Proceedings of the ASME 2010 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2010
August 15-18, 2010, Montreal, Quebec, Canada

DETC2010-28987

FINDING THE SHORTEST PATH FOR A MOBILE ROBOT WITH TWO ACTUATED SPOKE WHEELS BASED ON VARIABLE KINEMATIC CONFIGURATIONS

Ya Wang
RoMeLa: Robotics & Mechanisms Laboratory
Mechanical Engineering Department
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061
Email: ywang07@vt.edu

Dennis Hong[1]
RoMeLa: Robotics & Mechanisms Laboratory
Mechanical Engineering Department
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061
Email: dhong@vt.edu

ABSTRACT

Strategies for finding the shortest path for a mobile robot with two actuated spoke wheels based on variable kinematic configurations are presented in this paper. The optimal path planning strategy proposed here integrate the traditional constrained path planning tools and the unique kinematic configuration spaces of the mobile robot IMPASS (Intelligent Mobility Platform with Actuated Spoke System). IMPASS utilizes a unique mobility concept of stretching in or out individually actuated spokes in order to perform variable curvature radius steering using changing kinematic configuration during its movement. Due to this unique motion strategy, various kinematic topologies produce specific motion characteristics in the way of curvature radius-variable steering. Instead of traditional differential drive or Ackerman steering locomotion, combinational path geometry methods, *Dubins' curve* and *Reeds and Shepp's* curve are applied to classify optimal paths into known permutations of sequences consisting of various kinematic configurations. Numerical simulation is given to verify the analytical solutions provided by using Lagrange Multiplier.

1 INTRODUCTION

IMPASS is a novel leg-wheel robot developed at the Robotics and Mechanisms Lab (RoMeLa) at Virginia Tech [1-3]. The prototypes have been built to effectively demonstrate the actuated spoke wheel concept integrating into the leg-wheel robot. The latest IMPASS prototype is shown in Figure 1. Two rimless spoke wheels, mounted on a common axle, each contain three individually actuated spokes. These spokes each move through the hub, totaling in six feet on each hub. These six feet are situated evenly around the hub, with a common angle of 60 degrees between each spoke. Due to stretching in or out independently actuated spokes while it is walking, this highly mobile robot proves very valuable in application of traversing complex terrain in an intelligent manner by picking up the different kinematic topologies. This unique topology changing platform combines the efficiency of a wheeled robot and the mobility of a legged robot so that IMPASS is much more adaptable to wide range of unstructured ground environments than the wheeled robots and faster than the legged robots on smooth surfaces.

Figure 1. The latest IMPASS Prototype.

The changing topology of IMPASS produces twenty different mechanism cases during IMPASS's moving locomotion, classified using the numbers and sequences of the spokes touching the ground[4].As the topological configuration changes, the mobility and kinematic characteristics will vary from case to case because of the non-slip constraints at the spoke foot [4-6]. The kinematic analysis of gait and gait

1. Address all correspondence to this author.

transition for IMPASS is presented in [6], which lays the foundation for the later on path planning studies.

Due to the unique mobility platform, IMPASS performs the curvature variable steering in the use of kinematic topology switch by stretching in or out various spokes, rather than employ either the popular wheeled robot steering locomotion, Differential drive or Ackerman Steering, or a legged robot walking locomotion. The steering motion characteristics, with regard to steering curvature radius and steering angle, are discussed in [7] for typical kinematic topologies of IMPASS.

This paper studies the path planning strategies based on kinematic topologies for IMPASS, which is in general, a sequence of kinematic topology to be taken, although it could be more complicated. In robotics, path planning was initially referred to as the Piano Mover's Problem ---- how to move a piano from one place to another in a house without hitting anything. In artificial intelligence, planning was originally concerned with a search for a sequence of logical operators or actions that transform an initial world posture into a destination posture [8-9]. Even though each is originally considered as different problems, the fields of robotics and artificial intelligence have expanded their scope to share an interesting common ground. In this paper, the term path planning encompasses this common ground in a broad sense, which is defined as follows: "Given a robot and a description of an environment, plan a shortest path between two specific locations with orientations." Figure 2 illustrates the path planning problem statement for IMPASS from the initial posture (X_0, Y_0, σ_0) to the destination posture (X_N, Y_N, σ_N). The kinematic configurations are selected in an optimized way that tries to make this happen. The path planning strategies here ignore dynamics and other differential constraints and primarily focus on the translations and rotations required for IMPASS to produce curvature radius steering or straight line walking in an optimal way that respects the mechanical and kinematics limitations of the robot.

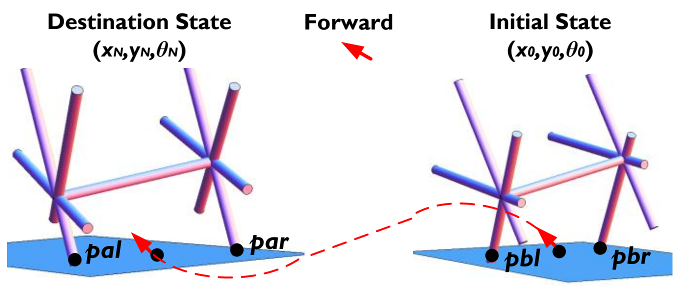

Figure 2. Path Planning Problem Definition for IMPASS.

However, regarding to the shortest path finding between the initial and destination postures, IMPASS cannot follow the *Dubin's* car or other wheel robot model, since IMPASS robot cannot change its turning curvature continuously and smoothly. Instead of that, IMPASS performs discrete steering by carrying on certain kinematic configuration transition for metamorphic topology transformation, so that the turning curvature of IMPASS has a much wider range but this makes its path discrete and discontinuously. Each kinematic configuration actually carries on certain motion mission, which will be discussed more in section IV and V.

