

# A hybrid model of obstacle-aided snake robot locomotion

Pål Liljebäck, Kristin Y. Pettersen, Øyvind Stavdahl, and Jan Tommy Gravdahl

**Abstract**—A snake can traverse cluttered and irregular environments by using irregularities around its body as push-points to aid the propulsion. This characteristic feature of biological snake locomotion, denoted *obstacle-aided locomotion*, is investigated for snake robot locomotion purposes in this paper. The paper presents a *hybrid model* of the dynamics of a planar snake robot interacting with obstacles in its environment. Obstacle contact forces are calculated by formulating and solving a *linear complementarity problem (LCP)*. The *existence* and *uniqueness* properties of the state evolution of the hybrid model are investigated. Simulation results validate the hybrid modelling approach.

## I. INTRODUCTION

Inspired by biological snake locomotion, snake robots carry the potential of meeting the growing need for robotic mobility in unknown and challenging environments. These mechanisms typically consist of serially connected joint modules capable of bending in one or more planes. The many degrees of freedom of snake robots make them difficult to control, but provides traversability in irregular environments that surpasses the mobility of the more conventional wheeled, tracked and legged forms of robotic mobility.

A unique feature of snake locomotion compared to other forms of robotic mobility is that irregularities on the ground are beneficial for the propulsion since they provide push-points for the snake. While *obstacle avoidance* is an important topic for wheeled, tracked and legged robots, the goal of snake locomotion is rather *obstacle exploitation*. The term *obstacle-aided locomotion* was introduced by Transeth *et al.* [1] and captures the essence of this concept.

Several empirical and analytical studies of biological snakes have been presented that shows the importance of external push-points on the locomotion of snakes [2]–[4]. Several mathematical models of the kinematics and dynamics of snake locomotion have been developed. Some models [5]–[8] assume that the links of the snake cannot move sideways (no-slip conditions achieved by e.g. mounting passive wheels along the snake body), while others assume isotropic or anisotropic friction conditions [9]–[12]. The works in [1], [13]–[15] present, to the authors' best knowledge, the only known models of snake robot dynamics that *also* include obstacle contact forces. However, except for the model in [15], these models are pure simulation models and not suited for synthesis of model-based control strategies. The work in [15] presents preliminary results by the authors in the form of an *analytical* model of a snake robot interacting

with obstacles. However, these preliminary results did not explicitly consider the hybrid nature of the model.

A long-term goal of the work presented in this paper is to synthesize model-based controllers for obstacle-aided locomotion with provable stability properties. To facilitate such developments, the underlying mathematical model should have a simple and analytical form. As a step in this direction, this paper presents a *hybrid model* of the dynamics of a planar snake robot interacting with obstacles in its environment. Obstacle interaction is modelled by imposing a *unilateral velocity constraint* on each contacted link of the snake robot. This is a novel approach since conventional models of mechanical systems with unilateral constraints calculate constraint forces *with respect to the normal direction of the obstacles* [16]. With the approach described in this paper, the shape of the obstacles does not have to be considered explicitly as we instead calculate constraint forces *with respect to the normal direction of the contacted links*. This simplifies the equations of motion. As an extension to our previous work in [15], we show how the equations governing the obstacle contact forces on the snake robot can be formulated as a *linear complementarity problem (LCP)*. This enables us to apply existing general results concerning existence and uniqueness of solutions to LCPs [17] to the model of the snake robot. Simulation results are presented that validate the hybrid modelling approach.

The paper is organized as follows. Section II presents essential background material regarding hybrid modelling and LCPs. Section III presents a 2D model of a snake robot without obstacle contact forces. Section IV extends the 2D model in order to include obstacle contact forces. Section V formulates the 2D model with obstacle contact forces as a *hybrid dynamical system*. Section VI presents a control strategy for obstacle-aided locomotion. Section VII presents simulation results, and Section VIII presents concluding remarks.

## II. BACKGROUND MATERIAL

This section presents the hybrid modelling framework employed in Section V and also the *linear complementarity problem*, which is central in the modelling of obstacle contact forces in Section IV.

### A. Modelling of hybrid dynamical systems

A *hybrid dynamical system* is a dynamical system that exhibits both continuous and discontinuous state evolution. A snake robot interacting with obstacles is a hybrid system since the impacts between the snake and the obstacles represent discrete events. Several modelling frameworks for hybrid systems exist [18]. In this work, we have chosen to employ the modelling framework described in [19] since it captures a wide variety of hybrid phenomena, and it also facilitates stability analysis of hybrid systems.

In accordance with [19], a hybrid system has a state vector  $x \in \mathbb{R}^n$  that can both *flow* (evolve continuously) and

Affiliation of Pål Liljebäck is shared between the Department of Engineering Cybernetics at the Norwegian University of Science and Technology, NO-7491 Trondheim, Norway, and SINTEF ICT, Dept. of Applied Cybernetics, N-7465 Trondheim, Norway. E-mail: Pal.Liljeback@sintef.no

K. Y. Pettersen, Øyvind Stavdahl, and Jan Tommy Gravdahl are with the Department of Engineering Cybernetics at the Norwegian University of Science and Technology, NO-7491 Trondheim, Norway. E-mail: {Kristin.Y.Pettersen, Oyvind.Stavdahl, Tommy.Gravdahl}@itk.ntnu.no

*jump* (evolve discontinuously). The data that determine the evolution of  $x$  is given by the four elements  $(C, F, D, G)$ , where  $C$  denotes the *flow set*,  $F$  denotes the *flow map*,  $D$  denotes the *jump set*, and  $G$  denotes the *jump map* of the hybrid system. Whenever the state  $x$  belongs to the *flow set*, it flows (or evolves continuously) according to  $F$ . During flows, the system acts as an ordinary continuous dynamical system. However, when  $x$  belongs to the *jump set*, it generally jumps according to  $G$  to a new value  $x^+$ , where superscript  $+$  denotes ‘the next value’. Hence, the general form of a hybrid dynamical system is given by

$$\begin{aligned} \dot{x} &= F(x, u) & \text{for all } x \in C \\ x^+ &= G(x) & \text{for all } x \in D \end{aligned} \quad (1)$$

where we have also included a control input,  $u \in \mathbb{R}^r$ .

*Existence* and *uniqueness* of solutions is a very important issue when modelling hybrid systems. From a given initial state,  $x_0$ , a hybrid system may have a single solution, several solutions, or no solution at all. For a general hybrid system, there are no easily verifiable necessary and sufficient conditions for *existence* and *uniqueness* of solutions. However, such conditions exist for special classes of hybrid systems, such as for *complementarity systems* (see Section II-B).

A hybrid system is simulated by letting the state vector,  $x$ , flow according to the flow map,  $F(x, u)$ , as long as  $x \in C$ . Whenever  $x \in D$ , a jump in the state vector,  $x^+$ , is calculated according to the jump map,  $G(x)$ , and the simulation of the flow map is *restarted* from the new initial value,  $x^+$ .

