

## Reactive Path Shaping: Local Path Planning for Autonomous Mobile Robots

Paul R. Schmitt  
(313) 741-9503

Wayne J. Book  
(404) 894-3247

paul@davinci.marc.gatech.edu    wayne.book@gtme.gatech.edu  
George W. Woodruff School of Mechanical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332-0405

### **Abstract**

*This paper introduces a path generator for autonomous nonholonomic mobile robots in partially known aisle environments. Reactive Path Shaping (RPS) is designed for autonomous vehicle obstacle avoidance and utilizes information about its environment through ultrasonic sensors. A path is constructed one piece at a time, based upon local environmental information. Each piece is initially considered to be rough and needs to be appropriately "shaped," or smoothed, to meet mobile robot nonholonomic kinematic constraints. Experiments on a simulated vehicle with ultrasonic sensors are briefly discussed.*

### **1.0 Introduction**

Many path planning techniques generate paths that are difficult or impossible for a nonholonomic vehicle to follow. Common situations may result in a desired motion that is outside of its kinematic and/or dynamic constraints. Other techniques designed for nonholonomic vehicles generate paths that are

rough due to discretization of steering angles or frequent changes of target points.

RPS derives its name from its ability to "react" to local surroundings. A path is constructed one piece at a time, based upon local environmental information. Each piece is initially considered to be rough and needs to be appropriately "shaped," or smoothed, to meet mobile robot nonholonomic kinematic or dynamic constraints. An autonomous vehicle path must meet curvature continuity constraints, to insure that it is feasible for the nonholonomic mobile robot.

RPS constructs curves from straight line segments and segments of a special class of curves called clothoids. Clothoids are continuous curves whose curvature varies linearly along its arc length. Clothoids have been used by researchers to satisfy constraints of nonholonomic mobile robots.<sup>1</sup>

RPS utilizes an environmental map to keep track of its surroundings. The *environmental map* efficiently stores and manipulates the large amount of range data from ultrasonic sensors. It is from this map that RPS generates a kind of potential field, keeping the path away from sensed obstacles.

## 2.0 Related Work

The following is an overview of other researchers' work related to RPS. It is not meant to be a comprehensive review as the area of mobile robot path planning is a broad and active one.

### 2.1 Potential Field Methods

Potential field methods provide a way to avoid costly searches.

One approach to this is the Virtual Force Field (VFF) method.<sup>2</sup> VFF, developed by Borenstein and Koren, combines certainty grids for obstacle representation and potential fields for navigation. The method is particularly well suited for ultrasonic sensing of objects.

Arkin's approach uses motor schemas for reactive/reflexive navigation.<sup>3, 4</sup> *Motor schemas* represent primitive motor behaviors, and examples include Move-to-goal, Avoid-static-obstacle, and Stay-on-path. A vector associated with a potential field is produced by each motor schema, which, after summation with the output of all active motor schemas, produces the motor commands for the robot. These schemas were inspired by motor behavior in animal systems and generate complex behaviors.

Another behavior based navigation system is the Distributed Architecture for Mobile Navigation (DAMN), used for driving a nonholonomic, cross-country vehicle through unmapped, open terrain. The DAMN approach allows for reactive planning while attempting to meet the vehicle nonholonomic constraints. A local map management module is used to maintain a representation of the environment in the vicinity of the vehicle. For navigation, several behaviors such as Goal-seeking, Avoid-obstacles, and Follow-road vote for or against a set of possible vehicle steering radii. For example, a nearby obstacle may cause the Avoid-obstacle behavior to vote for a sharp left-hand turn (i.e., sharp steering angle) while the Goal-seeking behavior favors its present course with little turning (straight steering angle). For convenience, the range of possible steering radii is discretized, most likely resulting in a jerky path as the vehicle jumps from one steering angle to the next. The path is suitable for a cross-country vehicle but may be rough for more sensitive warehouse or factory mobile robots.

Potential field methods, in general, suffer from the fact that they are difficult to apply to nonholonomic mobile robots. The combination of attractive and repulsive fields may easily generate a desired robot motion that is outside of its kinematic and/or dynamic constraints.

### 2.2 Aisle Navigation

Not much research has been performed in the area of aisle navigation. One exception is an obstacle avoidance system for mobile robots traveling through warehouse aisles developed by Borenstein et al. at the University of Michigan.<sup>5</sup> The system utilizes ultrasonic sensors to measure the width of the aisle and calculate the *loci* of the center points at 2 cm intervals. At every 2 cm of travel, the system attempts to steer toward successive center points to stay near the aisle middle.

