ellipse and the edge **P$_1$P$_2$**, respectively. Rearrange eq. (12) for μ in descending order,

$$K\mu^2+2L\mu+M=0 \qquad (13)$$

where K, L, M are

$$\begin{aligned}K&=(P_{2x}-P_{1x})^2(\delta^2+2d\delta)+(P_{2y}-P_{1y})^2(d+\delta)^2\\L&=P_{1x}(P_{2x}-P_{1x})(\delta^2+2d\delta)+P_{1y}(P_{2y}-P_{1y})^2(d+\delta)^2\\M&=P_{1x}^2(\delta^2+2d\delta)+P_{1y}^2(d+\delta)^2\end{aligned} \quad (14)$$

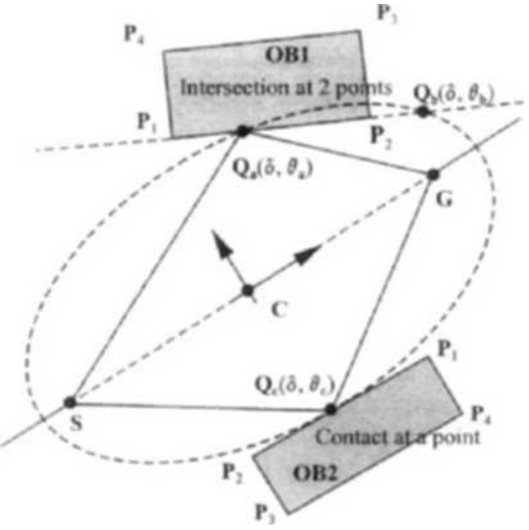


Fig. 5 Intersection or contact of obstacle edge with ellipse

For the prescribed value of δ, the parameters K, L, and M are constant, therefore solutions to eq. (13) can be obtained, and it is given by the following eq. (15).

$$\mu=\frac{-L\pm\sqrt{L^2-KM}}{K} \qquad (15)$$

The discriminant of eq. (15) exhibits three distinct cases: i) $L^2=KM$ indicates a contact, ii) $L^2>KM$ indicates intersection at two point, and iii) $L^2<KM$ indicates no interference occurs. Moreover actual calculation is necessary only if the numerator of eq. (15) is smaller than denominator since valid range of δ is $0\leq\delta\leq1$. This will effectively clear a singular case when K vanishes. The intersection point ($\mathbf{P}_x$, $\mathbf{P}_y$) can be obtained by substituting the resulting parameter μ into the line equation for the edge.

$$\begin{aligned}P_x&=(P_{2x}-P_{1x})\mu+P_{1x}\\P_y&=(P_{2y}-P_{1y})\mu+P_{1y}\end{aligned} \qquad (16)$$

The angular parameter, θ, of the point ($\mathbf{P}_x$, $\mathbf{P}_y$) which indicates angular location on ellipse can be calculated using eq. (9) where δ is known and ($\mathbf{P}_x$, $\mathbf{P}_y$) is substituted for (x, y).

When the discriminant of eq. (15) vanishes, a contact between the ellipse and the obstacle edge occurs. Expansion of the discriminant of eq. (15) is given as follows.

$$\begin{aligned}&\{P_{1x}(P_{2x}-P_{1x})(\delta^2+2d\delta)+P_{1y}(P_{2y}-P_{1y})^2(d+\delta)^2\}^2\\&-\{(P_{2x}-P_{1x})^2(\delta^2+2d\delta)+(P_{2y}-P_{1y})^2(d+\delta)^2\}\\&\cdot\{P_{1x}^2(\delta^2+2d\delta)+P_{1y}^2(d+\delta)^2-(\delta^2+2d\delta)(d+\delta)^2\}=0\end{aligned} \quad (17)$$

Re arranging eq. (17) for the unknown variable δ in descending order,

$$P\delta^2+2Pd\delta+R=0 \qquad (18)$$

where P and R is

$$\begin{aligned}P&=(P_{2x}-P_{1x})^2+(P_{2y}-P_{1y})^2\\R&=P_{1x}^2d^2-\{P_{1y}(P_{2x}-P_{1x})-P_{1x}(P_{2y}-P_{1y})\}\end{aligned} \qquad (19)$$

From eq. (18), δ can be obtained as follows,

$$\delta=-d\pm\sqrt{d^2-\frac{R}{P}} \qquad (20)$$

Since discriminant of eq (15) vanishes, μ is expressed as

$$\mu=-\frac{L}{K} \qquad (21)$$

Finally the contact point is obtained by substituting μ, the result of eq. (21), into eq. (16), and the angle, θ, at contact point can be calculated using eqs (5), (6) and (7).