### B. Complementarity systems

A hybrid system is called a *complementarity system* if the *flow* of the state,  $x$ , is constrained by a set of *complementarity conditions* [18]. The *complementarity condition* between two scalar variables,  $y$  and  $\lambda$ , is given as  $y \geq 0 \wedge \lambda \geq 0 \wedge y\lambda = 0$ . The condition states that both variables must be nonnegative and at least one variable is zero. Two vectors,  $y \in \mathbb{R}^m$  and  $\lambda \in \mathbb{R}^m$ , are said to be *complementary* if, for all  $i$ , the pair of variables  $(y_i, \lambda_i)$  is subject to a complementarity condition.

The constraint equations of a complementarity system can often be formulated as a *linear complementarity problem* (LCP). A LCP asks whether there exist two *complementary* vectors,  $y \in \mathbb{R}^m$  and  $\lambda \in \mathbb{R}^m$ , such that

$$\begin{aligned} y &= a + A\lambda \\ y \geq 0, \lambda \geq 0, y^T \lambda &= 0 \end{aligned} \quad (2)$$

for a given vector  $a \in \mathbb{R}^m$  and a matrix  $A \in \mathbb{R}^{m \times m}$ . The constraint equations of the snake robot are given in this form in Section IV. The following result is proved in [17]:

*Theorem 1:* The LCP in (2) is *uniquely solvable* for all data vectors  $a$  if and only if  $A$  is a *P-matrix*.

A matrix is a *P-matrix* if all its *principal minors* are positive. A *principal minor* of the matrix  $A$  is the *determinant* of a square submatrix of  $A$  consisting of the same set of rows and columns. A real *symmetrical* matrix is a *P-matrix* if and only if it is *positive definite*. For a real *symmetrical* matrix, one can therefore apply the standard criteria for positive definiteness in order to check if the matrix is a *P-matrix*. If the matrix is *not symmetrical*, one can e.g. apply the recursive algorithm in [20], which is  $O(2^m)$ , in order to check if the matrix is a *P-matrix*.

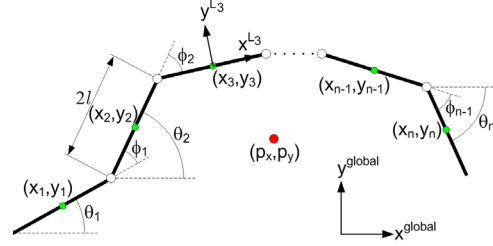


Fig. 1. Kinematic parameters for the snake robot.

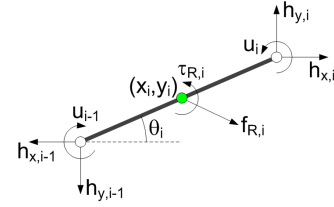


Fig. 2. Forces and torques acting on each link of the snake robot.

Several algorithms exist for solving the LCP in (2). A famous approach is the so-called *Lemke's algorithm* [17], which basically uses trial and error to find the non-zero elements of  $y$  and  $\lambda$ , but with clever rules for changing the non-zero elements between trials.

### III. MODEL OF A SNAKE ROBOT WITHOUT OBSTACLES

This section presents a *continuous* 2D model of a snake robot moving on a flat surface *without* obstacles. The model is based on [9], but whereas nonuniform ground friction was assumed in [9], we here employ a uniform friction model.

#### A. Notations and defined symbols

The snake robot consists of  $n$  links of length  $2l$  interconnected by  $n - 1$  joints. All  $n$  links have the same mass  $m$  and moment of inertia  $J = \frac{1}{3}ml^2$ . The total mass of the robot is therefore  $nm$ . The mass of each link is uniformly distributed so that the link CM (center of mass) is located at its center point (at length  $l$  from the joint at each side). The mathematical symbols defined in order to describe the kinematics and dynamics of the robot are illustrated in Fig. 1 and Fig. 2.

Vectors are either expressed in the global coordinate system or in the local coordinate system of link  $i$ . This is indicated by superscript *global* or  $L_i$ , respectively. If otherwise is not specified, a vector with no superscript is expressed in the global coordinate system. The following vectors and matrices are used in this paper:

$$A := \begin{bmatrix} 1 & 1 & & & \\ & \cdot & \cdot & \cdot & \\ & & \cdot & \cdot & \\ & & & 1 & 1 \\ & & & & \cdot \end{bmatrix}, D := \begin{bmatrix} 1 & -1 & & & \\ & \cdot & \cdot & \cdot & \\ & & \cdot & \cdot & \\ & & & 1 & -1 \\ & & & & \cdot \end{bmatrix}$$

where  $A \in \mathbb{R}^{(n-1) \times n}$  and  $D \in \mathbb{R}^{(n-1) \times n}$ . Furthermore,

$$\begin{aligned} e &:= [1 \ \dots \ 1]^T \in \mathbb{R}^n, \quad E = \begin{bmatrix} e & 0_{n \times 1} \\ 0_{n \times 1} & e \end{bmatrix} \in \mathbb{R}^{2n \times 2}, \\ \sin \theta &:= [\sin \theta_1 \ \dots \ \sin \theta_n]^T \in \mathbb{R}^n, \quad S_\theta := \text{diag}(\sin \theta) \in \mathbb{R}^{n \times n}, \\ \cos \theta &:= [\cos \theta_1 \ \dots \ \cos \theta_n]^T \in \mathbb{R}^n, \quad C_\theta := \text{diag}(\cos \theta) \in \mathbb{R}^{n \times n}. \end{aligned}$$

### B. Kinematics of the snake robot

The snake robot moves in the horizontal plane and has a total of  $n + 2$  degrees of freedom. The absolute angle,  $\theta_i$ , of link  $i$  is expressed with respect to the global  $x$  axis with counterclockwise positive direction. As seen in Fig. 1, the relative angle between link  $i$  and link  $i + 1$  is given by  $\phi_i = \theta_i - \theta_{i+1}$ . The local coordinate system of each link is fixed in the CM (center of mass) of the link with  $x$  (tangential) and  $y$  (normal) axis oriented such that they are oriented in the directions of the global  $x$  and  $y$  axis, respectively, when the link angle is zero. The rotation matrix from the global frame to the frame of link  $i$  is given by

$$R_{L_i}^{\text{global}} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{bmatrix} \quad (3)$$

The position of the snake robot,  $p$ , is described through the coordinates of its CM (center of mass) and is given by

$$p := \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} \frac{1}{nm} \sum_{i=1}^n m x_i \\ \frac{1}{nm} \sum_{i=1}^n m y_i \end{bmatrix} = \frac{1}{n} \begin{bmatrix} e^T x \\ e^T y \end{bmatrix} \quad (4)$$

It is shown in [15] that the position of the CM of each link along the global  $x$  and  $y$  axis, respectively, is given by

$$\begin{aligned} x &= -lN^T \cos \theta + e p_x \\ y &= -lN^T \sin \theta + e p_y \end{aligned} \quad (5)$$

$$N = A^T (DD^T)^{-1} D \in \mathbb{R}^{n \times n}$$

The linear velocities of the links are derived by differentiating (5). This gives

$$\begin{aligned} \dot{x} &= lN^T S_\theta \dot{\theta} + e \dot{p}_x \\ \dot{y} &= -lN^T C_\theta \dot{\theta} + e \dot{p}_y \end{aligned} \quad (6)$$

### C. Coulomb friction model

Each link is subjected to a ground friction force acting on the CM of the link and also a friction torque acting about the link CM. The friction forces and torques on all the links are contained in the vectors

$$f_R = \begin{bmatrix} f_{R,x} \\ f_{R,y} \end{bmatrix} \in \mathbb{R}^{2n} \quad (7)$$

$$\tau_R \in \mathbb{R}^n \quad (8)$$

where  $f_{R,x} \in \mathbb{R}^n$  and  $f_{R,y} \in \mathbb{R}^n$  contain the ground friction forces in the global  $x$  and  $y$  direction, respectively, and  $\tau_R$  contains the friction torques. Due to space limitation, we refer the reader to [15] for further details about the calculation of the friction forces and torques.

