Snake Robot Obstacle Aided Locomotion: Modeling, Simulations, and Experiments

Aksel A. Transeth, Remco I. Leine, Christoph Glocker, Kristin Y. Pettersen, and Pål Liljebäck

Abstract—**Snakes utilize irregularities in the terrain, such as rocks and vegetation, for faster and more efficient locomotion. This motivates the development of snake robots that actively use the terrain for locomotion, i.e. obstacle aided locomotion. In order to accurately model and understand this phenomenon, this paper presents a novel non-smooth (hybrid) mathematical model for wheel-less snake robots, which allows the snake robot to push against external obstacles apart from a flat ground. The framework of non-smooth dynamics and convex analysis allows us to systematically and accurately incorporate both unilateral contact forces (from the obstacles) and isotropic friction forces based on Coulomb's law using set-valued force laws. The mathematical model is verified through experiments. In particular, a back-to-back comparison between numerical simulations and experimental results is presented. It is furthermore shown that the snake robot is able to move forward faster and more robustly by exploiting obstacles.**

Index Terms—**Biomimetics, snake robot, time-stepping method, non-smooth dynamics, bio-inspired locomotion.**

I. Introduction

THE FASTEST biological snakes exploit roughness in the terrain for locomotion. They may push against rocks, branches, or other obstacles to move forward more efficiently [1], [2]. Snakes can also exploit the walls of narrow passages for locomotion. Their properties of mobility are interesting for snake robot locomotion and gives the motivation for investigating snake robot obstacle aided locomotion. In particular, if robots can be made able to traverse difficult terrain at a reasonable speed, they can be utilized in search and rescue missions in challenging environments such as collapsed buildings in earthquaked areas, or as inspection and intervention robots in possibly hazardous environments of industrial plants. This is the motivation for developing snake robots that exploit obstacles for locomotion. We define *obstacle aided locomotion* as snake robot locomotion where the snake robot utilizes walls or other external objects, apart from the flat ground, for means of propulsion. In order to develop such an obstacle aided locomotion scheme we need a mathematical model that includes the interaction between the snake robot and a terrain with external objects for the snake robot to push against.

Fig. 1. The NTNU/SINTEF snake robot 'Aiko'.

Research on snake robots has increased vastly during the past ten to fifteen years, and the published literature has mostly focused on snake robot modeling and locomotion. The fastest and most common serpentine motion pattern used by biological snakes is called lateral undulation. This has also been the most implemented motion pattern for snake robots [3]. Snakes exploit irregularities in the terrain to push against to move forwards by lateral undulation. This method of locomotion is attempted to be recreated for snake robots by adding passive caster wheels [4]–[8] or metals [9] on the underside of the snake robot body. However, these approaches result in that locomotion speed and efficiency are very dependent on the ground surface, and relatively fast locomotion is only obtained for snake robots with caster wheels on a solid smooth surface. For snake robots without wheels, the friction between the snake robot underside and the ground is important when moving by lateral undulation. This is because the friction property of the snake robot links must be such that the links slide easier forwards and backwards than sideways for efficient snake robot locomotion by lateral undulation. Hence, the friction model must be accurate to recreate the snake robot motion in simulations. The friction force between snake robots without wheels and the ground has been described by Coulomb friction and modeled using sign-functions in [7], [9]. However, only unidirectional Coulomb friction can be described by a sign-function (see Section V-B2).

The dependency on the ground surface can be relaxed by mimicking biological snakes and utilizing external objects to move forward. Obstacle aided locomotion for snake robots was first investigated by Hirose in 1976 and experiments with a snake robot with passive caster wheels moving through a

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors. This includes one multimedia MPEG format movie clip, which shows the snake robot Aiko during obstacle aided locomotion. This material is 11 MB in size.

A. A. Transeth and K. Y. Pettersen are with the Department of Engineering Cybernetics at the Norwegian University of Science and Technology, NO-7491 Trondheim, Norway. Fax: +47 73 59 45 99 (e-mail: Aksel.Andreas.Transeth,Kristin.Y.Pettersen@itk.ntnu.no).

R. I. Leine and Ch. Glocker are with the IMES-Center of Mechanics, ETH Zürich, CH-8092 Zürich, Switzerland (e-mail: Remco.Leine,Christoph.Glocker@imes.mavt.ethz.ch).

P. Liljebäck is with the Department of Applied Cybernetics at SINTEF IKT, NO-7465 Trondheim, Norway (e-mail: Pal.Liljeback@sintef.no).

winding track has been presented in [2], [10]. More recently, obstacle aided locomotion has been elaborated on for wheel-less snake robots [11], [12]. The dynamics of such locomotion has been simulated with the dynamic simulation software WorkingModel® in [11] where the rigid body obstacle contact is represented by a spring-damper model. A very high spring coefficient is needed to model a hard obstacle. In addition, it is not clear how to determine the dissipation parameters of the contact unambiguously when using a compliant model [13]. Moreover, a steep and smooth approximation of the sign-function together with the compliant contact model lead to stiff differential equations which are cumbersome to solve numerically. Hence, there is a need for a non-smooth model which correctly describes spatial Coulomb friction with stiction as well as the unilaterality of the obstacle contact.

This paper presents a non-smooth (hybrid) modeling approach particulary suitable for modeling snake robot obstacle aided locomotion. We use this approach to develop a 2D mathematical model of a snake robot that can push against external objects for locomotion. The mathematical model is described in the framework of non-smooth dynamics and convex analysis [13]–[15], which allows us to easily incorporate both the unilateral contact forces from the obstacles and the friction forces between the snake robot and the ground based on Coulomb's law using set-valued force laws (see Section V). Hence, stick-slip transitions with the ground and impacts with the obstacles are modeled as instantaneous transitions. This results in an accurate description of the Coulomb friction which is important for snake robot locomotion on a planar surface. Even though we model the snake robot as a hybrid system, we avoid an explicit switching between system equations (for example when a collision occurs) in this framework. Hence, this approach is advantageous for modeling obstacle aided locomotion in which the snake robot repeatedly collides with the obstacles. Furthermore, the model is verified through experiments. In particular, we present a back-to-back comparison between simulation and experimental results. The experiments were performed using the snake robot 'Aiko' in Fig. 1, which is a wheel-less snake robot with cylindrical links recently developed at the NTNU/SINTEF Advanced Robotics Laboratory. To the authors' best knowledge, this is the first time a mathematical model of the dynamics of a snake robot during obstacle aided locomotion has been developed and experimentally validated for an actual snake robot without wheels.

The paper is organized as follows. Section II gives a short introduction and motivation for the derivation of the mathematical model. The kinematics of the snake robot with obstacles is given in Section III. Section IV lays the foundation for finding the obstacle contact and ground friction forces. Section V describes the non-smooth dynamics of the snake robot, while Section VI outlines the numerical treatment of the model. A note on obstacle aided locomotion is given in Section VII. Simulations and experimental validations are given in Section VIII, and conclusions and suggestions for future work can be found in Section IX.

II. Introduction to the Mathematical Model

This section contains a brief outline of how to derive a non-smooth mathematical model of a snake robot. This preliminary section is meant to motivate and ease the understanding of the forthcoming deduction of the system equations.

The planar model of the snake robot consists of n links connected by $n-1$ one-degree-of-freedom (DOF) rotational joints. Let $\boldsymbol{u} \in \mathbb{R}^{3n}$ be a vector containing the translational and rotational velocities of all the links of the snake robot (the structure of the snake robot together with the coordinates and reference frames are described further in Section III). Let the differential measures $\mathrm{d}\boldsymbol{u}$ and $\mathrm{d}t$ be loosely described for now as a 'possible differential change' in $\boldsymbol{u}$ and time t, respectively, while a more precise definition is given in Section V. The use of differential measures allows for instantaneous changes in velocities which occur for impacts with obstacles. The system equations for the snake robot can now be written as

$$\mathbf{M}\mathrm{d}\boldsymbol{u} - \mathrm{d}\boldsymbol{R} = \boldsymbol{\tau}_C \mathrm{d}t, \tag{1}$$

which is called the *equality of measures* [16], where $\mathbf{M} \in \mathbb{R}^{3n\times 3n}$ is the mass matrix, $\boldsymbol{\tau}_C \in \mathbb{R}^{3n}$ contains all the joint actuator torques, and $\mathrm{d}\boldsymbol{R} \in \mathbb{R}^{3n}$ accounts for the normal contact forces/impulses from the obstacles, the Coulomb friction forces/impulses, and the bilateral constraint forces/impulses in the joints. Note: We allow in this paper for *instantaneous* changes in velocities usually associated with collisions. Hence, the (normal contact/friction/constraint) 'forces' are not always defined due to the infinite accelerations. In these cases we have impulses instead of forces. The non-smooth equality of measures (1) allows us to formulate in a uniform manner both the smooth and non-smooth phases of motion. This is achieved partly by representing the contact forces/impulses as *contact impulse measures*.

A substantial part of the beginning of this paper is devoted to deducting the force measure $\mathrm{d}\boldsymbol{R}$. Hence, let us briefly look at how to derive the contribution of the normal contact impulse measure between an obstacle j and the first link, in $\mathrm{d}\boldsymbol{R}$. Let $\mathrm{d}\boldsymbol{R}_1 \in \mathbb{R}^3$ be the sum of contact impulse measures (i.e. the representation of the contact forces and impulses) that directly affects link 1 (i.e the first three elements of $\mathrm{d}\boldsymbol{R}$), then

$$\mathrm{d}\boldsymbol{R}_1 = \boldsymbol{w}_H\, \mathrm{d}P_H + \left\{ \begin{array}{l} \text{friction and joint constraint} \\ \text{impulse measures} \end{array} \right\}, \tag{2}$$

where $\mathrm{d}P_H \in \mathbb{R}$ is the normal contact impulse measure from obstacle j on link 1, and $\boldsymbol{w}_H \in \mathbb{R}^3$ is the corresponding generalized force direction, i.e. a Jacobian (subscripts 'j' and '1' are omitted for brevity).

Let $g_H \in \mathbb{R}$ be a function giving the shortest distance between link 1 and the obstacle. Such a function is called a *gap function* [17] (see Fig. 3 for an illustration of the gap function between link i and obstacle j). The gap function is the starting point for the systematic approach of finding the impulse measures. The link and obstacle are separated if $g_H > 0$, are in contact if $g_H = 0$ and are penetrating each other if $g_H < 0$. Hence, the relative velocity between link 1 and the obstacle along the shortest line between the two objects can be defined as

$$\gamma_H := \dot{g}_H = \boldsymbol{w}_H^{\mathrm{T}} \boldsymbol{u}_1, \tag{3}$$

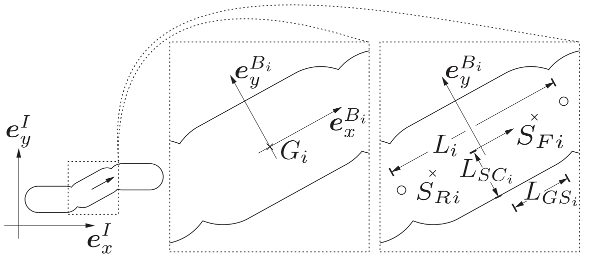


Fig. 2. Snake robot with three links.

where $\boldsymbol{u}_1$ is the velocity of link 1, and we have also shown how to find $\boldsymbol{w}_H$ in (2) which gives the direction of the generalized force acting on the link from the obstacle. The normal contact impulse measure $\mathrm{d}P_H$ is found from the relative velocity γ_H by employing a *set-valued force law* (see Section V-B1). The set-valuedness of the force laws allows us to write each constitutive law (force law) with a single equation and avoid explicit switching between equations (for example when a collision occurs). In addition, this formulation provides an accurate description of planar Coulomb friction (see Section V-B2). One reason for this is because the set-valued force law allows for a non-zero value of the friction force even though a body is not moving (i.e. during the stick-phase). This is not the case when using sign-functions to model Coulomb friction since a sign-function will result in a zero friction force for zero velocity. In the following three sections, we will elaborate on how to derive the elements that constitute (1), that is the various gap functions, relative velocities, and finally the forces/impulses included in the equality of measures.

III. Kinematics

In this section, we present the kinematics of the snake robot in an environment with obstacles. From the kinematics, we develop *gap functions* both for obstacle contact detection and for conforming to the bilateral joint constraints. The gap functions are later used as a basis for calculating the normal contact forces and the joint constraint forces.

First, we give an overview of the coordinates used to describe the position and orientation of the snake robot. The coordinates are chosen such that the mass matrix becomes constant. This is advantageous for the numerical treatment. Subsequently, the gap functions for describing the distance between the snake robot and the obstacles, and the ‘gaps’ in the joints, are found.

A. Snake Robot Description and Reference Frames

The snake robot model presented in this paper consists of n equal links connected by $n-1$ one-degree-of-freedom rotational joints. A snake robot with 3 links is depicted in Fig. 2. Link i is shaped as a rectangle of length $2L_{GS_i}$ and width $2L_{SC_i}$. It is assumed that there is a massless semicircle of radius L_{SC_i} connected to the ends of each link. The semicircles together with the straight sides define the surface (or contour) of a link used for contact with the external objects. The distance between two adjacent joints is L_i.