The intersection of a line segment and the ellipse is now derived with an illustration given in Fig. 6, in which the extended line $\mathbf{SP}_1$ intersects with the ellipse. It is more convenient to express the ellipse in local coordinate than in global coordinate. The calculation of intersection can be simplified by expressing the workspace environment in local coordinate. The start point, **S**, and the goal point, **G**, can be expressed in local coordinate easily as S(-d, 0) and G(d, 0), respectively. Vertices of obstacles can be expressed in local coordinate where the coordinate axes align with the ellipse major/minor axes, see Fig. 6. Assume a vertex $\mathbf{V}_1(\mathbf{V}_{1x}, \mathbf{V}_{1y})$ of an obstacle is expressed in the global coordinate, then the coordinate transformation into the local coordinate, $\mathbf{P}_1(\mathbf{P}_{1x}, \mathbf{P}_{1y})$, is given by

$$\begin{aligned} P_{1x} &= (V_{1x}-C_x)\cos\varphi + (V_{1y}-C_y)\sin\varphi \\ P_{1y} &= -(V_{1x}-C_x)\sin\varphi + (V_{1y}-C_y)\cos\varphi \end{aligned} \tag{22}$$

And the line equation $\mathbf{SP}_1$ in local coordinate is given by

$$\begin{aligned} x &= (P_{1x}+d)\mu - d \\ y &= P_{1y}\cdot\mu \end{aligned} \tag{23}$$

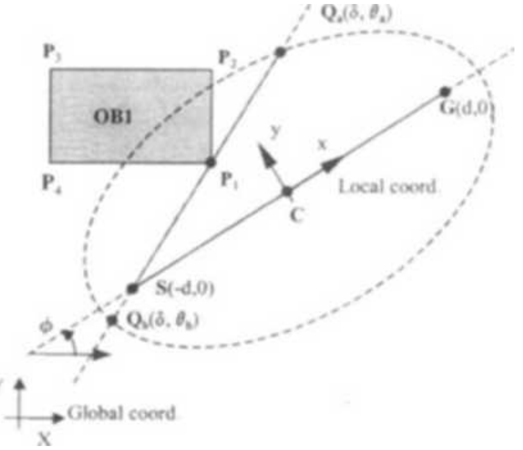


Fig. 6 Intersection between the extended line $\mathbf{SP}_1$ and the ellipse

In eq. (23), the parameter μ defines the line segment $\mathbf{SP}_1$. Now substitute eq. (23) into eq. (8), and arrange for μ in descending order,

$$A\mu^2 - 2B\mu - D = 0 \tag{24}$$

Where A, B, and D are constants that are given as follows.

$$\begin{aligned} A &= (d+\delta)^2 P_{1y}^2 + (\delta^2+2d\delta)(P_{1x}+d)^2 \\ B &= (\delta^2+2d\delta)(P_{1x}+d)\,d \\ D &= (\delta^2+2d\delta)^2 \end{aligned} \tag{25}$$

The parameters d, δ, and the vertex, $\mathbf{P}_1(\mathbf{P}_{1x}, \mathbf{P}_{1y})$, are all known. Hence the unknown parameter μ of eq. (24) can be obtained as follows.

$$\mu = \frac{B \pm \sqrt{B^2+AD}}{A} \tag{26}$$

Note that eq. (26) always returns double root since the discriminant is always positive, which can be seen from the fact that the line passing through S will intersect ellipse at two points. Having calculated two values of μ, the valid one is the one that is in the range of $0 \le \mu \le 1$. Fig. 6 shows a case where one of the value of μ is positive and the other is negative, and the one with positive value is valid one because it produces the intersection point that are of interest here. With the calculation of μ, the angular value of parameter, θ, can now be calculated using eq. (9).

Using above three basic operations, calculation of EWS boundary δ_{max} by eq. (11), contact colculation between an edge and ellipse by eqs. (20) and (21) and intersection calculation between an edge and ellipse by eq. (26), the EWS GM can now proceed. For example, in Fig 4, the blocked ranges of θ are shown by thick solid lines denoting the range of $\theta_1 \le \theta \le \theta_2$ and $\theta_a \le \theta \le \theta_b$. Because the blocked ranges of θ depend on the location of obstacles, it is required to categorize obstacles before calculating the parameter θ.

4. Categorization of Obstacles for Geometry Mapping

Previous section described three basic operations of obstacle interference check that are necessary for GM. As the prescribed value of the parameter, δ, changes, the interference range of the parameter, θ, also changes. The closed formulas derived in the previous section gives intersection and contact point on ellipse and allows to calculate the corresponding angle, however it does not provide any information about the blocked range of angle, θ. By categorizing obstacles, the blocked range of angle, θ, can be determined. This categorization also reduces calculations that are necessary for interference check. It is assumed that obstacles are convex in the categorization process. A concave obstacle can be decomposed to convex obstacles, and categorization can proceed afterward as obstacles OB2 and OB3 in Figs 11, 12 and 13.