### D. Equations of motion

It is shown in [15] that the complete model of the snake robot dynamics without obstacles is given as

$$\begin{aligned} M_\theta \ddot{\theta} &= W \dot{\theta}^2 + l S_\theta N f_{R,x} - l C_\theta N f_{R,y} + \tau_R + D^T u \\ nm \ddot{p} &= E^T f_R \end{aligned} \quad (9)$$

where  $\theta$  and  $p$  represent the  $n + 2$  generalized coordinates of the system,  $\dot{\theta}^2 = \text{diag}(\dot{\theta})\dot{\theta}$ , and

$$\begin{aligned} M_\theta &:= JI_{n \times n} + ml^2 S_\theta V S_\theta + ml^2 C_\theta V C_\theta \\ W &:= ml^2 C_\theta V S_\theta - ml^2 S_\theta V C_\theta \\ N &:= A^T (DD^T)^{-1} D \\ V &:= A^T (DD^T)^{-1} A \end{aligned} \quad (10)$$

These equations may be combined into the following single differential equation:

$$M(q) \ddot{q} = f_u(q, \dot{q}, u) \quad (11)$$

where subscript ‘ $u$ ’ denotes *unconstrained* and

$$\begin{aligned} q &= \begin{bmatrix} \theta \\ p \end{bmatrix} \in \mathbb{R}^{n+2} \\ M(q) &= \begin{bmatrix} M_\theta & 0_{n \times 1} & 0_{n \times 1} \\ 0_{1 \times n} & nm & 0 \\ 0_{1 \times n} & 0 & nm \end{bmatrix} \in \mathbb{R}^{(n+2) \times (n+2)} \\ f_u &= \begin{bmatrix} l S_\theta N & -l C_\theta N \\ e^T & 0_{1 \times n} \\ 0_{1 \times n} & e^T \end{bmatrix} f_R + \begin{bmatrix} W \dot{\theta}^2 + \tau_R + D^T u \\ 0 \\ 0 \end{bmatrix} \end{aligned} \quad (12)$$

## IV. MODELLING OF OBSTACLE CONTACT FORCES

This section extends the model from Section III in order to include contact forces from external obstacles in the environment around the snake robot.

### A. Overview of the contact modelling approach

The model of the snake robot should be kept as simple as possible for control design purposes. We therefore base the contact model of the robot on the following simplifying assumptions:

*Assumption 2:* All obstacles have a circular shape. This simplifies the specification of the obstacle environment.

*Assumption 3:* The distance between the edges of any two obstacles is greater than the link length  $2l$ . This prevents contact on both sides of a link.

*Assumption 4:* The friction coefficient between the snake robot and any obstacle is  $\mu_o$ .

*Assumption 5:* An obstacle contact force acts only on the CM of a link and contact torques about the link CM are negligible. This assumption simplifies the equations of motion considerably and is reasonable when the length of the links is small.

*Assumption 6:* Impacts between the snake robot and the obstacles are completely inelastic.

*Assumption 7:* All impacts are instantaneous in time and all impact forces are impulsive.

*Assumption 8:* During an impact, the configuration of the snake robot,  $q = (\theta, p)$ , remains unaltered while the velocity,  $\dot{q} = (\dot{\theta}, \dot{p})$ , will generally experience a jump.

*Assumption 9:* Obstacle friction forces are negligible during an impact.

The interaction between a snake robot link and an obstacle is modelled by introducing a *unilateral velocity constraint* for the link when it comes into contact with an obstacle. The constraint is *unilateral* (acts in one lateral direction only) since the constraint shall allow sideways motion of the link *away* from the obstacle, but prevent any sideways motion *towards* (and thereby into) the obstacle.

As illustrated in Fig. 3, the obstacle contact force on link  $i$  consists of a *constraint force*,  $f_{c,i} \in \mathbb{R}^2$ , acting in the normal direction of link  $i$  and away from the obstacle (parallel to the local  $y$  axis of link  $i$ ), and an obstacle *friction force*,  $f_{\mu,i} \in \mathbb{R}^2$ , acting in the tangential direction of link  $i$  and in the opposite direction of the tangential link velocity (parallel to the local  $x$  axis of link  $i$ ).

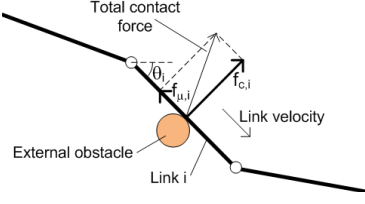


Fig. 3. The obstacle contact force on link  $i$  consisting of the normal direction constraint force,  $f_{c,i}$ , and the tangential direction friction force,  $f_{\mu,i}$ .

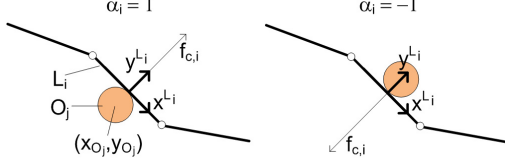


Fig. 4. The value of the contact parameter of link  $i$  is  $\alpha_i = 0$  when there is no contact,  $\alpha_i = 1$  when the constraint force points along the positive link  $y$  axis, and  $\alpha_i = -1$  when the constraint force points along the negative link  $y$  axis.

### B. Collision detection

The environment around the snake robot consists of  $k$  circular obstacles indexed by  $j \in \{1, \dots, k\}$ . The coordinates of the center of obstacle  $j$  is denoted by  $(x_{O_j}, y_{O_j})$ . The set  $O_j$  of points occupied by obstacle  $j$  is given by

$$O_j = \left\{ (x, y) \mid (x - x_{O_j})^2 + (y - y_{O_j})^2 \leq R_{O_j}^2 \right\} \quad (13)$$

where  $R_{O_j}$  is the radius of obstacle  $j$ . The set  $L_i$  of points occupied by link  $i$  (see Fig. 1) is given by

$$L_i = \{(x, y) \mid x = x_i + s \cos \theta_i, y = y_i + s \sin \theta_i, s \in [-l, l]\} \quad (14)$$

A collision between link  $i$  of the snake robot and obstacle  $j$  occurs whenever  $L_i \cap O_j \neq \emptyset$ , where  $\emptyset$  denotes an empty set. There is no collision if  $L_i \cap O_j = \emptyset$ .