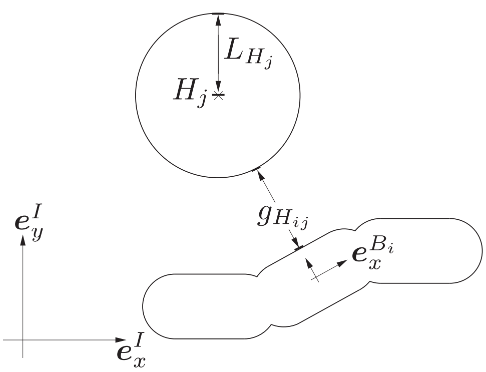


Fig. 3. Snake robot with obstacle j close to link i.

Let the inertial reference frame be approximated by an earth-fixed frame $I = \{O, \boldsymbol{e}_x^I, \boldsymbol{e}_y^I\}$ with the origin O attached to the ground surface. A general notation used throughout this paper is that a vector from the origin of frame I to a point A is given by $\boldsymbol{r}_A \in \mathbb{R}^2$ and a vector from point A to point B is written $\boldsymbol{r}_{AB}$. Let a vector $\boldsymbol{r}_A$ described in frame I be written as ${}_I\boldsymbol{r}_A$. Denote the point G_i as the centre of gravity (CG) of link i where the CG is assumed to be in the middle of the rectangle. Denote the body-fixed frame $B_i = \{G_i, \boldsymbol{e}_x^{B_i}, \boldsymbol{e}_y^{B_i}\}$, where the origin is at the point G_i, the axis $\boldsymbol{e}_x^{B_i}$ is pointing along the centre line of link i towards link $i+1$, and $\boldsymbol{e}_y^{B_i}$ is pointing transversal to the link following the right-hand convention. The centre point of the front and rear sphere that constitute the ends of link i is denoted S_{Fi} and S_{Ri}, respectively.

The position and orientation of link i is

$$\boldsymbol{q}_i = \begin{bmatrix} {}_I\boldsymbol{r}_{G_i} \\ \theta_i \end{bmatrix}, \tag{4}$$

where ${}_I\boldsymbol{r}_{G_i} \in \mathbb{R}^2$ is the position of the CG of link i and $\theta_i \in S^1$ is the angle between the axes $\boldsymbol{e}_x^I$ and $\boldsymbol{e}_x^{B_i}$. The velocity of link i is

$$\boldsymbol{u}_i = \begin{bmatrix} {}_I\boldsymbol{v}_{G_i} \\ \omega_i \end{bmatrix}, \tag{5}$$

where ${}_I\boldsymbol{v}_{G_i} := {}_I\dot{\boldsymbol{r}}_{G_i}$ and $\omega_i := \dot{\theta}_i$ when they exist (i.e. for impact free motion). All the positions and orientations, and velocities, are gathered in the two vectors

$$\boldsymbol{q} = \begin{bmatrix} \boldsymbol{q}_1 \\ \boldsymbol{q}_2 \\ \vdots \\ \boldsymbol{q}_n \end{bmatrix}, \text{ and } \boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \\ \vdots \\ \boldsymbol{u}_n \end{bmatrix}. \tag{6}$$

The transformation of a vector $\boldsymbol{r}$ between reference frames is given by ${}_I\boldsymbol{r} = \mathbf{R}_{B_i}^I \, {}_{B_i}\boldsymbol{r}$ where the rotation matrix is

$$\mathbf{R}_{B_i}^I = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix}. \tag{7}$$

External fixed cylinders are included in the model as objects for the snake robot to push against. Each obstacle j, $j = 1, 2, \ldots, \nu$, is shaped as a circle in the plane with radius L_{H_j} and midpoint H_j, see Fig. 3.

B. Obstacle Contact Detection

The gap function $g_{H_{ij}}$ gives the shortest distance between link i and obstacle j as illustrated in Fig. 3 and is used to detect whether the two bodies are in contact. In addition, the gap function will be used later as a basis for calculating the forces involved in the contact.

Link i can, at any time-instance, only touch a convex obstacle j at a single point, resulting in a contact point that may move across the entire surface of a link. The surface (or contour) of a link consists of three parts: the rectangle, the front semicircle and the rear semicircle. Hence, we see that we need to find two separate kinds of gap functions for each link depending on which part of the link is closest to the obstacle. In particular, we need to find the shortest distance between a rectangle and a circle, and between two circles. This approach provides us with a *moving* contact point. Hence, the obstacle can come in contact with the snake robot at any point and the effect the contact force has on a snake robot link is therefore modeled accurately.

1) Distance Between Rectangle Part and Obstacle: When the rectangle-shaped part of link i is closest to obstacle j, we need to investigate the distance between a line (the $\boldsymbol{e}_x^{B_i}$-axis) and the centre of the obstacle H_j. The shortest distance between the $\boldsymbol{e}_x^{B_i}$-axis and H_j is

$$d_{ij} = \left({}_I\boldsymbol{r}_{G_i} - {}_I\boldsymbol{r}_{Hj}\right)^{\mathrm{T}}\ {}_I\boldsymbol{e}_y^{B_i}. \tag{8}$$

Hence, the gap function when the rectangle part of link i is closest to obstacle j is

$$g_{H_{ij}} = |d_{ij}| - \left(L_{H_j} + L_{SC_i}\right). \tag{9}$$

2) Distance Between Circle part and Obstacle: The minimal distance between the circle part of link i and obstacle j is found from the gap function

$$g_{H_{ij}} = \left\|{}_I\boldsymbol{r}_{H_jS_i}\right\| - \left(L_{H_j} + L_{SC_i}\right), \tag{10}$$

where $\boldsymbol{r}_{H_jS_i} = \boldsymbol{r}_{G_i} + \boldsymbol{r}_{G_iS_{Fi}} - \boldsymbol{r}_{H_j}$ or $\boldsymbol{r}_{H_jS_i} = \boldsymbol{r}_{G_i} + \boldsymbol{r}_{G_iS_{Ri}} - \boldsymbol{r}_{H_j}$ depending on whether the front or rear part of the link is closest to the obstacle, respectively. In addition, $\boldsymbol{r}_{G_iS_{Fi}} = L_{GS_i}\boldsymbol{e}_x^{B_i}$, $\boldsymbol{r}_{G_iS_{Ri}} = -L_{GS_i}\boldsymbol{e}_x^{B_i}$, and we define $\|\cdot\| := \|\cdot\|_2$.

3) A Vector of All Obstacle Gap functions: We now gather the gap functions $g_{H_{ij}}$ for the combination of all n links and ν obstacles in the vector:

$$\boldsymbol{g}_H = \left[g_{H_{11}} \cdots g_{H_{n1}} g_{H_{12}} \cdots g_{H_{n2}} \cdots g_{H_{1\nu}} \cdots g_{H_{n\nu}}\right]^{\mathrm{T}}, \tag{11}$$

where $g_{H_{ij}}$ is found from either (9) or (10) depending on which part of link i is closest to obstacle j.

C. Bilateral Constraints - Joints

Each joint introduces two bilateral constraints in the model. These constraints keep the 'gap' in a joint equal to zero. An expression for this 'gap', a gap function, needs to be found in order to calculate the constraint forces. To find the gap function, we need to relate the position of joint i between link i and $i+1$ to both adjacent links. By inspecting Fig. 4, we see that the position of joint i going via link i and $i+1$ is given by ${}_I\boldsymbol{r}_{J_{Fi}} = {}_I\boldsymbol{r}_{G_i} + \frac{1}{2}L_i\,{}_I\boldsymbol{e}_x^{B_i}$ and ${}_I\boldsymbol{r}_{J_{Ri+1}} = {}_I\boldsymbol{r}_{G_{i+1}} - \frac{1}{2}L_{i+1}\,{}_I\boldsymbol{e}_x^{B_{i+1}}$, respectively. Now, the gap functions for the gaps in the joints are

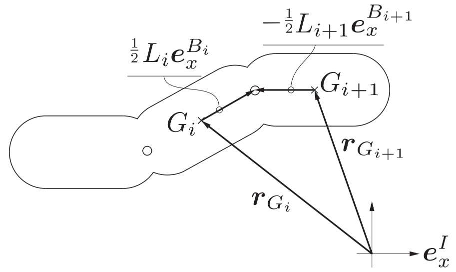


Fig. 4. Illustration of how the joint gap function $g_{J_{\chi i}}$ is found for $i = 2$. We see that the vectors are drawn for link 2 and 3 for the snake robot in the picture.

$$g_{J_{\chi i}} = \left({}_I\boldsymbol{e}_\chi^I\right)^{\mathrm{T}}\left({}_I\boldsymbol{r}_{J_{Fi}} - {}_I\boldsymbol{r}_{J_{Ri+1}}\right), \tag{12}$$

for $\chi = x, y$ and $i = 1, \ldots, n-1$.

IV. Contact Constraints on Velocity Level

In this section the relative velocities between the snake robot, and the obstacles and the ground, are found by taking the time-derivative (when they exist) of the corresponding gap functions. The relative velocities are used to calculate the normal contact forces involved in the contact between the snake robot and the obstacles [18], [19]. Also, the relative velocities are employed to find the friction forces between the ground and the snake robot, and the bilateral constraint forces in the joints. This is shown in Section V-B.

A. Relative Velocity Between an Obstacle and a Link

The relative velocity between a link i and an obstacle j is defined as $\gamma_{H_{ij}} := \dot{g}_{H_{ij}}$ (when the time-derivative exists). As for the gap functions, the expression for the relative velocity depends on which part of the link is closest to the obstacle.

1) Rectangle Part Closest to Obstacle: If the rectangle-shaped part of link i is closest to obstacle j, then the relative velocity between these two convex objects $\gamma_{H_{ij}} := \dot{g}_{H_{ij}}$ is found, by employing (9), as

$$\gamma_{H_{ij}} = \frac{\mathrm{d}}{\mathrm{d}t}|d_{ij}| \Longrightarrow \gamma_{H_{ij}} = \boldsymbol{w}_{H_{ij}}^{\mathrm{T}}\boldsymbol{u}_i, \tag{13}$$

where d_{ij} is given in (8) and the Jacobian is

$$\boldsymbol{w}_{H_{ij}}^{\mathrm{T}} = \mathrm{sign}\left(d_{ij}\right)\left[\left({}_I\boldsymbol{e}_y^{B_i}\right)^{\mathrm{T}}, \left({}_I\boldsymbol{r}_{H_j} - {}_I\boldsymbol{r}_{G_i}\right)^{\mathrm{T}}\begin{bmatrix}\cos\left(\theta_i\right)\\ \sin\left(\theta_i\right)\end{bmatrix}\right]\boldsymbol{u}_i. \tag{14}$$

for $i = 1, \ldots, n$ and $j = 1, \ldots, \nu$.

2) Circle Part Closest to Obstacle: The gap function (10) is employed to define the relative velocity $\gamma_{H_{ij}} := \dot{g}_{H_{ij}}$ when one of the circle-shaped parts of link i is closest to obstacle j. By definition, the relative velocity is

$$\gamma_{H_{ij}} = \frac{\mathrm{d}}{\mathrm{d}t}\left\|{}_I\boldsymbol{r}_{H_jS_i}\right\| \Longrightarrow \gamma_{H_{ij}} = \boldsymbol{w}_{H_{ij}}^{\mathrm{T}}\boldsymbol{u}_i, \tag{15}$$

where the Jacobian is, for $i = 1, \dots, n$ and $j = 1, \dots, \nu$,

$$\boldsymbol{w}_{H_{ij}}^{\mathrm{T}} = \frac{{}_I\boldsymbol{r}_{H_jS_i}}{\left\|{}_I\boldsymbol{r}_{H_jS_i}\right\|} \left[\mathbf{I}_{2\times 2} \quad \mathbf{R}_{B_i}^{I}\mathbf{D}\,{}_{B_i}\boldsymbol{r}_{G_iS_i}\right], \tag{16}$$

$$\mathbf{D} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{I}_{2\times 2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{17}$$

and ${}_I\boldsymbol{r}_{H_jS_i}$ depends on whether the front or rear part of the link is closest to the obstacle as described in Section III-B2. This is also the case for ${}_I\boldsymbol{r}_{G_iS_i}$ which is either equal to ${}_I\boldsymbol{r}_{G_iS_{Fi}}$ or ${}_I\boldsymbol{r}_{G_iS_{Ri}}$. Moreover, we have employed the identity

$$\dot{\mathbf{R}}_{B_i}^{I} = \mathbf{R}_{B_i}^{I}\mathbf{D} \tag{18}$$

to calculate (16).

3) Vector of All Relative Velocities for Obstacles: We gather the relative velocities $\gamma_{H_{ij}}$ for all n links and ν obstacles in the vector

$$\boldsymbol{\gamma}_H = \mathbf{W}_H^{\mathrm{T}}\boldsymbol{u} \in \mathbb{R}^{n\cdot\nu}, \tag{19}$$

where

$$\boldsymbol{\gamma}_H = \left[\gamma_{H_{11}} \cdots \gamma_{H_{n1}} \gamma_{H_{12}} \cdots \gamma_{H_{n2}} \cdots \gamma_{H_{1\nu}} \cdots \gamma_{H_{n\nu}}\right]^{\mathrm{T}}, \tag{20}$$

$$\mathbf{W}_H = \begin{bmatrix} \mathbf{W}_{H_1} & \cdots & \mathbf{W}_{H_\nu} \end{bmatrix} \in \mathbb{R}^{3n\times n\cdot\nu}, \tag{21}$$

and

$$\mathbf{W}_{H_j} = \begin{bmatrix} \boldsymbol{w}_{H_{1j}} & \mathbf{0}_{3\times1} & \cdots & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} & \ddots & & \vdots \\ \vdots & & \ddots & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} & \cdots & \mathbf{0}_{3\times1} & \boldsymbol{w}_{H_{nj}} \end{bmatrix} \in \mathbb{R}^{3n\times n}. \tag{22}$$

Depending on which part of the link is closest to the obstacle, $\boldsymbol{w}_{H_{ij}}$ is obtained either from (14) or (16).