Table 1 Obstacle categorization and interference range of angle θ

Cases	Condition	Interference Range	Equations	Reference Figure
OB1	all vertices outside ellipse, does not intersect line seg. **SG**	None	None	Figure 7
OB2	all vertices outside ellipse, intersect line segment **SG**	All $(0° \leq \theta \leq 360°)$	None	Figure 7
OB3	all vertices outside ellipse, intersect line outside **G** of **SG**	$0° \leq \theta_2$ $\theta_1 \leq 360$℃	(14), (15), (16), (5), (6), (7)	Figure 7
OB4	all vertices outside ellipse, intersect line outside **S** of **SG**	$\theta_1 \leq \theta \leq_2$	(14), (15), (16), (5), (6), (7)	Figure 7
OB5	all vertices outside ellipse, does not intersect line **SG**	$\theta_1 \leq \theta \leq \theta_2$	(14), (15), (16), (5), (6), (7)	Figure 7
OB6	intersect ellipse, does not intersect line **SG**	$\theta_1 \leq \theta \leq \theta_2$	(14), (15), (16), (23), (25), (26), (5), (6), (7)	Figure 8
OB7	intersect ellipse, intersect line outside **G** of **SG**	$0° \leq \theta_2$ $\theta_1 \leq 360°$	(14), (15), (16), (23), (25), (26), (5), (6), (7)	Figure 8
OB8	intersect ellipse, intersect line outside **S** of **SG**	$\theta_1 \leq \theta \leq \theta_2$	(14), (15), (16), (23), (25), (26), (5), (6), (7)	Figure 8
OB9	intersect ellipse both side, intersect line segment **SG**	All $(0° \leq \theta \leq 360°)$	None	Figure 8
OB10	intersect ellipse upside, intersect line segment **SG**	$\theta_1 \leq \theta \leq \theta_2$	(23), (25), (26), (5), (6), (7)	Figure 9
OB11	intersect ellipse downside, intersect line segment **SG**	$\theta_1 \leq \theta \leq \theta_2$	(23), (25), (26), (5), (6), (7)	Figure 9
OB12	inside ellipse, above line **SG**	$\theta_1 \leq \theta \leq_2$ $\theta_3 \leq \theta \leq \theta_4$	(23), (25), (26), (5), (6), (7)	Figure 9
OB13	inside ellipse, below line **SG**	$\theta_1 \leq \theta \leq \theta_2$ $\theta_3 \leq \theta \leq \theta_4$	(23), (25), (26), (5), (6), (7)	Figure 10
OB14	inside ellipse, intersect line outside **S** of **SG**	$0° \leq \theta \leq 1$ $\theta_2 \leq \theta \leq 360°$	(23), (25), (26), (5), (6), (7)	Figure 10
OB15	inside ellipse, intersect line outside **G** of **SG**	$\theta_1 \leq \theta \leq \theta_2$	(23), (25), (26), (5), (6), (7)	Figure 10
OB16	inside ellipse, intersect line segment **SG**	$\theta_1 \leq \theta \leq \theta_2$ $\theta_3 \leq \theta \leq \theta_4$	(23), (25), (26), (5), (6), (7)	Figure 10

Figures 7 through 10 shows categorization of obstacle and Table 1 summarizes the conditions for obstacle categorization and interference range with related equations to calculate the range of the angle, θ. The condition to categorize obstacles whether outside, intersect or within ellipse is determined by eq. (10). The intersection of line **SG** and obstacle edge can be determined by calculating interference check between obstacle edges and line **SG** or by determining location of vertices of obstacle about line **SG**. Discrimination of a vertex whether it is upside or lower side of line **SG** and the intersection of obstacle edge with the line SG is used in the process of obstacle categorization.

The interference range of angel, θ, can be computed for prescribed values of δ increasing from 0 to $\delta_{\max}$ for each obstacle. The accuracy of obstacle image produced by this process depends on the resolution of the parameter δ. Since the parameter δ is discretized, in some case it is necessary to calculate the accurate value of δ that is in between the digitized values when ellipse and obstacle are in contact. Equations (14), (19), (20), (21) and (5), (6), (7) can be used to calculate the contact between ellipse and obstacle which provides more accurate parameter values for the obstacle image.

