Proceedings of the 2006 IEEE International Conference on Robotics and Automation
Orlando, Florida - May 2006

Extension of Reeds & Shepp Paths to a Robot with Front and Rear Wheel Steer

Siddharth Sanan+, Darshan Santani+, K Madhava Krishna* and Henry Hexmoor[$]

+Manipal Institute of Technology, Manipal, India. sidsanan@yahoo.com
*International Institute of Information Technology, Hyderabad, India. mkrishna@iiit.ac.in
$ CSCE Department, University of Arkansas Fayetteville, AR

Abstract - **This paper presents an algorithm for extending RS paths for a robot with both front and rear wheel steer. We call such robots as FR steer. The occurrence of such paths is due to the additional maneuver possible in such a robot which we call parallel steer, in addition to the ones already present in a vehicle with only front wheel steering. Hence we extend the optimal path set Λ, containing only a single element to a set Ω, containing n elements, thereby extending its configuration set along the optimal path from the initial to the final configuration. This extension of the set Λ to set Ω is made possible by introducing a special set, which we call the Parallel Steer (PS) Set. Such an extension of the configuration set would increase the size of the final configuration set achievable by a path that is optimal in free space. In the following discussion, we shall term all paths whose length is equal to an RS path as optimal.**

Index Terms - Reeds and Shepp (RS) Paths, Parallel steer, FR steer, motion planning

I. Introduction

The problem of motion planning constitutes the construction of a feasible path for a robot to move along, such that it achieves the final configuration from the initial configuration. The term feasible here implies that the path conforms to the kinematics of the robot at every point. Such problems are not only constrained to the domain of mobile robotics but have been an area of research since the very beginning of the Robotic Arm. One of the main features of motion planning in mobile robotics is non-holonomic motion planning [1], [2]. The essential characteristic of a nonholonomic constraint is that it reduces the space of achievable velocities for a robot to a dimension smaller than its configuration space. The essential outcome of this characteristic is that a geometrically possible path for the robot is not kinematically feasible.

In this paper, we extend the RS paths to a robot with both front and rear wheeled steer that we call as FR steer. A FR steer robot finds utility due to its smaller turning radius than a normal car-like-robot that helps in navigating through tighter spaces. This smaller turning radius is achieved by appropriate combination of front and rear steering angles. Trajectories of a FR steer robot are a combination of straight lines and circles as its kinematics reveal. Hence optimal paths for a car in the form of RS curves are also optimal trajectories for a FR steer. The FR steer robot also has an extra maneuver called the parallel steer which is achieved when the front and rear steering angles are same in magnitude and direction. This maneuver results in the extension of the optimal path set Λ containing a single element for a normal car robot to a parallel steer (PS) set containing an arbitrary number of n elements, each of them giving the same distance between a given initial and target configuration as would the original RS trajectory for a normal car. The primary advantage of this extension is that where there is an obstacle along the original RS trajectory for the normal car, one could still come up with an equivalent optimal path that avoids the obstacle but retaining the optimal distance measure. This is achieved by selecting any one of the arbitrary n elements in the extended PS set.

II. Related Works

Motion planning with non-holonomic constraints first appeared in robotic literature through the seminal work of Laumond [3]. Later Reeds and Shepp showed by geometric methods synthesis of optimal paths in their famous paper [4] for a car-like robot that can move forward and back. They illustrated that a shortest path motion could always be achieved by means of trajectories of a special kind, namely concatenation of at most five pieces each of which is a straight line or an arc and that these concatenations can be classified into 48 three parameter families. By combining techniques of optimal control such as Pontryagins Maximum Principle (PMP) and geometric methods like Lie Algebraic analysis of trajectories, Sussman and Tang recovered the RS results and improved them by lowering the 48 to 46 [5]. A similar effort was also published by Boissonat *et al.* [6] on minimum paths for an RS car. In [1], Barraquand and Latombe describe a planner based on randomized techniques for a multi-body nonholonomic robot in presence of obstacles and in [7] a motion planner for the RS car was presented. With the arrival of randomized techniques, motion planning with nonholonomic constraints has been attacked under the broader context of kinodynamic planning [8]. However randomized techniques do not generate optimal trajectories both in presence and absence of obstacles. In [11] an adaptive back stepping controller for a non holonomic robot with unknown parameters was designed and simulated.

