Steering of a Robotic Snake

K.Sarrigeorgides and K.J.Kyriakopoulos
Control Systems and Intelligent Automation Laboratory
Department of Mechanical Engineering
National Technical University of Athens, Athens 10682
Greece
e-mail: kkyria@central.ntua.gr

Abstract

In this paper we develop a methodology for steering on the plane the wheeled "snake" robot designed by Migadis and Kyriakopoulos [MK97]. This mechanical system is subject to nonholonomic constraints. The kinematic model of the mobile robot is derived taking into consideration these constraints. The nonholonomic motion planning is solved by converting the multiple input system to a multiple-chain, single generator chained form via state feedback and a coordinate transformation. Simulation results are provided for a test case.

1 Introduction

The mechanical design of the "snake" robot (fig. 1) was motivated by the increasing need for robotic devices that are able to crawl into places too dangerous or too small for human beings. For example, such kind of devices can be used for inspection tasks inside a nuclear reactor, for fire-fighting reconnaissance, maneuvering through the rubble to look for survivors after an earthquake, investigation or minor repair of leaks or other problems in municipal sewer systems or utility tunnels etc. In [YY93] the design and motion planning of a mechanical snake is presented. In [CJ95] the kinematics of hyperredudant robot locomotion over uneven solid terrain are considered.

Our"snake" robot is depicted in Figure 1. The basic structure of its body consists of three repeated modules with four active joints on each module. Further details about the mechanical structure can be found in [MK97]. The mobile robot is designed to perform multiple tasks. A kind of clutch is used in order to convert an active joint on a module to a passive one and conversely. The passive joint can be used, for example, in motion on a plane. The active can be activated in order to overcome an obtacle or in the case of locomotion on an uneven terrain. Each module has an active joint having the ability to rotate the wheels so that the snake can change the motion plane.

planning for several car-like mobile robots has been extensively investigated. In [RS93], the example of a car pulling n trailers with 2 inputs is considered. In these developments, the control system is converted into a chained form for which the solution to the steering problem is straightforward. In [Sor93] the kinematic model is locally converted into, a nilpotent, chained form. In [DTS95] the machinery of exterior differential forms is developed in order to convert the 2 input n trailer system into the Goursat normal form which is dual to the chained form. These results were extended to systems with three inputs by [BTS93b] and to systems with multiple inputs [DTS94].

The examples used though in those developments consider axle-to-axle hitching between the trailers. Our robot can be considered as a 2 trailer system with 2 kingpin hitches, where the axles are connected by a kingpin between the bodies. This kind of system cannot be converted into chained form [Bus95] due to the fact that sush a system is not flat [Fli93]. We overcome this problem by adding 2 more controls in the 2 trailers.

In section 2 we state the control system associated with our model. While the derivation of the modeling equations of the motion on a plane is presented in Section 3. In Section 4, we convert our model to the extended Goursat normal form. A method of steering the system is presented in section 5, while simulation results are presented in section 6. Finally in Section 7, we discuss issues of further research. In the appendix we provide the reader with the neccesary background on exterior differential forms.

2 Mathematical problem statement

We are interested in steering a mechanical system with the nonholonomic constraints

$$\omega_i(x) \cdot \dot{x} = 0 \qquad i = 1, 2, \dots k \qquad (1)$$

where $x \in \Re^n$ is the state of the system and $\omega_i(x) \in \Re^n$ are row vectors. We assume that vec-