In Figs 7 through 10, the blocked range of angle for each obstacle is marked with θ_1 and θ_2. These values are either the biggest or the smallest value of angle of interference either with an edge of obstacle or with a vertex of obstacle. Because each obstacle returns a blocked range of angle, a sorting process is necessary to get the combined

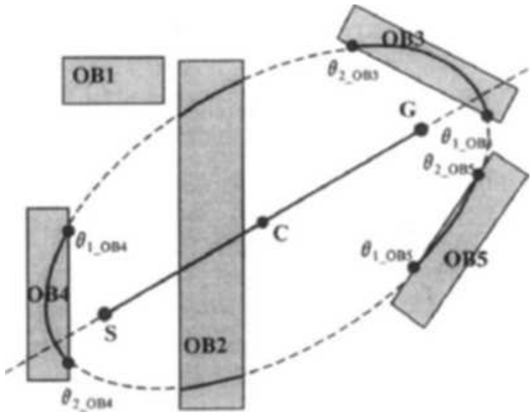


Fig. 7 Categorization of Obstacles — All vertices located outside of the ellipse (cases 1~5)

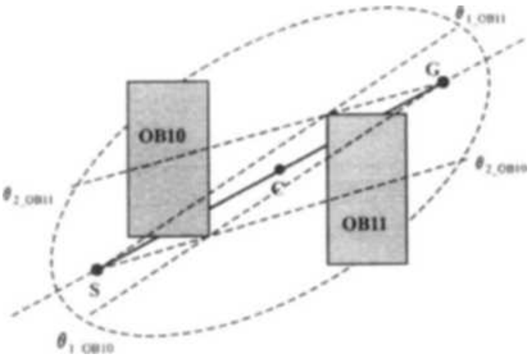


Fig. 9 Categorization of Obstacles — Some vertices are located within the ellipse (case 10, 11)

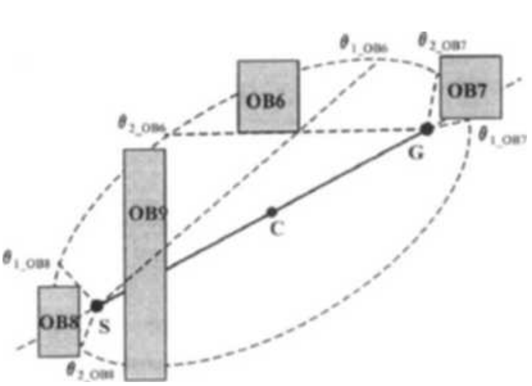


Fig. 8 Categorization of Obstacles — Some vertices are located within the ellipse (case 6~9)

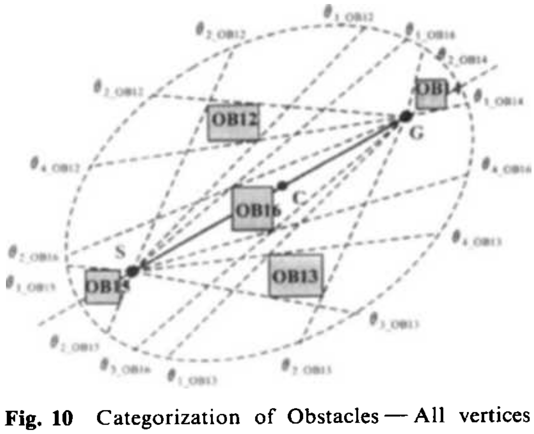


Fig. 10 Categorization of Obstacles — All vertices are located inside the ellipse (cases 12~16)

blocked range of angle for a prescribed value of parameter δ.

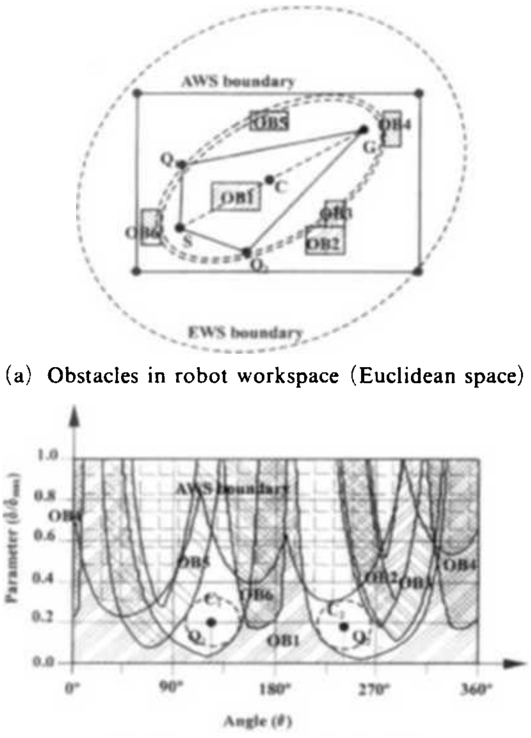
5. Simulation of EWS GM and Images of Obstacles in PS

A simulation of EWS was performed to measure the performance of the EWS GM algorithm. The parameter $\delta/\delta_{\max}$, instead of δ for dimensionless term, is used and discretized for resolution of $\Delta\delta=0.02$ and 0.1. The simulation results presented in Fig. 11 through 13 are those with $\Delta\delta=0.02$. The results of $\Delta\delta=0.1$ showed similar plot, but a little more coarse image. Table 2 presents a summary of performance of the EWS GM. The execution time clearly depends on the

Table 2 Summary of EWS GM simulation results

(unit : sec.)