The remainder of this paper is organized as follows: The preliminaries of path planning with term definition are stated in section II. Section III presents the state of art of the wheeled robot locomotion. In section IV, the general kinematic configurations are briefly concluded case by case due to the changing topology mechanisms of the spoke wheel robot IMPASS. In section V, the motion characteristics of each kinematic topology case are discussed, and the general sequence-evolution expression of the foot position with orientation at each time-step is formulated in the use of the shifting angle and the step length of each stride in the forward direction, which satisfies all the kinematic configurations. In section VI, the shortest path finding strategies based on *Dubin's* law are proposed and stated. Numerical simulation is given to verify the analytical solutions provided by using Lagrange Multiplier. In the end, conclusions and future work are discussed in section VII.

2 PRELIMINARIES

The path planning problem spans a broad subject field in robotics. However, there are several basic terminologies that arise throughout all of the topics covered as part of path planning.

2.1 Posture:

The posture, in this paper, represents the position and orientation of the robot. Path planning problems here involve a state space that captures all possible situations that could arise.

2.2 Kinematic Configurations:

Kinematic Configuration (K.C.) describes the specific mechanism topology at the particular moment. Kinematic Topology Case (K.T.C) is classified by how many left/right spokes touching the ground at certain moving direction[4]. The same K.T.C.s could have two different K.C.s due to the contrary moving directions: forward or backward. K.C. indicates the specific configuration, which leads to the certain motion-circular turning or straight line walking, which will be concluded in detail in section IV .

2.3 A Criteria:

This encodes the desired outcome of a plan in terms of the posture and K.Cs that are executed:

1) Feasibility: find a plan that could reach the destination posture, regardless of its efficiency.

2) Optimality: find an optimal plan in some carefully specified performance, in addition to arriving in a destination posture.

The majority of literature in robotics and related fields focuses on optimality. In this paper, the shortest path finding is the primary focus.

2.4 A Plan:

In general, a plan imposes a specific strategy or behavior on a decision maker. In this paper, a plan simply specifies a sequence of K.C.s to be taken, although it could be more complicated.

2.5 A Gait:

A gait is characterized as the sequence of lift and release events of the individual legs.

3 OVERVIEW IF THE WHEELED ROBOT STEERING LOCOMOTION

There are various mechanisms existing for the wheeled robot locomotion, and a number of these are discussed in [10-11], involving differential drive, steering wheels (tricycle, bicycles, wagon) and Ackerman steering. However, no matter which mechanisms they are using, the wheels exploit friction or ground contact to enable the robot to move. This makes difference from IMPASS, which is considered performing discrete walking locomotion. But it shares some common steering concepts with differential drive and Ackerman steering.

3.1 Differential Drive:

Differential drive is perhaps one of the easiest and the most popular drive mechanism, which consists of two rigidly mounted motors/servos with wheels attached. There is usually the roller-ball for balance and stability. It has two motors and drive wheels mounted co-linearly with the center of the base. It can turn about its center and move forward and backward. While we can vary the velocity of each wheel, for the robot to perform rolling motion, the robot must rotate about a point that lies along their common left and right wheel axis. The point that the robot rotates about is known as the *ICC*- Instantaneous Center of Curvature, as shown in Figure 3 and Figure 4.

The velocity difference of the two wheels varies the trajectories that the robot takes. Because the rate of rotation about the *ICC* must be the same for both wheels, the following equations yield:

$$R = \frac{l}{2}\frac{V_r + V_l}{V_r - V_l}, \omega = \frac{V_r - V_l}{l}. \quad (1)$$

where l is the distance between the centers of the two wheels, V_r, V_l are the right and left wheel velocities along the ground, and R is the instantaneous curvature radius of the robot trajectory (distance from *ICC* to the midpoint between the two wheels). From eq. (1), we know that:

1) When $V_r = V_l$, ω is zero, and R becomes infinite, the robot will move in a straight line;
2) When $V_r = -V_l$, R becomes zero, the robot will rotate about the midpoint of the wheel axis;
3) When V_r or $V_l = 0$, then the wheel robot will rotate about the right/left wheel.

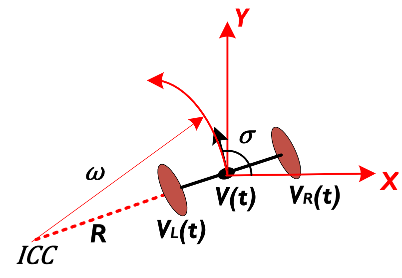

Figure 3. Kinematics for Differential Drive.

We assume that the posture of the wheel robot is *P(X, Y,σ)*, which indicates the position *(X, Y)*, and the heading direction with the *X* axis σ , as shown in Figure 4. Then the location of ICC would be: $ICC(x - R\sin(\sigma), y + R\cos(\sigma))$. Here the control input would be (v, ω), where *v* is the linear velocity of the robot, and *w* is corresponding to the angular velocity of the robot (notice: not for each wheel).

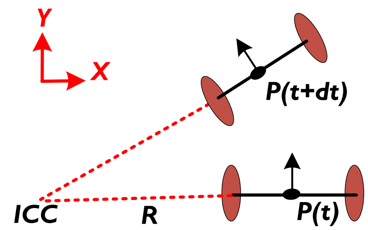

Figure 4. Differential Drive Forward Kinematics.

3.2 Ackerman Steering:

Ackerman steering, also known as car driving, is the type of steering found on most automobiles. One set of wheels, usually the drive wheels, are fixed while the other set pivots to steer the robot. The robot can have several wheels, but mush always have a single *ICC*, where all of its zero motion lines must meet, as shown in Figure 5.