We now introduce a vector of *contact parameters*,  $\alpha \in \mathbb{R}^n$ . The contact parameter of link  $i$ , denoted by  $\alpha_i \in \{-1, 0, 1\}$ , is a discrete state value that determines if the link is in contact with an obstacle and also on which side of the link there is contact. As shown in Fig. 4,  $\alpha_i = 0$  when there is no contact,  $\alpha_i = 1$  when the obstacle constraint force points along the positive link  $y$  axis, and  $\alpha_i = -1$  when the obstacle constraint force points along the negative link  $y$  axis.

Whenever link  $i$  impacts an obstacle, the contact parameter,  $\alpha_i$ , is updated according to  $\alpha_i = \tilde{\alpha}(i)$ , where

$$\tilde{\alpha}(i) = -\text{sgn} \left( [0, 1] \left( R_{L_i}^{\text{global}} \right)^T \left( \min_{j \in \{1, \dots, k\}} r_{L_i, O_j} \right) \right) \quad (15)$$

and  $r_{L_i, O_j} \in \mathbb{R}^2$  is the vector from link  $i$  to obstacle  $j$ . In (15), the global frame vector from link  $i$  to the *closest* obstacle is first found (by use of the  $\min$  operator). Subsequently, this vector is transformed to the frame of link  $i$  using the rotation matrix  $R_{L_i}^{\text{global}}$ . Finally, the  $y$  component of this vector is extracted. The sign of the  $y$  component determines on which side of the link the obstacle is located.

### C. Unilateral constraint equations

This section derives the unilateral velocity constraints for the links of the snake robot that are in contact with an obstacle. With reference to Fig. 4, the unilateral velocity constraint imposed on link  $i$  during contact with an obstacle may be compactly expressed as

$$\alpha_i v_{n,i} \geq 0 \quad (16)$$

where  $v_{n,i}$  is the normal direction velocity of link  $i$ . This constraint prevents any sideways motion towards (and thereby into) the obstacle. Using (3), the velocity constraint is given in the global frame as

$$\alpha_i (-\dot{x}_i \sin \theta_i + \dot{y}_i \cos \theta_i) \geq 0 \quad (17)$$

By assembling the velocity constraints for all links in matrix form, inserting (6), and rearranging we get

$$\text{diag}(\alpha) C(q) \dot{q} \geq 0 \quad (18)$$

where  $\text{diag}(\alpha) \in \mathbb{R}^{n \times n}$  is a diagonal matrix containing the elements of  $\alpha$  along its diagonal and  $C(q) \in \mathbb{R}^{n \times n+2}$  is given by

$$C(q) = \begin{bmatrix} -l(S_\theta N^T S_\theta + C_\theta N^T C_\theta) & -\sin \theta & \cos \theta \end{bmatrix} \quad (19)$$

We denote the number of contacted links by  $m \in \{0, 1, \dots, n\}$ . In order to easily select the velocity constraints from (18) that correspond to contacted links, we define a *selection matrix*  $S_c(\alpha) \in \mathbb{R}^{m \times n}$ , which simply contains all  $m$  rows from the matrix  $\text{diag}(\alpha)$  that contain a nonzero element. This enables us to write the equation of velocity constraints for all links that are in contact with an obstacle as

$$\bar{C}(q, \alpha) \dot{q} \geq 0 \quad (20)$$

where  $\bar{C}(q, \alpha) = S_c(\alpha) C(q) \in \mathbb{R}^{m \times n+2}$ . The following rank property is important in order to uniquely determine the contact forces acting on the snake robot.

*Property 10:* The matrix  $\bar{C}$  has full rank ( $\text{rank}(\bar{C}) = m$ ) for all  $(q, \alpha)$ .

*Remark 11:* Due to the complexity of the elements in  $\bar{C}$ , it is difficult to present a purely mathematical proof that Property 10 holds. However, we can argue from a physical perspective that this property must hold. Assume that  $\text{rank}(\bar{C}) < m$ . This implies linear dependence between some of the  $m$  rows of  $\bar{C}$ , i.e. there must exist a row of  $\bar{C}$ , denoted  $\bar{C}_i$ , such that

$$\bar{C}_i = \sum_{j \in \{1, \dots, m\} \setminus \{i\}} k_j \bar{C}_j \quad (21)$$

where  $k_j \in \mathbb{R}$ . The scalar  $\bar{C}_i \dot{q}$  is the magnitude of the normal direction velocity of link  $i$ , denoted by  $|v_{n,i}|$ . Multiplying (21) by  $\dot{q}$  therefore gives

$$|v_{n,i}| = \bar{C}_i \dot{q} = \sum_{j \in \{1, \dots, m\} \setminus \{i\}} k_j |v_{n,j}| \quad (22)$$

which states that the normal direction velocity of link  $i$  can be written as a linear combination of the normal direction velocities of all other contacted links of the snake robot. From a physical perspective, such a dependence could never occur unless all links are parallel since the couplings between the link velocities must naturally be given in terms of both normal *and* tangential link velocities. This contradicts

(22) since the relationship in (22) only contains normal direction velocities. This leaves the case of parallel links ( $\theta_1 = \theta_2 = \dots = \theta_n$ ) as the only way for (22) to be true. A straightforward calculation of  $\overline{C}$  in e.g. *Matlab Symbolic Toolbox* shows that  $\overline{C}$  always has full rank when the link angles are equal. We therefore conclude that  $\overline{C}$  never drops rank.

This subsection can be summarized as follows. At any given time instant the snake robot is in contact with  $m$  obstacles. The interaction between the snake robot and these  $m$  obstacles is modelled by imposing a unilateral velocity constraint on each of the  $m$  contacted links. These  $m$  velocity constraints are given by (20).

#### D. Continuous contact dynamics

We will now use the unilateral velocity constraints in (20) to derive the resulting equations of motion of the snake robot. We assume that the  $m$  contact points between the links and the obstacles have *already* been established, i.e. we consider the *continuous contact dynamics* of the snake robot over a time interval where the set of contacted links remains constant. The *discrete impact dynamics* occurring when a non-contacted link comes into contact with an obstacle is treated in Section IV-E. We first consider the frictionless case in Section IV-D.1, followed by contact forces *with* friction in Section IV-D.2.