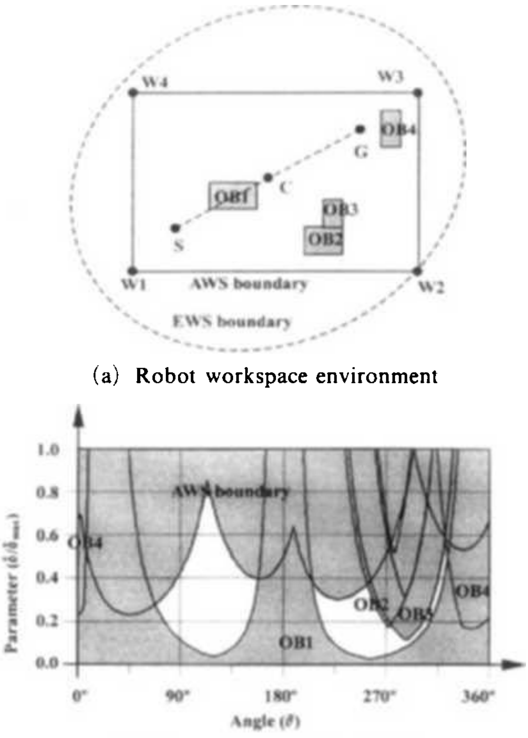
Reference figure	PentiumIV/2.53 GHz system		PentiumIII (Celeron) / 533 MHz system		Pentium/90 MHz system	
	$\Delta\delta=0.02$	$\Delta\delta=0.1$	$\Delta\delta=0.02$	$\Delta\delta=0.1$	$\Delta\delta=0.02$	$\Delta\delta=0.1$
Figure 11 (case 1)	0.001656	0.000344	0.005600	0.001210	0.137363	0.021978
Figure 12 (case 2)	0.001219	0.000266	0.004220	0.000820	0.104396	0.016484
Figure 13 (case 3)	0.001516	0.000329	0.005170	0.001090	0.126374	0.021978



(a) Obstacles in robot workspace (Euclidean space)

(b) Images of obstacles in PS

Fig. 11 Simulation of CM (case 1) — robot workspace and images of each obstacle in PS



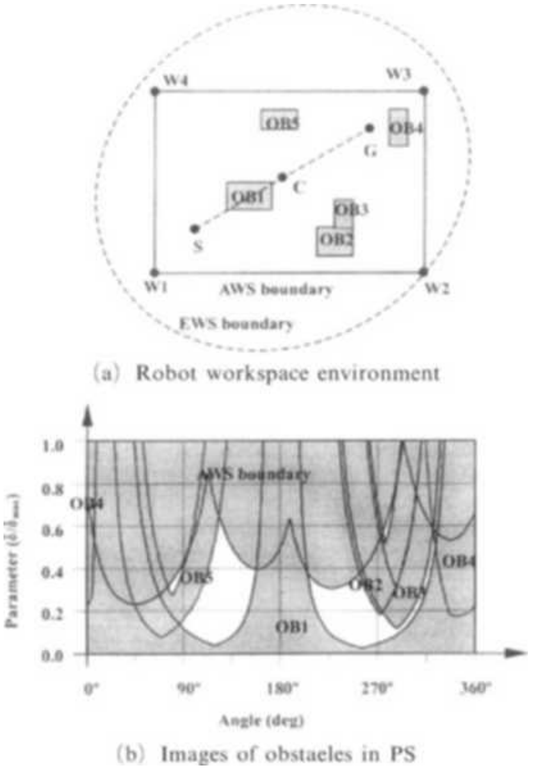
(a) Robot workspace environment

(b) Images of obstacles in PS

Fig. 12 Simulation of CM (case 2) — robot workspace and images of each obstacle in PS

number of obstacles and location of S and G.

In Fig. 11, obstacles and AWS boundary in Euclidian space are mapped into curved images, and filled with different hatching patterns in order to indicate the mapping between the obstacle and image. Two paths one from above **SG** line and another from below **SG** line are shown in Fig. 11. These paths indicate the safest path from each side of line **SG**. These paths correspond to center point of circle C1 and C2 of PS in which the circles represent the largest inscribing circle of clear area from each side of line **SG**. Figs. 12 and 13 show different simulation settings. In Fig. 12, OB5 and OB6 are removed from Fig. 11. In Fig. 13, OB6 is removed from Fig. 11. The determination of failure may depend on the resolution of GM since accuracy of images in PS depends on the resolution of parameter δ. The time to finish a complete GM is also depends on the resolution of parameter δ.