Since the system drives toward a new point calculated from ultrasonic range information at frequent intervals, the resulting path is choppy and rough.

### 2.3 Nonholonomic Path Shapes

A lesser amount of studies have been done on generating paths and path shapes for nonholonomic vehicles.

#### 2.3.1 Circular Arcs

Laumond *et al.* construct piece-wise smooth paths constructed from line segments and arcs of circles. Reeds and Shepp have demonstrated that such curves are shortest paths for car-like vehicles.<sup>6</sup> The algorithm assumes that obstacles are known *a priori*.

Laumond's earlier work studied the problem of constructing a trajectory without backing up maneuvers for a circular, but car-like vehicle.<sup>7</sup> The trajectory avoids obstacles. The method constructs curves from line segments and circular arcs as before, and obstacles are assumed known *a priori*.

Paths consisting solely of straight line segments and arcs have inherent difficulties for nonholonomic mobile robots. When reaching the point of transition between a circular arc and straight line segment, a nonholonomic mobile robot must either alter its steering wheels orientation and/or velocity instantaneously or slow to a stop before adjusting the steering wheels. Unfortunately, both require awkward, jerky motion for the robot.

#### 2.3.2 Clothoids and Cubic Spirals

For non-stop vehicle motion, Kanayama introduces clothoids for connecting vehicle configurations.<sup>8</sup> The method assumes total knowledge of the surroundings for safe path development. Kanayama with Hartman took this method a step further by introducing a new set of curves, called *cubic spirals*.<sup>9</sup> These curves are generated by minimizing curvature in a cost function. They are better than clothoids in terms of their maximum curvature, but require extra computation for connecting positions and orientations in general.

### 2.4 Related Work Summary

Many alternative path planning techniques generate paths that are difficult or impossible for a nonholonomic vehicle to

follow. Common situations may result in a desired motion that is outside of its kinematic and/or dynamic constraints. Other techniques designed for nonholonomic vehicles generate paths that are rough due to discretization of steering angles or frequent changes of target points.

### 3.0 Vehicle Paths and Clothoids

#### 3.1 Nonholonomic vehicles

Often an autonomous mobile vehicle can be characterized as *nonholonomic*, or is said to have *nonholonomic* constraints. Such constraints can have several effects on autonomous vehicles.

- A vehicle is limited in how it can move. It has fewer local degrees of freedom than it has globally. In other words, while a position and orientation (i.e., a posture) may be achievable, a vehicle's current configuration often limits its ability to reach the posture without careful maneuvering or judicious use of its local degrees of freedom.
- Variables describing vehicle position and orientation are dependent on each other. For example, an automobile (a tricycle-like vehicle) cannot alter its orientation without changing its position. It cannot spin "on a dime." An aspect that must be considered for such a vehicle is its minimum turning radius described later in this paper.<sup>10</sup>

A primary difficulty in developing a path generator for a nonholonomic vehicle lies in giving a path to the vehicle that fits within its local degrees of freedom.

#### 3.2 Significance of Curvature Continuity

For continuous motion along a continuous path, torques given to the vehicle actuators are finite so long as curvature is also continuous. If the path curvature is not continuous at  $s$ , where  $s$  is the arc length, the actuator torques approach infinity for continuous motion. If stopping is allowed, the vehicle must pause at the point of discontinuity to alter its body or wheel angle.

#### 3.3 Clothoids

Clothoids are useful curves for generating smooth trajectories. A *clothoid* is a spiral whose curvature,  $\kappa$ , is a linear function of its arc length  $s$ ,

$$(1) \quad \kappa = k_c s + \kappa_0$$

where  $\kappa_0$  is the initial curvature and  $k_c$  a constant characterizing the shape of the clothoid.<sup>11</sup>  $\kappa_0$  is zero for the remainder of this paper.