1) *Contact dynamics without obstacle friction*: The equations of motion of the snake robot when the links are subjected to the velocity constraints in (20) are given by [15], [21]

$$M\ddot{q} = f_u + \overline{C}^T \lambda \quad (23)$$

where  $\lambda \in \mathbb{R}^m$  is a vector of scalars known as *Lagrange multipliers* [21]. The Lagrange multipliers are important because multiplier  $\lambda_j$  equals the magnitude of the constraint force that ensures compliance with the  $j$ th constraint. This means that if the  $j$ th constraint in (20) corresponds to the velocity constraint on link  $i$ , then  $\lambda_j$  equals the magnitude of the constraint force  $f_{c,i}$  acting on link  $i$ . Since the velocity constraints in (20) are *unilateral*, we require that  $\lambda \geq 0$ . This means that the constraint forces can only point *away* from the obstacles. We will handle this directional requirement by employing the theory of *linear complementarity problems* (LCPs) introduced in Section II-B. This approach is based on the work in [22]. An important observation is that the normal direction velocity of a contacted link and the corresponding constraint force are subjected to a *complementarity condition*, i.e. we have that

$$\overline{C}\dot{q} \geq 0, \quad \lambda \geq 0, \quad \lambda^T \overline{C}\dot{q} = 0 \quad (24)$$

The normal direction *acceleration* of each contacted link in the direction *away* from each obstacle, denoted by  $\overline{a}_n \in \mathbb{R}^m$ , is given by

$$\overline{a}_n = \frac{d}{dt} (\overline{C}\dot{q}) = \overline{C}\ddot{q} + \dot{\overline{C}}\dot{q} \geq 0 \quad (25)$$

By solving (23) for  $\ddot{q}$  and inserting into (25), we arrive at the following model of the *continuous contact dynamics* of the snake robot with *frictionless* obstacles:

$$M\ddot{q} = f_u + \overline{C}^T \lambda \quad (26)$$

$$\begin{aligned} \overline{a}_n &= \overline{C}M^{-1}f_u + \dot{\overline{C}}\dot{q} + \overline{C}M^{-1}\overline{C}^T \lambda \\ \overline{a}_n &\geq 0, \quad \lambda \geq 0, \quad \lambda^T \overline{a}_n = 0 \end{aligned} \quad (27)$$

Equation (27) is in the form of the general LCP given in (2) with  $A = \overline{C}M^{-1}\overline{C}^T$  and  $a = \overline{C}M^{-1}f_u + \dot{\overline{C}}\dot{q}$ . In order to calculate the dynamics of the snake robot at any given time instant, this LCP must be solved for the unknowns  $\overline{a}_n$  and  $\lambda$  subject to the complementarity conditions. The calculated  $\lambda$  gives the constraint forces from the obstacles and is plugged into (26) in order to calculate  $\ddot{q}$ . The following result concerns the existence and uniqueness properties of the LCP in (27):

*Proposition 12*: The LCP in (27) always possesses a unique solution  $(\overline{a}_n, \lambda)$ .

*Proof*: From Theorem 1, the proof is complete if we can show that  $A = \overline{C}M^{-1}\overline{C}^T$  is a  $P$ -matrix. We first note that  $A$  is symmetrical. In addition, we have that  $\overline{C}$  has full rank (by Property 10) and  $M > 0$  (the inertia matrix is always positive definite). We can therefore conclude that  $A = \overline{C}M^{-1}\overline{C}^T > 0$ , which implies that  $A$  is a  $P$ -matrix. This completes the proof. ■

2) *Contact dynamics with obstacle friction*: A Coulomb friction model is employed in order to describe the gliding friction force between the links and the obstacles. In accordance with Fig. 3, we define the obstacle friction forces on all the links of the snake robot as

$$f_\mu = \begin{bmatrix} f_{\mu,x} \\ f_{\mu,y} \end{bmatrix} = -\mu_o \begin{bmatrix} C_\theta \\ S_\theta \end{bmatrix} \text{diag}(\text{sgn}(v_t)) |f_c| \quad (28)$$

where  $\mu_o \in [0, 1]$  is the Coulomb friction coefficient of the obstacles,  $f_{\mu,x} \in \mathbb{R}^n$  and  $f_{\mu,y} \in \mathbb{R}^n$  contain the obstacle friction forces in the global  $x$  and  $y$  direction, respectively,  $f_c \in \mathbb{R}^n$  contains the constraint forces in the local  $y$  direction of each link, and  $v_t \in \mathbb{R}^n$  contains the tangential link velocities in the local  $x$  direction of each link. The mapping between the friction forces and the link accelerations,  $\ddot{q}$ , is identical to the mapping between ground friction forces,  $f_R$ , and  $\ddot{q}$  given in (12) since  $f_R$  and  $f_\mu$  both act on the CM of the links. By using that  $|f_c| = S_c(\alpha)^T \lambda$ , where  $S_c(\alpha)$  is the selection matrix introduced in (20), we may write the link accelerations due to the obstacle friction forces, temporarily denoted  $\ddot{q}_{f_\mu}$ , as

$$\ddot{q}_{f_\mu} = -\mu_o \Lambda \lambda \quad (29)$$

where  $\Lambda \in \mathbb{R}^{n+2 \times m}$  is given by

$$\Lambda = \begin{bmatrix} lS_\theta N & -lC_\theta N \\ e^T & 0_{1 \times n} \\ 0_{1 \times n} & e^T \end{bmatrix} \begin{bmatrix} C_\theta \\ S_\theta \end{bmatrix} \text{diag}(\text{sgn}(v_t)) S_c(\alpha)^T \quad (30)$$

By adding (29) to the equations of motion in (26) and following the exact same approach that led to the LCP in (27), we get

$$M\ddot{q} = f_u + (\overline{C}^T - \mu_o \Lambda) \lambda \quad (31)$$

$$\overline{a}_n = \overline{C}M^{-1}f_u + \dot{\overline{C}}\dot{q} + \overline{C}M^{-1}(\overline{C}^T - \mu_o \Lambda) \lambda \quad (32)$$

$$\overline{a}_n \geq 0, \quad \lambda \geq 0, \quad \lambda^T \overline{a}_n = 0$$

We again identify (32) as a LCP of the general form given in (2) with  $A = \overline{C}M^{-1}(\overline{C}^T - \mu_o \Lambda)$  and  $a = \overline{C}M^{-1}f_u + \dot{\overline{C}}\dot{q}$ . When obstacle friction is present, we can no longer guarantee

existence and uniqueness of the solution to the LCP in (32) since it is no longer evident that the matrix  $A$  is a  $P$ -matrix (see Section II-B). We are unable to provide an analytical upper bound of  $\mu_o$  due to the complexity of determining if  $A$  is a  $P$ -matrix ( $A$  is not symmetrical, which complicates the  $P$ -matrix check). However, we can still state the following result:

*Proposition 13:* For a given  $(q, \dot{q}, \alpha)$ , there exists a  $\mu_o^* > 0$  such that the LCP in (32) possesses a unique solution  $(\bar{a}_n, \lambda)$  for  $\mu_o \in [0, \mu_o^*]$ .