B. Tangential Relative Velocity

The tangential relative velocity between link i and the ground is presented here. It will later be employed to find the friction forces between the link and the ground. Note that 'tangential' in this case refers to 'tangential to the ground surface' and not 'tangential to a link' (i.e. parallel to the link) which is a notation sometimes employed for snake robots.

We start by finding the tangential relative velocities parallel to the $\boldsymbol{e}_x^I$- and $\boldsymbol{e}_y^I$-axis. These are denoted for link i by $\gamma'_{T_{xi}}$ and $\gamma'_{T_{yi}}$, respectively. However, in order to easily find the friction forces longitudinal and lateral to each link, we need to derive the tangential relative velocities longitudinal $\gamma_{T_{xi}}$ and lateral $\gamma_{T_{yi}}$ to each link.

The tangential relative velocities parallel to the $\boldsymbol{e}_x^I$- and $\boldsymbol{e}_y^I$-axis are

$$\gamma'_{T_{\chi i}} = \left({}_I\boldsymbol{e}_\chi^I\right)^{\mathrm{T}} {}_I\boldsymbol{v}_{G_i} \Longrightarrow \gamma'_{T_{\chi i}} = \left(\boldsymbol{w}'_{T_{\chi i}}\right)^{\mathrm{T}}\boldsymbol{u}_i, \tag{23}$$

where the Jacobian is

$$\left(\boldsymbol{w}'_{T_{\chi i}}\right)^{\mathrm{T}} = \left[\left({}_I\boldsymbol{e}_\chi^I\right)^{\mathrm{T}} \quad 0\right], \tag{24}$$

for $\chi = x, y$ and $i = 1, \dots, n$. The two relative velocities are combined in one vector so that

$$\boldsymbol{\gamma}'_{T_i} = \left(\mathbf{W}'_{T_i}\right)^{\mathrm{T}}\boldsymbol{u}_i, \tag{25}$$

where

$$\boldsymbol{\gamma}'_{T_i} = \begin{bmatrix} \gamma'_{T_{xi}} \\ \gamma'_{T_{yi}} \end{bmatrix} \in \mathbb{R}^2, \ \mathbf{W}'_{T_i} = \begin{bmatrix} \boldsymbol{w}'_{T_{xi}} & \boldsymbol{w}'_{T_{yi}} \end{bmatrix} \in \mathbb{R}^{3\times2}. \tag{26}$$

The tangential relative velocities along the $\boldsymbol{e}_x^{B_i}$- and $\boldsymbol{e}_y^{B_i}$-axis can now be found as follows. Define $\boldsymbol{\gamma}_{T_i} = \begin{bmatrix} \gamma_{T_{xi}} & \gamma_{T_{yi}} \end{bmatrix}^{\mathrm{T}}$. Then, by employing the rotation matrix $\mathbf{R}_{B_i}^I$ in (7), we have that $\boldsymbol{\gamma}'_{T_i} = \mathbf{R}_{B_i}^I\,\boldsymbol{\gamma}_{T_i}$. Hence,

$$\boldsymbol{\gamma}_{T_i} = \mathbf{W}_{T_i}^{\mathrm{T}}\boldsymbol{u}_i, \tag{27}$$

where

$$\mathbf{W}_{T_i} = \mathbf{W}'_{T_i}\mathbf{R}_{B_i}^I, \tag{28}$$

for $i = 1, \dots, n$. We gather all the tangential relative velocities as

$$\boldsymbol{\gamma}_T = \mathbf{W}_T^{\mathrm{T}}\boldsymbol{u}, \tag{29}$$

where $\boldsymbol{\gamma}_T = \begin{bmatrix} \boldsymbol{\gamma}_{T_1}^{\mathrm{T}} & \cdots & \boldsymbol{\gamma}_{T_n}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$, and $\mathbf{W}_T \in \mathbb{R}^{3n\times2n}$ is found on the same form as (22) by replacing the $\boldsymbol{w}_{H_{ij}}$-vectors with $\mathbf{W}_{T_i}$ and replacing the $\mathbf{0}_{3\times1}$-vectors with $\mathbf{0}_{3\times2}$-matrices.

C. Bilateral Constraints - Joints

We need the relative velocities of the 'gaps' in the joints to find the forces involved in the bilateral constraint forces on the links imposed by the joints. The relative joint velocities for joint i along the $\boldsymbol{e}_x^I$- and $\boldsymbol{e}_y^I$-axis are defined as $\gamma_{J_{xi}} := \dot{g}_{J_{xi}}$ and $\gamma_{J_{yi}} := \dot{g}_{J_{yi}}$ (when they exist). Hence, by employing (12) we find that

$$\gamma_{J_{\chi i}} = \boldsymbol{w}_{J_{\chi i}}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{u}_i \\ \boldsymbol{u}_{i+1} \end{bmatrix}, \tag{30}$$

for $\chi = x, y$ and $i = 1, \dots, n-1$, where

$$\boldsymbol{w}_{J_{xi}}^{\mathrm{T}} = \left[\left({}_I\boldsymbol{e}_x^I\right)^{\mathrm{T}}, \tfrac{-L_i}{2}\sin(\theta_i), \ -\left({}_I\boldsymbol{e}_x^I\right)^{\mathrm{T}}, \tfrac{-L_{i+1}}{2}\sin(\theta_{i+1})\right] \tag{31}$$

$$\boldsymbol{w}_{J_{yi}}^{\mathrm{T}} = \left[\left({}_I\boldsymbol{e}_y^I\right)^{\mathrm{T}}, \tfrac{L_i}{2}\cos(\theta_i), \ -\left({}_I\boldsymbol{e}_y^I\right)^{\mathrm{T}}, \tfrac{L_{i+1}}{2}\cos(\theta_{i+1})\right]. \tag{32}$$

By defining $\boldsymbol{\gamma}_{J_i} = \begin{bmatrix} \gamma_{J_{xi}} & \gamma_{J_{yi}} \end{bmatrix}^{\mathrm{T}}$, we gather all the relative joint velocities as

$$\boldsymbol{\gamma}_J = \mathbf{W}_J^{\mathrm{T}}\boldsymbol{u}, \tag{33}$$

where $\boldsymbol{\gamma}_J = \begin{bmatrix} \boldsymbol{\gamma}_{J_1}^{\mathrm{T}} & \cdots & \boldsymbol{\gamma}_{J_{n-1}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$,

$$\mathbf{W}_J^{\mathrm{T}} = \begin{bmatrix} \mathbf{W}_{J_1}^{\mathrm{T}} & \mathbf{0}_{2\times3} & \cdots & \mathbf{0}_{2\times3} \\ \mathbf{0}_{2\times3} & \mathbf{W}_{J_2}^{\mathrm{T}} & & \mathbf{0}_{2\times3} \\ \vdots & & \ddots & \mathbf{0}_{2\times3} \\ \mathbf{0}_{2\times3} & \cdots & \mathbf{0}_{2\times3} & \mathbf{W}_{J_{n-1}}^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{3n\times2(n-1)}, \tag{34}$$

and $\mathbf{W}_{J_i} = \begin{bmatrix} \boldsymbol{w}_{J_{xi}} & \boldsymbol{w}_{J_{yi}} \end{bmatrix}$ for $i = 1, \dots, n-1$.

V. Non-Smooth Dynamics

The starting point for describing the dynamics of the snake robot is the *equality of measures* as introduced in [16]. The equality of measures includes both the equations of motion for impact free motion and the impact equations. The impact equations give rise to impulsive behaviour [17]. In this section, we employ all the previous results to find the various components of the equality of measures.

A. The Equality of Measures

The equality of measures describes the dynamics of the snake robot within the context of non-smooth dynamics. The non-smoothness allows for instantaneous velocity jumps, usually associated with impacts. The velocity $\boldsymbol{u}(t)$ is assumed to admit a right $\boldsymbol{u}^+$ and left $\boldsymbol{u}^-$ limit for all t in the (short) time-interval $I = [t_A, t_E]$ [16], and its time-derivative $\dot{\boldsymbol{u}}$ exists for almost all $t \in I$. The differential measure $\mathrm{d}\boldsymbol{u}$ is assumed to be decomposed into two parts to be able to obtain $\boldsymbol{u}$ from integration:

$$\mathrm{d}\boldsymbol{u} = \dot{\boldsymbol{u}}\mathrm{d}t + \left(\boldsymbol{u}^+ - \boldsymbol{u}^-\right)\mathrm{d}\eta, \tag{35}$$

where $\mathrm{d}t$ denotes the Lebesgue measure and $\mathrm{d}\eta$ denotes the atomic measure where $\int_{t_i} \mathrm{d}\eta = 1$ if $\boldsymbol{u}^+(t_i) - \boldsymbol{u}^-(t_i) \neq 0$. The total increment of $\boldsymbol{u}$ over a compact subinterval $[t_1, t_2]$ of I is

$$\int_{[t_1,t_2]} \mathrm{d}\boldsymbol{u} = \boldsymbol{u}^+(t_2) - \boldsymbol{u}^-(t_1), \tag{36}$$

and is due to a continuous change (i.e. impact free motion) stemming from $\dot{\boldsymbol{u}}$ as well as possible discontinuities in $\boldsymbol{u}$ within the time-interval $[t_1, t_2]$. Equation (36) is also valid when the time-interval reduces to a singleton $\{t_1\}$, so if a velocity jump occurs for $t = t_1$ then (36) gives a nonzero result. Note that if the obstacles are removed from the model (thus, no impacts will occur), then we have $\boldsymbol{u}^+(t) = \boldsymbol{u}^-(t)\,\forall t$, and (35) becomes $\mathrm{d}\boldsymbol{u} = \dot{\boldsymbol{u}}\mathrm{d}t$.

The planar motion of the snake robot is given by the Newton-Euler equations written as an equality of measures:

$$\mathbf{M}\mathrm{d}\boldsymbol{u} - \mathrm{d}\boldsymbol{R} = \boldsymbol{\tau}_C\mathrm{d}t, \tag{37}$$

where the force measure of possibly atomic impact impulsions $\mathrm{d}\boldsymbol{R}$ and the vector of applied joint torques $\boldsymbol{\tau}_C$ will be described in the following. The mass matrix is

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{M}_n \end{bmatrix} \in \mathbb{R}^{3n\times 3n}, \tag{38}$$

where $\mathbf{M}_i = \operatorname{diag}\left(\begin{bmatrix} m_i & m_i & \Theta_i \end{bmatrix}\right) \in \mathbb{R}^{3\times 3}$, and m_i and Θ_i is the mass and moment of inertia of link i, respectively. Hence, the mass matrix is diagonal and constant.

The force measure $\mathrm{d}\boldsymbol{R}$ accounts for all contact forces and impulses. Let $\mathcal{I}$ be the set of all active contacts with the obstacles

$$\mathcal{I}(t) = \{a \mid g_{H_a}(\boldsymbol{q}(t)) = 0\} \subseteq \{1, 2, \ldots, n\nu\}, \tag{39}$$

where g_{H_a} is the a-th element of the vector $\boldsymbol{g}_H$ in (11). The force measure can now be written as

$$\mathrm{d}\boldsymbol{R} = \sum_{a\in\mathcal{I}} (\mathbf{W}_H)_a\, \mathrm{d}P_{H_a} + \mathbf{W}_T\mathrm{d}\boldsymbol{P}_T + \mathbf{W}_J\mathrm{d}\boldsymbol{P}_J, \tag{40}$$

where $\mathrm{d}P_{H_a}$ is the normal contact impulse measure between a link and an obstacle, $\mathrm{d}\boldsymbol{P}_T$ is the tangential contact impulse measure (the friction) between the ground and the links, and $\mathrm{d}\boldsymbol{P}_J$ is the contact impulse measures due to the bilateral constraints in the joints. The order of the elements in the vectors $\mathrm{d}\boldsymbol{P}_T$ and $\mathrm{d}\boldsymbol{P}_J$ corresponds to the vectors of relative velocities (29) and (33), respectively. The notation $(\mathbf{W}_H)_a$ denotes the a-th column of the $\mathbf{W}_H$-matrix in (21). The contact impulse measures are, similarly to the velocity measure, decomposed into a Lebesgue-measurable force and a purely atomic impact impulse. This is written for $\mathrm{d}P_{H_a}$ as

$$\mathrm{d}P_{H_a} = \lambda_{H_a}\mathrm{d}t + \Lambda_{H_a}\mathrm{d}\eta, \tag{41}$$

where λ_{H_a} is a Lebesgue-measurable force due to a *continuous contact* between a link and an obstacle, and Λ_{H_a} is a purely atomic impact impulse caused by a *collision* between the two objects. The same decomposition can be performed for the other contact impulse measures. The Lebesgue measurable force and the purely atomic impact impulse for the normal contact with the obstacles and the friction are found from the force laws given in Section V-B.