For a two dimensional world coordinate  $(x, y)$  system, clothoids can be defined parametrically in terms of Fresnel integrals as

$$(2) \quad x = aC(t) = \int_0^t \cos(\pi u^2/2) du$$

$$(3) \quad y = aS(t) = \int_0^t \sin(\pi u^2/2) du$$

where the scaling factor  $a > 0$  and  $t$  varies from  $0 \rightarrow \infty$ . Fast but accurate approximations for these integrals have been found by Heald.<sup>12</sup> A useful property of clothoids is that the tangent angle,  $\theta$ , at  $t$  is given by

$$(4) \quad \theta = \frac{\pi}{2} t^2$$

### 4.0 Reactive Path Shaping (RPS)

RPS is outlined in Figure 1. Several inputs to RPS are given off-line: minimum turning radius, goal point, and spring model parameters. When running (on-line), RPS takes data from ultrasonic or other environmental sensors and stores it in a map (See 4.1 Environmental Map below). RPS takes the map data and generates a short path. The mobile robot moves over this short path and the process starts again until the goal point is reached.

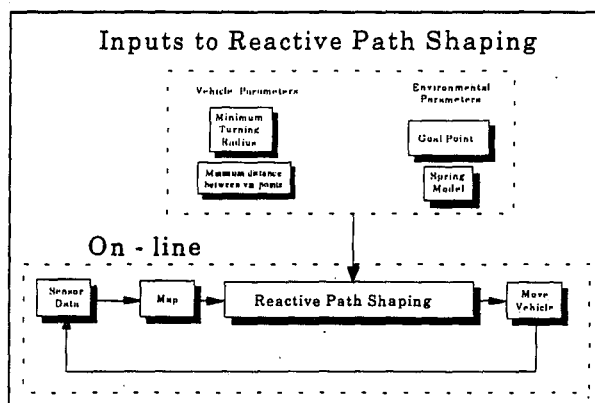


Figure 1- Reactive Path Shaping Flow Chart

#### 4.1 Environmental Map

An *environmental map* is used for the purpose of efficiently storing range sensor information.<sup>13, 14, 15, 16, 17</sup> An unfortunate characteristic of ultrasonic sensors, the type of sensor used in this paper, is that they are often noisy and inaccurate. To offset this, many sensors are installed at strategic locations and repeatedly activated. Thus to gain a more accurate representation of a robot's surroundings, a map is needed to quickly handle and store large amounts of data.

The map is a local one, with the vehicle always located at the map center. Thus, obstacles appear to "move" in the map while the vehicle center is fixed at the map center. To reduce errors

and computational overhead, map obstacles translate only; they do not rotate about the vehicle. Thus, a vehicle moving west will have map obstacles moving east, and map obstacles will not move for a vehicle that is rotating about its center.

An environmental map has proven to be useful and accurate in quickly and efficiently building an environmental map from a large amount of sensor data.

## 4.2 Via Points

Via points are intermediate points between the mobile robot's initial position and the goal point. They are initially positioned and then allowed some freedom to move towards safer, less crowded positions. Each via point corresponds to a potential new turn in the path.

### 4.2.1 Initialization

RPS positions via points an equal distance apart along the line connecting the mobile robot and the given goal point (See Figure 2(a)). This distance ( $d_A$ ) is based upon a given minimum distance between via points, ( $d_G$ ), and is given by

$$(5) \quad d_A = D_{goal} / \text{ceiling}(D_{goal}/d_G)$$

where  $D_{goal}$  is the distance between the initial and goal configurations, *ceiling* returns an integer by rounding up, and  $\text{ceiling}(D_{goal}/d_G)$  is the number of via points.

The via point closest to the robot is specially positioned to be in line with the robot initial orientation (at angle  $\alpha$ ) at a distance of  $d_G$  because the mobile robot, in general, will not be in line with the goal point initially (See Figure 2 (b)). The last via point is the goal point itself and remains fixed throughout RPS.

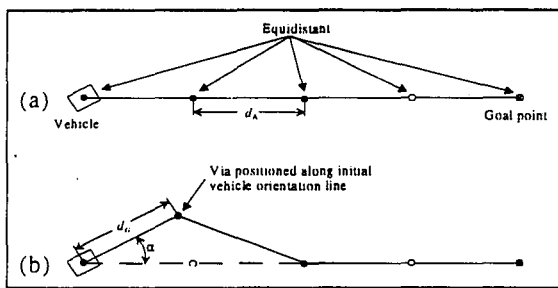


Figure 2 - Initial placement of via points.

The development of this initialization method stemmed from the aisle inspection scenario. The mobile robot starts at one end of an aisle; the goal is at the other end; and the desired motion is close to a straight line between the two points. The robot deviates from this line to avoid possible collisions with sensed obstacles by positioning via points according to the following method.

