A step in the right direction

Modelling and analysis of a robotic leg with an energy storage system

Berend van der Grinten

28th February 2013

Summary

Dike monitoring is currently done via a labour intensive network of trained people. The recent unforeseen floods in Stein en Wilnis support the statement that the current monitoring system is not flawless. The goal of project RObotic SEnsor (ROSE) is to develop a team of walking robotic sensors that will autonomously inspect dikes. An energy storage system has been designed in order to increase the energy efficiency of the robot. Energy (such as impact energy) is stored in a rotational spring above the robotic leg. There are two actuation beams, which can reconfigure, attached to the spring. It has been researched to what extent this system can behave as an actuator and what the limitations are. The energy storage configuration has been analytically modelled using a method based on Lagrange's theorem applied to a multi-body variational method based on virtual work and generalised force. The model has been numerically checked. One of the main conclusions was that the system has its limitations. In order to let the robot travel along a chosen trajectory, continuous reconfiguration of the actuation beams is necessary. Not all trajectories can be travelled, limitations for the available paths of the leg are due to both the maximum elongation of the actuation beams, and the chosen restriction of adding energy to the system during reconfiguration. Furthermore, it is concluded that in order for the system to be useful in both stance and swing period, the payload of the leg should be relatively close to the mass of the leg itself. Despite the limitations mentioned before, it is expected that the designed system will create an energetic advantage. This advantage is expected because it is not necessary to use stored energy in any direction. Other actuators, which are necessary in the real world to account for friction losses, could be used only to actuate in the directions which are not possible with continuous reconfiguration.

Samenvatting

Dijkbewaking gebeurt op dit moment via een arbeidsintensief netwerk van getraind personeel. De recente onvoorziene overstromingen in Stein en Wilnis ondersteunen de stelling dat het huidige systeem van toezicht niet foutloos is. In project Robotic Sensor (ROSE) is het doel om een team van lopende robots te ontwikkelen die op autonome wijze dijken kunnen inspecteren. Een energieopslagsysteem is ontworpen om de energie-efficiëntie van de robot te verhogen. Energie (zoals impactenergie) wordt opgeslagen in een rotatieveer boven het been van de robot. Er zijn twee balken, die bevestigd zijn aan de veer die een kracht kunnen uitoefenen op de poot. Het is onderzocht in hoeverre dit systeem zich kan gedragen als een actuator voor de poot, en wat de beperkingen zijn. Het systeem is analytisch gemodelleerd met een methode die is gebaseerd op Lagrange's Theorem toegepast op een multi-body methode van virtuele arbeid en kracht. Het model is numeriek gecontroleerd. Geconcludeerd wordt dat het systeem zijn beperkingen heeft. Om de robot in een gekozen traject te laten volgen, is continue herconfiguratie van de actuatie balken noodzakelijk. Beperkingen voor de beschikbare paden komen vanwege zowel de maximale verlenging van de actuatie balken en van een gekozen beperking van het toevoegen van energie aan het systeem tijdens herconfiguratie. Geconcludeerd wordt ook dat, om het systeem effectief te gebruiken tijdens de periode dat het been in de lucht is, én de periode dat het been de robot draagt, de lading van het been relatief dicht bij de massa van het been zelf moet liggen. In de praktijk wordt verwacht dat het ontworpen systeem een energetisch voordeel zal hebben, ondanks de eerder genoemde beperkingen. Dit voordeel wordt verwacht omdat het niet nodig is om opgeslagen energie in elke richting te gebruiken. Andere actuatoren zijn sowieso nodig vanwege energieverlies dankzij wrijving, deze actuatoren kunnen gebruikt worden voor aandrijving in de richtingen die niet mogelijk zijn met continue herconfiguratie.

Preface

A step in the right direction. By means of this research I hope to contribute to a breed of energy efficient walking robots. Starting up the research was not easy. At the end of March 2012, one day after I agreed to start with the assignment, fate struck, when a good friend crushed my ankle and lower leg by accident. This, and some major recovering setbacks, made that the starting date had to be moved up to May, July, September and eventually October. Luckily matters got not too complicated with regards to the understanding and adaptability of my supervisor, Douwe. Moreover, when I started, the moments of consult and brainstorm felt more like teamwork rather than a boss-employee relationship. I would like to thank Douwe for his greatly stimulating and friendly coordination of this assignment. I would also like to thank the people of the research group for the friday-afternoon-drinks, the morning coffee, and the attentive people who made the work very enjoyable.