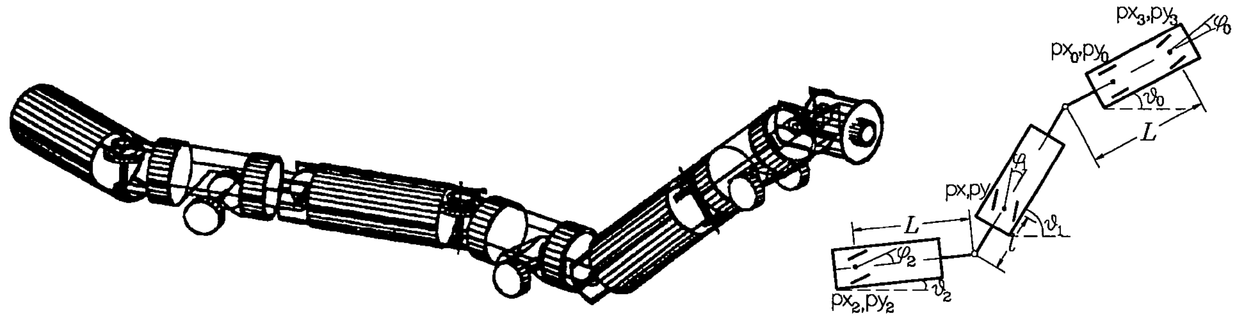

Figure 1: a) the "snake" robot b) model of the "snake" robot

can find an $(n-k)$-dimensional distribution $\Delta(x) = \text{span}\{g_1(x), \dots, g_{n-k}(x)\}$ with $g_i(x) \in \Re^n$, $i = 1, \dots, n-k$ such that $\Delta = \Omega^{\perp}$ i.e. $\omega_i(x) g_j(x) = 0 \; \forall \omega_i \in \Omega,\; g_j \in \Delta$. Then the system with the above velocity constraints is represented as a system with inputs u_i

$$\dot{x} = g_1(x)u_1(t) + \dots + g_{n-k}(x)u_{n-k}(t) \tag{2}$$

The motion planning problem can be stated as: find a control law $u = (u_1(t), \dots, u_{n-k}(t))$ to steer the system from the initial state $x(0) = x_0$ to $x(T) = x_f$ on the time interval $[0, T]$. We want to convert the system into multi-chain single-generator form since there are methods for steering such systems.

3 Modeling Equations

We want to derive the kinematic model of our system. The states of the system is $x = [p_x \; p_y \; \theta_0 \; \theta_1 \; \theta_2 \; \phi_0 \; \phi_1 \; \phi_2]^T$ where : θ_i is the orientation of the i-th module of the robot with respect to the horizontal axis of the inertial frame, ϕ_i is the steering angle of the i-th module, (p_x, p_y) are the cartesian coordinates of the rear axle of the second module.

The distance between the axle and joint of each module is L and the distance between the joint of one module and the axle of the next module is l. This gives the holonomic constraints:

$$p_{x_0} = p_x + L\cos\theta_1 + l\cos\theta_0 \tag{3}$$

$$p_{y_0} = p_y + L\sin\theta_1 + l\sin\theta_0 \tag{4}$$

$$p_{x_3} = p_x + L\cos\theta_1 + L\cos\theta_0 \tag{5}$$

$$p_{y_3} = p_y + L\sin\theta_1 + L\sin\theta_0 \tag{6}$$

$$p_{x_2} = p_x - l\cos\theta_1 - L\cos\theta_2 \tag{7}$$

$$p_{y_2} = p_y - l\sin\theta_1 - L\sin\theta_2 \tag{8}$$

In order to simplify our kinematic model, each pair of wheels is modeled as a single wheel centered at the midpoint of the axle. In fact the two wheels have different angles and their normals intersect at a single point [AM89]. The nonslipping requirement of the wheels gives the four linear velocity constraints :

$$\dot{p}_x\sin\theta_0 - \dot{p}_y\cos\theta_0 - L\cos(\theta_1-\theta_0)\dot{\theta}_1 - l\dot{\theta}_0 = 0$$

$$\dot{p}_x\sin(\phi_1+\theta_1) - \dot{p}_y\cos(\phi_1+\theta_1) = 0$$

$$\begin{aligned}&\dot{p}_x\sin(\phi_2+\theta_2) - \dot{p}_y\cos(\phi_2+\theta_2)\\ &\quad + l\dot{\theta}_1\cos(\phi_2+\theta_2-\theta_1) + L\dot{\theta}_2\cos\phi_2 = 0\end{aligned}$$

The constraints can be written in the (1) if

$$\begin{aligned}
\omega_1(x) = {} & [\sin(\phi_0+\theta_0) \quad -\cos(\phi_0+\theta_0) \quad -L\cos\phi_0 \\
& -L\cos(\phi_0+\theta_0-\theta_1) \quad 0 \quad 0 \quad 0 \quad 0] \\
\omega_2(x) = {} & [\sin\theta_0 \quad -\cos\theta_0 \quad -l \quad -L\cos(\theta_1-\theta_0) \\
& 0 \quad 0 \quad 0 \quad 0] \\
\omega_3(x) = {} & [\sin(\phi_1+\theta_1) \quad -\cos(\phi_1+\theta_1) \quad 0 \quad 0 \quad 0 \\
& 0 \quad 0 \quad 0] \\
\omega_4(x) = {} & [\sin(\phi_2+\theta_2) \quad -\cos(\phi_2+\theta_2) \quad 0 \\
& l\cos(\phi_2+\theta_2-\theta_1) \quad L\cos\phi_2 \quad 0 \quad 0 \quad 0]
\end{aligned}$$