To the best of authors' knowledge, there has not been any article related to motion planning for a robot with FR steer. In

[9], a technique for planning a velocity profile on an *a priori* given path for a four-wheel steer robot is presented. But [9] does not talk of a motion planner as such. In this regard, the authors believe that this could be one of the first attempts to present a motion planner for an FR steer robot that is optimal in the sense of RS path lengths.

III. Problem Definition

All previous work dealing with minimal length optimal path planning [3], [4], [5], [6], [10] has dealt with a car model of a mobile robot with front wheel steering only. The problem here is to find a motion planning scheme for a variation of the RS car model with both front and rear wheel steering which we refer to as the FR steer model.

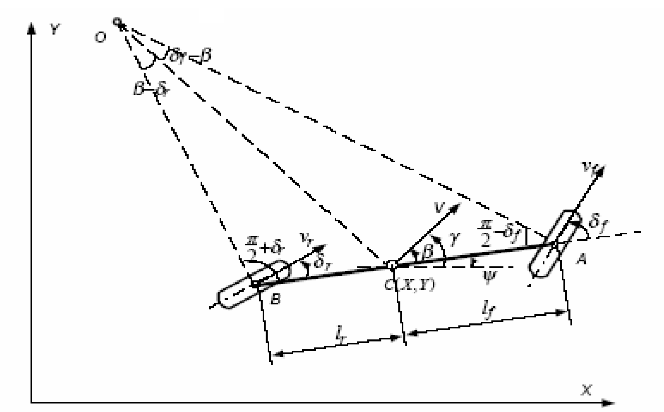

Fig. 1 Bicycle model of the FR steer car

A. *Model of the Car*

The configuration space of the robot is in $\mathbf{R}^2\times\mathbf{S}^1$ and given by the variables [X Y ψ]. The kinematic model of an FR steer robot is considered here under assumptions of non-slippage of tire, planar motion and rigid body. An FR steer car under such assumptions can be modeled as a bicycle [9] as illustrated in Fig. 1. Reference point C is chosen at the center of gravity of the body and all kinematic analysis is with reference to this point. The following parameters are needed in such a model:

- ψ $\rightarrow$ heading angle: the angle made by the longitudinal axis of the vehicle and the X-axis
- γ $\rightarrow$ course angle: the angle made by the velocity of point *C* with the X-axis
- β $\rightarrow$ side slip angle: the angle of the velocity of point C with the longitudinal axis of the car
- R $\rightarrow$ the distance between the point C and the instant centre of rotation (ICR) O
- v_f $\rightarrow$ velocity of forward wheel
- v_r $\rightarrow$ velocity of rear wheel
- δ_f $\rightarrow$ steering angle of the front wheel
- δ_r $\rightarrow$ steering angle of the rear wheel

To define the trajectory of the car at any time, we place the robot in a Cartesian plane and represent the time derivatives of its configuration parameters in terms of the above parameters illustrated in Fig. 1. The kinematics of the car based on the above parameters is defined as:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} v\cos(\psi+\beta) \\ v\sin(\psi+\beta) \\ \dfrac{v\cos\beta(\tan\delta_f - \tan\delta_r)}{l_f + l_r} \end{bmatrix} \tag{1}$$

where

$$\beta = \arctan\frac{l_f\tan\delta_r + l_r\tan\delta_f}{l_f + l_r}$$

and

$$v = \frac{v_f\cos\delta_f + v_r\cos\delta_r}{2\cos\beta}$$

In the above model, there are four control variables, namely v_f, v_r the two velocity inputs and δ_f, δ_r the two steering inputs. Integrating (1) confirms that the car traverses a circle with turning radius $\dfrac{l_f + l_r}{\cos\beta\left(\tan\delta_f - \tan\delta_r\right)}$ centered at O (Fig. 1). This radius is the distance from O to C. The parallel steer condition occurs when $\delta_f = \delta_r$ resulting in an infinite turn radius. This implies that the car translates without changing its orientation.