The $n-1$ joint actuators are modeled as applied torques. The total torque applied to link i is

$$\tau_{C_{3i}} = \tau_i - \tau_{i-1}, \tag{42}$$

for $i = 1, \ldots, n$ where $\tau_i \in \mathbb{R}$ is the torque applied to joint i and $\tau_0 = \tau_n = 0$. The torques are found with a PD-controller:

$$\tau_i = -K_P\left(\phi_{i,d} - \phi_i\right) - K_D\left(\dot{\phi}_{i,d} - \omega_i\right), \tag{43}$$

where $K_P \in \mathbb{R}^+, K_D \in \mathbb{R}^+$ are positive constants, $\phi_i = \theta_{i+1} - \theta_i$, and $\phi_{i,d}$ is the desired joint angle for joint i for $i = 1, \ldots, n-1$. The joint torques are included in the equality of measures through the vector $\boldsymbol{\tau}_C = \begin{bmatrix}\mathbf{0}_{1\times 2} & \tau_{C_3} & \mathbf{0}_{1\times 2} & \tau_{C_6} & \cdots & \mathbf{0}_{1\times 2} & \tau_{C_{3n}}\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{3n}$.

B. Constitutive Laws for the Contact Forces

In this section, we will introduce set-valued force laws for normal contact with the obstacles, and Coulomb friction. The set-valuedness of the force laws allows us to write each constitutive law with a single equations and avoids explicit switching between equations in the numerical treatment. These laws will all be formulated on velocity level using the relative contact velocities γ given by (19) and (29). Subsequently, the set-valued force laws are formulated as equalities in Section V-B3 using the so-called 'proximal point function' in order to include the force laws in the numerical simulation [15].

1) Normal Contact Force: The normal contact between a link and an obstacle is described by the unilateral constraint

$$g_H \geq 0,\ \lambda_H \geq 0, \quad g_H\lambda_H = 0, \tag{44}$$

which is known as Signorini's law [15]. Here, λ_H is the normal contact force and g_H is the gap function. The subscripts 'i' and 'j' are temporarily removed for brevity. This set-valued force law states that the contact is impenetrable, $g_H \geq 0$, the contact can only transmit pressure forces $\lambda_H \geq 0$ and the contact force λ_H does not produce work $g_H\lambda_H = 0$. The force law can be expressed on different kinematic levels: displacement level (44), velocity level, and acceleration level. In the following we express all force laws for a closed contact on velocity level, while all forces vanish for open contacts. Then, by employing concepts of convex analysis, the relationship between the relative velocity and the Lebesgue measurable normal contact force (not an impulse) may be written for a closed contact $g_H = 0$ as an inclusion on velocity level

$$-\gamma_H \in N_{C_H}(\lambda_H), \tag{45}$$

where the convex set $C_H = \{\lambda_H \mid \lambda_H \geq 0\} = \mathbb{R}^+$ is the set of admissible contact forces, and N_{C_H} is the normal cone to C_H at λ_H [15]. The inclusion (45) is equivalent to the condition

$$\gamma_H \geq 0,\ \lambda_H \geq 0, \quad \gamma_H \lambda_H = 0, \tag{46}$$

for a closed contact $g_H = 0$. Before explaining the above force law (45), let us first mention that this force law describes the impenetrability of sustained contact, i.e. $g_H = 0$ and $\gamma_H = 0$, as well as detachment: $\gamma_H > 0 \Rightarrow \lambda_H = 0$. However, (45) does not cover impacts (where we have impulses instead of forces). For impacts we need a similar *impact law* described at the end of this subsection.

From the definition of a normal cone $N_C(\boldsymbol{x})$ to a convex set C at the point $\boldsymbol{x} \in \mathbb{R}^n$ [15], [20], we have that $N_C(\boldsymbol{x}) = \{0\}$ for $\boldsymbol{x} \in \operatorname{int} C$, and $N_C(\boldsymbol{x}) = \{\emptyset\}$ for $\boldsymbol{x} \notin C$. If $\boldsymbol{x}$ is on the *boundary* of C, then $N_C(\boldsymbol{x})$ is the set of all vectors $\boldsymbol{y} \in \mathbb{R}^n$ that are normal to $\boldsymbol{x}$. For example, the forces acting on a snake robot link from an obstacle is, by definition, always pointing away from the obstacle. Hence, the contact force is in the set $C_H = \mathbb{R}^+$ when the two objects are touching. So if the two objects are in contact, then the relative velocity γ_H between them is zero. Hence, e.g. $-\gamma_H = N_C(\lambda_H = 2) = 0$ in (45). In addition, the force law (45) also covers detachment (i.e. the moment the link moves away from the obstacle after having been in contact). For this case we have a contact force $\lambda_H = 0$ and a positive relative velocity: $-\gamma_H \in N_{C_H}(\lambda_H = 0) = \mathbb{R}^-$.

The force law (45) only covers finite-valued contact efforts during impulse free motion, i.e. all velocities are absolutely continuous in time. When a collision occurs in a rigid-body setting, then the velocities will be locally discontinuous in order to prevent penetration. The velocity jump is accompanied by an impact impulse Λ_H, for which we will set up an impact law. The relative velocity admits, similarly to the velocities $\boldsymbol{u}$, a right γ_H^+ and a left γ_H^- limit. The impact law for a completely inelastic impact at a closed contact can now be written as

$$-\gamma_H^+ \in N_{C_H}(\Lambda_H), \qquad C_H = \{\Lambda_H \mid \Lambda_H \geq 0\} = \mathbb{R}^+, \tag{47}$$

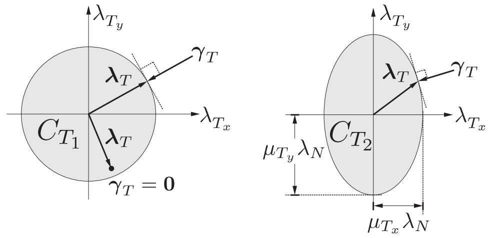
which is equivalent to the condition

$$\gamma_H^+ \geq 0,\ \Lambda_H \geq 0, \quad \gamma_H^+ \Lambda_H = 0. \tag{48}$$

2) Coulomb Friction Force: Similarly to the force law (45) for normal contact, we describe the constitutive description for friction using an inclusion on a normal cone. The friction force $\boldsymbol{\lambda}_T = \begin{bmatrix}\lambda_{T_x} & \lambda_{T_y}\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^2$ between the ground and a link (we omit subscript 'i' for brevity), in the two-dimensional tangent plane to the contact point, is modeled as Coulomb friction with a set-valued force law

$$-\boldsymbol{\gamma}_T \in N_{C_T}(\boldsymbol{\lambda}_T), \tag{49}$$

where $\boldsymbol{\gamma}_T \in \mathbb{R}^2$ is a relative sliding velocity, C_T is a convex set of admissible friction forces, and N_{C_T} is the normal cone to C_T at $\boldsymbol{\lambda}_T$. In the following, we show how to describe isotropic and anisotropic spatial Coulomb friction with (49) by changing the set C_T. First, isotropic friction will be elaborated on. Then, we extend the force law (49) to describe a particular form of anisotropic friction, called orthotropic friction.



(a) (b)

Fig. 5. Relationship between tangential relative velocity and friction force for (a) isotropic friction with the set C_{T_1} (in grey) given by (50) and (b) anisotropic friction where C_{T_2} is found from (52).

Isotropic friction indicates that the friction forces are not dependent on the orientation of the snake robot links while sliding. We employ the force law (49) to describe isotropic friction by choosing the convex set C_T equal to

$$C_{T_1} = \{\boldsymbol{\lambda}_T \mid \|\boldsymbol{\lambda}_T\| \leq \mu_T \lambda_N\}, \tag{50}$$

where $\mu_T > 0$ is the friction coefficient and $\lambda_N = mg$. The resulting relationship between the tangential relative velocities and the friction forces is illustrated in Fig. 5 (a) where a possible resulting friction force for $\boldsymbol{\gamma}_T = \boldsymbol{0}$ is included, and the relationship is elaborated on in the following.

The set-valued force law (49) with $C_T = C_{T_1}$ contains to the cases of stick and slip

$$\begin{array}{lll} \text{stick}: & \boldsymbol{\gamma}_T = \boldsymbol{0}, & \|\boldsymbol{\lambda}_T\| \leq \mu_T \lambda_N \\ \text{slip}: & \boldsymbol{\gamma}_T \neq \boldsymbol{0}, & \boldsymbol{\lambda}_T = -\mu_T \lambda_N \frac{\boldsymbol{\gamma}_T}{\|\boldsymbol{\gamma}_T\|}, \end{array} \tag{51}$$

The advantage of formulating the friction law as the inclusion (49) now becomes apparent. A spatial friction law such as (49), which is equivalent to (51) by choosing $C_T = C_{T_1}$, can not properly be described by a set-valued sign-function. Some authors model the spatial contact with two sign-functions for the two components of the relative sliding velocity using two friction coefficients μ_{T_x} and μ_{T_y} [6], [9]. This results however directly in an anisotropic friction law, as the friction force and the sliding velocity do no longer point in opposite directions (even for $\mu_{T_x} = \mu_{T_y}$). Then, such a double-sign-function force law corresponds to (49) with C_T being a rectangle with length $2\mu_{T_x}\lambda_N$ and width $2\mu_{T_y}\lambda_N$. Also, the (set-valued) sign function can be approximated with a smoothening function, for example some arctangent function. This results in a very steep slope of the friction curve near zero relative velocity. Such an approach is very cumbersome for two reasons. First of all, stiction can not properly be described: an object on a slope will with a smoothened friction law always slide. Secondly, the very steep slope of the friction curve causes the differential equations of motion to become numerically stiff. Summarising, we see that (49) or (51) describes spatial Coulomb friction taking isotropy and stiction properly into account. We prefer using (49) instead of (51), because the latter becomes not well conditioned for very small $\boldsymbol{\gamma}_T$ when used in numerics. Note also that (49) and (45) have the same mathematical form. Moreover, the inclusion (49) is much more general since we can easily change the convex set C_T to get a different (and hence anisotropic) friction model. To achieve this, we can instead choose the set C_T as

$$C_{T_2} = \left\{ \boldsymbol{\lambda}_T \mid \frac{\lambda_{T_x}^2}{(\mu_{T_x}\lambda_N)^2} + \frac{\lambda_{T_y}^2}{(\mu_{T_y}\lambda_N)^2} \le 1 \right\}, \tag{52}$$

where $\mu_{T_x}, \mu_{T_y} > 0$ are directional friction coefficients along the $\boldsymbol{e}_x^{B_i}$- and $\boldsymbol{e}_y^{B_i}$-axis, respectively. The relationship between the direction of the tangential relative velocity and the direction of the corresponding friction force is illustrated in Fig. 5 (b) for $\mu_{T_x} < \mu_{T_y}$. Note that for $\mu_{T_x} = \mu_{T_y} = \mu_T$, the set C_{T_2} is equal to C_{T_1}.

While the force law (49) only describes the Coulomb friction during impulse free motion, we also need a force law for impact impulses $\boldsymbol{\Lambda}_T$. These are found from the exact same form as (49) by replacing $\boldsymbol{\gamma}_T$ with its right limit $\boldsymbol{\gamma}_T^+$ and inserting $\boldsymbol{\Lambda}_T$ instead of $\boldsymbol{\lambda}_T$ both in (49) and in the convex set C_T.

3) Constitutive Laws as Projections: An inclusion can not be directly employed in numerical calculations. Hence, we transform the force laws (45) and (49), which have been stated as an inclusion to a normal cone, into an equality. This is achieved through the so-called proximal point function $\text{prox}_C(\boldsymbol{x})$, which equals $\boldsymbol{x}$ if $\boldsymbol{x} \in C$ and equals the closest point in C to $\boldsymbol{x}$ if $\boldsymbol{x} \notin C$. The set C must be convex. Using the proximal point function we transform the force laws into implicit equalities (see [15])

$$-\gamma_\kappa \in N_{C_\kappa}(\lambda_\kappa) \iff \lambda_\kappa = \text{prox}_{C_\kappa}(\lambda_\kappa - r_\kappa \gamma_\kappa), \tag{53}$$

where $r_\kappa > 0$ for $\kappa = H, T$.

VI. Numerical Algorithm - Time-Stepping

The numerical solution of the equality of measures (37) is found with an algorithm introduced in [16] (see also [15], [17]) called the time-stepping method described in the following. By employing this method together with set-valued force laws we avoid having to explicitly switch between equations when impacts occur. Section VI-A and VI-B are based on [19], except for the novel direct calculation of the bilateral contact impulses which we have introduced in [18], [21].