Since the corresponding codistribution $\Omega(x)$ is 4-dimensional and the state space is 8-dimensional, we can find a 4-dimensional distribution $\Delta(x) = \Omega^{\perp}(x) = \text{span}\{g_1(x)\; g_2(x)\; g_3(x)\; g_4(x)\}$ where :

$$g_1(x) = \begin{bmatrix} \cos\theta_0 + A\cdot L\sin\theta_1 + l\frac{\sin\theta_0\tan\phi_0}{L-l} \\ \sin\theta_0 - A\cdot L\cos\theta_1 - l\frac{\cos\theta_0\tan\phi_0}{L-l} \\ \frac{\tan\phi_0}{L-l} \\ A \\ B \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{aligned}
g_2(x) &= [0\,0\,0\,0\,0\,1\,0\,0]^T \\
g_3(x) &= [0\,0\,0\,0\,0\,0\,1\,0]^T \\
g_4(x) &= [0\,0\,0\,0\,0\,0\,0\,1]^T
\end{aligned}$$

where $A = \frac{1}{L}\sin(\theta_0 - \phi_1 - \theta_1)\sec\phi_1 - \frac{l}{L(L-l)}\cos(\phi_1+\theta_1-\theta_0)\sec\phi_1\tan\phi_0$, $B = \frac{1}{L}\sin(\theta_0 - \phi_2 - \theta_2)\sec\phi_2 - \frac{l}{L(L-l)}\cos(\phi_2+\theta_2-\theta_0)\sec\phi_2\tan\phi_0 - A\frac{L+l}{l}\sec\phi_2\cos(\phi_2+\theta_2-\theta_1)$.

One can easily write now the model in the form (2)

We plan to convert the above system into the multi-chained form. To do so we need tools from exterior algebra and Pfaffian systems. Those tools are presented in Appendix A.

Our system has 4 inputs and 8 states. In the next section we apply the proofs of the theorems 2 and 3 in order to convert the control system of the mobile robot into the extended Goursat normal form.

4 Converting the "snake" robot system into Extended Goursat Normal Form

Our mechanical system is depicted in 1. In order to derive the Pfaffian system associated with the above system we write the nonholonomic constraints, taking into consideration the holonomic ones, as the one forms:

$$\begin{array}{rcl} a^0 & = & \sin\theta_0 dp_x - \cos\theta_0 dp_y - l d\theta_0 - L\cos(\theta_0-\theta_1)d\theta_1 \\ a^1 & = & \sin(\phi_1+\theta_1)dp_x - \cos(\phi_1+\theta_1)dp_y \\ a^2 & = & \sin(\phi_2+\theta_2)dp_x - \cos(\phi_2+\theta_2)dp_y + \\ & & L\cos\phi_2 d\theta_2 + l\cos(\theta_2+\phi_2-\theta_1)d\theta_1 \\ a^3 & = & \sin(\phi_0+\theta_0)dp_x - \cos(\phi_0+\theta_0)dp_y - \\ & & L\cos\phi_0 d\theta_0 - L\cos(\theta_0+\phi_0-\theta_1)d\theta_1 \end{array}$$

The derived flag of the Pfaffian system is:

$$I = I^{(0)} = \left\{a^0, a^1, a^2, a^3\right\} \quad I^{(1)} = \left\{a^0\right\} \quad I^{(2)} = \{0\}$$

Calculations of the exterior products give:

$$\begin{array}{l} da^1 \wedge \alpha = h_1(\theta_0,\theta_1,\theta_2,\phi_0,\phi_1,\phi_2)d\phi_1 \wedge ds \\ da^2 \wedge \alpha = h_2(\theta_0,\theta_1,\theta_2,\phi_0,\phi_1,\phi_2)d\phi_2 \wedge ds \\ da^3 \wedge \alpha = h_3(\theta_0,\theta_1,\theta_2,\phi_0,\phi_1,\phi_2)d\phi_0 \wedge ds \end{array}$$

where $\alpha = a^0 \wedge a^1 \wedge a^2 \wedge a^3$, $ds = d\theta_0 \wedge d\theta_1 \wedge d\theta_2 \wedge dp_x \wedge dp_y$. Thus the Pfaffian system $\left\{I^{(0)}, d\theta_0\right\}$ is integrable. We also have $da^0 \wedge d\theta_0 = 0$ and therefore $\left\{I^{(1)}, d\theta_0\right\}$ is integrable, too. According to theorem 3 our system can be converted to extended Goursat normal form. We now follow the proof of theorem 3 in order to find a basis I which satisfies the Goursat congruences (27).