*Proof:* Recall from Section II-B that  $A = \bar{C}M^{-1}(\bar{C}^T - \mu_o\Lambda)$  is a  $P$ -matrix if all principal minors of  $A$  are positive. We know from Proposition 12 that the LCP in (32) *always* possesses a unique solution for  $\mu_o = 0$  since  $A$  is a  $P$ -matrix in this case. All principal minors of  $A$  must therefore be positive for  $\mu_o = 0$ . Assume now that we increase  $\mu_o$  until a principal minor of  $A$  becomes zero, and denote the corresponding value of the friction coefficient by  $\mu_o^* > 0$ . It is then evident that the  $P$ -matrix property of  $A$  must be preserved for  $\mu_o < \mu_o^*$ , i.e. existence and uniqueness of the solution to the LCP in (32) must hold for  $\mu_o < \mu_o^*$ . This completes the proof. ■

*Remark 14:* During our numerical treatments of the LCP in (32) so far, we have not yet encountered a single instance where  $A$  has failed to be a  $P$ -matrix. The authors therefore conjecture that  $\mu_o$  must have an unrealistically high value in order for  $A$  to no longer be a  $P$ -matrix, and that the LCP in (32) will always be uniquely solvable during our simulations of the snake robot.

### E. Discrete impact dynamics

An *inelastic impact* occurs when a link comes into contact with an obstacle. We calculate the impulsive constraint forces and the post-impact velocity of the snake robot by following the approach presented in [18]. The impact dynamics is given by [18]

$$M(q)(\dot{q}^+ - \dot{q}^-) = \bar{C}^T(q, \alpha^+) \lambda \quad (33)$$

where  $\dot{q}^-$  and  $\dot{q}^+$  denote the generalized velocities immediately *before* and *after* the impact, respectively, and  $\lambda \in \mathbb{R}^m$  is a vector of *impulsive constraint forces*. Note that if link  $i$  impacts an obstacle, then  $\alpha_i^- = 0$ . In order to include this link in the impact dynamics, we must calculate  $\bar{C}$  based on the value of  $\alpha_i$  *after* the impact, i.e.  $\alpha_i^+ = \tilde{\alpha}(i)$ , where  $\tilde{\alpha}(i)$  is given by (15).

The post-impact velocity and the impulsive constraint forces are complementary, i.e. we have that

$$\bar{C}(q, \alpha^+) \dot{q}^+ \geq 0, \quad \lambda \geq 0, \quad \lambda^T \bar{C}(q, \alpha^+) \dot{q}^+ = 0 \quad (34)$$

Solving (33) for  $\dot{q}^+$  and premultiplying by  $\bar{C}(q, \alpha^+)$  gives

$$\bar{C} \dot{q}^+ = \bar{C} \dot{q}^- + \bar{C} M^{-1} \bar{C}^T \lambda \quad (35)$$

Denoting the normal direction velocities of each of the contacted links by the vector  $\bar{v}_n \in \mathbb{R}^m$ , we may combine (34) and (35) into the following LCP describing the impact dynamics of the snake robot:

$$\begin{aligned} \bar{v}_n^+ &= \bar{v}_n^- + \bar{C} M^{-1} \bar{C}^T \lambda \\ \bar{v}_n^+ &\geq 0, \quad \lambda \geq 0, \quad \lambda^T \bar{v}_n^+ = 0 \end{aligned} \quad (36)$$

The LCP in (36) is in the general form of the LCP given in (2) with  $A = \bar{C} M^{-1} \bar{C}^T$  and  $a = \bar{v}_n^-$ , and must be solved

for the unknowns  $\bar{v}_n^+$  and  $\lambda$ . Subsequently, the post-impact velocity of the snake robot is found by solving (33) for  $\dot{q}^+$  and inserting the calculated  $\lambda$ . The following result concerns the existence and uniqueness properties of the LCP in (36):

*Proposition 15:* The LCP in (36) always possesses a unique solution  $(\bar{v}_n^+, \lambda)$ .

*Proof:* The proof is identical to the proof of Proposition 12. ■

This subsection is now summarized. The *discrete impact dynamics* of the snake robot when link  $i$  impacts an obstacle and the state immediately before the impact is  $(q^-, \dot{q}^-, \alpha^-)$ , is given by

$$\begin{aligned} \alpha_j^+ &= \begin{cases} \tilde{\alpha}(i), & j = i \\ \alpha_j^-, & j \neq i \end{cases} \\ \dot{q}^+ &= \dot{q}^- + M^{-1}(q) \bar{C}^T(q, \alpha^+) \lambda \end{aligned} \quad (37)$$

where  $j \in \{1, \dots, n\}$ ,  $\tilde{\alpha}(i)$  is given by (15), and  $\lambda$  is calculated from the LCP in (36).

## V. HYBRID MODEL OF A SNAKE ROBOT WITH OBSTACLES

In this section, we employ the framework of a *hybrid dynamical system* (see Section II-A) in order to encapsulate the *continuous* and the *discrete* dynamics of the snake robot into a single *hybrid* model. The *jump set* ( $D$ ), *jump map* ( $G$ ), *flow set* ( $C$ ), and *flow map* ( $F$ ) of the model are presented in the first four subsections, respectively, followed by a summary of the hybrid model in the last subsection. We define the state vector of the hybrid model as

$$x := (q, \dot{q}, \alpha) \in \mathbb{R}^{3n+4} \quad (38)$$

### A. Jump set

A *jump* in the state vector  $x$  of the model occurs when a link *impacts* an obstacle (jump in  $\dot{q}$  and  $\alpha$ ) or when a link *detaches* from an obstacle (jump in  $\alpha$ ). By employing the notation from Section IV-B, the *jump set* corresponding to an *impact* between link  $i$  and an obstacle is expressed as

$$D_{L_i}^{\text{Impact}} = \{x | L_i \cap O_j \neq \emptyset, j \in \{1, \dots, k\}, \tilde{\alpha}(i) C_i \dot{q} < 0\} \quad (39)$$

where  $\tilde{\alpha}(i)$  is given by (15) and  $C_i$  denotes the  $i$ th row of the matrix  $C(q)$  in (19). We use  $\tilde{\alpha}(i)$  instead of  $\alpha_i$  in (39) because  $\alpha_i = 0$  before the impact has taken place. The *jump set* corresponding to link  $i$  *detaching* from an obstacle may be expressed as

$$D_{L_i}^{\text{Detach}} = \{x | L_i \cap O_j = \emptyset, j \in \{1, \dots, k\}, \alpha_i \neq 0\} \quad (40)$$

The jump sets comprising the *impacts* and the *detachments* of all the links, respectively, are given by

$$D^{\text{Impact}} = \bigcup_{i \in \{1, \dots, n\}} D_{L_i}^{\text{Impact}}, \quad D^{\text{Detach}} = \bigcup_{i \in \{1, \dots, n\}} D_{L_i}^{\text{Detach}} \quad (41)$$

The complete *jump set* of the hybrid model may now be compactly expressed as

$$D = D^{\text{Impact}} \cup D^{\text{Detach}} \quad (42)$$

### B. Jump map

The *jump map* corresponding to the *impact* between link  $i$  and an obstacle is presented in (37). The *jump map* corresponding to link  $i$  *detaching* from an obstacle involves simply setting  $\alpha_i$  to zero. We may therefore express the complete *jump map* of the model as

$$x^+ = G(x) = (q^+, \dot{q}^+, \alpha^+) \quad \text{for all } x \in D \quad (43)$$

where

$$\alpha_i^+ = \begin{cases} \tilde{\alpha}(i) & \text{when } x \in D_{L_i}^{\text{Impact}} \\ 0 & \text{when } x \in D_{L_i}^{\text{Detach}} \\ \alpha_i^- & \text{when } x \notin \left( D_{L_i}^{\text{Impact}} \cup D_{L_i}^{\text{Detach}} \right) \end{cases}$$

$$\dot{q}^+ = \begin{cases} \dot{q}^- + M^{-1}(q) \bar{C}^T(q, \alpha^+) \lambda & \text{when } x \in D^{\text{Impact}} \\ \dot{q}^- & \text{when } x \notin D^{\text{Impact}} \end{cases} \quad (44)$$

The value of  $\tilde{\alpha}(i)$  is given by (15) and  $\lambda$  is calculated from the LCP in (36).