A. Time Discretization

Let t_l denote the time at time-step $l = 1, 2, 3, \ldots$ where the step size is $\Delta t = t_{l+1} - t_l$. Consider the (usually very short) time interval $I = [t_A, t_E]$, and let $t_l = t_A$. Define $\boldsymbol{q}_A = \boldsymbol{q}(t_A)$, $\boldsymbol{u}_A = \boldsymbol{u}(t_A)$ which are admissible with respect to both the unilateral and bilateral constraints. Our goal is now to find $\boldsymbol{q}_E = \boldsymbol{q}(t_E)$. We use the states of the system at the *mid-point* $t_M = t_A + \frac{1}{2}\Delta t$ of the time-interval I to decide which contact points are active (i.e. whether or not any links are touching an obstacle). The coordinates (positions and orientations) at t_M are found from

$$\boldsymbol{q}_M = \boldsymbol{q}_A + \frac{\Delta t}{2}\boldsymbol{u}_A. \tag{54}$$

The approximation of the matrices $\mathbf{W}_\Xi$, where $\Xi = H, T, J$, on the time-interval I is given as $\mathbf{W}_{\Xi M} := \mathbf{W}_\Xi(\boldsymbol{q}_M)$. A numerical approximation of the equality of measures (37) over the time-interval I can now be written as

$$\mathbf{M}(\boldsymbol{u}_E - \boldsymbol{u}_A) - \boldsymbol{S} - \mathbf{W}_{JM}\boldsymbol{P}_J = \boldsymbol{\tau}_C \Delta t, \tag{55}$$

where

$$\boldsymbol{S} = \sum_{a\in\mathcal{I}} (\mathbf{W}_{HM})_a P_{H_a} + \mathbf{W}_{TM}\boldsymbol{P}_T, \tag{56}$$

and P_{H_a}, $\boldsymbol{P}_T$, and $\boldsymbol{P}_J$ are the contact impulses during the time-interval I. They consist of forces λ acting *during* I, and possible impulses Λ acting *in* the time-interval I. To find the positions and orientations $\boldsymbol{q}_E$ at the end of the time-interval, we need to solve (55) for $\boldsymbol{u}_E$ and the contact impulses. The contact impulses associated with obstacle and ground frictional contact are found using the prox-functions described in Section V-B3 for the set of active contact points $\mathcal{I}$. Hence, the constitutive laws (53) for the obstacle contact and friction impulses may now be written as

$$P_{H_a} = \text{prox}_{C_H}(P_{H_a} - r_H \gamma_{HE_a}), \ a \in \mathcal{I}, \tag{57}$$

$$\boldsymbol{P}_{T_b} = \text{prox}_{C_T}(\boldsymbol{P}_{T_b} - r_T \boldsymbol{\gamma}_{TE_b}), \ b = 1, \ldots, n \tag{58}$$

where $r_H, r_T > 0$, γ_{HE_a} is the a-th element of the vector $\boldsymbol{\gamma}_{HE}$, $\boldsymbol{\gamma}_{TE_b}$ is the vector of the $(2b-1)-$th and $2b$-th element of $\boldsymbol{\gamma}_{TE}$, and

$$\boldsymbol{\gamma}_{HE} := \boldsymbol{\gamma}_H(\boldsymbol{q}_M, \boldsymbol{u}_E) = \mathbf{W}_{HM}^{\mathrm{T}}\boldsymbol{u}_E, \tag{59}$$

$$\boldsymbol{\gamma}_{TE} := \boldsymbol{\gamma}_T(\boldsymbol{q}_M, \boldsymbol{u}_E) = \mathbf{W}_{TM}^{\mathrm{T}}\boldsymbol{u}_E. \tag{60}$$

The constitutive laws (57)-(60) are valid for completely inelastic impacts.

The constraints on the joints are bilateral and it therefore holds that

$$\boldsymbol{\gamma}_{JE} := \boldsymbol{\gamma}_J(\boldsymbol{q}_E, \boldsymbol{u}_E) = \mathbf{W}_{JM}^{\mathrm{T}}\boldsymbol{u}_E = \mathbf{0}, \ \forall t. \tag{61}$$

This allows us to directly compute the associated contact impulses $\boldsymbol{P}_J$ by employing (55) and (61):

$$\begin{aligned}\boldsymbol{P}_J = &-\left(\mathbf{W}_{JM}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{W}_{JM}\right)^{-1}\mathbf{W}_{JM}^{\mathrm{T}} \cdot \\ &[\boldsymbol{u}_A + \mathbf{M}^{-1}(\boldsymbol{S} + \boldsymbol{\tau}_C\Delta t)].\end{aligned} \tag{62}$$

The equations (55)-(60) and (62) constitute a set of non-smooth equations where the number of unknowns $\boldsymbol{u}_E$, P_{H_a}, $\boldsymbol{P}_T$, and $\boldsymbol{P}_J$, equals the number of equations. An algorithm to solve this set of equations is described in Section VI-B. After having solved for $\boldsymbol{u}_E$ we find the position at the end of the time-step as

$$\boldsymbol{q}_E = \boldsymbol{q}_M + \frac{\Delta t}{2}\boldsymbol{u}_E. \tag{63}$$

B. Solving for the Contact Impulses

The numerical integration algorithm used in this paper is called a time-stepping method which allows for a simultaneous treatment of both impulsive and non-impulsive forces during a time-step. The frictional contact problem, defined by (55)-(60) and (62), needs to be solved for each time-step t_l. A Modified Newton Algorithm [22] has been chosen to solve the nonlinear problem iteratively because of its simplicity. Let the superscript $^{(k)}$ denote the current iteration of the Modified

Newton Algorithm, and initialize all contact impulsions (for active contacts) with the value they had the last time their corresponding contact point was active. Let those that *were* active be initialized with their previous values. Now, the algorithm may be written as

1) Solve

$$\boldsymbol{P}_J^{(k)} = -\left(\mathbf{W}_{JM}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{W}_{JM}\right)^{-1}\mathbf{W}_{JM}^{\mathrm{T}} \cdot \left[\boldsymbol{u}_A + \mathbf{M}^{-1}\left(\boldsymbol{S}^{(k)} + \boldsymbol{\tau}_C\Delta t\right)\right], \tag{64}$$

where

$$\boldsymbol{S}^{(k)} = \sum_{a\in\mathcal{I}}(\mathbf{W}_{HM})_a\, P_{H_a}^{(k)} + \mathbf{W}_{TM}\boldsymbol{P}_T^{(k)}, \tag{65}$$

2) Solve $\boldsymbol{u}_E^{(k+1)}$ from

$$\mathbf{M}\left(\boldsymbol{u}_E^{(k+1)} - \boldsymbol{u}_A\right) - \boldsymbol{S}^{(k)} - \mathbf{W}_{JM}\,\boldsymbol{P}_J^{(k)} = \boldsymbol{\tau}_C\Delta t \tag{66}$$

3) Solve for $a \in \mathcal{I}$ and $b = 1, \ldots, n$

$$P_{H_a}^{(k+1)} = \mathrm{prox}_{C_H}\left(P_{H_a}^{(k)} - r_H\gamma_{HE_a}^{(k+1)}\right), \tag{67}$$

$$\boldsymbol{P}_{T_b}^{(k+1)} = \mathrm{prox}_{C_T}\left(\boldsymbol{P}_{T_b}^{(k)} - r_T\boldsymbol{\gamma}_{TE_b}^{(k+1)}\right), \tag{68}$$

where

$$\boldsymbol{\gamma}_{HE}^{(k+1)} = \mathbf{W}_{HM}^{\mathrm{T}}\boldsymbol{u}_E^{(k+1)}, \quad \boldsymbol{\gamma}_{TE}^{(k+1)} = \mathbf{W}_{TM}^{\mathrm{T}}\boldsymbol{u}_E^{(k+1)}. \tag{69}$$

Repeat steps 1. to 3. until

$$\|\boldsymbol{P}_H^{(k+1)} - \boldsymbol{P}_H^{(k)}\| + \|\boldsymbol{P}_T^{(k+1)} - \boldsymbol{P}_T^{(k)}\| < \epsilon, \tag{70}$$

where $\epsilon > 0$ is a user-defined tolerance. Subsequently, $\boldsymbol{q}_E$ is calculated from (63) and the calculation of the time-step is finished. Usually, a higher value of the parameters r_H, r_T yields a faster convergence rate at the risk of divergence. However, a general convergence result for the Modified Newton Algorithm does not exist. The constitutive laws (67)-(69) used to describe the contact impulses are given on *velocity level*. This means that the bilateral constraints on position level are in general not satisfied. A solution to this problems is suggested in the following.

C. Constraint Violation

The contact impulses $\boldsymbol{P}_J$ are calculated such that the bilateral constraints are satisfied on velocity level. In order to prevent drift problems, we project the positions of the links so that the bilateral constraints are satisfied on position level after the Modified Newton Algorithm has converged and $\boldsymbol{q}_E$ has been found from (63). The projection is performed by keeping the position of link 1 fixed together with the orientations of all links. Then, the links are moved so that all the gap functions (12) associated with the joints are equal to zero, i.e. $g_{J_{\chi i}} = 0$ for $\chi = x, y$ and $i = 1, \ldots, n-1$.

The projection performed to satisfy the bilateral constraints on position level adds additional computations to the numerical treatment of the snake robot model. An alternative approach is to reduce the number of generalized coordinates (from a non-minimal description) by expressing the bilateral constraints analytically and then incorporating them implicitly into the model. However, some sort of projection algorithm will in some cases still be necessary to avoid that the links drift apart depending on how the link positions are found from the resulting generalized coordinates and the accuracy of the numerical integrator. Another approach would be to develop the model from a set of minimal coordinates directly, but then the system matrices will grow very large when the number of links is increased. With the modeling approach described in this paper, we avoid having to rewrite the model to a new (minimal) set of coordinates and the system matrices are kept simple. Hence, we have the positions of all links directly available and the appearance of the model makes it easy to understand and to implement for numerical integration. Moreover, it is less demanding to extend the model to 3D where an implicit incorporation of the bilateral constraint in the model would prove cumbersome and result in even larger system matrices.

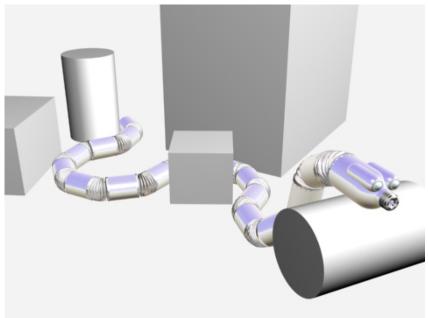


Fig. 6. A snake robot in a cluttered environment (motivating illustration).

VII. Obstacle Aided Locomotion

This section treats various aspects of obstacle aided locomotion with snake robots in order to provide a better understanding of the simulation and experimental results presented later in this paper. A short elaboration on the basic principles of obstacle aided locomotion is given followed by a discussion on how to implement this for snake robots together with a description of an experimental setup for studying the phenomenon of obstacle aided locomotion.

A. Motivation

Biological snakes exploit objects and irregularities in their environment to achieve more efficient propulsion. A comparison with legged creatures provides an indication of this statement. While the gait pattern displayed by a legged creature is highly dependent on the current speed of this creature, the gait pattern displayed by a snake will mainly be dependent on the properties of the environment surrounding the snake. The flexibility and robust mobility of snakes in cluttered environments is a significant motivation for research on snake robots. However, in order to fully benefit from the advantages of snake locomotion in such environments, as illustrated in Fig. 6, snake robots must learn to sense and understand the geometries surrounding the robot at any given time.

In this paper, the word obstacle is used to denote an object or an irregular surface in the path of the snake robot that can be utilized for propulsion. This may seem like a contradiction since it is in fact not an obstacle from the point of view of the snake robot. However, the characterization is valid in the sense of mobile robotics in general. The use of this denotation therefore helps emphasize one of the fundamental differences between snake locomotion and other traditional means of mobility, such as wheeled, tracked, and legged robots. While in traditional mobile robotics the aim is typically to avoid obstacles, a snake robot should rather seek out and make contact with obstacles since they represent push-points that can be utilized for more efficient propulsion. Hence, for snake robots the aim is therefore not *obstacle avoidance*, but rather *obstacle utilization*.

B. Understanding Obstacle Aided Locomotion

The undulatory gait pattern called lateral undulation [23] represents a common form of snake locomotion. During this motion, a series of sinusoidal curves are propagated backwards through the snake body. These waves interact with irregularities in the environment of the snake in such a way that the resulting forces propel the snake forward. Progression by lateral undulation is highly dependent on interaction forces from the environment. The study of this motion pattern therefore provides a good understanding of how a snake robot may act on its environment in order to move forwards.

A study of lateral undulation and its dependency on external push-points was given by J. Gray as far back as 1946 [1]. Gray depicted a scenario similar to the one illustrated in Fig. 7. It shows an articulated mechanism with three links interconnected by two joints. Torques are applied to the two joints so that each link is pushed towards a perfectly smooth obstacle, giving rise to contact forces acting in the lateral direction of each link. Gray showed that this mechanism will glide in the forward direction if the angle α (between link 1 and 2) is greater than the angle β (between link 2 and 3) and in the backward direction if α is smaller than β. This may be understood by studying the sum of the contact forces to the right in the figure. It is assumed that the resultant of the contact forces $\boldsymbol{F}_{sum}$ is large enough to overcome the friction forces from the ground.