### 4.2.2 Positioning

Now that the via points have been initialized, the task at hand is to position them for collision avoidance. Obstacles shape the path by "pushing" via points away. The positioning method described can be done on-line and without a prior model of the obstacles.

Each occupied cell in the map exerts a "force" on the via points. Occupied cells that are far away are of less importance and affect the curve less than those closer but with the same occupancy value. The force between the map cell and via point varies linearly (within a range of action) with both distance and the occupancy value. It is represented by a spring of length  $L$  placed between a via point and each map cell (see Figure 3). This is a purely compressive spring so it only repels via points. Cells outside a circle of radius  $L$  from the via point do not contribute any forces to the via point.

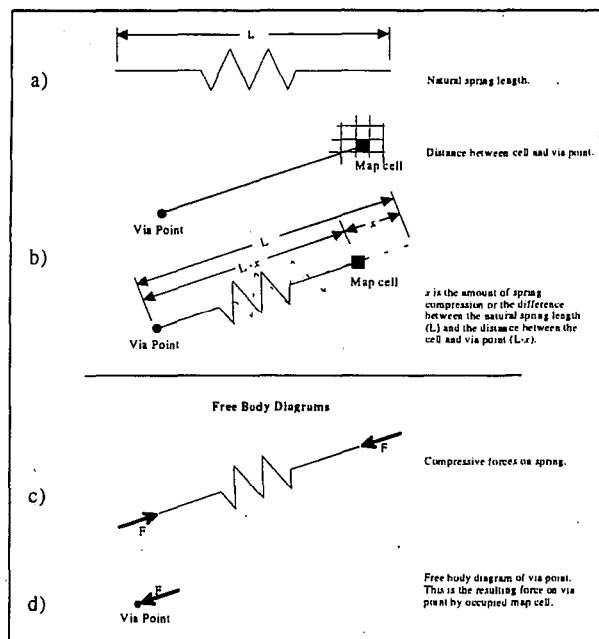


Figure 3 - Force effect of occupied cell on via point.

The forces from all map cells are vectorially summed yielding a *resultant force*,  $F$ . To minimize the chance that this resultant force pushes a via point over an adjacent via point, only a small percentage of the resultant force is applied to the via point in the direction of connecting the vehicle's starting point and the goal point,  $L_1$ . Let the angle of this line with respect to the resultant force be  $\beta$ . After a number of trials, 10 percent was found to be small enough to reduce the chance of overlap, but still allow a little freedom in the  $L_1$  direction.

The objective is to position the via point at a safe location; that is, where the forces are below a threshold magnitude,  $e$ .

This location is referred to as an *equilibrium* location. Due to the piecewise linear spring profile (a compressive spring), the equilibrium location may not lie in the direction of  $F_N$ . To find its location, a set of non-linear equations must be solved. To reduce computation time at the expense of accuracy, an incremental method is employed. The via point is constrained to move a distance equivalent to the map resolution in the direction of  $F_N$ .  $F_N$  is then recalculated and the process repeated until the threshold magnitude,  $\epsilon$ , or a local minimum is reached.

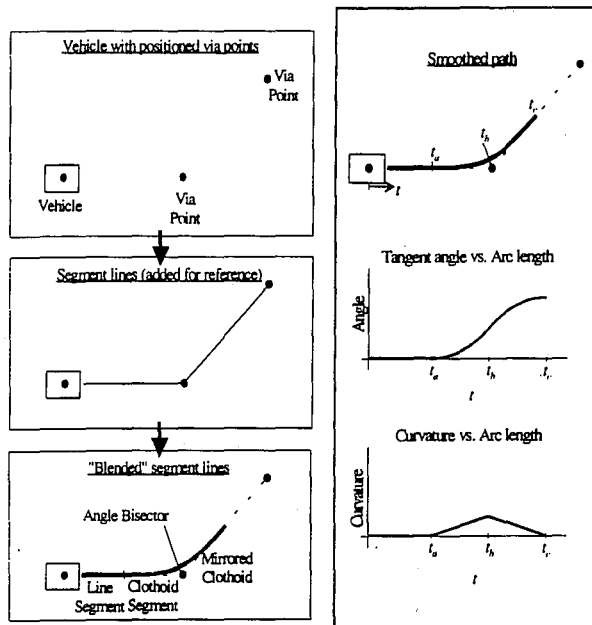


Figure 4 - Smoothing a path. Figure 5 - Path tangent angle and curvature profiles.