(a) Robot workspace environment

(b) Images of obstacles in PS

Fig. 13 Simulation of CM (case 3) —robot workspace and images of each obstacle in PS

Recall that in EWS GM, the smaller the value of parameter δ, the shorter the total length of path. This allows finding the shortest collision-free path in minimum time. In other word, the construction of PS can proceed by increasing $\delta=0.0$ to $\delta=\delta_{\max}$, and as soon as a collision-free path is found, the mapping stops and return with the collision-free path. This would take the minimum time to find the shortest collision-free path that connects the start point S and the goal point G.

6. Conversion of PS Between EWS GM and CWS GM

In (Namgung, 97), Circular Workspace (CWS) GM based on polar coordinate (θ_p, ρ) of intermediate connection point was developed. In this paper, Elliptic Workspace (EWS) GM based on elliptic locus (δ, θ_e) of intermediate connection point is developed. The CWS GM was carried out for prescribed value of parameter (θ_p) of the connection point and interference range of (ρ), while EWS GM was carried out for prescribed value of parameter (δ) and obtain the interference range of angle (θ_e). Note that the both PS is the same because both are based on the same base curve, only the mapping methodology is different. Hence it is of interest to know how they are related, and how one mapping can be converted to the other. Conversion from CWS GM (θ_p, δ) to EWS GM (ρ, θ_e), or vice versa can be obtained by equating the defining formulas for CWS GM and EWS GM.

$$\begin{aligned} x&=\rho\cos\theta_c=(d+\delta)\cos\theta_e \\ y&=\rho\sin\theta_c=(\delta^2+2d\delta)^{\frac{1}{2}}\sin\theta_e \end{aligned} \tag{27}$$

From eq. (27), the conversion from EWS GM to CWS GM is given as follows.

$$\theta_p=\tan^{-1}\frac{\sqrt{\delta^2+2d\delta}\sin\theta_e}{(d+\delta)\cos\theta_e} \tag{28}$$

$$\rho=\{(d+\delta)^2\cos^2\theta_e+(\delta^2+2d\delta)\sin^2\theta_e\}^{\frac{1}{2}} \tag{29}$$

where d=|**T**−**S**|/2. The conversion from CWS

GM to EWS GM is expressed as follows.

$$\delta=\frac{1}{2}\left[\{(\rho\cos\theta_p+d)^2+\rho^2\sin^2\theta_p\}^{\frac{1}{2}}+\{(\rho\cos\theta_p-d)^2+\rho^2\sin^2\theta_p\}^{\frac{1}{2}}\right]-d \tag{30}$$

$$\theta_e=\tan^{-1}\frac{(d+\delta)\sin\theta_p}{\sqrt{\delta^2+2d\delta}\cos\theta_p} \tag{31}$$

Where δ in eq. (31) is substituted by the result of eq. (30), and d=|**G**−**S**|/2. The conversion between the two mapping changes the shape of obstacle images in the resulting PS, as well as the defining area, a rectangular area defining PS mapping area, changes after conversion in either direction.

Although it was not treated in this paper, a Cartesian Workspace Geometry Mapping (RWS GM) is also possible. Fig. 14 shows the basic concepts. The intermediate connection point is organized in Cartesian coordinate (x, y). In RWS GM, the Cartesian workspace is defined to enclose AWS boundary. A local coordinate is set at the center point of **S** and **G**, and the x-axis is set toward goal point G, see Fig. 14. For the prescribed value of x, the interference ranges of parameter y for each obstacle are computed, and it is the process of RWS GM. The conversion from CWS GM (ρ, θ_c) to RWS GM (x_R, y_R) is given by eq. (27) and that from RWS GM (x_R, y_R) to CWS GM (ρ, θ_c) is as follows.

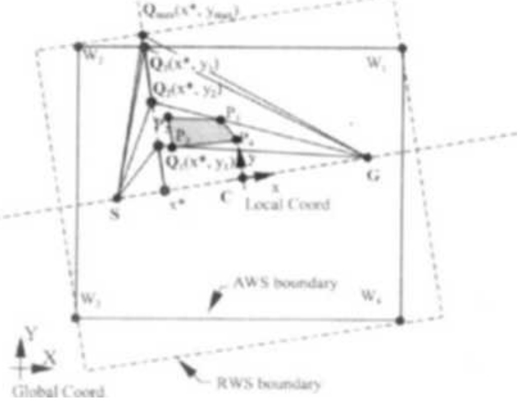


Fig. 14 Cartesian coordinate of intermediate connection point and RWS GM

$$\rho=\sqrt{x_R^2+y_R^2}$$
$$\theta_c=\tan^{-1}\frac{y_R}{x_R} \tag{32}$$

The conversion from EWS GM (δ, θ_e) to RWS GM (x_R, y_R) is given by eq. (27) and that from RWS GM (x_R, y_R) to EWS GM (δ, θ_c) is as follows.

$$\tan^2\theta_e-\left(\frac{y_R}{x_R}\right)^2-\sin^2\theta_e\left(\frac{d}{x_R}\right)^2=0 \tag{33}$$

$$\delta^2+2d\delta+d^2\cos^2\theta_e-x_R^2-y_R^2=0 \tag{34}$$

Eq. (33) can be solved using half-tangent law. Once the parameter θ_e is obtained, δ can be obtained from eq. (34). Note that the valid range of parameter for θ_e is $0\leq\theta_e\leq360$, and that for δ is $0\leq\delta\leq\delta_{max}$. Although δ_{max} is not known in the conversion from RWS GM (x_R, y_R) to EWS GM (δ, θ_e), other limits are known, and that can be used in the calculation of eqs (33) and (34).