The basis of the last nontrivial system $I^{(1)}$ is a^0. The definition of the derived flag $I^{(2)}$ and the fact that $\left\{I^{(1)}, d\theta_0\right\}$ is integrable imply that

$$da^0 \not\equiv 0 \bmod I^{(1)} \qquad da^0 \equiv 0 \bmod \left\{I^{(1)}, d\theta_0\right\} \quad (9)$$

Combining those equations we obtain the congruence $da^0 \equiv d\theta_0 \wedge \beta^0 \bmod I^{(1)}$ for some $\beta^0 \not\equiv 0 \bmod I^{(1)}$. From this and the definition of the derived flag $da^0 \equiv 0 \bmod I^{(0)}$ we obtain $\beta^0 \in I^{(0)}$. Applying this procedure to our system we obtain $\beta^0 = \bar{a}^1 = a^3 \csc\phi_0$ such that $da^0 \equiv d\theta_0 \wedge \bar{a}^1 \bmod a^0$. We choose $\bar{a}^3 = a^1$ such that $\left\{a^0, \bar{a}^1, a^2, \bar{a}^3\right\}$ is a basis of I which is not only adapted to the derived flag, but also satisfies the Goursat congruences:

We see that the form a^0 has rank 1 since $da^0 \wedge a^0 \neq 0$ and $da^0 \wedge da^0 \wedge a^0 = 0$. Therefore, according to theorem 1 there are f_1, f_2 such that $da^0 \wedge a^0 \wedge df_1 = 0$, $a^0 \wedge df_1 \neq 0$, $a^0 \wedge df_1 \wedge df_2 = 0$, $df_1 \wedge df_2 \neq 0$. Integration of these equations gives: $f_1 = \theta_0$,

$$f_2 = z_3^1 = p_x \sin\theta_0 - p_y\cos\theta_0 + L\sin(\theta_0-\theta_1) \quad (11)$$

So the form a^0 can be written as : $a^0 = df_2 - g \cdot df_1 = dz_3^1 - z_2^1 dz^0$, with

$$\begin{array}{l} z^0 = d\theta_0 \\ z_2^1 = l + p_x\cos\theta_0 + p_y\sin\theta_0 + L\cos(\theta_0-\theta_1) \end{array} \quad (12)$$

From the Goursat congruences (10) we can use a^0 and $\bar{a}^1$ in order to put $\bar{a}^1$ in the form $dz_2^1 - z_1^1 dz^0$. From the equation $\tilde{a}^1 = b_0 a^0 + b_1 \bar{a}^1$ we obtain $b_0 = -\cot\phi_0$ and $b_1 = 1$. So we have $\tilde{a}^1 = dz_2^1 - z_1^1 dz^0$ and

$$z_1^1 = (L-l)\cot\phi_0 + p_y\cos\theta_0 - p_x\sin\theta_0 - L\sin(\theta_0-\theta_1) \quad (13)$$

We can modify $a^2, \bar{a}^3$ using forms from I in order to put them in the form $dz_2^2 - z_1^2 dz^0$ and $dz_2^3 - z_1^3 dz^0$, respectively. We choose

$$z_2^2 = p_y, \qquad z_2^3 = \theta_2 \quad (14)$$

We set $\tilde{a}^2 = k_0 a^0 + k_1\tilde{a}^1 + k_2 a^2 + k_3\bar{a}^3 = dz_2^2 - z_1^2 dz^0$ and easily obtain k_0, k_1, k_2, k_3. Thus

$$\begin{array}{rcl} z_1^2 & = & \csc\phi_0 \sec\phi_1 \sin(\phi_1+\theta_1)\left[L\cos\phi_0\cos(\theta_0-\theta_1) - \right. \\ & & \left. -l\cos(\phi_0+\theta_0-\theta_1)\right] \end{array} \quad (15)$$

In the same way we set $\tilde{a}^3 = c_0 a^0 + c_1\tilde{a}^1 + c_2 a^2 + c_3\bar{a}^3 = dz_2^3 - z_1^3 dz^0$ and choose $z_2^3 = \theta_2$. Algebraic calculations provide c_0, c_1, c_2, c_3 and we obtain

$$\begin{array}{rll} z_1^3 = & \frac{1}{L} & \cot\phi_0 \sec\phi_1 \sec\phi_2 \cdot \\ & \{ & l\cos(\phi_2-\theta_1+\theta_2)\sin(\phi_1-\theta_0+\theta_1) + \\ & & L\cos(\theta_0-\theta_1)\sin(\phi_1-\phi_2+\theta_1-\theta_2)\} + \\ & \frac{l}{L^2}\{ & l\cos(\phi_2+\theta_0-\theta_1+\theta_2)\sin(\phi_0-\phi_1-\theta_1) - \\ & & L\cos(\phi_0+\theta_0-\theta_1)\sin(\phi_1-\phi_2+\theta_1-\theta_2)\} \cdot \\ & & \cdot \csc\phi_0 \sec\phi_1 \sec\phi_2 \end{array} \quad (16)$$