Yes, I liked working on this assignment, which is the first major individual thesis I had to write. I have learned to plan small research, to work individually, and most important, I have learned a bit about how to step out of a problem and try to look at the bigger picture in order to solve problems. I also learned it is quite hard to do 40 hours a week of modelling and debugging instead of 40 hours of regular study. I admit, there where some days where I thought that it would never work.

To all the readers of this thesis, I have done my very best to complete an interesting research. I do hope you will enjoy reading it.

Berend van der Grinten

Contents

1. Introduction

For reasons of soil fertility, transport and food supply a high percentage of the world population lives close to rivers, lakes, seas and oceans. Protection against tides and seasonal rises takes place in the forms of dunes, flood barriers, dikes and dams. A few inhabited areas lie below sea level, solely protected by natural or engineered structures. The result is a higher risk of casualties and economical impact when flooding occurs. In those areas monitoring of flood defence structures has to be done extensively.

1.1. Monitoring of dikes in the Netherlands

In the Netherlands over 9 million inhabitants live in areas which need flood defence mechanisms. These areas cover the majority of the Netherlands (Figure 1.1) . The areas and its inhabitants account for over 65% of the Gross National Product [1] giving major social and economical importance to flood defence structures. Monitoring of dikes is done with an extensive network of experts who try to spot early signs of possible failures such as the mechanisms in Table 1.1. Failures such as *piping* and *erosion* can be difficult to spot because those mechanisms happen inside the dike or below water levels. Recent unforeseen floods in the Netherlands as in Wilnis in 2003 and Stein in 2004, support the notion that the current monitoring system is not flawless.

1.2. Robotic monitoring of dikes

Project ROSE (RObotic SEnsor) has participants in the research group 'Robotics and Mechatronics' at the University of Twente and 'Discrete Technology & Production Automation' at the University of Groningen. The goal of the ROSE project is to develop a team of robotic sensors that will autonomously acquire data about the composition, consistency and condition of a dike. The focus is realisation of a team of ground vehicles. For reasons of slipping on steep slopes and possible harm to the surface of the dike,

Water level difference across structure	Erosion of surface by overflow
	Slip/slide
	Piping and/or internal erosion
	Erosion of landward or downstream slope
	Erosion at transition between structures
	Crest level too low - Overflow
Wave loading	Erosion of seaward face/slope
	Erosion of landward face/slope
	Local surface failure
	Crest level too low - Over topping
Lateral flow velocity	Erosion (scour) of bed or bank

Table 1.1.: Failure mechanisms for sea dikes[2]

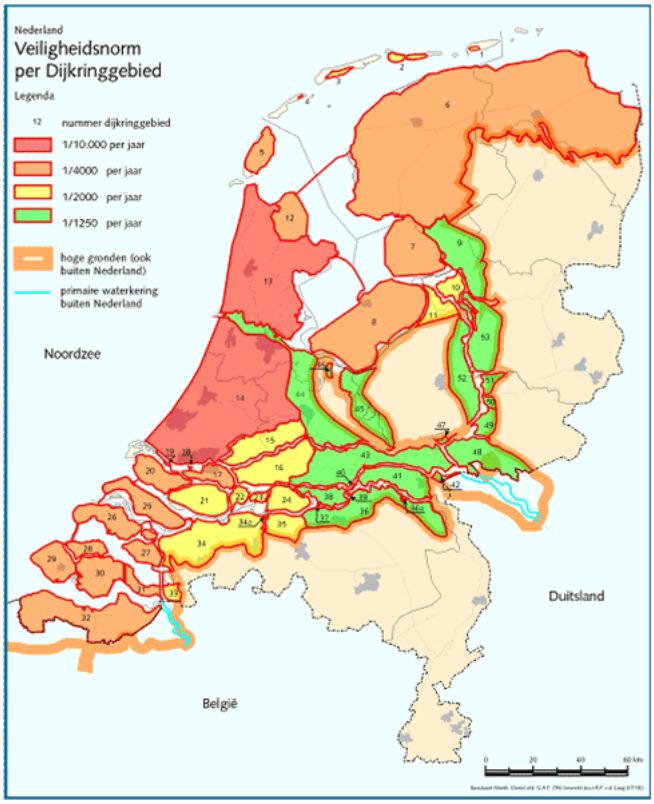


Figure 1.1.: Areas in the Netherlands below sea level including their safety standards [1]

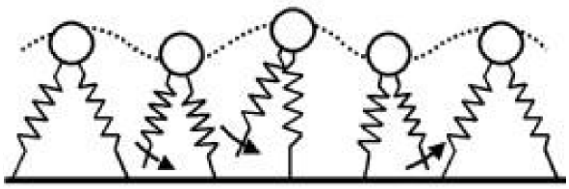


Figure 1.2.: Schematic figure of a robot which reuses its impact energy[3]

legged locomotion is preferred over wheeled and tracked locomotion. A challenge which arises with the choice for legged locomotion, is the relative low energy efficiency of current legged robots. With few resources on a dike and the aim to work completely independent for long periods of time, energy efficiency is a crucial topic for designing this robot.