7. Conclusion

The collision-free path investigated here is a series of connected line segment between two points, the start point, **S**, and the goal point, **G**, through a third point called the intermediate connection point, **Q**. The connection point is organized in elliptic locus (elliptic coordinate defined by angle, θ, and distance from focus along major axis, δ). Interference check between path and obstacles is carried out to map obstacles from Euclidian Space into images in PS. The mapping produces PS that belongs to PS_2 class of path space, which is based on two linear curves. The connection point acts as a vehicle for the mapping. Because the connection point is organized in elliptic coordinate, the mapping is call EWS GM.

To facilitate closed form solution for obstacle interference, obstacles are categorized depending on the location and the blocked range of parameter, θ, are assessed based on the categorization. This helps reduce the calculation time for interference check as well. The mapping process creates overlapping images of obstacles in PS, and a point from clear area of PS that is not mapped out by obstacle images identifies collision-free path.

The algorithm has a significant characteristic in that finding the shortest path in minimum time is possible with EWS GM. Because the radial distance (δ) is proportional to the total length of the path, the shortest path can be obtained by finding the smallest value of parameter (δ_{min}). Therefore, the search for a collision-free path can proceed by starting the radial parameter δ from $\delta=0$ to δ_{max}, as soon as a path is found the mapping process can stop. This way the search time can be minimized while obtaining the shortest collision-free path.

Simulation of the algorithm revealed that the execution time on a PC (PentiumIV/2.53 GHz CPU) is in the range of 0.0012 sec to 0.0022 sec for the problem setting shown in Figs 11~13. The fast execution time is achieved by the closed form derivation of interference calculation, such as intersection of obstacle edge or contact with obstacle vertex.

This paper presented the construction of different GMs that belong to the same PS_2. The relati onship between CWS GM, EWS GM and RWS GM was derived as well. A higher order GM that belong to PS_3 is next stage of research and a development of recursive algorithm for higher order PS another direction of future research.

References

Alsultan, K. S. and Aliyu, M. D. S., 1996, "A New Potential Field-Based Algorithm For Path Planning," *Journal of Intelligent & Robotic Systems*, Vol. 17, No. 3, pp. 265~282.

Barbehenn, M. and Hutchinson, S., 1995, "A Efficient Search and Hierarchical Motion Planning by Dynamically Maintaining Single-Source Shortest Paths Trees," *IEEE Transactions on Robotics and Automation*, Vol. 11, No. 2, pp. 198~214.

Barraquand, J. and Latombe, J. C., 1991, "Robot Motion Planning : A Distributed Representation Approach," *International Journal of Robotics Research*, Vol. 10 No. 6, pp. 460~468.

Bernabeu, E. J. and Tornero, J., 2000, "Optimal geometric modeler for robot motion planning," *Journal of Robotic Systems,* Vol. 17, No. 11, pp. 593~608.

Bezier, P., 1972, *Numerical Control-Mathematics and Application*, Translated by A. R. Forrest and A. F. Pankhurst, John Wiley & Sons, New York.

Fujimura, K., 1994, "Motion Planning Amid Transient Obstacles," *International Journal of Robotics Research*, Vol. 13 No. 5, pp. 395~407.

Ge, S. S. and Cui, Y. J., 2000, "New Potential Functions for Mobile robot Path Planning," *IEEE Transactions on Robotics & Automation*, Vol. 16, No. 5, pp. 615~620.

Ge, S. S. and Cui, Y. J., 2002, "Dynamic Motion Planning for Mobile Robots Using Potential Field Method," *Autonomous Robots*, No. 13, Vol. 3, pp. 207~222.

Hwang, Y. K. and Ahuja, N., 1992, "A Potential Field Approach to Path Planning," *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 1, pp. 23~32.

Jiang, K. C., Seneviratne, L. D. and Earles, S. W. E., 1999, "A Shortest Path Based Path Planning Algorithm for Nonholonomic Mobile Robots," *Journal of Intelligent & Robotic Systems*, Vol. 24, No. 4, pp. 347~366.

Jun, S. and Shin, K. G., 1991, "Shortest Path Planning in Discretized Workspaces Using Dominance Relation," *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 3, pp. 342~518.