### 4.3 Connecting Lines with Clothoids

Now that the via points are positioned, the task becomes generating a path suitable for a nonholonomic vehicle. The method used by RPS "blends" one straight line segment connecting a pair of via points into the next by introducing an appropriate clothoid pair. A clothoid pair consists of a clothoid and its mirror image. A simple illustration is given in Figure 4. Both clothoids join the line segments with continuous tangent angle and continuous curvature. The resulting tangent angle and curvature profiles are given in Figure 5.

The following sections detail the process of generating the clothoids between line segments: determining the starting and ending points, and the scaling factor. For simplicity, the clothoid calculations are shown first for the simple case where

the vehicle initial tangent angle is zero and then generalized for any initial angle.

#### 4.3.1 Generating the Clothoid Pair

The problem is summarized as this: finding a pair of clothoids between two lines such that the clothoids join each other and the straight lines with continuous curvature. The following details how the first half of the clothoid pair is generated. The second half is the mirror image of the first about the angle bisector as shown in Figure 4. The minimum turning radius,  $\rho$ , is given. Using this radius of curvature at the angle bisector yields a curve that comes as close as possible to the desired via point. The tightest part of a turn is in the middle.

A unique clothoid is found by calculating the appropriate values for the parameters  $t_0$ ,  $D$ , and  $a$ . The parameter  $t$  (from the Fresnel equations (2) and (3)) varies from zero, at the clothoid start, to infinity. We are interested in finding  $t_0$ , the point at which the clothoid segment ends and its mirror image takes over.<sup>18</sup> This is the point at which the curvature is the largest. The curvature at  $t_0$  is set equal to  $\rho$ , the minimum turning radius.  $D$  is the distance between the via point,  $V$ , and the clothoid start. See Figure 6. Parameter  $a$  is the scaling factor. The angle subtended by the line segments,  $\alpha$ , is assumed known.

By symmetry, the tangent angle,  $\theta$ , at  $t = t_0$ , must be

$$(6) \quad \theta = \frac{1}{2}(\pi - \alpha)$$

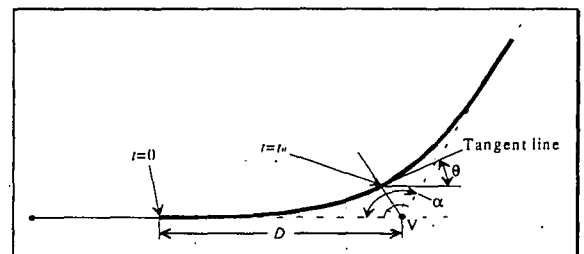


Figure 6- Clothoid parameters.

Substituting (6) into (4) for  $\theta$  yields

$$\frac{1}{2}(\pi - \alpha) = \frac{\pi}{2} t^2$$

Solving for  $t$  gives us the value of  $t$  at the desired clothoid endpoint,  $t_0$ , or

$$(7) \quad t_0 = \sqrt{1 - \frac{\alpha}{\pi}}$$

The distance,  $D$ , is

$$(8) \quad D = aC(t_0) + aS(t_0)\cot(\alpha/2)$$

where  $a$  is the scaling factor,

$$(9) \quad a = \pi \rho t_0$$

and  $C(t)$  and  $S(t)$  are the Fresnel equations.

#### 4.3.2 Generalizing: Non-Zero Initial Tangent Angles

The above equations assume a starting point at the origin and a zero tangent angle. Now generalizing equations the Fresnel equations  $x(t)=C(t)$  and  $y(t)=S(t)$  for a given initial rotation,  $\beta$ , about the x-axis and translation to  $(x_i, y_i)$  yields

$$(10) \quad x(t) = x_0 + a(C(t)\cos\beta - gS(t)\sin\beta)$$

$$(11) \quad y(t) = y_0 + a(C(t)\sin\beta + gS(t)\cos\beta)$$

where  $\beta$  is the angle between the first segment and the world coordinate system x-axis and  $g$  represents the concavity of the needed curve. If it is concave up, then set  $g = 1$ , else set it to  $-1$ . Parameter  $g$  also represents the direction of turn (i.e., left or right).

#### 4.4 A Step-by-Step Example

The following discussion takes a step-by-step look at a vehicle moving towards a goal point with RPS. For clarity only one obstacle is used. The figure symbols used in this section are outlined in Figure 7.