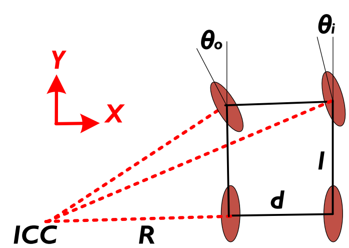

Fig. 5. Kinematics for Ackerman Steering.

Under Ackerman steering, the vehicle rotates about a point lying on the line passing through the rear axle a distance R from the centerline of the wheel robot. At the same time, in order for the wheels to exhibit rolling motion, the other steering wheel must be rotated through an angleθ_0, so that the kinematics geometry gives us the following equation:

$$cot\,\theta_i - cot\,\theta_0 = \frac{d}{l}. \qquad (2)$$

where d and l are the longitudinal wheel distance between the centers of the two wheels, θ_i, θ_o are the relative steering angles of the inside and outside wheels. R is the instantaneous curvature radius of the robot trajectory (distance from *ICC* to the midpoint between the two wheels).

4 GENERAL KINEMATIC TOPOLOGY CASES

Historically, the changing topology analysis has been conducted using the terminology or concept of Mechanism with Variable Topologies (MVT) [12-14], Metamorphic Mechanism [15-16], or Kinematotropic Mechanism (KM) [17]. In our previous paper [7], the classification of general K.T.C.s has been presented in detail. However, the concept is simply restated here for the convenient understanding. Three basic motion types are produced for the path planning performance using three K.C. types, see in Figure 6. For instance, the positive Y axis indicates the forward direction of IMPASS, the left side K.T.C. produces left steering turn, while the same K.T.C. on the right side produces the right steering turn. Both of these kinematic configurations belong to the same 1-1 Parallel Unequal Case, but the effective spoke length ratio of the left and right spokes determines the direction of steering turn. The top K.T.C in Figure 6, leads to the straight line walking motion, since the effective spoke length of both sides of spokes touching the ground is equal. Here we define the effective spoke length as the distance from the ground contact point *(A*, *B* or *C* in Figure 6*)* to the respective hub center of each wheel.

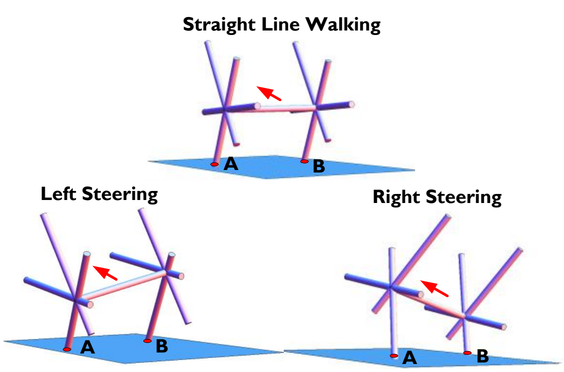

Figure 6. K.T.C.s for IMPASS's Walking and Steering Locomotion.

Figure 7 shows the transition K.C.s of IMPASS. Any transitions between any K.C.s of Figure 6 have to go through the transition K.C.s in Figure 7 to complete the motion transition between left steering turn, right steering turn and straight line walking.

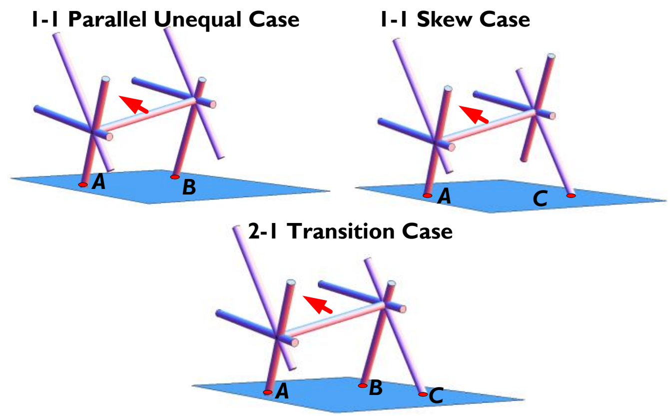

Figure 7. K.T.C.s for IMPASS's Transition Locomotion.

The K.C. describes the numbers and the sequence of the spokes touching the ground at each specific time moment [7]. Each specific topological configuration is named by *n1-n2* case (*n1*, *n2*=0, 1, 2, or 3).

Table I. Nomenclature of Metamorphic Configuration

Symbol	Quantity
L	Length of the IMPASS axle
${L_{Ri-1}}^i, {L_{Ri}}^i$	Right Effective Spoke Length during Step *i**
${L_{Li-1}}^i, {L_{Li}}^i$	Left Effective Spoke Length during Step *i **
k_i	Respective parallel left/right effective spoke length ratio during Step *i*
R_{Ri}	Turning radius from the turning center to the right spoke foot at Gait *i*
R_{Li}	Turning radius from the turning center to the left spoke foot Gait *i*
${R_i}^i$	Curvature radius from the turning center to the midpoint of the left & right spoke feet during Step *i*
φ_i	Turning angle about the turning center during Step *i*
ψ_i	Heading angle during Step *i*
L	Length of the IMPASS axle
${L_{Ri-1}}^i, {L_{Ri}}^i$	Right Effective Spoke Length during Step *i**
${L_{Li-1}}^i, {L_{Li}}^i$	Left Effective Spoke Length during Step *i **
k_i	Respective parallel left/right effective spoke length ratio during Step *i*
R_{Ri}	Turning radius from the turning center to the right spoke foot at Gait *i*
R_{Li}	Turning radius from the turning center to the left spoke foot Gait *i*

*Step *i*: the step from Gait *i-1* to Gait *i*

With respect to the forward direction of IMPASS, *n1* denotes the number of right wheel spokes, which are touching the ground, while *n2* stands for the number of the left wheel spokes touching the ground. Each K.C. indicates the specific mechanism configuration at each time moment, which produces the specific motion—such as left or right steering turn with different radius curvature or straight line walking with variable stride. The walking direction of IMPASS— right/ left turn or straight line walking is decided by the respective parallel right / left effective spoke length ratio k_i (in Table 1): k_i >1 indicates Left Steering Turn; k_i <1is corresponding to Right Steering Turn; and k_i =1refers to Straight Line Walking.