B. *Properties of the Car*

For implementation purpose, consider a simplified model, in our case $v_f = v_r$ always and $\delta_r = \pm\delta_f$. Due to this simplification, we can now set the reference point C on the robot at the midpoint of the longitudinal axis. The condition $\delta_r = -\delta_f$ gives the least turning radius and is always used when having to traverse a circular arc. For the parallel steer maneuver, we have $\delta_r = \delta_f$ and while moving along the regular straight line path $\delta_r = \delta_f = 0$.

We also add the following inequality constraints to the control variables δ_f,δ_f.

$$-\delta_{max} \le \delta_f \le \delta_{max} \tag{2}$$

which implies

$$-\delta_{max} \le \delta_r \le \delta_{max} \tag{3}$$

where δ_{max} is the mechanical limit of the steering system.

This constraint also causes the slip angle β to be bounded by the following condition:

$$-\delta_{max} \le \beta \le \delta_{max} \tag{4}$$

Constraints (2) and (3) impose a limitation on the lower bound of the radius of curvature *r*. We shall assume without any loss of generality for generating RS equivalent paths that

$$r_{min} = \pm 1 \qquad (5)$$

(+1 when the robot goes forward left or backward right). This assumption is routinely used [4], [5].

With the model completely defined, we now look at the problem of finding minimal length optimal paths for such a robot between an initial configuration [0 0 0] to a final configuration of [X Y ψ].

V. Optimal Paths

In [4], the optimal paths for a car like robot are found to consist of i) straight lines ii) circular arcs of radius of curvature r_{min}. They proved that the optimal path belongs to a minimal and sufficient set consisting of 48 paths or 9 path families. These are given below:

$$\text{C|C|C, CC|C, CSC, CC}_u\text{|C}_u\text{C, C|C}_u\text{C}_u\text{|C, C|C}_{(\pi/2)}\text{SC,}$$

$$\text{C|C}_{(\pi/2)}\text{SC}_{(\pi/2)}\text{|C, C|CC, CSC}_{(\pi/2)}\text{|C} \qquad (a)$$

Here C indicates motion along a circular arc, S indicates motion along a straight line, | indicates a cusp point or a point where the direction of movement is reversed and the subscript indicates the length of segment on the particular arc or straight line.

We use the modified notations of Sussman and Tang given in [5] to represent a particular path. l^+ indicates a left turn in the forward direction, r^+ indicates a right turn forward, the letter l indicates a steer which causes an increase in the state space variable ψ and the letter r indicates a steer causing a reduction in the same, the superscript + or – indicates the direction i.e. forward or backward. On substituting each of the above 9 families with the letters defined above, a total of 48 paths are obtained. We divide our path planning procedure into two phases. Phase I generates standard RS paths between the initial configuration and final configuration. Phase II searches for paths where multiple optimal solutions can exist and generates these alternate paths.

A. The Parallel Steer Maneuver

In the simplified model of the robot considered, we trace the mid-point of the longitudinal axis.

The state space of the FR steer robot consists of three variables X, Y and ψ. A path along an arc of a circle causes a change in all the three state space variables while a path along a straight line causes a change in only two Cartesian space variables X and Y. For a front wheel steer robot, the direction of the straight line coincides with the orientation ψ of the robot. However as seen in Fig. 2, the robot can move in a direction with an angular offset of $\pm\beta$ from the orientation ψ of its longitudinal axis. Such a maneuver is known as parallel steer shown in Fig. 2 and Fig. 3.

In Fig. 2, the origin of the vector indicates the robots position [X Y] and the direction of the vector indicates the configuration variable ψ. Such a maneuver introduces what we call the Parallel Steer (PS) set which we shall discuss in the following section. In Fig. 3, it can be seen that in such a maneuver $\delta_f = \delta_r = \beta$ and we call this angle δ and β interchangeably dropping subscript *f* and *r* from δ.