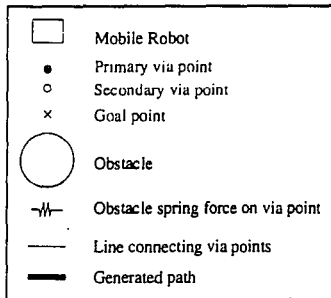


Figure 7 - Legend for symbols found in Figure 8.

The example is set up in Figure 8 part *a*. The mobile robot is given a goal point. The via points are initialized in part *b*. The first via point is positioned along the vehicle orientation and the rest are placed equidistant along the line to the goal point. Note that the first three via points are *primary* via points (PVPs) which consist of the mobile robot and the next two via points. Later, as the vehicle moves to the next via point, the set of PVPs will be indexed to the next set of three. It is always the case that the first two PVPs have been previously positioned and the third needs to be positioned.

The third PVP experiences forces from the detected object in part *c*, and is positioned at an equilibrium location in part *d*. Now that the PVPs are positioned, the path is smoothed using a line segment and clothoid pair in part *e*. In part *f*, the robot traverses the path to the end of the clothoid pair. The robot is

now tangent with the second line segment. Note that the set of PVPs is indexed to the next set of three via points.

The third PVP is positioned by the detected object in parts *g* and *h*. Due to the distance between the object and the via point, it is not positioned far from its initialized location. A smooth path is generated in part *i* that brings the robot tangent to the third line segment in part *j*. Note that the set of PVPs is indexed again and includes the goal point.

The goal point is assumed not to need repositioning, and the final path connecting the robot to the goal point is generated in *k*. Since the PVPs are nearly collinear, only a small piece of the clothoid curve is needed to appropriately smooth the path.

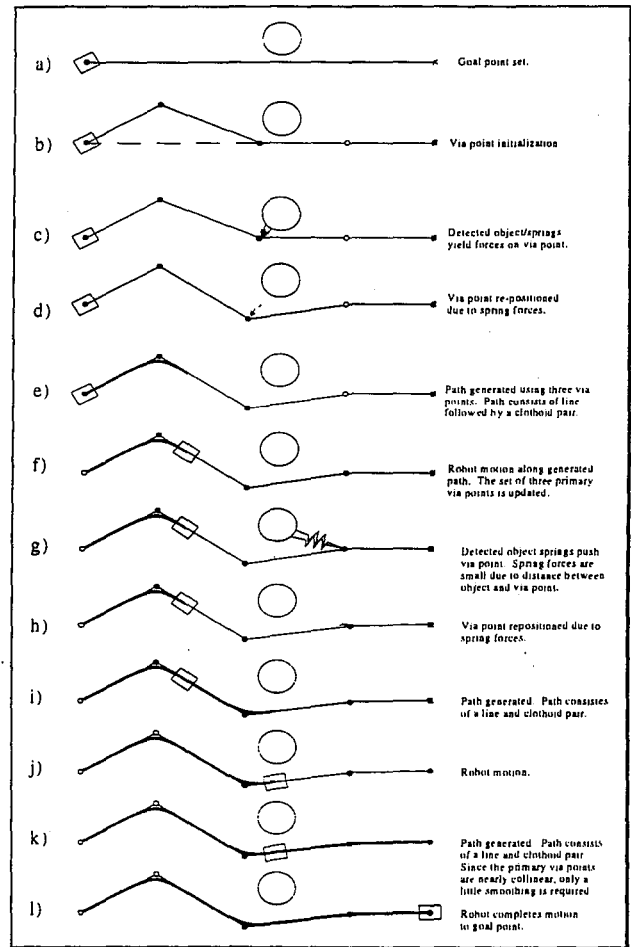


Figure 8 - Reactive Path Shaping overview. A mobile robot follows a generated path to a goal point.

## 4.5 RPS Parameters

Setting the input parameters is an important aspect of RPS. Poor parameter values may cause poor obstacle avoidance. Setting them appropriately requires an understanding of how they affect the resulting path. The three RPS parameters are the minimum turning radius, the minimum distance between via points, and the uncompressed spring length.

### 4.5.1 Minimum turning radius

This is the radius of the smallest diameter circle able to be traveled.

### 4.5.2 Minimum distance between via points

The minimum distance between via points determines how reactive RPS is to the environment. Large values yields large distances between via points and the generated path is not responsive to changes in the environment. Small values are desirable so that the vehicle is responsive to changes in the environment but may result in frequent backing up when combined with a large minimum turning radius.