A snake robot is constructed by serially connecting multiple mechanisms similar to the one in Fig. 7. The structure of such a snake robot is depicted in Fig. 8 (a) where the snake robot posture and obstacles in Fig. 7 have been duplicated. Three push-points are needed for locomotion [23], and we see from the Fig. 7 that obstacle 1 is associated with the normal reaction force that has an advantageous contribution in the (upwards) direction of propulsion. This suggests that obstacle 1 should also be utilized for locomotion of the interconnected structure in Fig. 8 (a). Hence, a natural choice of push-points for obstacle aided locomotion by lateral undulation are the three occurrences of obstacle 1 depicted in Fig. 8 (b). Moreover, we see from the figure that the push-points are close to the centre line of the snake robot posture. This corresponds to the utilization of walls for snakes moving by lateral undulation through a winding track depicted in [1]: The sequence of pictures clearly suggest that the snake pushes against the

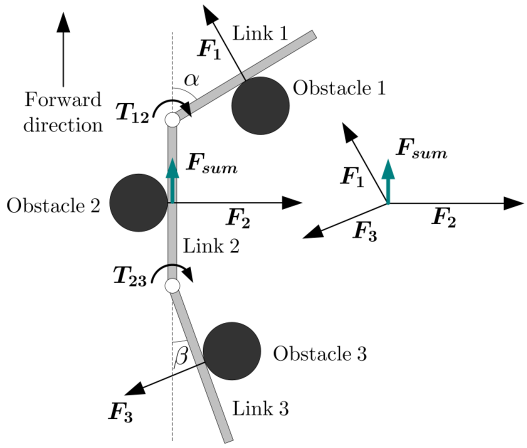


Fig. 7. A snake robot with three links interconnected by two joints. The resultant of the contact forces from the obstacles propels the snake. The propelling force points in the forward direction if $\alpha > \beta$, in the backward direction if $\alpha < \beta$, and is zero if $\alpha = \beta$.

walls of the track approximately along the centre line of its undulatory motion. Hence, the angle between the snake body at the contact point and the direction of locomotion is large. This results in an efficient utilization of the push-points which is explained in the following. The reaction force from an obstacle can be decomposed into two parts: a longitudinal and a lateral part with respect to the direction of locomotion. If the push-points are far from the centre line of the snake robot posture (for example if the bottom two obstacles in Fig. 8 (b) were located next to link 3 and link 6 instead), then the angles between the links pushing against the obstacles and the centre line are decreased, and a substantial portion of the normal contact reaction force points in the lateral direction and does not contribute to moving the snake robot forwards. In addition, larger joint torques are needed to obtain the same longitudinal reaction forces compared to if the obstacles are moved closer to the centre line. Hence, moving the contact point to the centre line of the snake robot posture optimizes the longitudinal contribution of the normal contact reaction forces and reduces the necessary joint torques.

C. Requirements for Intelligent Obstacle Aided Locomotion

In the view of the authors of this paper, efficient obstacle aided locomotion with a snake robot in a cluttered and irregular environment requires three main physical properties. These are a *smooth exterior body surface*, a *contact force sensing system*, and a *obstacle locating system*. The first property will give the robot the ability to glide forward despite the irregularities in its environment. Moreover, it will mimic the skin of a biological snake more closely. Any non-smooth body surface will potentially obstruct the gliding motion of the robot. The second property is crucial for intelligent obstacle utilization. Contact force sensing allows the robot to detect when the body is in contact with a push-point and also control the force exerted on a push-point. Since the sum of contact forces along the snake body is what propels the robot forward, the ability to measure these forces is important in order to

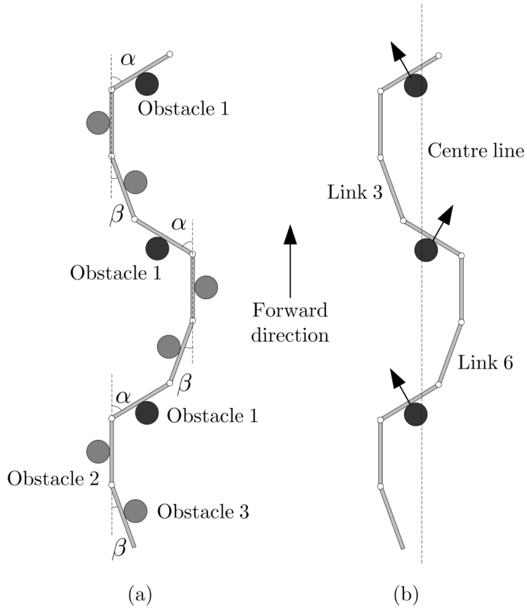


Fig. 8. Snake robot (a) with all obstacles from Fig. 7, and (b) with only the necessary and most suited obstacles for locomotion.

be able to control the propulsion. The third property enables the snake robot to search for new push-points which it is not currently able to touch (e.g. sense with the force sensors).

A control strategy for obstacle aided locomotion should in general consist of two continuous tasks running in parallel. The function of the first task should be high-level motion planning. This task should establish some sort of map of the environment surrounding the snake robot and use it to plan a path through this environment. In contrast to conventional path planning strategies for mobile robots, the goal should not be to avoid obstacles, but rather to move relative to these irregularities in a way that enables the snake robot to utilize push-points for more efficient propulsion.

The actual obstacle utilization should be continuously carried out in the second task as a function of the current force measurements along the body and the desired direction of motion. This task should basically control the joints of the snake robot so that the sum of the contact forces along the snake body is pointing toward the desired direction of motion.

The purpose of this paper is mainly to present a mathematical framework for the study of obstacle aided locomotion. This paper will therefore not treat the specific implementation of the motion controller discussed above. However, we note that all the three above main physical properties are in effect available in the mathematical model presented in this paper. Hence, the model can be employed to test intelligent obstacle aided locomotion when such future control schemes are developed.

D. Experimental Observation of Obstacle Aided Locomotion

As described earlier in this chapter, the study of lateral undulation provides a good understanding of how a snake robot may utilize interaction forces from its environment in order to progress forward. The shape of a snake robot displaying this gait pattern may be described by the 'serpenoid curve' [2]. A snake robot may approximate this shape by setting its joint angles as

$$\phi_{i,d} = A_h \sin\left(\omega_h t + (i-1)\,\delta_h\right) \tag{71}$$

where $\phi_{i,d}$ is the desired relative angle between link i and $i+1$, A_h is the amplitude of joint oscillation, ω_h is the angular frequency of oscillation of the joint, t is the time, and δ_h is the phase-shift between adjacent joints [9].

Lateral interaction forces from the environment are needed for this gait pattern to progress the snake forward. The sum of these forces must act forwards for the snake robot to move in this direction. A simple way to study obstacle aided locomotion is therefore to curve the snake robot to the initial shape of this gait pattern and place obstacles along the snake robot similar to the position of the obstacles in Fig. 8 (b). When the snake robot resumes the gait pattern from its initial shape, it will progress forward by lateral undulation due to the reaction forces from the obstacles. Note from (71) and Fig. 8 (b) that the chosen amplitude A_h and phase-shift δ_h depends on the design of the snake robot since it needs three push-points to move forward. Hence, we can choose from a bigger variety of A_h and δ_h for a snake robot with many links, since it might be able to utilize the obstacles for many different sinus-like shapes. However, for a snake robot with fewer links, the choices of A_h and δ_h are more pre-determined. This is because a small δ_h will not result in a sinus-like shape, but rather a C-like shape. The snake robot will not be able to utilize the three obstacles in a C-like shape based on the elaboration above.

A real rescue operation in a shattered building will require a snake robot with a smooth and sealed exterior. This avoids that small pieces of rubble get stuck in the joints of the snake robot and hinders it to move. However, the principle motion pattern for how to move in such an environment can be studied in a laboratory setting without having developed a robust snake robot design first. To this end, we simply choose the obstacles large enough so that they do not get stuck in the snake robot joints. This is advantageous since it reduces the development time and cost of the snake robot used in the experiments. Moreover, the basics of obstacle aided locomotion are the same for small and large obstacles since the fundamental goal is still to allocate enough push-points such that the snake robot is able to move forward.

VIII. Simulations and Experimental Validation

In this section, we first present the parameters used in the simulation, then simulation results are presented, subsequently the experimental setup is described, and finally results from simulations and experiments are presented and compared.

A. Simulation Parameters

The parameters that describe the mathematical model of the snake robot are for $i = 1, \ldots, n$: $n = 11$ links, $L_i = 0.122$ m, $L_{SC_i} = 0.0525$ m, $L_{GS_i} = 0.0393$ m, $m_i = 7.5/11\text{kg} \approx 0.682$ kg, and $J_i = 1.32 \times 10^{-3}$ kg m^2. The controller parameters are $K_P = 800$ Nm and $K_D = 2$ Nm. The simulation parameters are $r_H = 0.01$ and $r_T = 1.3$ in (67) and (68), respectively. The Coulomb friction coefficient

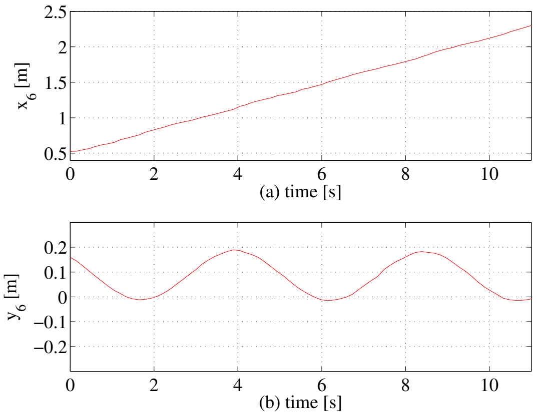


Fig. 9. Position of link 6 during obstacle aided locomotion: simulation results.

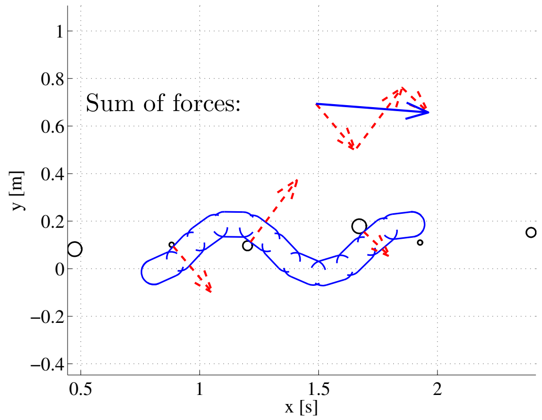


Fig. 10. Snake robot at $t = 5.2$ s where contact forces are indicated by arrows (dashed line). The obstacles are given by the black circles. The snake robot is moving to the right.

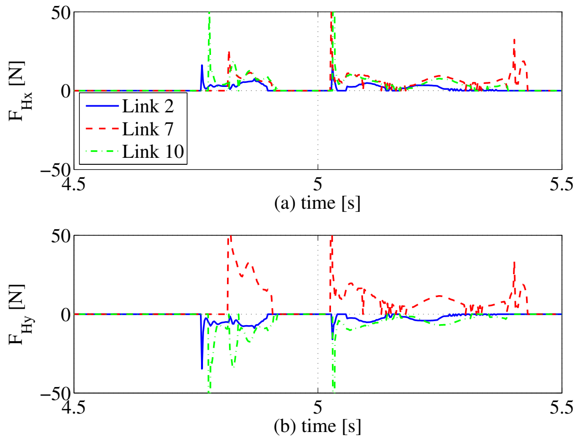


Fig. 11. Linear contact force acting on the centre of gravity of links 2, 7, and 10.

was measured to $\mu_T = 0.2$ by dragging the actual snake robot from a scale along the ground surface used in the experiments.

In the following, we will simulate the motion pattern lateral undulation with and without obstacles for one set of motion pattern parameters A_h, δ_h, and ω_h. The parameters A_h and δ_h determines the shape of a snake robot, and our particular choice of these parameters is based on the discussion in Section VII where it is noted that the snake robot needs at least three contact points to move forward by lateral undulation by pushing against obstacles. Hence, the shape of the snake robot must be 'curled enough' so that these three contact points are realizable. An elaboration of this concept is also given in the following.

B. Simulation Results

Before we proceed to compare the simulation and experimental results, we will first present simulation results of obstacle aided locomotion with small obstacles. We present these results to show that the snake robot model works for small obstacles, and to compare the simulation results with the discussion made on obstacle aided locomotion in Section VII. Experimental results with such small obstacles are not provided since the obstacles would get stuck in the snake robot joints. Hence, to perform an experimental validation of this setup, the contact surface the snake robot has with the obstacles will have to be redesigned.

The obstacles are placed based on the elaboration in Section VII-B and VII-D. That is, the obstacles are placed according to the initial configuration of the snake robot based on the choice of motion pattern parameters A_h and δ_h given in the beginning of Section VIII-D, and they are located close the centre line of the sinus-like shape of the snake robot. The radii were chosen for obstacle $j = 1, \ldots, 7$ (all sizes in metres) as: $0.02, 0.03, 0.01, 0.02, 0.03, 0.01, 0.02, 0.03$, respectively. The snake robot was set to move with the motion pattern given by (71) with the initial value of all relative angles equal to $\phi_{0i} = \phi_i(t = 0)$ for $i = 1, \ldots, n-1$. The motion pattern parameters were $A_h = 40\pi/180$ rad, $\omega_h = 80\pi/180$ rad/s, and $\delta_h = -50\pi/180$ rad and a discussion of the choice of these are given in the end of this section. The position of link 6 during locomotion is given in Fig. 9.

We quantify the instantaneous value of the contact force that acts on the snake robot from the obstacles as

$$\boldsymbol{F}_H = \frac{1}{\Delta t}\sum_{a\in\mathcal{I}}(\mathbf{W}_{HM})_a\, P_{H_a}. \tag{72}$$

Hence, $\boldsymbol{F}_H$ consists of both forces and possible impulses (see the explanation of (56)) and it is found from the integral of forces over each time-step.

Fig. 10 depicts the snake robot at $t = 5.2$ s in an environment with circular obstacles. The snake robot is moving to the right in the illustration and we observe that this is also the direction in which the sum of forces, acting on the snake robot from the obstacles, points. The contact forces found from (72) correspond to the illustration of obstacle aided locomotion in Fig. 8, hence it is the sum of contact forces that results in the forward (toward the right) motion of the snake robot.