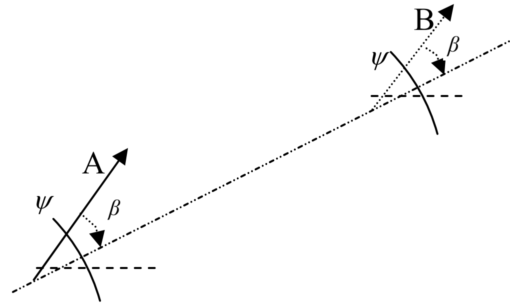

Fig. 2 Parallel steer at β slip

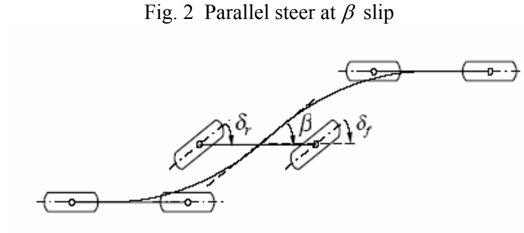

Fig. 3 The parallel steer maneuver

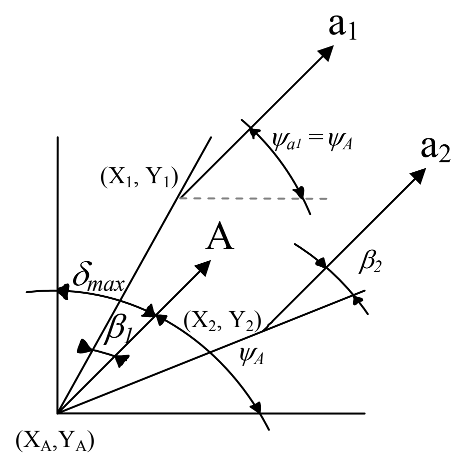

Fig. 4 Elements of a PS set for a configuration

B. The Parallel Steer Set

In the RS paths, the configuration set which is the set of all configurations achieved by the robot in moving from the initial A to the final configuration B contains elements, which define the path of the robot. Let us name this set Λ_{AB}. The possible paths are listed in (a). The Parallel Steer Set (€) is generated at the end points of the path i.e. between A and B by constructing paths from A and B by using the PS maneuver only. We shall prove later that the PS sets thus generated are sufficient and contain elements that would yield a path that is equivalent in length to the RS path. Each element of the PS set is defined by:

$$€_A = \{X, Y, \psi \mid X = X_A + K\cos\beta,\ Y = Y_A + K\sin\beta,\ \psi = \psi_A\} \qquad (6)$$

Here K is a parameter representing the distance traversed on the PS steer maneuver and β satisfies (4). The physical description of these elements is given in Fig. 4 where a_1 and a_2 are two elements of the PS set at A ($€_A$).

Without any loss of generality, we can limit the construction at B only. The configurations of the robot along these paths constitute the elements of the PS set as defined in (6). Our objective is to find paths that have a nonempty PS set whose lengths are equivalent to RS paths.

At this stage, we can remodel our problem to i) finding a pair of elements (a,b) where a is an element of the PS set at A (ϵ_A) and b is an element of the PS set at B (ϵ_B), ii) finding an RS path between these two elements formed from the configuration set Λ_{ab}, and iii) connecting a, A and b, B with PS maneuver such that the length of the path found is equivalent to the length of the optimal RS path between the two configurations.

Hence, the total configuration set in any path of the robot may be defined as

$$\Omega \equiv (\epsilon_A \cup \epsilon_B) \cup \Lambda_{ab} \quad (7)$$

for optimal set Ω_o

$$\Omega_o \equiv (\epsilon_{Ao} \cup \epsilon_{Bo}) \cup \Lambda_{abo} \quad (8)$$

such that

$$\text{length}\{\Omega_o\} = \text{length}\{\Lambda_{AB}) \quad (9)$$

Within the PS set, every path is a straight line with $v = \pm 1$ and β obeys (4). Here the speed v is taken as ±1 without any loss of generality.

C. Phase I

We now proceed to develop a methodology to generate RS paths between any two configurations through the method described in [10]. We develop an efficient algorithm to reduce the search among the families given in (a). We achieve this by developing a domain plane that contains the boundaries for all path families listed in (a) for every final orientation ψ_f using [10] considering the initial orientation to be $\psi = 0$. This is obtained by applying the backward and time-flip transforms to all the paths in the domain map of [10] since in [10], the initial configuration is set as [X Y ψ] and the final configuration is set as [0 0 0] while in our model, the robot moves from [0 0 0] to final configuration [X Y ψ]. For a further understanding of these transforms, the reader is referred to [10].