### 4.5.3 Uncompressed Spring length

The uncompressed spring length determines how much the sensed environment affects the generated path. Higher values for this parameter may cause the generated path to deviate significantly from the initialized via point path, as more distant obstacles contribute forces on via points. With high values there may exist multiple local minima. Low uncompressed spring length values result in paths that deviate little from the initialized via point path and may not avoid significant obstacles. A good value for this parameter is the vehicle width plus the map resolution (cell width) plus a safety margin to account for the greater profile long vehicles have when turning.

## 5.0 Implementation

RPS has been implemented in a simulator that simulates motion of a variety of vehicles and range sensors.<sup>19</sup> The code that comprises RPS is outlined in pseudo-code in Figure 9.

## 6.0 Experiments

Numerous simulation experiments have been run with several types of aisle environments and a pair of nonholonomic vehicles. The objectives were to determine good parameter values and determine the utility of RPS. An example experiment is given in Figure 10. The figure shows the environmental map atop a view of the simulated environment. The environment consists of two long wavy but parallel lines of pallets.

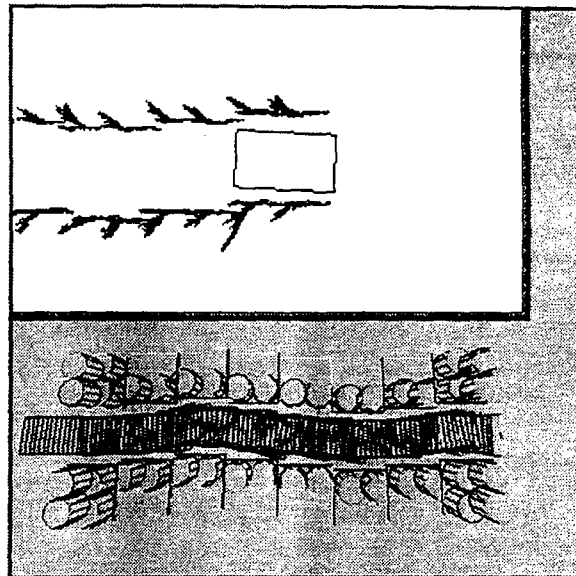


Figure 10 - Nonholonomic vehicle in aisle environment

Results have shown that the best spring lengths are slightly larger than the vehicle width but less than half the aisle width. The best lengths are those that result in a narrow "valley" field down the aisle middle. In the bottom of the valley spring forces are zero and suitable for via point equilibrium locations.

Further experiments show that small distances between via points are needed since on-board ultrasonic range sensors have low angle accuracy and provide a poor map of the environment beyond a meter.<sup>20</sup> Small distances between via points result in a more reactive path as well, responsive to changes in the environment.

```
Set up environment, vehicle, vehicle initial configuration,
and map.
Set up path planner, read parameters.
  Read goal point.
  Read minimum turning radius.
  Read minimum distance between via points.
  Read uncompressed spring length.
Initialize via points.
while(current_via_point less than number_via_points){
  Activate sensors, send range info to map.
  Position via points reactively.
  Generate line segment and clothoid pair.
  while(not at end of line-clothoid pair){
    Increment along path.
    Translate map.
    Activate sensors, send range info to map.
  }
  Go to next via point and increase current_via_point by
  1.
}
```

Figure 9 - RPS Pseudo-code

## 7.0 Conclusions

Reactive Path Shaping (RPS), a new path generator for nonholonomic mobile robots in partially known environments has been introduced.

RPS generates paths that meet the kinematic constraints of several types of nonholonomic vehicle types, the Wheel, Dual Wheel, and Tricycle. Alternative methods produce paths that may be rough or impossible to follow with real vehicle dynamics except by bringing the vehicle to a stop.

While RPS is tested in simulation only, it is felt that the design of RPS allows for real-time implementation on a mobile test vehicle. RPS is structured such that the environmental data gathering and path generation modules run in parallel, independent of each other, saving computational time.

RPS can be likened to a potential field methods in which the field "acts" upon the path instead of the vehicle. Repulsive forces act upon a part of the path ahead of the vehicle so there is time to make the path suitable for a nonholonomic vehicle.

An extension to this work could provide velocity commands based upon the path curvature such that the mobile robot slows down for sharp turns and speeds up for shallow ones. RPS should be implemented on a test vehicle with ultrasonic range sensors to confirm the simulation studies. Future work should compare RPS and other path generators for nonholonomic vehicles.