5 MOTION CHARACTERISTICS FOR EACH KINEMATIC CONFIGURATION

Figure 6 and Figure 7 illustrate the bidirectional transformation relationships between different kinematic configurations, which produce various motion types. This section will focus on the motion characteristics of each K.C.
First of all, the right / left effective spoke length ratio k_i is an important parameter, which could be preset before the robot takes any action. Again, the change of the k_i value produces the changing of steering curvature radius or the steering /heading direction of the IMPASS robot. This changing process of k_i value requires the robot to go through 2-1 transition case and 1-1 skew case, as shown in Figure 7.

5.1 1-1 Parallel Unequal Steering Case:

The topology in this case can produce the left/right steering motion with/without changing the curvature radius. As shown in Figure 8, the right and left turning radius is calculated by:

$$R_{Ri} = \frac{1}{1-1/k_i}\sqrt{L^2 + (L_{Ri} - L_{Li})^2},$$

$$R_{Li} = \frac{1}{1-k_i}\sqrt{L^2 + (L_{Ri} - L_{Li})^2}. \quad (3)$$

where, R_{Ri}/R_{Li} indicates the distance from the turning center *Oi* to the right/left spoke foot touching the ground at Gait *i*, as shown in Table 1. Positive means the spoke foot locates at the first/fourth quadrant, while negative represents the location at the second/third quadrant. L is the axel length, and $L_{Ri}/\,L_{Li}$ is corresponding to the right/left effective spoke length at Gait *i*. Here, we define the curvature radius R_i as the distance from the turning center O_i to the midpoint of the right, left spoke foot:

$$R_i = \frac{k_i+1}{2(k_i-1)}\sqrt{L^2 + (L_{Ri} - L_{Li})^2}. \quad (4)$$

Obviously, $k_i > 1$ demonstrates left turn, whereas $k_i < 1$ means right turn. For$k_i = 1$, it belongs to the 1-1 Parallel Equal Case, which will be introduced in the next subsection.

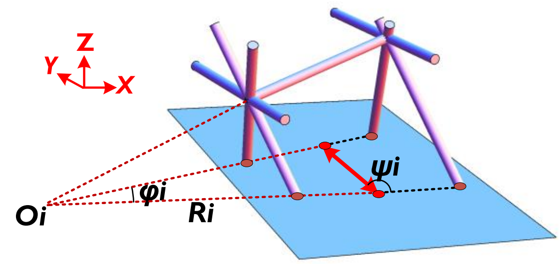

Figure 8. Kinematic Topology of 1-1 Parallel Unequal Case during Step *i*.

Considering the constraint condition of k_i, we rewrite eq. (4) the curvature radius during step *i* as:

$$R_i^{\,i} = \frac{k_i+1}{2(1-k_i)}\sqrt{L^2 + (k_i-1)^2 {L_{Li}^{\,i}}^2}. \quad (5)$$

We have to notice that, Step *i* represents the gait transition from Gait *i* -1 to Gait *i*. Then during Step *i+1*, L_{Li} has different value from step *i*, that is $L_{Li}^{\,i} \neq L_{Li}^{\,i+1}$, which caused the discrete changing of curvature radius from $R_i^{\,i}$ to $R_i^{\,i+1}$. Here, $R_i^{\,i}$ and $R_i^{\,i+1}$ denote the discrete changing curvature radius at step *i* and step *i+1*.

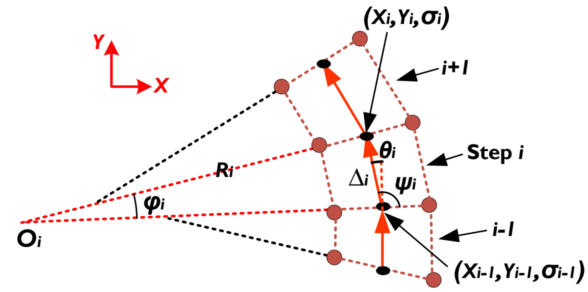

Figure 9. Motion Characteristics of 1-1 Parallel Unequal Case.

The *turning angle* φ_i is the anti-clockwise rotating angle about the turning center O_i starting from the first pivot line (the line passing through the first pair of spoke feet touching the ground) during Step *i* (in Figure 8and Figure 9), which is obtained by:

$$\varphi_i = ArcCos\frac{2L^2+(k_i-1)^2 L_{Li-1}^{\,i} L_{Li}^{\,i}}{2\sqrt{[L^2+(k_i-1)^2 {L_{Li-1}^{\,i}}^2][L^2+(k_i-1)^2 {L_{Li}^{\,i}}^2]}}. \quad (6)$$

The *heading angle* ψ_i is the step orientation angle in anti-clockwise direction departing from the first spoke feet pivot line during step *i* (in Figure 8 and Figure 9), which is expressed by:

$$\psi_i = ArcCos\frac{(1-k_i)L_{Li-1}^{\,i}(2L_{Li-1}^{\,i} - L_{Li}^{\,i})}{2\sqrt{[L^2+(k_i-1)^2 {L_{Li-1}^{\,i}}^2][{L_{Li-1}^{\,i}}^2 - L_{Li-1}^{\,i} L_{Li}^{\,i} + {L_{Li}^{\,i}}^2]}}. \quad (7)$$

The *step length* Δi (in Figure 8 and Figure 9) is the stride during step *i* which yields:

$$\Delta_i = \frac{k_i+1}{2}\sqrt{{L_{Li-1}}^{i^2} - {L_{Li-1}}^{i}{L_{Li}}^{i} + {L_{Li}}^{i^2}}. \qquad (8)$$