Once the domain map is built, it is seen that all paths have mutually exclusive regions of optimality. The algorithm, thus developed, yields an optimal RS path between any two given configurations. The graphical simulation of one such path is shown in Fig. 5.

D. Phase II

Referring to Fig. 6, A is the initial configuration of the robot, B is the final configuration of the robot and C is an intermediate configuration of the robot. An RS path in the above situation constitutes movement of the robot along circle C1, then catching the transverse tangent to circle C2 and traversing the shortest arc on C2 to reach configuration B.

Such a path would occur in the following path families:

1. CSC, 2. $C|C_{\pi/2}SC_{\pi/2}|C$, 3. $C|C_{\pi/2}SC$, 4. $CSC_{\pi/2}|C$ (b)

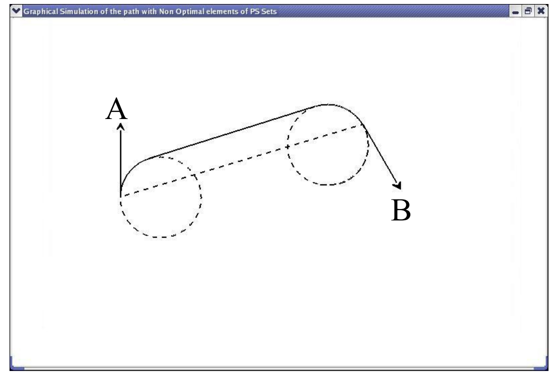

Fig. 5 Simulation of RS Paths

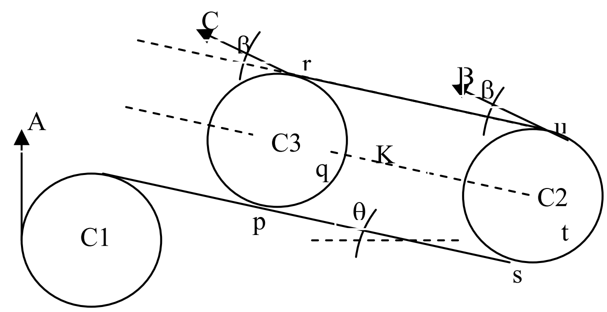

Fig. 6 Path with PS set element at B

In a front-only steered robot, one such path would be the unique-distance-minimal path. However in an FR steer robot, where the parallel steer maneuver is possible, an intermediate configuration C which is an element of the PS set ϵ_B is obtainable which is connected by a standard RS path with the initial configuration (Fig. 6). The configuration C and B where C,B $\in \epsilon_B$ are finally connected through the parallel steer maneuver by steering to a slip angle δ such that the movement of the robot is along a line that is at an inclination β. It is required to find C to satisfy (4) and (9), hence we have the condition

$$\psi + \beta = \theta$$

$$\Rightarrow \quad \beta = \theta - \psi \quad (10)$$

It can be seen by simple geometry that arc(p q r) on C3 = arc(s t u) on C2 and ru = ps. Hence the length of the path connecting the configurations A-C-B would be equal in length to the path connecting the configurations A-B within a given range of K, K being the length of parallel steer. It is obvious that for equivalent paths, K < length of tangent from circle C1 to C2.

Of the four path families mentioned in (b), it should be observed that path family 2 will never generate a path with parallel steer maneuver for a robot with a finite turn circle radius. In path families 1, 3 and 4, the path between the CSC letters would be checked for a possible parallel steer path that exists within the given bounds of δ and β ($|\beta|$ is always lesser than 90°). A path would also be searched for in the backward and time-flipped transforms of each path i.e. by generating an optimal PS set ϵ_A and yielding a configuration element C which has orientation parallel to A. If either of the searches yields a parallel steer path, a value of K within the permissible range would be set and a Parallel Steer path returned to the motion planner.

Any value of K within the permissible range can be taken and an equivalent path found.

VI. Simulation Results

A graphical simulation of the actual path generated by the optimal configuration set Ω_o is shown in the Fig. 7 and computational results are illustrated in Table I.