The linear contact forces found from (72) are plotted from $t = 4.5$ s to $t = 5.5$ s for links 2, 7, and 10 in Fig. 11. Hence, the figure depicts how the contact forces develop around the time the snake robot was as illustrated in Fig. 10. The spikes in

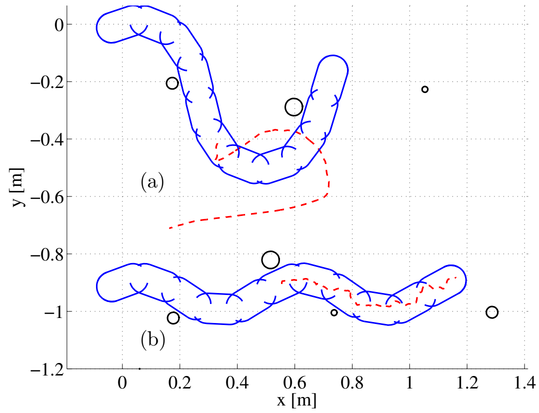


Fig. 12. Posture at $t = 0$ s of snake robot for (a) $\delta_h = -30\pi/180$ rad and (b) $\delta_h = -70\pi/180$ rad. The posititions of the middle link from $t = 0$ s to $t = 5$ s are indicated by the dashed line.

the forces arise from impulses due to the link colliding with an obstacle. The maximum values of these spikes have not been included in the plot to depict the contact force during the rest of the time-interval more accurately. We see from Fig. 11 (b) that the average applied force to link 7 is higher than to the other two links. Comparing to Fig. 10, we see that the difference is because link 7 is the only link that is subjected to a force pointing in the negative $\boldsymbol{e}_y^I$-direction. Hence, this contact force balance the contact forces acting on the other two links (links 2 and 10).

We see from Fig. 10 that the snake robot is barely 'curled enough' to be able to take advantage of the necessary three contact points discussed in Section VII. To this end, a comment on the choice of the motion pattern parameters A_h, δ_h, and ω_h is imminent. We see from Fig. 12 (a) that a too low value of δ_h results in that the snake robot is no longer able to be in contact with three push-points simultaneously, and this renders it unable to move forward by lateral undulation. However, if we increase the value of δ_h as illustrated in Fig. 12 (b), then snake robot is able to move forward, but the angle between the link in contact with an obstacle and the direction of progress becomes small. Hence, most of the reaction force from the obstacle will point normal to the direction of motion (since it points transversal to the link), and this leads to very inefficient locomotion since larger joint torques are needed to propel the snake robot forward (i.e. the contact force needs to be higher to obtain the same amount of force tangential to the direction of motion with three push-points). Moreover, the increased contact force will result in that the snake robot can get easier stuck since the friction forces between the obstacles and the snake robot will increase. The increase in contact force for a posture with 'more curls' corresponds to the increase in the constraint forces on the wheels for a wheeled snake robot with the same posture during lateral undulation (without obstacles) as described in [24]. The angle between the link in contact with the obstacle and the direction of locomotion will also be smaller if we choose to keep δ_h at its original value, but lower the amplitude of joint oscillation A_h. Hence, the value of δ_h should be chosen such that the snake robot is just 'curled enough' to utilize the push-points for locomotion, and the amplitude of oscillation should, up to some extent, be chosen as high as possible to obtain a large angle between the direction of locomotion and the links in contact with the push-points. However, it is important to be aware of that if the obstacles are located far from each other, a larger value of δ_h is necessary to allow the snake robot to reach all three necessary push-points. Moreover, the wave propagation speed is dependent on δ_h, so this also has to be taken into account when choosing the motion pattern parameters.

It now remains to comment on the value of the angular frequency ω_h. For an infinitely strong snake robot (both with respect to actuator torques and outer shell) and fixed obstacles that do not yield, a larger ω_h will result in a greater forward speed since the propagating wave is propagated faster toward the back of the snake robot. However, the larger the speed, the harder the impact between the snake robot and the obstacles will become. Also, the speed will be limited by the amount of torque the actuators are able to produce. Moreover, the obstacles may not always be fixed to the ground, such as a chair or a small rock, hence it may in some cases be necessary to limit the forces exerted on the obstacle to keep them fixed. This is also achieved by limiting the angular frequency ω_h. From the discussion above, we choose to keep the motion parameters as they are for the upcoming experiments.

1) What if We Remove One Obstacle?: We briefly explore by simulation the scenario where obstacle 3 (counting from the beginning of the track) is partly and completely removed. First, obstacle 3 was moved one snake robot link diameter to the left: $r_{H_3}^{new} = r_{H_3}^{old} + 2L_{SC_i}\,{}_I\boldsymbol{e}_y^I$. The simulation result showed that the snake robot started sliding to the left and rotating anti-clockwise when it was supposed to collide with the moved obstacle. However, the deviation from its original path was not substantial enough to hinder the snake robot in reaching obstacle 4 sufficiently accurate to continue its path along the track. Secondly, obstacle 3 was completely removed. Then, the snake robot also rotated anti-clockwise and slid to the left when it was supposed to hit obstacle 3, but this time it was unable to complete the track since it never collided with the third obstacle, and therefore never regained a third contact point needed for locomotion as discussed in Section VII.

We might expect that the snake robot fails to move through the entire track with one obstacle moved or missing since the joint pattern is pre-programmed, however these results show that some repositioning is possible. Hence, a control algorithm for intelligent obstacle aided locomotion does not need to provide a motion pattern fully accurate compared to the position of the obstacles. The snake robot still manages to move through the track even if it touches some obstacles in a far from perfect manner. The downside can be that larger joint torques are needed as discussed in Section VII-B.

C. The Snake Robot and Experimental Setup

The snake robot Aiko (Fig. 1) used in the experiments was built in 2006 at the NTNU/SINTEF Advanced Robotics Laboratory in Trondheim, Norway. The length L_i and mass m_i of the links are the same as given for the mathematical model in Section VIII-A. There are however a few differences between the model and the robot: 1) The outer shell of most of the links are not as smooth and round as in the model, but the front link has a aluminum hemisphere as a 'head' in the front. 2) The snake robot has ten 2 DOF cardan joints, but the angles of the joints that control the vertical movement are set to zero in the experiments. 3) All links are shaped as cylinders so the robot may roll. However, this was not a problem once at least one of the joints were moved slightly away from its mid-position.

Each 2 DOF cardan joint was actuated by two 6 W DC-motors. The two motors were controlled by a controller with dynamic feedback (see [25]) implemented on an Atmel ATmega128 microcontroller. The snake robot joint reference angles were sent with a frequency of 10 Hz from a Pentium-M 1.8 GHz laptop, via a CAN-bus to the microcontrollers. The position of the middle link (link 6) was tracked using a Vicon MX Motion Capture System with 4 cameras (MX3) together with Matlab Simulink. The Vicon programme (Vicon iQ 2.0) ran on a computer with 4 Intel Xeon 3 GHz processors and 2 GB RAM. Logging of motion data was synchronized in time through a TCP-connection between the laptop and the Vicon-computer at the start-up of the transmission of the desired snake robot joint angles. Data logging was performed at 20 Hz. We have chosen to let the position of link 6 represent the position of the snake robot, because then we filter out any transient behaviour of the snake robot that might occur at its ends and it is thus easier to focus on the general motion of the snake robot.

The snake robot moves on a particle board in the experiments. The obstacles are cut from a PVC-pipe with 0.25 m outer diameter.

D. Simulation and Experimental Results

In this section the results from simulations and experiments are presented and compared. Lateral undulation without obstacles is included to validate the mathematical model and to confirm the necessity of projections from the ground to move forward effectively on a plane with isotropic friction. The snake robot was set to move with the motion pattern given by (71) with the initial value of all relative angles equal to $\phi_{0i} = \phi_i(t = 0)$ for $i = 1, \ldots, n-1$ for all simulations and experiments. The motion pattern parameters were $A_h = 40\pi/180$ rad, $\omega_h = 80\pi/180$ rad/s, and $\delta_h = -50\pi/180$ rad, and the choice of these are based on the discussion in Section VII and Section VIII-B.

1) Lateral Undulation and Isotropic Friction: Although not commonly used, it is possible for a snake robot to use lateral undulation for locomotion when the friction between the ground and the snake robot is isotropic [26]. In this section, we validate the model of the snake robot by letting it move by lateral undulation on a surface with isotropic friction. The simulation and experimental data are compared by letting the initial position of link 6 be the same for both cases. Fig. 13 shows both the position of link 6 calculated from the mathematical model in the simulation, and the position logged from the experiment with the snake robot. We see that the snake robot moves slowly backwards along the e_x^I-axis in both the simulation and the experiment. The average speed of the link is approximately -1 cm/s. We see that the numerical simulations and the experimental results compare quite well even though locomotion is purely based on stick-slip contact with the ground. The reason for the slight difference in y-position may partly arise from the deadband in the DC-motor gears (see Section VIII-E).

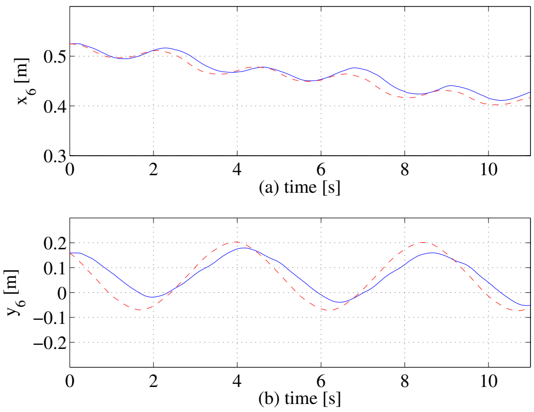


Fig. 13. Position of link 6 during lateral undulation on a flat plane: Simulation (dashed line) and experimental (solid line) results.

We see from the simulation and experimental results that the snake robot moves backwards and this compares well with the theory presented in [26]. Moreover, the motion of the snake robot exhibits a clear similarity to simulation and experimental results presented for a snake-like robot in [27]. In this latter paper, a model of a snake-like robot was simulated and experimental results were given for a snake-like robot where the longitudinal friction coefficient was *higher* than the transversal friction coefficient.

2) Obstacle Aided Locomotion: In this section we present results from an experiment with the snake robot without wheels moving using obstacle aided locomotion. We note that the snake robot is set to move with exactly the same motion pattern as in the previous subsection, however now it moves forward and much faster because of the obstacles that it pushes against. The position of link 6 for the simulation and the experiment are shown in Fig. 14. The obstacles are placed based on the elaboration in Section VII-B and VII-D. Moreover, the diameter of the obstacles are chosen such that they will not get stuck in the joints of the snake robot. All obstacles $j = 1, \ldots, \nu$, where $\nu = 7$, have radii $L_{H_j} = 0.0125$ m. The exact positions of the obstacles are included here for completeness. The (x, y)-positions are for obstacles $j = 1, \ldots, 7$ (all sizes in metres): (0.0507,0.1761), (0.4215, 0.0029), (0.8037, 0.1851), (1.1532, 0.0031), (1.5298, 0.1783), (1.8877, 0.0020), and (2.2588, 0.1745). The positions of the obstacles were used to synchronize (in the horizontal plane) the positions logged in the experiment and the positions found in the simulation. The snake robot was placed among the obstacles before it started to move as shown in Fig. 15 for $t = 0$. We see from Fig. 14 that the snake robot traverses the obstacles along the e_x^I-axis with approximately the same speed (ca. 15 cm/s) in the simulation as in the experiments. This is 15 times faster, and in the opposite direction, compared to the results without the obstacles. Another way of increasing the velocity of a snake robot on a flat surface is to attach passive caster wheels on the underside of the robot. However, this approach has some downsides. First of all the snake robot is more dependent on the ground surface condition for effective locomotion. Secondly, the speed of the snake robot is dependent on the angular frequency ω_h in (71) [9], and for a snake robot with passive wheels, the side-ways slip has to be considered when setting ω_h [4]. However, the only practical limitation for the wave propagation speed for obstacle aided locomotion with obstacles that do not yield is the speed of the joint motors. It is therefore possible to gain great velocities using obstacle aided locomotion. Note that the forward speed of the snake robot is also dependent on the motion pattern parameter δ_h. Hence, a different choice of δ_h, and thus a different setup of the obstacles, since the obstacle positions are dependent on the shape of the robot, would result in a different forward speed. Note that the path shown in Fig. 14 is almost identical to the simulation depicted in Fig. 9 with smaller obstacles. This is because the contact points between the snake robot and the obstacles are located at approximately the same position and the same motion pattern parameters were employed.

The actual snake robot is a little behind (ca 3 to 4 cm) the simulation results for any given point in time. This may be due to reasons discussed in Section VIII-E.

We see from comparing Fig. 13 and Fig. 14 that the main propulsive forces comes from the contact with the obstacles and not due to interaction with the ground since the snake robot moves forward much faster by exploiting the obstacles. Hence, the ability to move forward is less dependent on the condition of the friction with the ground surface when employing obstacles to push against. This is a desirable property of mobility and makes snake robot locomotion more robust with respect to the condition of the ground surface.

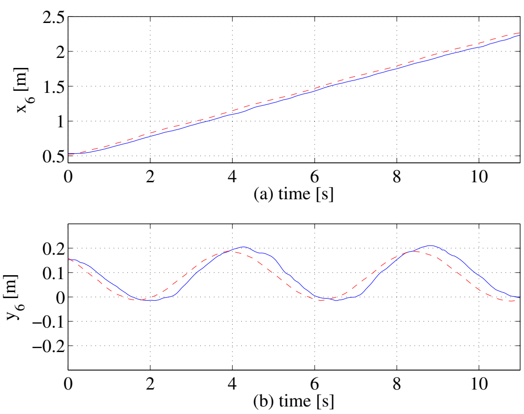


Fig. 14. Position of link 6 during obstacle aided locomotion: Simulation (dashed line) and experimental (solid line) results.