In fact, the position and orientation of the robot at each time-step (X_i, Y_i, σ_i) at the Global Coordinate can be formulated as discrete optimization problems using the characteristic parameters the $shifting\ angle\ \ \theta_i$, and the $step\ length\ \Delta i$, as shown in Figure 9. This general sequence-evolution formulation is deducted using transformation matrix:

$$\begin{cases} X_i = X_{i-1} - \Delta i Cos(\sigma_i) \\ Y_i = Y_{i-1} + \Delta i Sin(\sigma_i)\ . \\ \sigma_i = \sigma_{i-1} + \theta_i \end{cases} \qquad (9)$$

where, the $shifting\ angle\ \ \theta_i$is defined as the shifting angle from the line vector of stride *i-1* to the line vector of stride *i:*

$$\theta_i = \varphi_{i-1} + \psi_i - \psi_{i-1}. \qquad (10)$$

From the general sequence-evolution formulation, we know that the position and orientation of the robot at each time-step (X_i, Y_i, σ_i) is only the function of the $shifting\ angle\ \ \theta_i$ and the $step\ length\ \Delta i$.

From the eq. (6-8) and (10), we can obtain that, for 1-1 Parallel Unequal Case:

$$\begin{aligned} \theta_i &= f({L_{Li-1}}^{i}, {L_{Li}}^{i}) \\ \Delta i &= f({L_{Li-1}}^{i}, {L_{Li}}^{i}). \end{aligned} \qquad (11)$$

Again, as indicated in Figure 6, the topology of 1-1 Parallel Unequal Case produces curvature radius-variable left turn or right turn, which depends on the right / left effective spoke length ratiok_i. When k_i >1, left turn will be generated, as shown in Figure 9. If k_i <1, right turn will be produced. When k_i = 1, the robot will perform straight line walking. That is, the walking direction of the robot is controlled by the changing of the k_i value, which leads to the mechanisms with variable topology of IMPASS. This process of changing topology (changing k_i value) will not be completed until the robot carries out the 2-1 Transition Case and 1-1 Skew Case in Figure 7. This will be discussed in detail in the next subsection.

5.2 2-1 Transition Case and 1-1 Skew Case:

Imagine that, in Figure 8, at Gait *i*, if the left spoke touches the ground first, instead of both of the right and left spokes touching the ground at the same time, 2-1 Transition case will be generated. And then, if we lift the right spoke at Gait *i-1*, 1-1 Skew Case will be produced. Both of these two transition cases are brought out during Step *i,* which is illustrated in Figure 10. This figure illustrated the switch from left steering to right steering of the robot, that is the changing from k_i *>1* to k_i *<1.*

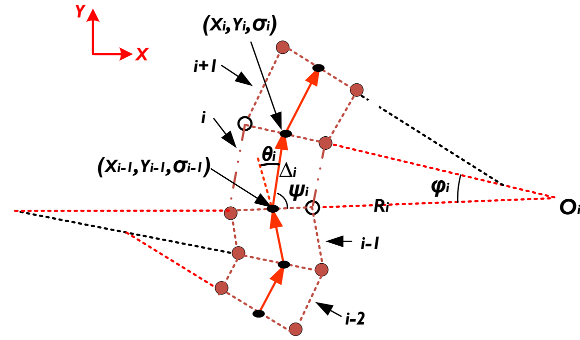

Figure 10. Motion Characteristics of 2-1 Transition Case and 1-1 Skew Case.

For the transition topology, the position and orientation of the robot during step *i*, (X_i, Y_i, σ_i) also follows the general sequence-evolution formulation. However, the characteristic parameters will have different expression from eq. (6-8). The equation is not given here for concision. Please contact the author if you want more details. But they are all the function of the effective spoke length of the spokes actually touching the ground (1-1 Skew Case), and the right / left effective spoke length ratiok_i:

$$\begin{aligned} \theta_i &= f\left(k_i, {L_{L_{i-1}}}^{i}, {L_{Ri}}^{i}\right) \\ \Delta i &= f\left(k_i, {L_{L_{i-1}}}^{i}, {L_{Ri}}^{i}\right). \end{aligned} \qquad (12)$$

5.3 1-1 Parallel Equal Case:

When k_i is changed to be 1, the topology of this 1-1 Parallel Equal Case will conduct straight line walking motion. Figure 11 illustrates the motion characteristics transactions from 1-1 Parallel Unequal Case to 2-1 Transition case and 1-1 Skew Case to 1-1 Parallel Equal Case. The motion characteristics of 1-1 Parallel Equal case will actually follow all the equations from (6-8), but much easier, since $k_i = 1$:

$$\begin{aligned} \theta_i &= 0°, \\ \Delta i &= \sqrt{{L_{Li-1}}^{i^2} - {L_{Li-1}}^{i}{L_{Li}}^{i} + {L_{Li}}^{i^2}}. \end{aligned} \qquad (13)$$

The position and orientation of the robot at each time-step *i* (X_i, Y_i, σ_i) also follows the general sequence-evolution formulation.

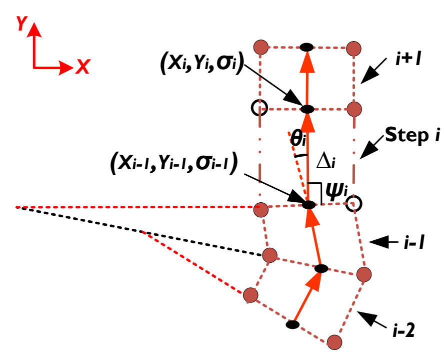

Figure 11. Motion Characteristics of 1-1 Parallel Equal Case.