1.3. A flexible energy storage system

Almost all walking robots lose energy when the foot touches the ground. Both the leg and the body of the robot can have a vertical speed which comes to an abrupt stop at each step. The energy of this impact is lost, thus the total energy efficiency of the robot will decrease. Some robots have been designed with a spring attached to the end of the leg in order to reuse the impact energy (See Figure 1.2). A major disadvantage of this simple configuration is that it only allows the stored energy to be used in the direction of the leg. A configuration which could use the stored energy in every direction is more desirable.

When using a rotational spring in combination with a rhombus (diamond) shaped

leg, it is possible to use the stored energy in any direction (details about the design are explained in Section 2.3). This conceptual model may be used in the leg of the dike inspection robot in order to increase its energy efficiency. This bachelor thesis aims to analyse and analytically model a robotic leg with a flexible energy storage system in order to evaluate the usability of this storage system.

1.4. Research questions

In order to scope the analysis three research questions have been set.

1. How should the system change its configurations during a walking cycle in order to effectively use an energy storage mechanism with a rotational spring?
2. What are the limitations of the system of the configuration to actuate the robot solely via the energy in the spring?
3. What are the design limitations for the mass of the body in relation with the mass of the leg in order for the stored energy to be used in the most versatile way?

1.5. Methodology

Using Maple©the system has been analysed using an analytical method found in Haug [4]. This method for calculation of the equations of motions is explained in section 2.5. The analytical approach is preferred over a numerical approach because the model of the system is highly conceptual. It is expected that for this conceptual model, analytic parameter correlations can give more information about design criteria and limitations. When working with a numeric modelling program such as 20-sim, parameter correlations can be hard to obtain.

The method of Haug has been chosen because it is specially designed for 2D constrained systems. In order to validate the model, the equations of motions have been set in 20-sim, with 3D-animation the visual output can be compared with the expected output. Evaluation of the model has been done with an algebraic simplified static version of the model and the 3D-animation resulting from the equations of motions.

1.6. Demarcation

This study only aims to analyse the idea of the rhombus shaped leg, not to change it. The mathematical displacement correlations between the endpoint of the leg and the corresponding configuration which was used in Chapter 2 was available.

For reasons of time limitations, a frictionless system has been analysed. In the model the origin has been taken as the endpoint of the leg which touches the ground. During stance period this is correct, for the swing period the same model with a massless robot has been used. This model is analogue to a model which origins at the robot.

2. Background

2.1. Walk cycles

Stability and energy efficiency are important properties of most robots. A useful tool to analyse these properties is the limit cycle, as it has been successfully used for different robots [5]. According to Stramigioli [6] a limit cycle is a periodic solution of a differential equation with the additional property that it is isolated. Periodic means that the system follows a closed trajectory, for example the path of the endpoint of a constant walking robotic leg. Isolated means that there are no other closed solutions near the limit cycle, so solutions will spiral towards the closed trajectory (stable limit cycle), diverge away from the closed trajectory (unstable) or a combination of both (semi-stable). Modelling of the system in this study has been done without friction or dissipation of energy, giving endless possible closed trajectories. So linguistically the leg which is modelled cannot be described by a limit cycle. In this study it will be referred to as *walk cycle*.

2.2. Phases of a walk cycle

To analyse the outcome of the results, the breakdown of a walk cycle in its phases is useful. In this study the breakdown as in Nishii [7] has been used:

Stride period The period of the movement cycle of one leg

Swing period The period for which a leg is not in contact with the ground

Stance period The period for which a leg applies a force to the ground to support the body