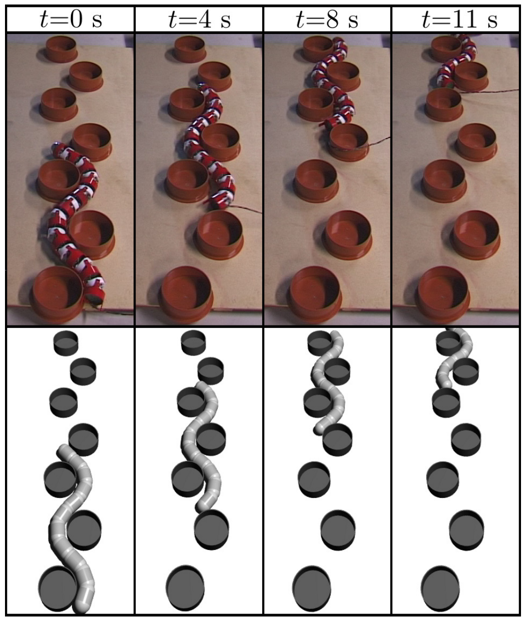


Fig. 15. Simulation results (bottom) and experimental validation (top) of obstacle aided locomotion.

E. Discussion of the Experimental Validation

Fig. 13 and Fig. 14 suggest that the mathematical model covers both the qualitative and quantitative aspects of the motion of the real snake robot. However, the figures depict a noticeable difference between the simulation and experimental results which may be explained by a combination of reasons: 1) There was a time-delay between the start-up of the snake robot and the start-up of the logging of position data. This delay ranged between 50 and 150 ms. 2) The model of the robot did not include a saturation in joint torques. Hence, the model might have been able to accelerate faster in the beginning of the obstacle aided locomotion than the actual robot. 3) There is a deadband of about $2 - 3$ degrees in the DC-motor gears. This results in that the control of the joint angles are not completely accurate and the joint angle might not be able to reach its desired angle. 4) It was difficult to place the snake robot in the exact same position on the particle board relative to the obstacles compared the placement of the snake robot in the simulation.

As mentioned earlier, approaches to how to control a snake robot during obstacle aided locomotion have been presented [2], [10]–[12]. However, a back-to-back comparison between simulation and experimental results has not been presented. We have shown that our model resembles obstacle aided locomotion for a real robot adequately and therefore it can be employed to test and further develop the locomotion schemes from e.g. [2], [10]–[12] together with the discussion given in this paper.

IX. Conclusions and Further Work

The results presented in this paper are steps towards developing snake robots that can navigate efficiently in narrow and chaotic environments, for example when searching for people in a collapsed building or inspecting narrow and possibly dangerous areas of industrial plants. In addition, the results can be used to better understand the working principle of biological snake locomotion.

A novel mathematical 2D model based on the framework of non-smooth dynamics and convex analysis has been developed for a snake robot in an environment with external objects in this paper. The model describes planar snake robot motion where the snake robot can push against external obstacles over the entire surface of each link. Hence, the effect of the contact forces between the obstacle and each link is modelled precisely. It has been shown how to easily incorporate the normal contact and friction forces into the system dynamics (i.e. the equality of measures) by set-valued force laws. The modeling framework allowed for accurately describing rigid body contact and collisions in a hybrid manner without having to explicitly switch between system equations. Moreover, it has been shown how to include correct directional Coulomb friction.

Experiments were performed with the snake robot Aiko, a robot with cylindrical links and no wheels. To validate the mathematical model, a back-to-back comparison was performed between numerical simulations and experimental results. The serpentine motion pattern 'lateral undulation' was used as a pre-programmed motion pattern for locomotion with and without obstacles in the simulations and in the experiments. The experimental and simulation results compared well both for obstacle aided locomotion and for locomotion without obstacles where the snake robot only interacted with the ground surface through gliding and sticking. These interaction forces between the snake and the ground make it harder to obtain a correct model of the environment compared to, for example, non-slip wheel-based locomotion, and the experimental validation indicates that the modeling approach presented in this paper is well suited for developing correct (and efficient) mathematical models for snake robot locomotion. Furthermore, we have illustrated from the results that a key to robust snake locomotion is to learn the snake robot (in the future) how to use obstacles to move forward both faster and more robustly compared to just undulating on a flat surface.

This paper has provided a framework based on non-smooth dynamics for the development, analysis, and testing of forthcoming motion planning and control approaches to obstacle aided locomotion. Further work will focus on developing motion patterns based on the mathematical model and observations of real snakes, that enable the snake robot to detect and exploit obstacles by itself for efficient locomotion.

X. Acknowledgment

We thank Terje Haugen and Hans Jørgen Berntsen for invaluable help in getting the mechanical parts of the snake robot finished in time. We thank Kristoffer Nyborg Gregertsen and Sverre Brovoll for the hardware and software design and implementation needed to get the snake robot to work. We thank the anonymous reviewers for valuable feedback.

References

[1] J. Gray, "The mechanism of locomotion in snakes," *J. Exp. Biol.*, vol. 23, no. 2, pp. 101–120, 1946.

[2] S. Hirose, *Biologically Inspired Robots: Snake-Like Locomotors and Manipulators*. Oxford: Oxford University Press, 1993.

[3] A. A. Transeth and K. Y. Pettersen, "Developments in snake robot modeling and locomotion," in *Proc. IEEE Int. Conf. Control, Automation, Robotics and Vision*, Dec 2006, pp. 1393–1400.

[4] P. Wiriyacharoensunthorn and S. Laowattana, "Analysis and design of a multi-link mobile robot (serpentine)," in *Proc. IEEE Int. Conf. Robotics, Intelligent Systems and Signal Processing*, vol. 2, December 2002, pp. 694–699.

[5] J. Ostrowski and J. Burdick, "Gait kinematics for a serpentine robot," in *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 2, April 1996, pp. 1294–1299.

[6] S. Ma, N. Tadokoro, B. Li, and K. Inoue, "Analysis of creeping locomotion of a snake robot on a slope," in *Proc. IEEE Int. Conf. Robotics and Automation*, September 2003, pp. 2073–2078.

[7] S. Ma, "Analysis of creeping locomotion of a snake-like robot," *Adv. Robotics*, vol. 15, no. 2, pp. 205–224, 2001.

[8] C. Ye, S. Ma, B. Li, and Y. Wang, "Turning and side motion of snake-like robot," in *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 5, 2004, pp. 5075–5080.

[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] S. Hirose and Y. Umetani, "Kinematic control of active cord-mechanism with tactile sensors," in *Proc. 2nd RoMAnSy Symp.*, Warsaw, 1976, pp. 241–252.

[11] 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.

[12] Z. Y. Bayraktaroglu, A. Kilicarslan, A. Kuzucu, V. Hugel, and P. Blazevic, "Design and control of biologically inspired wheel-less snake-like robot," in *Proc. IEEE/RAS-EMBS Int. Conf. On Biomed. Robotics Biomech.*, February 2006, pp. 1001–1006.

[13] B. Brogliato, *Nonsmooth Mechanics*, 2nd ed. London: Springer, 1999.

[14] Ch. Glocker, *Set-Valued Force Laws, Dynamics of Non-Smooth Systems*, ser. Lecture Notes in Applied Mechanics. Berlin: Springer-Verlag, 2001, vol. 1.

[15] R. I. Leine and H. Nijmeijer, *Dynamics and Bifurcations of Non-Smooth Mechanical Systems*, ser. Lecture Notes in Applied and Computational Mechanics. Berlin: Springer Verlag, 2004, vol. 18.

[16] J. J. Moreau, "Unilateral contact and dry friction in finite freedom dynamics," in *Non-Smooth Mechanics and Applications*, ser. CISM Courses and Lectures, J. J. Moreau and P. D. Panagiotopoulos, Eds. Wien: Springer Verlag, 1988, vol. 302, pp. 1–82.

[17] Ch. Glocker and C. Studer, "Formulation and preparation for numerical evaluation of linear complementarity systems in dynamics," *Multibody System Dynamics*, vol. 13, pp. 447–463, 2005.

[18] A. A. Transeth, R. I. Leine, Ch. Glocker, and K. Y. Pettersen, "Non-smooth 3D modeling of a snake robot with external obstacles," in *Proc. IEEE Int. Conf. Robotics and Biomimetics*, Dec 2006, pp. 1189–1196.

[19] C. Le Saux, R. I. Leine, and Ch. Glocker, "Dynamics of a rolling disk in the presence of dry friction," *Journal of Nonlinear Science*, vol. 15, no. 1, pp. 27–61, 2005.

[20] R. T. Rockafellar, *Convex Analysis*, ser. Princeton Landmarks in Mathematics. Princeton, New Jersey: Princeton University Press, 1970.

[21] A. A. Transeth, R. I. Leine, Ch. Glocker, and K. Y. Pettersen, "Non-smooth 3D modeling of a snake robot with frictional unilateral constraints," in *Proc. IEEE Int. Conf. Robotics and Biomimetics*, Dec 2006, pp. 1181–1188.

[22] P. Alart and A. Curnier, "A mixed formulation for frictional contact problems prone to Newton like solution methods," *Comput. Method. Appl. M.*, vol. 92, pp. 353–375, 1991.

[23] C. Mattison, *The Encyclopaedia of Snakes*. London: Cassell Paperbacks, 2002.

[24] H. Date, Y. Hoshi, and M. Sampei, "Locomotion control of a snake-like robot based on dynamic manipulability," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2000.

[25] O. Egeland, *Servo Engineering*. Tapir Forlag, 1993, in Norwegian.

[26] M. Nilsson, "Serpentine locomotion on surfaces with uniform friction," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2004, pp. 1751–1755.

[27] D. P. Tsakiris, M. Sfakiotakis, A. Menciassi, G. lla Spina, and P. Dario, “Polychaete-like undulatory robotic locomotion,” in *IEEE Int. Conf. Robotics and Automation*, 2005, pp. 3018 – 3023.

Aksel A. Transeth Aksel Andreas Transeth received his MSc degree in 2004 from the Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU), Trondheim, Norway. He is currently employed as a research scientist by the Norwegian research organization SINTEF at the Department of Applied Cybernetics. His research interests include modeling and control of dynamical systems, biologically inspired locomotion, and robotics.

Remco I. Leine Remco Leine is currently Privatdozent of Mechanics at the ETH Zurich. He received his MSc degree in Mechanical Engineering from the Technical University in Delft in 1996, and received his PhD under the guidance of Professor D. van Campen at the Technical University Eindhoven in 2000. In the following years, he visited the Technical University Munich (Professor F. Pfeiffer), the INRIA Rhône-Alpes (Professor B. Brogliato) and the ETH Zurich (Prof. Ch. Glocker) as a postdoctoral researcher. Since 2003, he is active as Senior Lecturer in the Institute of Mechanical Systems, ETH Zurich. In 2007 he obtained his Habilitation and was appointed as ‘Privatdozent’.

His research interests are on the intersection of Non-smooth Dynamics and Nonlinear Dynamics: stability theory for non-smooth systems, friction induced vibration and nonlinear phenomena, discontinuous bifurcations in non-smooth dynamical systems, multibody systems with impact and friction.

Christoph Glocker Christoph Glocker is currently Full Professor of Mechanics at ETH Zurich. He received the Diploma and Doctoral degree in mechanical engineering from Technische Universitt Mnchen in 1989 and 1995, respectively. In 1996 he has been granted with a Feodor Lynen Fellowship supported by the Alexander von Humboldt Foundation and spent one year at Aristotle University of Thessaloniki. After this stay, he returned in 1997 to TU Mnchen as a senior engineer to obtain in 2001 the Certificate of Habilitation in Mechanics by a postdoctoral thesis devoted to non-smooth dynamics, and to move to ETH Zurich afterwards.

A central point in his research is non-smooth dynamics of mechanical systems with finite degrees of freedom which includes, for example, the friction and impact problem in multibody systems, providing a generalization of classical analytical mechanics. He is interested in both, theoretical and practical questions concerning the mechanical modeling, the mathematical formulation, and the numerical treatment of systems with discontinuities, as well as their applications to industrial problems.

Kristin Y. Pettersen Kristin Ytterstad Pettersen received her MSc and PhD degrees in Electrical Engineering at the Norwegian University of Science and Technology (NTNU), Trondheim, Norway, in 1992 and 1996 respectively. She became Associate Professor in 1996 and in 2002 Professor at the Department of Engineering Cybernetics, NTNU. In 1999 she was a Visiting Fellow at the Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ. She has published more than 80 conference and journal papers. In 2006 she received the IEEE Transactions on Control Systems Technology Outstanding Paper Award. She is a senior member of IEEE and holds several board positions. Her research interests include nonlinear control of mechanical systems with applications to robotics, satellites, AUVs and ships.

Pål Liljebäck Pål Liljebäck received his MSc degree in 2004 from the Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU). He has been working as a research scientist for the Norwegian research organization SINTEF from 2004 to 2007. He is currently a PhD-candidate at the Department of Engineering Cybernetics at NTNU. His interests and main fields of expertise include modeling and control of dynamic systems and mechatronic design.