6 PATH PLANING STRATEGIES

A plan simply specifies a sequence of K.C.s o be taken, although it could be more complicated. In order to solve path planning problem, two sub problems must be solved: the motion characteristics of the K.C.s, and the planning strategies. The planning strategies are defined as ill-posed problems because they do not have a unique solution, that is, there is an unlimited number of paths linking the initial and destination postures.

The traditional tools to solve the constrained path-planning problems are primary the combinational methods based on path geometry. Finding the shortest path under the curvature constraint was first brought out by Dubins [18], where he characterized such paths in two dimensions in the absence of obstacles. The well-known *Dubins' car* refers to a vehicle with a minimum turning radius that only moves forward. Reeds and Shepp [19] then introduced its variant (still in the absence of obstacles), the *Reeds and Shepp's car*.

6.1 Dubins Curves:

Let a *Dubins' car-like* robot pursue a continuously differentiable path from an initial position with an orientation to a terminal position with orientation, and the shortest path can always be expressed as a combination of no more than three motion primitives [8, 18, 20]

1) An arc, followed by a line and then an arc.
2) A sequence of three arcs of circles.
3) A sub path of category 1 or 2.

Assume that the *S* primitive drives the robot straight ahead, the *L* and *R* primitives turn as sharply as possible to the left and right, respectively. Using these symbols, each possible kind of shortest path can be designated as a sequence of three symbols that corresponds to the order in which the primitives are applied. There is no need to have two consecutive primitives of the same kind because they can be merged into one. Under this observation, fifteen types of optimal path are possible. Category 1 includes the paths of (*RSR, RSL, LSR,* and *LSL).* Category 2 contains *(RLR, LRL),* and those in Category 3 as *(R, S, L, RS, RL, SR, SL, LR, LS)*. According to Fig.6, unlike the car-like robot, *R, S, L* segments are all performed by choosing different kinematic configurations of IMPASS based on metamorphic configurations.

6.2 Reeds and Shepp's Curves:

Compared to *Dubins' car*, the only difference in comparison to the Dubins car is that moving backward is now allowed. Both *Dubins' car* and *Reeds and Shepp's car* have optimal paths that could be classified into known permutations of a sequence consisting of bang-bang controls-moving straight (forward or/and backward), turning fully right or fully left).

6.3 Problem Formulation:

We have explained earlier, for the purpose of this work, our algorithm computes the optimal trajectory by minimizing an objective function, which represents the shortest path to reach the destination. This shortest path is commonly used in path planning and robotics to find an optimal trajectory of the robot to the destination, given motion and geometry constraints due to the physical construction.

The problem solved in this work can be summarized as follows:

Problem Statement:
Given the following:
Initial posture (X_0, Y_0,σ_0); destination posture (X_N,Y_N,σ_N).

The cost function:

$$Min\, f(\Delta_i) = \sum_{i=1}^{N} \Delta_i + \sum_{i=1}^{N} \frac{1}{(\Delta_i - a)(b - \Delta_i)} + \sum_{i=1}^{N} \frac{1}{(\Theta_k - \alpha)(\beta - \Theta_k)}$$

The constraint conditions:

$$g_1(\Delta_i, \Theta_i) = \sigma_N - \sigma_0 - \Theta_N = 0$$

$$g_2(\Delta_i, \Theta_i) = X_N - X_0 + \sum_{i=1}^{N} \Delta_i \; Cos(\Theta_i) = 0$$

$$g_3(\Delta_i, \Theta_i) = Y_N - Y_0 - \sum_{i=1}^{N} \Delta_i \; Sin(\Theta_i) = 0$$

$$g_4(\Delta_i) = \Delta_i - a \le 0$$

$$g_5(\Delta_i) = b - \Delta_i \le 0$$

$$g_6(\Theta_i) = \Theta_i - \alpha \le 0$$

$$g_7(\Theta_i) = \beta - \Theta_i \le 0$$

Find the shortest path between the initial and destination position with orientation.

Currently, we consider only static case. We are looking for the optimal input sequence Δ_i and θ_i of the IMPASS robot under given constraints described above to minimize the cost function. The variable Δ_i represents the collection of input sequences Δ_i for all steps with *i Є [1, N],* and so does Θ_i.

We can compactly represent the cost function and constraint function by writing the *Lagrangian*:

$$\Lambda(\Delta_i, \Theta_i, \lambda_j) = f(\Delta_i) + \sum_{j=1}^{3} \lambda_j \; g_j(\Delta_i, \Theta_i) = 0 \tag{14}$$

And the points we want are located where:

$$\nabla\Lambda(\Delta_i, \Theta_i, \lambda_j) = \nabla \sum_{i=1}^{N} \Delta_i + \nabla \sum_{i=1}^{N} \frac{1}{(\Delta_i - a)(b - \Delta_i)} + \nabla \sum_{i=1}^{N} \frac{1}{(\Theta_k - \alpha)(\beta - \Theta_k)} + \nabla \sum_{j=1}^{3} \lambda_j \; g_j(\Delta_i, \Theta_i) = 0 \tag{15}$$

Equation (15) can be rewritten as

$$\nabla \sum_{i=1}^{N} \Delta_i + \nabla \frac{1}{\mathrm{b-a}} \sum_{i=1}^{N} \frac{1}{(\Delta_i - a)} + \frac{1}{(b-\Delta_i)} + \nabla \frac{1}{\beta-\alpha} \sum_{i=1}^{N} \frac{1}{(\beta-\Theta_k)} + \frac{1}{(\Theta_k-\alpha)} + \nabla \sum_{j=1}^{3} \lambda_j \, g_j(\Delta_i, \theta_i) = 0 \tag{16}$$