So we converted our control system in the four input, three-chained system:

$$\begin{array}{ll} \dot{z}^0 = v_1 & \dot{z}_1^2 = v_3 \\ \dot{z}_1^1 = v_2 & \dot{z}_2^2 = z_1^2 v_1 \\ \dot{z}_2^1 = z_1^1 v_1 & \dot{z}_1^3 = v_4 \\ \dot{z}_3^1 = z_2^1 v_1 & \dot{z}_2^3 = z_1^3 v_1 \end{array} \quad (17)$$

5 Steering the"snake" robot

There are a number of algorithms that can be used to sinusoidal method for multiple-input systems was proposed. Another steering method uses polynomial inputs and is presented fot the multiple-input case in [DTS94]. In [TC93] the method of piecewise constant inputs is used. The method of steering nonholonomic systems using piecewise constant inputs was first introduced in [MNC92] as multirate digital control. All of these algorithms steer the system from the initial to the desired state. The resulting trajectories may look "nicer" for some of those methods compared to others. Through simulations we found that the trajectories generated using the method of multirate controls are reasonable.

The theoretical framework that we adopted is based on [CP95] since it is powerful for systems admitting an exact discretization, because in those cases, an exact point-to-point trajectory can be generated.

We consider our system: $\dot{x} = \sum_{i=1}^{4} g_i(x) u_i$ driven by[1] $v_i(t) = v_i^D(k)\ t \in [k\delta, (k+1)\delta]$, $k \geq 0$, $i = 1, \ldots 4$. If the operator $X_i = v_1^D L_{g_1} + v_{i1}^D L_{g_2} + v_{i2}^D L_{g_3} + v_{i3}^D L_{g_4}$ is defined and the choice $v_{33}^D = v_{23}^D$, $v_{32}^D = v_{22}^D$ is made, one can proove [CP95, TC93] that

$$z(k+1) = e^{\bar{\delta} X_1} \circ e^{\bar{\delta} X_2} \circ e^{\bar{\delta} X_3} (I_d)|_{z(k)} \tag{18}$$

where $e^X \equiv \sum_{k \geq 0} \frac{1}{k!} X^k$ is the exponential Lie operator, $\bar{\delta} = \delta/3$. I_d is the identity function and "$\circ$" is the composition operator. We choose the controls:

$$\begin{aligned}
v_1(t) &= v_1^D \quad t \in [k\delta,\ (k+1)\delta] \\
v_2(t) &= \begin{cases} v_{11}^D & t \in \left[k\delta,\ (k+\frac{1}{3})\delta\right] \\ v_{21}^D & t \in \left[(k+\frac{1}{3})\delta\ (k+\frac{2}{3})\delta\right] \\ v_{31}^D & t \in \left[(k+\frac{2}{3})\delta,\ (k+1)\delta\right] \end{cases} \\
v_3(t) &= \begin{cases} v_{12}^D & t \in \left[k\delta,\ (k+\frac{1}{3})\delta\right] \\ v_{22}^D & t \in \left[(k+\frac{1}{3})\delta,\ (k+1)\delta\right] \end{cases} \\
v_4(t) &= \begin{cases} v_{13}^D & t \in \left[k\delta,\ (k+\frac{1}{3})\delta\right] \\ v_{23}^D & t \in \left[(k+\frac{1}{3})\delta,\ (k+1)\delta\right] \end{cases}
\end{aligned} \tag{19}$$

Using (18) with $\bar{\delta} = \delta/3$, the discretization of the system in chained form coordinates is exact, given by:

$$\begin{aligned}
z^0(k+1) &= z^0(k) + 3\bar{\delta} v_1^D \\
z_1^1(k+1) &= z_1^1(k) + \bar{\delta}(v_{11}^D + v_{21}^D + v_{31}^D) \\
z_1^2(k+1) &= z_1^2(k) + \bar{\delta}(v_{12}^D + 2v_{22}^D) \\
z_1^3(k+1) &= z_1^3(k) + \bar{\delta}(v_{13}^D + 2v_{23}^D) \\
z_2^1(k+1) &= z_2^1(k) + 3\bar{\delta} v_1^D z_1^1(k) + \\
&\quad \tfrac{\bar{\delta}^2}{2!}(5v_{11}^D + 3v_{21}^D + v_{31}^D) v_1^D \\
z_2^2(k+1) &= z_2^2(k) + 3\bar{\delta} v_1^D z_1^2(k) + \tfrac{\bar{\delta}^2}{2!}(5v_{12}^D + 4v_{22}^D) v_1^D \\
z_2^3(k+1) &= z_2^3(k) + 3\bar{\delta} v_1^D z_1^3(k) + \tfrac{\bar{\delta}^2}{2!}(5v_{13}^D + 4v_{23}^D) v_1^D \\
z_3^1(k+1) &= z_3^1(k) + 3\bar{\delta} v_1^D z_2^1(k) + \tfrac{9}{2!}\bar{\delta}^2 (v_1^D)^2 z_1^1(k) \\
&\quad + \tfrac{\bar{\delta}^3}{3!}(19v_{11}^D + 7v_{21}^D + v_{31}^D)(v_1^D)^2
\end{aligned} \tag{20}$$

The motion planning problem is recasted now to finding the controls $v_1^D, v_{11}^D, v_{211}^D, v_{31}^D, v_{12}^D, v_{22}^D, v_{13}^D, v_{23}^D$ to lead the system $z(t) = [z^0\ z^1\ z^1\ z^1\ z^2\ z^2\ z^3\ z^3]$ from an initial state $z(k) = z_0$ to the desired state $z(k+1) = z_f$. This can be easily done by simple algebra.

6 Simulation results

The simulation of the "snake" robot is performed using the system in the chained form. Then the actual state x and input u trajectories are obtained from z and the v_is via the inverse of transformations (11,12,13,14,16).

We considered a "snake" with $L = 1$, $l = 0.5$". The case study we chose is a transition from an initial state $x(0) = x_0 = [0\,0\,0\,0\,0\,0.2\,0\,0]^T$ to the desired $x_f = [6\,4\,\pi/2\,1.5\,\pi/2\,0.07\,0\,0]^T$. Notice that we chose $\phi_0(0) = 0.2 \neq 0$ to avoid singularity of certain coordinate transformations. The corresponing chained form coordinates are $z(0) = z_0 = [0\,2.46658\,1.5\,0\,0\,0\,0.25\,0]^T$ and $z_f = [\pi/2\ 1.06045\ 5.4974\ 6.07\ 7.13\ 4\ -0.5085\ \pi/2]^T$

Substituting these to the (20) and solving we obtain:

$$\begin{array}{lll}
v_{11}^D = 1.74597 & v_{12}^D = 0.339106 & v_{13}^D = 2.00568 \\
v_{21}^D = -2.55373 & v_{22}^D = 3.39584 & v_{23}^D = -1.3821 \\
v_{31}^D = -0.59837 & &
\end{array}$$

States x, y, θ_0 of the system can be shown in fig. 2.

7 Issues of further research

We plan to address a number of issues following the problem that we tackled. Stabilization around a point, trajectory tracking and motion planning in a maze-like environment are our immediate step while steering the robot on $SO(3)$ is our ultimate task.

A APPENDIX : Pfaffian systems and Goursat normal forms

We present some issues on exterior algebra and Pfaffian systems and the theorems providing the conditions to convert a Pfaffian system into the extended Goursat normal form ([BCG+91]).

A nonholonomic system can be defined by a codistribution $I = \{a^1, \ldots a^k\}$ on the cotangent space to the configuration manifold, specifying the directions that the system is not allowed to move. This codistribution generates a Pfaffian system, and can be analyzed using exterior differential systems. This formulation is dual of this of sec. 4 in the sence that codistribution I annihilates distribution Δ, i.e. $I = \Delta^{\perp}$ or, $a^i(g_j) = 0\ \forall i, j$. A basis for I is written in coordinates by taking each $a^i = 0$ as the i-th nonholonomic constraint.

We will need the notion of *congruence modulo* of a Pfaffian system. We say that

$$\eta \equiv \xi \bmod I \quad \text{if} \quad \eta \equiv \xi + \sum_{i=1}^{k} \vartheta^i \wedge a^i \tag{21}$$