Jung, D. and Gupta, K. K., 1997, "Octree-Based Hierarchical Distance Maps For Collision Detection," *Journal of Robotic Systems*, Vol. 14, No. 11, pp. 789~806.

Kambhampati, S. and Davis, L. S., 1986, "Multiresolution Path Planning for Mobile Robot," *IEEE Journal of Robotics and Automation*, Vol. RA-2, No. 3, pp. 135~145.

Khatib, O., 1986, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," *International Journal of Robotic Research*, Vol. 5, No. 1, pp. 90~98.

Khosla, P. and Volpe, R., 1988, "Superquadric Artificial Potential for Obstacle Avoidance and Approach," IEEE Proc. Int. Conf. on Robotics and Automation, Philadelphia, pp. 1778~1784.

Koditschek, D. E. and Rimon, E., 1992, "Exact Robot Navigation Using Artificial Potential Function," *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 5, pp. 501~518.

Li, Z. X. and Bui, T. D., 1998, "Robot Path Planning Using Fluid Model," *Journal of Intelligent & Robotic Systems*, Vol. 21, No. 1, pp. 29~50.

Lozano-Perez, T. and Wesley, M. A., 1979, "An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles," *Communications of ACM*, Vol. 22, No. 10, pp. 560~570.

Lozano-Perez, T., 1981, "Automatic Planning of Manipulator Transfer Movements," IEEE Trans. Systems, Man, and Cybernetics, Vol. SMC-11, No. 10, pp. 681~698.

Lynch, K. M., Shiroma, N., Arai, H. and Tanie, K., 2000, "Collision-Free Trajectory Planning For A 3-Dof Robot With A Passive Joint," *International Journal of Robotics Research*, Vol. 19, No. 12, pp. 1171~1184.

Mansor, M. A. and Morris, A. S., 1999, "Path Planning in Unknown Environment with Obstacles Using Virtual Window," *Journal of Intelligent & Robotic Systems*, Vol. 24, No. 3, pp. 235~251.

Moutarlier, P., Mirtich, B. and Canny, J., 1996, "Shortest Paths for a Car-like Robot to Manifolds in Configuration Space," *International Journal of Robotics Research*, Vol. 15, No. 1, pp. 36~60.

Namgung, I. and Duffy, J., 1997, "Two Dimensional Collision-Free Path Planning Using Linear Parametric Curve," *Journal of Robotic Systems*, Vol. 14, No. 9, pp. 659~673.

Namgung, I., 1996, "A Global Collision-Free Path Planning Using Parametric Parabola Through Geometry Mapping of Obstacles in Robot Work Space," *KSME International Journal*, Vol. 10, No. 4, pp. 443~449.

Namgung, I., 1989, "Planning Collision-Free Paths with Applications to Robot Manipulators," Ph. D. Dissertation, University of Florida.

Nearchou, A. C. and Aspragathos, N. A., 1998, "Collision-Free Cartesian Trajectory Generation Using Raster Scanning And Genetic Algorithm," *Journal of Intelligent & Robotic Systems*, Vol. 23 No. 2-4, pp. 351~377.

Ong, C. J. and Gilbert, E. G., 1998, "Robot path planning with penetration growth distance," *Journal of Robotic Systems*, Vol. 15, No. 2, pp. 57~74.

Rogers, D. F. and Adams, J. A., 1990, *Mathematical Elements for Computer Graphics, 2nd Ed.*, McGraw-Hill, New York.

Singh, J. S. and Wagh, M. D., 1987, "Robot Path Planning Using Intersecting Convex Shapes: Analysis and Simulation," *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 2, pp. 101~108.

Szczerba, R. J., Chen, D. Z. and Uhran, J. J., 1998, "Planning Shortest Paths among 2D and 3D Weighted Regions Using Framed-Subspaces," *International Journal of Robotics Research*, Vol. 17, No. 5, pp. 531~546.

Wang, D. and Hamam, Y, 1992, "Optimal Tranjectory Planning of Manipulators With Collision Detection and Avoidance," *International Journal of Robotics Research*, Vol. 11 No. 5, pp. 460~468.

Wu, Y. F., Widmayer, P., Schlag, M. D. F. and Wong, C. K., 1987, "Rectilinear Shortest Path and Minimum Spanning Trees in the Presence of Rectilinear Obstacles," *IEEE Trans. On Computer*, Vol. C-36, No. 3, pp. 321~331.

Xue, Q., Sheu, P. C. Y., Maciejewski, A. A. and Chien, S. Y. P., 1996, "Planning Of Collision-Free Paths For A Reconfigurable Dual Manipulator Equipped Mobile Robot," *Journal of Intelligent & Robotic Systems*, Vol. 17, No. 3, pp. 223~242.

Zelinsky, A., 1994, "Using Path Transforms to Guide the Search for Findpath in 2D," *International Journal of Robotics Research*, Vol. 13, No. 5, pp. 395~407.