We know that:

$$\frac{\partial \Lambda}{\partial \Delta_N} = 1 + \lambda_2 \, Cos(\Theta_N) - \lambda_3 \, Sin(\Theta_N) + \frac{1}{b-a}[\frac{1}{(b-\Delta_N)^2} - \frac{1}{(\Delta_N - a)^2}] = 0$$

$$\frac{\partial \Lambda}{\partial \Theta_N} = -\lambda_1 - \lambda_2 \Delta_N Sin(\Theta_N) - \lambda_3 \Delta_N \, Cos(\Theta_N) + \frac{1}{\beta-\alpha}\left[\frac{1}{(\beta-\Theta_N)^2} - \frac{1}{(\Theta_N-\alpha)^2}\right] = 0 \tag{17}$$

And we get:

$$\Theta_N = \sigma_N - \sigma_0$$

when $k \neq N$:

$$\frac{\partial \Lambda}{\partial \Delta_k} = 1 + \lambda_2 \, Cos(\Theta_k) - \lambda_3 \, Sin(\Theta_k) + \frac{1}{b-a}[\frac{1}{(b-\Delta_k)^2} - \frac{1}{(\Delta_k - a)^2}] = 0$$

$$\frac{\partial \Lambda}{\partial \Theta_k} = -\lambda_2 \Delta_k Sin(\Theta_k) - \lambda_3 \Delta_k \, Cos(\Theta_k) + \frac{1}{\beta-\alpha}\left[\frac{1}{(\beta-\Theta_k)^2} - \frac{1}{(\Theta_k-\alpha)^2}\right] = 0 \tag{18}$$

So the solution would be:

$$\Theta_k = \pm ArcCos\left[-\frac{\lambda_2 + \sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}{\lambda_2{}^2+\lambda_3{}^2}\right]$$

$$\Delta_k = \frac{\lambda_1\lambda_3}{\sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}$$

Or

$$\Theta_k = \pm ArcCos\left[\frac{-\lambda_2 + \sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}{\lambda_2{}^2+\lambda_3{}^2}\right]$$

$$\Delta_k = -\frac{\lambda_1\lambda_3}{\sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}$$

when $k = N$:

$$\frac{\partial \Lambda}{\partial \Delta_k} = 1 + \lambda_2 \, Cos(\Theta_N) - \lambda_3 \, Sin(\Theta_N) = 0$$

$$\frac{\partial \Lambda}{\partial \Theta_k} = -\lambda_1 - \lambda_2 \Delta_N Sin(\Theta_N) - \lambda_3 \Delta_N \, Cos(\Theta_N) = 0 \tag{19}$$

Solution:

$$\Theta_N = \pm ArcCos\left[-\frac{\lambda_2 + \sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}{\lambda_2{}^2+\lambda_3{}^2}\right]$$

$$\Delta_N = \frac{\lambda_1\lambda_3}{\sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}$$

Or

$$\Theta_N = \pm ArcCos\left[\frac{-\lambda_2 + \sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}{\lambda_2{}^2+\lambda_3{}^2}\right]$$

$$\Delta_N = -\frac{\lambda_1\lambda_3}{\sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}$$

When $k = N-1$:

$$\frac{\partial \Lambda}{\partial \Delta_k} = 1 + \lambda_2 \, Cos\left(\sum_{j=0}^{N-1} \theta_j\right) - \lambda_3 \, Sin\left(\sum_{j=0}^{N-1} \theta_j\right) = 0$$

$$\frac{\partial \Lambda}{\partial \theta_k} = -\lambda_1 - \lambda_2 \Delta_N Sin(\Theta_N) - \lambda_3 \Delta_N \, Cos(\Theta_N) - \lambda_2 \Delta_{N-1} Sin(\Theta_{N-1}) - \lambda_3 \Delta_{N-1} Cos(\Theta_{N-1}) = -\lambda_2 \Delta_{N-1} Sin(\Theta_{N-1}) - \lambda_3 \Delta_{N-1} Cos(\Theta_{N-1}) = 0 \tag{20}$$

Solution:

$$\Theta_{N-1} = \pm ArcCos\left[\frac{-\lambda_2}{\lambda_2{}^2+\lambda_3{}^2}\right] = ArcSin\left[\frac{\lambda_3}{\lambda_2{}^2+\lambda_3{}^2}\right]$$

$$\Delta_{N-1} = \forall$$

Or

$$\Theta_{N-1} = \pm ArcCos\left[\frac{\pm\lambda_2 + \sqrt{\lambda_3{}^2(-1+\lambda_2{}^2+\lambda_3{}^2)}}{\lambda_2{}^2+\lambda_3{}^2}\right]$$

$$\Delta_{N-1} = 0$$

When $k < N-1$: same solution as above.

7 SIMULATION

The algorithm used to compute shortest trajectories combines numerical methods based on MATLAB. The outline of the algorithm is presented as follows:

Initialization

Set initial posture (X_0, Y_0, σ_0)*, destination posture* (X_N, Y_N, σ_N)*.*

Define constraint of the stride Δ_i *and the switching angle* θ_i*.*

Main Loop

If new posture is available, reiterate until the robot has reached their destination posture.

As we did for the Lagrangian and its gradient, we provide the MATLAB function *fmincon* with the constraints of the robot as well as their gradient with respect to the control variable Δ_i and θ_i. Providing the gradient of the constraints to the function *fmincon* leads to a more accurate solution, than allowing *fmincon* to compute its own numerical approximation to the gradient of the constraints. Figure 11 shows the shortest path generated from the initial position $(0, 0, \frac{2\pi}{3})$ to the destination position $(25, 20, -\frac{\pi}{3})$.