TABLE I
Lengths of Paths with PS Sets for Various Values of K, Ψ
K = 0 corresponds to RS path and
K>0 corresponds to equivalent RS paths obtained by PS maneuver
β = 153.17° β = 152.54°

K \ Ψ	30º	–30º
	Path : $r^+s^+l^+$	Path : $r^+s^+r^+$
	Path Length	
0	21.627403	21.613615
1	21.627403	21.613617
2	21.627403	21.613617
3	21.627405	21.613617
4	21.627403	21.613617

In Fig. 7, the path connecting the configurations A-B by an $r^+s^+r^+$ path is the RS path while the path connecting the configurations A-B with an intermediate configuration C is a path with the PS maneuver equivalent in length to the RS path. The different circles shown in Fig. 7 and Fig. 8 correspond to different values of K which is the length of the PS maneuver.

E. Non optimal elements in PS sets

The elements in the PS sets which do not satisfy (10), if selected, generate nonoptimal paths i.e. paths which are not equal in length to RS paths. This fact was first realized by computational means using the algorithm developed. The computational results for various values of δ are shown in Table II.

We have seen through extensive computational simulations these elements belonging to PS set do not give RS equivalent paths but paths that are longer. Based on computational simulations and resulting data we find (9) is violated for a path for which (10) is not satisfied and the paths are longer.

In (b) and (d), we stated that the PS sets shall be generated at the end configurations A and B only and these sets shall be sufficient to generate a path with the RS path length however these sets are not exhaustive with respect to forming paths of length equal to RS paths. We do not attempt

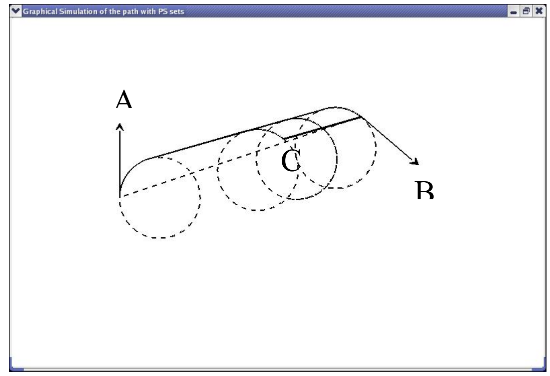

Fig. 7 Simulation of Path with optimal elements of PS set

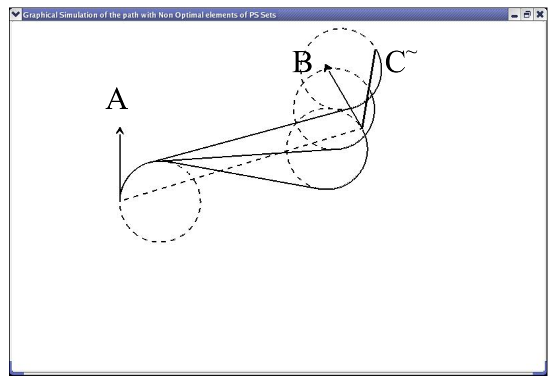

Fig. 8 Simulation of Paths with non-optimal elements of PS set

TABLE II
Lengths of Paths with nonoptimal elements of PS set
K = 0 corresponds to RS path and
K>0 corresponds to paths obtained by PS maneuver
β = 30°

K \ Ψ	30º	60º
	Path : $r^+s^+r^+$	Path : $r^+s^+r^+$
	Path Length	
0	21.627403	21.810835
1	23.526035	23.807905
2	25.433754	25.805252
3	27.349457	27.802841
4	29.272200	29.800638

to prove this fact, but the case of PS sets at any configuration along the arcs of a CSC path satisfying (9) may be sited as an instance of this fact.

A consequence of parallel steer is that it generates RS path length even in the presence of obstacles. Such an instance is shown in Fig. 9. In figure 9 an RS path is constructed between two configurations A and B along circles C1 and C2. Due to the presence of an obstacle, (shaded rectangle) this path is not permissible. By using the method described in the previous sections, a path is constructed using the optimal elements of the PS set at B. This path consists of a forward right turn on C1 (r^+), a forward path connecting the circles C1 and C3 (s^+), a forward left turn on C3 (l^+) and then a PS maneuver with β satisfying (6) and $v = -1$. This path is topologically permissible. The length of this path is equal to the original RS path. This is a consequence of the PS steer maneuver.