## 8.0 Acknowledgment

This research was sponsored in part by Westinghouse Savannah River Corporation through the Educational Research and Development Association (ERDA) of Georgia Universities through order number 94050.

## 9.0 References

- <sup>1</sup> Kanayama, Y. and B.I. Hartman, "Smooth Local Path Planning for Autonomous Vehicles", *IEEE International Conference on Robotics and Automation*, 1989, p. 1265-1270.
- <sup>2</sup> Borenstein, Johann and Koren, Yorem, "Real-Time Obstacle Avoidance for Fast Mobile Robots", *IEEE Transactions on Systems, Man, and Cybernetics*, 1989, vol. 19, no. 5, p. 1179-1186.
- <sup>3</sup> Arkin, R., "Motor Schema Based Navigation for a Mobile Robot: An Approach to Programming by Behavior", *Proceedings of the IEEE International Conference of Robotics and Automation*, 1987, Raleigh, NC pp. 264-271.
- <sup>4</sup> Arkin, R., "Chapter 5: Autonomous Mobile Robots", *Knowledge Engineering, Volume II Applications*, McGraw-Hill Publishing Company, pp. 116-160.
- <sup>5</sup> Borenstein, J. et al., "Mobile Robot Navigation in Narrow Aisles with Ultrasonic Sensors," *ANS 6th Topical Meeting on Robotics and Remote Systems*, Monterey, California, February 5-10, 1995.
- <sup>6</sup> J.A Reeds and R. A. Shepp, "Optimal paths for a car that goes both forwards and backwards," *Pacific J. Mathematics*, vol. 145, no. 2, 1990, pp. 367-393.
- <sup>7</sup> Laumond, Jean-Paul, "Finding Collision-Free Smooth Trajectories for a Non-Holonomic Mobile Robot," *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, August 1987, pp. 1120-1123.
- <sup>8</sup> Kanayama, Y. and N. Miyake, "Trajectory Generation for Mobile Robots", *Robotics Research: Third International Symposium.*, 8: p. 333-340.
- <sup>9</sup> Kanayama, Y. and B. I. Hartman, "Smooth Local Path Planning for Autonomous Vehicles", *IEEE International Conference on Robotics and Automation*, 1989, p. 1265-1270.
- <sup>10</sup> Murray, Richard M. *et al.*, *A Mathematical Introduction to Robotics Manipulation*, CRC Press, 1994.
- <sup>11</sup> Fleury, S et al., "Primitives for Smoothing Mobile Robot Trajectories", *IEEE International Conference on Robotics and Automation*, 1993.
- <sup>12</sup> Heald, M. A., "Rational Approximations for the Fresnel Integrals", *Mathematics of Computation*, 1985, p. 459-461.
- <sup>13</sup> Borenstein, J. and Y. Koren, "Histogrammic In-Motion Mapping for Mobile Robot Obstacle Avoidance", *IEEE Transactions on Robotics and Automation*, 1991. 7(3): p. 535-539.
- <sup>14</sup> Moravec, H. and M. Blackwell, "Learning Sensor Models for Evidence Grids".
- <sup>15</sup> Moravec, H. P. and A. Elfes. "High Resolution Maps from Wide Angle Sonar", *IEEE International Conference on Robotics and Automation*, 1985, St. Louis, Missouri.
- <sup>16</sup> Borenstein, J. and Y. Koren, "The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robots", *IEEE Transactions on Robotics and Automation*, 1991. 7(3): p. 278-288.
- <sup>17</sup> Borenstein, J. and Y. Koren, "Real-Time Obstacle Avoidance for Fast Mobile Robots", *IEEE Transactions on Systems, Man, and Cybernetics*, 1989. 19(5): p. 1179-1186.
- <sup>18</sup> Meek, D. S. and D. J. Walton, "The use of Cornu Spirals in Drawing Planar Curves of Controlled Curvature", *Computational and Applied Mathematics*, 1989. 25: p. 69-78.
- <sup>19</sup> Schmitt, P. R., et al., "Integrated Modeling, Simulation, and Animation of Rigid Arms and Vehicles through Object Oriented Programming", *MATHMOD*, 1994, Vienna, Austria.
- <sup>20</sup> Schmitt, P.R., Reactive Path Shaping: Local Path Planning for Autonomous Mobile Robots in Aisles, Thesis, Georgia Institute of Technology, submitted March, 1996.