### C. Flow set

We define the *flow set* of the model so that the state vector  $x$  always flows as long as the *jump set* is empty. The *flow set* is therefore simply given as

$$C = \{x | x \notin D\} \quad (45)$$

### D. Flow map

The *flow map* of  $q$  is simply  $\dot{q}$  and the *flow map* of  $\dot{q}$  is given by (31). The contact vector,  $\alpha$ , remains unchanged between jumps of  $x$ . The *flow map* of  $\alpha$  is therefore the zero vector. The complete *flow map* of the model is given by

$$\dot{x} = F(x, u) = (\dot{q}, \ddot{q}, 0_{n \times 1}) \quad \text{for all } x \in C \quad (46)$$

where

$$\ddot{q} = M^{-1} \left( f_u(q, \dot{q}, u) + \left( \bar{C}^T - \mu_o \Lambda \right) \lambda \right) \quad (47)$$

and  $\lambda$  is calculated from the LCP in (32). Note that we have not yet specified the control input,  $u$ , i.e. the torques applied to the joints of the snake robot.

### E. Summary of the complete hybrid model

In accordance with Section II-A, the complete hybrid model is given by

$$\begin{aligned} \dot{x} &= F(x, u) & \text{for all } x \in C \\ x^+ &= G(x) & \text{for all } x \in D \end{aligned} \quad (48)$$

The following result summarizes the existence and uniqueness properties of the model.

*Proposition 16:* The evolution of the state  $x$  of the model in (48) from any initial state can always be uniquely determined when the obstacles are frictionless ( $\mu_o = 0$ ). With obstacle friction, there exists a  $\mu_o^* > 0$  such that existence and uniqueness of the evolution of  $x$  is guaranteed for  $\mu_o \in [0, \mu_o^*)$ , but not guaranteed for  $\mu_o \geq \mu_o^*$ .

*Proof:* From (45), the *flow* and *jump set* are mutually exclusive, so we can always uniquely determine whether  $x$  should *flow* or *jump*. By Proposition 15, the *jump map* of  $x$  is always unique. By Proposition 12, the *flow map* of  $x$  is always unique with frictionless obstacles. By Proposition 13, the *flow map* of  $x$  is always unique for some  $\mu_o = \mu_o^* > 0$ . This completes the proof. ■

## VI. CONTROL OF OBSTACLE-AIDED LOCOMOTION

The authors have previously presented a control strategy for obstacle-aided locomotion [15] that will be employed during the simulation of the snake robot in Section VII. Due to space limitations, we only summarize the basic idea behind the controller.

A major challenge during obstacle-aided locomotion is to prevent the snake robot from being *jammed* between the obstacles. In a jammed situation, the *propulsive* components of the obstacle contact forces are too small to overcome the friction forces, and hence the forward motion of the snake robot stops. The control strategy proposed in [15] consists of a *leader-follower* scheme and a *jam resolution* scheme, and also a supervisory mechanism for switching between these two schemes. The *leader-follower* scheme, whose single goal is to propel the snake robot along the global positive  $x$  axis, is carried out as long as the snake robot is able to move without being jammed between the obstacles. If the snake robot is jammed, the *jam resolution* scheme is carried out in order to effectively ‘unlock’ the jammed joints. The complete controller produces the torque input vector,  $u \in \mathbb{R}^{n-1}$ , for the  $n-1$  joints of the snake robot. See [15] for more details about the controller.

## VII. SIMULATION RESULTS

The main purpose of the simulation results presented in this section is to validate the hybrid modelling approach, but also to show how the controller described in Section VI performs with this hybrid model. The model and the controller were implemented and simulated in *Matlab R2008b* on a laptop running *Windows XP*. The *ode45* solver in Matlab was employed with a relative and absolute error tolerance of  $10^{-3}$ .

The parameters characterizing the simulated snake robot were  $n = 10$ ,  $l = 0.07$  m,  $m = 1$  kg, and  $J = 0.0016$  kgm<sup>2</sup>. The various controller parameters were set according to the controller described in [15]. The ground and obstacle friction coefficients were set to  $\mu = 0.3$  and  $\mu_o = 0.2$ , respectively. The initial link angles and position of the snake robot were set to  $\theta_0 = [7, -32, -57, -46, -8, 33, 53, 45, 12, -23]^T$  [deg] and  $p_0 = [0, 0]^T$ , respectively. At each timestep of the simulation, we verified that the solution to the LCP in (32) was unique by employing the  $P$ -matrix test presented in [20].

The motion of the snake robot was simulated both in a *structured* and *unstructured* obstacle environment. The initial ( $t = 0$  s) and final ( $t = 30$  s) shape and position of the snake robot in these two environments are shown in Fig. 5, while a plot of the  $x$  direction velocity of the snake,  $\dot{p}_x$ , is shown in Fig. 6. We see that the velocity of the snake robot varies around  $10 \text{ cm/s}$  in both environments and that the snake robot manages to crawl about 2.5 m along the global  $x$  axis.

In order to give an idea of the forces involved in obstacle-aided locomotion, Fig. 7 provides a plot of the constraint forces on the center link (link 5) of the snake robot,  $f_{c,5}$ , and the torque input applied to joint 5,  $u_5$ , during motion in the structured environment.

## VIII. CONCLUSIONS AND FUTURE WORK

This paper has presented a *hybrid model* of the dynamics of a planar snake robot interacting with obstacles in its

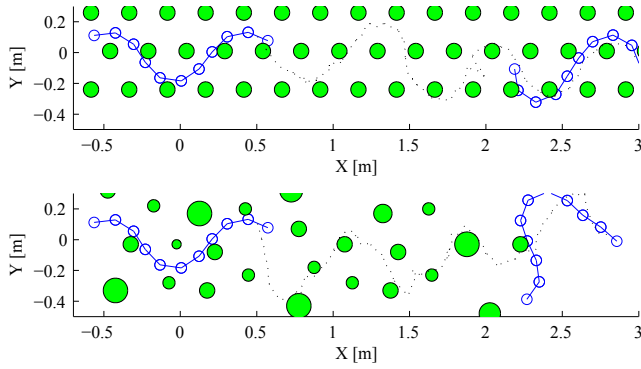


Fig. 5. The initial ( $t = 0$  s) and final ( $t = 30$  s) shape and position of the snake robot during motion in the structured (top) and unstructured (bottom) obstacle environment. The trace of the head is indicated with a dotted line.