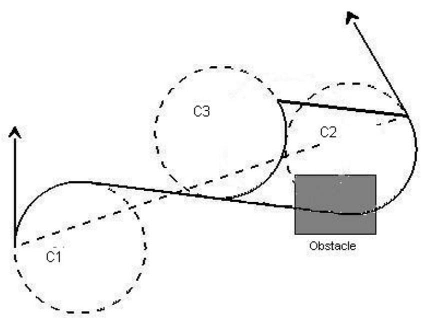

Fig. 9 Construction of paths with optimal elements of PS set in an environment with obstacles

The Path planner was made for our FR Steer Robot that we call FRBot. The FRBot robot has a unique 4-wheel Steering system (double Ackermann) and has a four wheel independent suspension. The frame replicates the characteristic of Monster trucks built in the U.S.A. It is built to move on multiple terrain types from smooth surfaces to rough terrain with high frequency surface variations.

The Bot is being used as a research platform/tested at our lab. It was developed in-house at our university through off-the-shelf components. It has provisions for integrating sonar and CCR cameras and the API for these have been developed. Several clients can connect to it wirelessly and the low level control is run on-board a Motorola's DSP based processor.

The algorithm was coded in C language using OpenGL (Mesa 3D Library) on the Linux platform (Fedora Core 3.0).

VII. Conclusions

We find that paths which are equivalent in length to RS paths in free space can be constructed by using the optimal elements of the PS sets. Such paths lead to the expansion of the configuration set for obtaining a path which is equal in length to an RS path in free space. An interesting consequence of the extra degree of freedom obtained through reverse steering is that a path equivalent to RS can still be found while the original path is not collision free. This interesting property can be made use in motion planning in presence of obstacles while retaining the path length of RS.

References

[1] J. Barraquand and J. C. Latombe, "Nonholonomic multibody mobile robots: Controllability and motion planning in presence of obstacles," *Algorithmica*, vol. 10, pp 121-155, 1993.

[2] "Robot Motion Planning and Control," J. P. Laumond, editor, LAAS-CNRS, Toulouse, France, August 1997.

[3] J. P. Laumond, "Feasible trajectories for mobile robots with kinematic and environmental constraints," in *Intelligent Automation Systems*, F. C. A. Groen, ed: North Holland, 1987.

[4] J. A. Reeds and L. A. Shepp, "Optimal paths for a car that goes both forwards and backwards," Pacific Journal of Mathematics, vol. 145, no 2. pp 367-393, 1990.

[5] Héctor J. Sussman and Guoqing Tang, "Shortest paths for the Reeds-Shepp car: a worked out example of the use of geometric techniques in nonlinear optimal control," *Technical Report SYCON-91-10*, Department of Mathematics, Rutgers University, 1991.

[6] J. D. Boissonnat, A. Cerezo, and J. Leblond, "Shortest paths of bounded curvature in the plane," *IEEE International Conference on Robotics and Automation,* Nice, France, 1992.

[7] J. P. Laumond, et al., "A nonholonomic motion planner for mobile robots," *IEEE Transactions on Robot. and Automat.,* 10(5): 577-593.

[8] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," *Proc. IEEE Int'l Conf. on Robotics and Automation*, 473-479, 1999.

[9] Danwei Wang and Feng Oi, "Trajectory planning for a four-wheel-steering vehicle," *Proceedings of 2001 IEEE International Conference on Robotics and Automation*, May 2001, Korea, pp. 3320-3325.

[10] Philippe Souéres and Jean-Paul Laumond. "Shortest Path Synthesis for a Car-Like Robot," *IEEE Transactions on Automatic Control,* 41(5): 672–688, May 1996.

[11] T Fukao, H Nakagawa and N Adachi, "Adaptive Tracking Control of a Non Holonomic Mobile Robot", *IEEE Transactions on Robotics and Automation*, 16(5):609-615