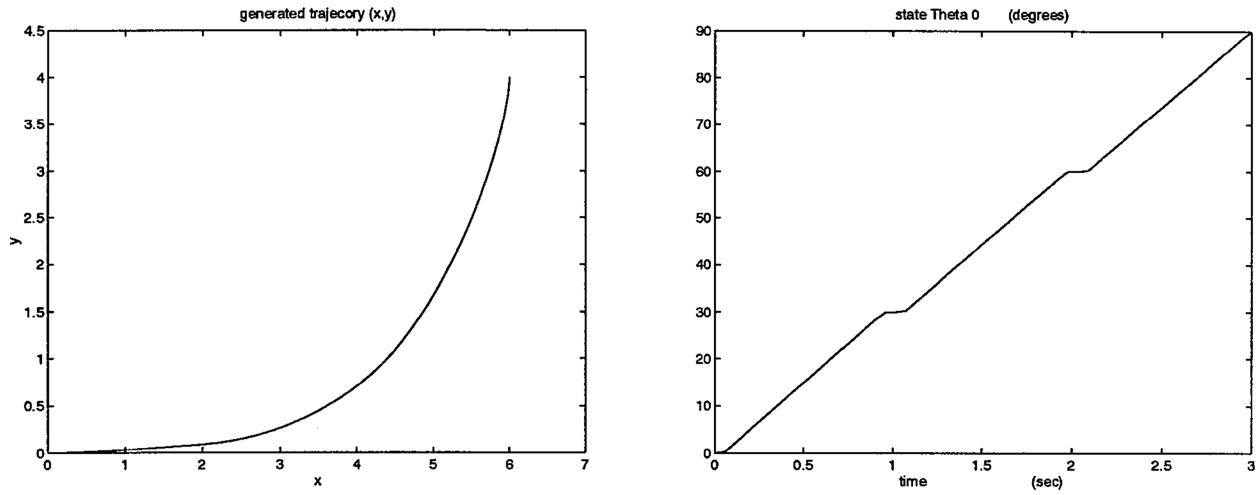

Figure 2: a) x-y trajectory of the middle link b) orientation θ_0 of the front link w.r.t time

The *derived flag* is defined to be the nested chain of codistributions given by $I^{(0)} = I$ and

$$I^{(k+1)} = \left\{ \omega \in I^{(k)} : d\omega \equiv 0 \bmod I^{(k)} \right\} \tag{22}$$

The construction is assumed to terminate at some N, when $I^{(N)} = I^{(N+1)}$.

One way to find integral curves for Pfaffian systems is to transform the system into a normal form. If such a normal form can be found, then there is a 1-1 correspodence between the integral curves of the normal form and the original system. A pertinent theorem is:

Theorem 1 *Pfaff's Theorem. Suppose a is a one-form on $\Re^n$ which satisfies $(da)^{r+1} \wedge a = 0$, $(da)^r \wedge a \neq 0$. Then there exist local coordinates z such that*

$$a = dz_1 + z_2 dz_3 + \ldots + z_{2r} dz_{2r+1} \tag{23}$$

In the case $r = 1$, the proof [BCG+91] shows that there exist two functions f_1, f_2 which satisfy the p.d.e

$$da \wedge a \wedge df_1 = 0 \quad a \wedge df_1 \neq 0 \tag{24}$$

$$a \wedge df_1 \wedge df_2 = 0 \quad df_1 \wedge df_2 \neq 0 \tag{25}$$

If suitable f_1, f_2, are found, a can be scaled so that

$$a = df_2 + g \cdot df_1 \tag{26}$$

In this paper we consider the case of multi-steering [BTS93b, TS94, DTS94], with a Pfaffian system of codimension four. We have the following definition:

Definition 1 *Extended Goursat Normal Form. A Pfaffian system I on $\Re^{n+m+1}$ of codimension $m+1$ is in extended Goursat normal form if it is generated by n constraints of the form:*

We note that all solution trajectories of I are determined by the $m+1$ functions $z^0(t), z_1^1(t), \ldots z_1^m(t)$ and their derivatives with respect to time t. Sufficient and necessary conditions for converting a two-input nonholonomic system into Goursat normal form are given in [Mur92]. For the multi-input case there are conditions due to [Mur93] for converting a system into extended Goursat normal form. The proof of the next theorem can be also found in [TS94].

Theorem 2 *Extended Goursat Normal Form. [Mur93] Let I be a Pfaffian system of codimension $m+1$. If there exists a set of generators $\left\{ a_i^j : j = 1, \ldots m; i = 1, \ldots s_j \right\}$ for I and an integrable one-form π such that*

$$\begin{array}{l} da_i^j \equiv -a_{i+1}^j \wedge \pi \bmod I^{s_j - i} \; i = 1, \ldots s_j - 1 \\ da_{s_j}^j \not\equiv 0 \bmod I \end{array} \tag{27}$$

for all j, then there exists a set of coordinates z such that I is in Goursat normal form ,

$$I = \left\{ dz_i^j - z_{i+1}^j dz^0 \; : \; j = 1, \ldots m; i = 1, \ldots s_j \right\} \tag{28}$$