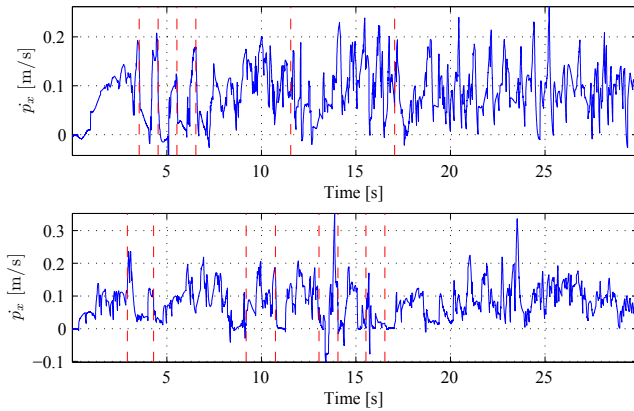


Fig. 6. The global  $x$  direction velocity of the snake robot during motion in the structured (top) and unstructured (bottom) obstacle environment. Vertical dashed lines indicate time instants where the *jam resolution* scheme is initiated.

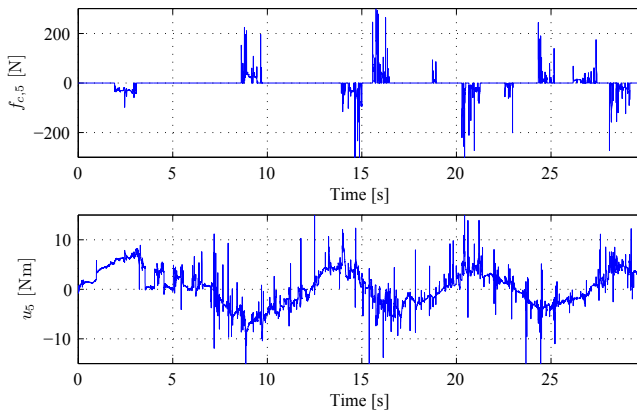


Fig. 7. The constraint forces on link 5 (top) and the torque input applied to joint 5 (bottom) during motion in the structured obstacle environment.

environment where obstacle contact forces are calculated by formulating and solving a *linear complementarity problem* (LCP). The *existence* and *uniqueness* properties of the state evolution of the hybrid model were considered. In particular, it was shown that the evolution of the state  $x$  of the model from any initial state can always be uniquely determined when the obstacles are frictionless ( $\mu_o = 0$ ). With obstacle friction, we showed that there exists a  $\mu_o^* > 0$  such that existence and uniqueness of the evolution of  $x$  is guaranteed for  $\mu_o \in [0, \mu_o^*)$ , but not guaranteed for  $\mu_o \geq \mu_o^*$ .

In future work, the authors will use the hybrid model in order to synthesize model-based controllers for obstacle-aided locomotion with provable stability properties.

## REFERENCES

- [1] A. A. Transth, R. I. Leine, C. Glocker, K. Y. Pettersen, and P. Liljebäck, "Snake robot obstacle aided locomotion: Modeling, simulations, and experiments," *IEEE Trans. Robot.*, vol. 24, no. 1, pp. 88–104, February 2008.
- [2] J. Gray, "The mechanism of locomotion in snakes," *J. Exp. Biol.*, vol. 23, no. 2, pp. 101–120, 1946.
- [3] S. Hirose, *Biologically Inspired Robots: Snake-Like Locomotors and Manipulators*. Oxford: Oxford University Press, 1993.
- [4] B. Moon and C. Gans, "Kinematics, muscular activity and propulsion in gopher snakes," *Journal of Experimental Biology*, vol. 201, pp. 2669–2684, 1998.
- [5] P. Prautsch and T. Mita, "Control and analysis of the gait of snake robots," in *Proc. IEEE Int. Conf. Control Applications*, Kohala Coast, HI USA, 1999, pp. 502–507.
- [6] H. Date, M. Sampei, and S. Nakaura, "Control of a snake robot in consideration of constraint force," in *Proc. IEEE Int. Conf. on Control Applications*, 2001, pp. 966–971.
- [7] S. Ma, "Analysis of creeping locomotion of a snake-like robot," *Adv. Robotics*, vol. 15, no. 2, pp. 205–224, 2001.
- [8] F. Matsuno and H. Sato, "Trajectory tracking control of snake robots based on dynamic model," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2005, pp. 3029–3034.
- [9] M. Saito, M. Fukaya, and T. Iwasaki, "Serpentine locomotion with robotic snakes," *IEEE Contr. Syst. Mag.*, vol. 22, no. 1, pp. 64–81, February 2002.
- [10] K. McIsaac and J. Ostrowski, "Motion planning for anguilliform locomotion," *IEEE Trans. Robot. Autom.*, vol. 19, no. 4, pp. 637–625, August 2003.
- [11] G. P. Hicks, "Modeling and control of a snake-like serial-link structure," Ph.D. dissertation, North Carolina State University, 2003.
- [12] P. Liljebäck, Ø. Stavdahl, and K. Y. Pettersen, "Modular pneumatic snake robot: 3D modelling, implementation and control," in *Proc. 16th IFAC World Congress*, July 2005.
- [13] Z. Bayraktaroglu and P. Blazevic, "Understanding snakelike locomotion through a novel push-point approach," *J. Dyn. Syst. - Trans. ASME*, vol. 127, no. 1, pp. 146–152, March 2005.
- [14] I. Tanev, T. Ray, and A. Buller, "Automated evolutionary design, robustness, and adaptation of sidewinding locomotion of a simulated snake-like robot," *IEEE Trans. on Robotics*, vol. 21, no. 4, pp. 632–645, August 2005.
- [15] P. Liljebäck, K. Y. Pettersen, and Ø. Stavdahl, "Modelling and control of obstacle-aided snake robot locomotion based on jam resolution," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2009, pp. 3807–3814.
- [16] B. Brogliato, *Nonsmooth Mechanics*, 2nd ed. London: Springer, 1999.
- [17] R. W. Cottle, J. S. Pang, and R. E. Stone, *The linear complementarity problem*. Academic Press, 1992.
- [18] A. J. v. d. Schaft and J. M. Schumacher, *An Introduction to Hybrid Dynamical System*, M. Thoma and M. Morari, Eds. Springer, 2000.
- [19] R. Goebel, R. Sanfelice, and A. Teel, "Hybrid dynamical systems," *IEEE Control Systems Magazine*, vol. 29, no. 2, pp. 28–93, April 2009.
- [20] M. J. Tsatsomeros and L. Li, "A recursive test for P-matrices," *Bit Numerical Mathematics*, vol. 40, pp. 410–414, 2000.
- [21] H. Goldstein, C. Poole, and J. Safko, *Classical Mechanics - Third Edition*. Addison Wesley, 2002.
- [22] P. Lötstedt, "Mechanical systems of rigid bodies subject to unilateral constraints," *SIAM J. Appl. Math.*, vol. 42, no. 2, pp. 281–296, April 1982.