2.3. Conceptual design of the energy storage system

In most robotic legs, mechanical storage of energy is preferable over electrical storage of energy. This is because of the energy loss when gearboxes and motors are used to account for the high torques and small angle deflections. Mechanical energy storage and reuse of this energy within a two-dimensional robotic leg is needed. A design where all actuators are located close to each other is preferable. In this way, the mechanical energy stored via one direction could be used more easily in another direction since energy does not have to be transported. The available model which has such properties is shown in Figure 2.1. Where $l_{13} = l_{24}$ and $l_{12} = l_{34}$, which forms a rhombus. Point C can be controlled solely by the angles in point 1. Using basic geometry the relationship can be calculated:

$$\begin{bmatrix} Cx \\ Cy \end{bmatrix} = \begin{bmatrix} -l_{12}\sin(\phi_{a1}) - (l_{24} + l_{c4})\sin(\phi_{b1}) \\ -l_{12}\cos(\phi_{a1}) - (l_{24} + l_{c4})\cos(\phi_{b1}) \end{bmatrix} \tag{2.1}$$

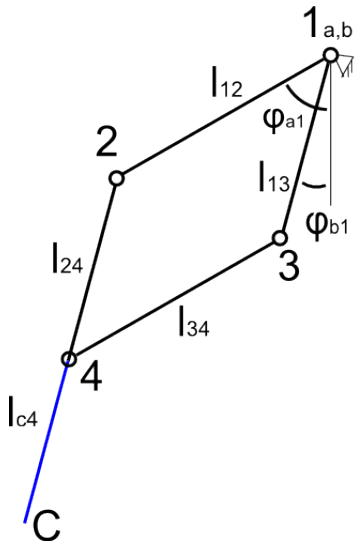


Figure 2.1.: A four bar linkage system configuration of a leg

2.3.1. Rotational spring storage system

For energy storage, a rotational spring is used which is positioned in the conceptual design at an 45° angle with respect to the x-axis as can be seen in Figure 2.2. Furthermore, beams A and B have been stretched and fitted with a rail system so the beams with lengths l_{57} and l_{67} can slide over respectively rail l_{15} and rail l_{16}. Rotational spring K has a locking system. Thus beams with lengths l_{57} and l_{67} can each freely rotate when spring K is locked, these beams also can change and lock its lengths. Although this model is mechanically challenging, it is worth analysing because theoretically it has some advantages:

- The position and movement of point C can be determined by changing the angles and lengths of l_{57} and l_{67};
- The spring can exchange its potential energy with kinetic energy in every direction of the 2D-plane;
- When the spring is locked the system can freely reconfigure in order to change the direction of movement when the lock is removed.

Using these properties a robotic leg can, in theory, be solely actuated by this single spring K, and impact energy losses of the body can be zero if the system reconfigures properly.

2.4. Evaluation of the design

With equation 2.1, one can calculate the angles of the bars in point 1. The correlations between the angles ϕ_{a1}, ϕ_{b1} and ϕ_{a7}, ϕ_{b7} can be created using the cosine rule and basic goniometric formulas:

$$\begin{bmatrix} \phi_{a1} \\ \phi_{b1} \end{bmatrix} = \begin{bmatrix} 1/4\,\pi + \arccos\left(1/2\,\frac{{l_{17}}^2+{l_{16}}^2-{l_{67}}^2}{l_{17}l_{16}}\right) \\ 1/4\,\pi - \arccos\left(1/2\,\frac{{l_{17}}^2+{l_{15}}^2-{l_{57}}^2}{l_{17}l_{15}}\right) \end{bmatrix} \tag{2.2}$$

Where:

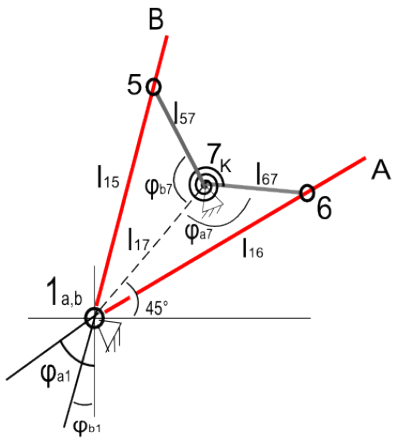


Figure 2.2.: Conceptual design of a flexible energy storage system

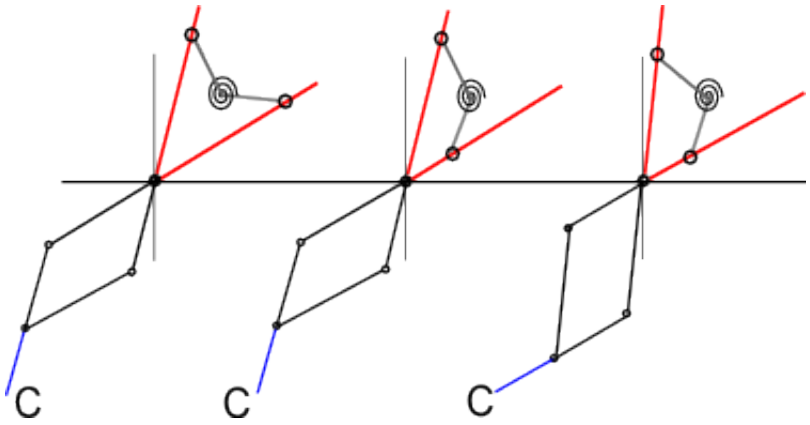


Figure 2.3.: Examples of possible configurations with same position for end connector C.