The conditions (27) are called *Goursat congruenses.*

An algorithm for converting systems to extended Goursat normal form is given in [BTS93a]. There is also another set of conditions for converting a system into Goursat normal form, which are easier to check, since they do not require finding a basis which satisfies the Goursat congruences but only one which is adapted to the derived flag. A basis of one-forms a^j for I is said to be adapted to the derived flag if a basis for each derived system $I^{(i)}$ can be taken to be some subset of the a^j's. One special case of this theorem is proved in [Slu92]. The proof can also be found in [TS94].

converted to Goursat Normal Form if and only if there exists a one-form π *such that* $\{I^{(k)}, \pi\}$ *is integrable for* $k = 0, \dots N-1$.

Consider all the one-forms in I which are integrable modulo the entire codistribution: $\{a \,:\, da = 0 \bmod I\}$. If this set is the entire codistribution I, then I is integrable. The extended Goursat normal form is the dual of the multi-input, single-generator chained form presented in [BTS93b]. Sufficient conditions for converting the system to the above form are given in [BTS93b].

References

[AM89] J. Alexander and J. Maddocks. On the kinematics of wheeled mobile robots. *International Journal of Robotics Research*, pages 15–27, October 1989.

[BCG+91] R.L. Bryant, S.S. Chern, R.B. Gardner, H.L. Goldschmidt, and P.A. Griffiths. *Exterior differential systems.* Springer, 1991.

[BTS93a] L. Bushnell, D. Tilbury, and S.S. Sastry. Extended goursat normal forms with applications to nonholonomic motion planning. In *IEEE Conference on Decision and Control*, pages 3447–3452, 1993.

[BTS93b] L. Bushnell, D. Tilbury, and S.S. Sastry. Steering three input chained form nonholonomic systems using sinusoids:the firetruck example. In *European Control Conference*, pages 1432–1437, 1993.

[Bus95] L. Bushnell. An obstacle avoidance algorithm for a car pulling trailers with off-axle hitching. In *34th IEEE Conference on Decision and Control*, December 1995.

[CJ95] G.S. Chirikjian and J.W.Burdick. Kinematically optimal hyper-redudant manipulator configurations. *IEEE Transactions on Robotics and Automation*, 11(6), December 1995.

[CP95] A. Chelouah and M. Petitot. Finitely discretizable nonlinear systems :concepts and definitions. In *34th IEEE Conference on Decision and Control*, December 1995.

[DTS94] L.Bushnell D. Tilbury, O. Sordalen and S. Sastry. A multi-steering trailer system :conversion into chained form using dymamic feedback. In *IFAC Symposium on Robot Control*, September 1994.

[Fli93] M. Fliess. Flatness,motion planning,and trailer systems. In *32th IEEE Conference on Decision and Control*, pages 2700–2705, December 1993.

[MK97] G. Migadis and K. J. Kyriakopoulos. Design and forward kinematic analysis of a robotic snake. In *1997 IEEE International Conference on Robotics and Automation*, April 1997.

[MNC92] S. Monaco and D. Normand-Cyrot. An introduction to motion planning under multirate digital control. In *1992 IEEE Conference on Decision and Control*, pages 1780–1785, 1992.

[Mur92] R.M. Murray. Nilpotent bases for a class of nonholonomic distributions with applications to trajectory generation for nonholonomic systems. Technical Report CIT/CDS 92-002, California Institute of Technology, 1992.

[Mur93] R.M. Murray. Applications and extensions of goursat normal forms to control of nonlinear systems. In *IEEE Conference on Decision and Control*, pages 3425–3430, 1993.

[RS93] Murray R.M. and S. Sastry. Nonholonomic motion planning : Steering using sinusoids. *IEEE Transactions on Robotics & Automation*, May 1993.

[Slu92] S.L. Sluis. *Absolute Equivalence and its Applications to Control Theory.* PhD thesis, University of Waterloo, 1992.

[Sor93] O.J. Sordalen. Conversion of the kinematics of a car with n trailers into a chained form. In *1993 IEEE International Conference on Robotics and Automation*, pages 382–387, April 1993.

[TC93] D. Tilbury and A. Chelouah. Steering a three-input nonholonomic system using multirate controls. In *European Control Conference*, pages 1428–1431, 1993.

[TS94] D. Tilbury and S. Sastry. On coursat normal forms ,prolongations and control systems. In *IEEE Conference on Decision and Control*, December 1994.

[YY93] Y.Koren and Y.Shan. Design and motion planning of a mechanical snake. *IEEE Transactions on Systems, Man and Cyber-*