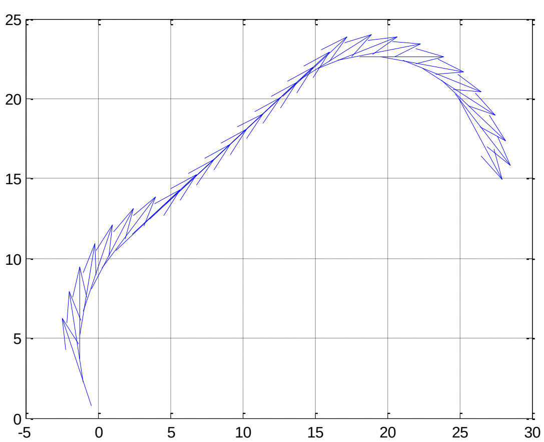

Fig. 12. Shortest Path Finding using Matlab

8 CONCLUSIONS

The shortest path finding strategies based on variable kinematics configurations are presented in this paper for a novel mobile robot that uses two actuated spoke wheels. The optimal planning strategies are proposed integrating the traditional constrained path planning tools and the unique kinematic configuration spaces of the mobile robot IMPASS. Due to this unique motion strategy, various kinematic topologies produce specific motion characteristics in the way of curvature variable steering. Instead of traditional differential drive or Ackerman steering locomotion, combinational path geometry methods, Dubins' curve as well as Reeds and Shepp's curve are applied to classify optimal paths into known permutations of sequences consisting of various kinematic configurations. Numerical simulation is given in the end to verify the analytical solutions provided by using Lagrange Multiplier.

9 ACKNOWLEGMENTS

This work was supported by the National Science Foundation under Grant No. 0535012.

10 REFERENCES

[1] Laney, D. and Hong, D.W.,"Kinematic Analysis of a Novel Rimless Wheel with Independently Actuated Spokes", 29th ASME Mechanisms and Robotics Conference, Long Beach, California, September 24-28, 2005.

[2] Hong. D.W. and Laney, D., "Preliminary Design and Kinematic Analysis of a Mobility Platform with Two Actuated Spoke Wheels", US-Korea Conference on Science, Technology and Entrepreneurship (UKC 2006), Mechanical Engineering & Robotics Symposium, Teaneck, New Jersey, August 10-13, 2006.

[3] Laney, D. and Hong, D.W., "Three-Dimensional Kinematic Analysis of the Actuated Spoke Wheel Robot". 30th ASME Mechanisms and Robotics Conference, Philadelphia, Pennsylvania, September 10-13, 2006.

[4] Y. Wang, P. Ren, D. Hong, "Mobility and Geometrical Analysis of a Two Actuated Spoke Wheel Robot Modeled as a Mechanism with Variable Topology," 32nd ASME Mechanisms and Robotics Conference, August 3-6, 2008, Brooklyn, New York, USA.

[5] P. Ren, Y. Wang, D. Hong, "Three-dimensional Kinematic Analysis of a Two Actuated Spoke Wheel Robot Based on its Equivalency to A Serial Manipulator," 32nd ASME Mechanisms and Robotics Conference, August 3-6, 2008, Brooklyn, New York, USA.

[6] Y. Wang, *et al.*, "Gait and Gait Transition for a Robot with Two Actuated Spoke Wheels," presented at the 33rd ASME Mechanisms and Robotics Conference, San Diego, California, USA, 2009.

[7] Y. Wang and D. Hong, "Motion Characteristics for the Kinematics Topologies of the IMPASS Robot," presented at the the 6th International Conference on Ubiquitous Robots and Ambient Intelligence, Gwangju, Korea, 2009.

[8] [8] S. M. LaValle, *Planning algorithms*. Cambridge ; New York: Cambridge University Press, 2006.

[9] [9] Y. K. H. a. N. Ahuja, "Gross motion planning – A survey," *ACM Compt. Surveys,* vol. 24, No.3, pp. 219-291, September 1992.

[10] [10] R. Siegwart and I. R. Nourbakhsh, *Introduction to autonomous mobile robots*. Cambridge, Mass.: MIT Press, 2004.

[11] [11] G. Dudek and M. Jenkin, *Computational principles of mobile robotics*. New York: Cambridge University Press, 2000.
[12] [12] H.-S. Yan and C.-H. Kuo, "Topological representations and characteristics of variable kinematic joints," *Journal of Mechanical Design, Transactions of the ASME,* vol. 128, pp. 384-391, 2006.
[13] H.-S. Yan and C.-H. Kuo, "Representations and identifications of structural and motion state characteristics of mechanisms with variable topologies," *Transactions of the Canadian Society for Mechanical Engineering,* vol. 30, pp. 19-40, 2006.
[14] H.-S. Yan and N.-T. Liu, "Joint-codes representations for mechanisms and chains with variable topologies," *Transactions of the Canadian Society for Mechanical Engineering,* vol. 27, pp. 131-143, 2003.
[15] J. S. Dai and J. R. Jones, "Kinematics and mobility analysis of carton folds in packing manipulation based on the mechanism equivalent," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science,* vol. 216, pp. 959-970, 2002.
[16] J. S. Dai and J. Rees Jones, "Mobility in metamorphic mechanisms of foldable/erectable kinds," *Journal of Mechanical Design, Transactions of the ASME,* vol. 121, pp. 375-382, 1999.
[17] K. K. Wohlhart, "Kinematotropic Linkages," in *Recent Advances in Robot Kinematics*, ed: Kluwer Academic Publishers, 1996, pp. 359–368.
[18] L. E. Dubins, "On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents," *American Journal of Mathematics,* vol. 79, p. 20, July 1957.
[19] J. A. Reeds and L. A. Shepp, "Optimal Paths for a Car That Goes Both Forwards and Backwards," *Pacific Journal of Mathematics,* vol. 145, pp. 367-393, Oct 1990.
[20] J. C. B. Richard H. Bsrtels, and Brian A. Barsky, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling*: Morgan Kaufmann Publishers Inc, 1987.