$$l_{16} = \sqrt{{l_{17}}^2 + {l_{67}}^2 - 2\, l_{17} l_{67} \cos\left(\phi_{a7}\right)} \tag{2.3}$$

And:

$$l_{15} = \sqrt{{l_{17}}^2 + {l_{57}}^2 - 2\, l_{17} l_{57} \cos\left(\phi_{b7}\right)} \tag{2.4}$$

Solving these formulas to create the correlations between Cx, Cy and ϕ_{a7}, ϕ_{b7} gives 8 solutions. This result is according to what was expected, because there are 2 different possible configurations for the same Cx and Cy. ϕ_{a1} and ϕ_{b1}, and for ϕ_{a1} and ϕ_{b1} each give 2 different configurations with ϕ_{a7} and ϕ_{b7}. As an example, three of these possible configurations are shown in Figure 2.3 .

2.5. Modelling methodology

The 2D-method found in Chapter 6 in Haug[4] has been used to analyse the analytical model. This method is based on *Lagrange's theorem* applied to the multi-body variational method based on *virtual work* and *generalised force*. This section will provide an introduction to this method.

D'Alembert's principle states that for the system to be in dynamic equilibrium the sum of differences between internal and external forces F_i is zero along virtual displacements. In other words, constraint forces add zero virtual work to the system.

$$\sum (m_i a_i - F_i)\, \delta\, r_i = 0 \tag{2.5}$$

This principle can be displayed in a more compact vectorial form for multi body systems as:

$$\delta \mathbf{q}^{\mathsf{T}} \left(\mathbf{M}\ddot{\mathbf{q}} - \mathbf{Q} \right) = 0 \tag{2.6}$$

Where:

$$\delta\mathbf{q} = \begin{bmatrix} \delta x_i \\ \delta y_i \\ \delta \phi_i \end{bmatrix}, \ddot{\mathbf{q}} = \begin{bmatrix} \ddot{x}_i \\ \ddot{y}_i \\ \ddot{\phi}_i \end{bmatrix}, \mathbf{M} = \begin{bmatrix} m_i & 0 & 0 \\ 0 & m_i & 0 \\ 0 & 0 & I_i \end{bmatrix}, \mathbf{Q} = \begin{bmatrix} F_{x_i} \\ F_{y_i} \\ \tau_{\phi_i} \end{bmatrix}$$

The kinematic and actuation constraints are written in the form of a constraint matrix:

$$\mathbf{\Phi}\left(\mathbf{q}, t\right) = 0 \tag{2.7}$$

Since virtual displacements $\delta\mathbf{q}$ occur with time held fixed, one can rewrite Eq. 2.7 to:

$$\mathbf{\Phi_q} \delta\mathbf{q} = 0 \tag{2.8}$$

Where matrix $\mathbf{\Phi_q}$ is the partial derivative of constraint vector $\mathbf{\Phi}\left(\mathbf{q}, t\right) = 0$ with respect to its state vector $\mathbf{q}$. So in order for the system to be *kinematically admissible,* Eq. 2.7 is evaluated with time held constant at a state $\mathbf{q}$. Eq. 2.6 holds for all virtual displacements $\delta\mathbf{q}$ that satisfy Eq. 2.8. These equations are known as the *variational equations of motions.* The matrix $\mathbf{\Phi_q}$ is better known as the Jacobian of the system.

2.6. Lagrangian constraints

Lagrange's theorem states that if $\mathbf{b}$ is an n-vector of constants, $\mathbf{x}$ is an n-vector of variables and $\mathbf{A}$ is an $n \times m$-matrix where:

$$\mathbf{b}^{\mathsf{T}}\mathbf{x} = 0 \tag{2.9}$$

which holds for all $\mathbf{x}$ that satisfy:

$$\mathbf{A}\mathbf{x} = 0 \tag{2.10}$$

there exists an m-vector of *Lagrange multipliers* such that:

$$\mathbf{q}^{\mathsf{T}}\mathbf{x} + \boldsymbol{\lambda}^{\mathsf{T}}\mathbf{A}\mathbf{x} = 0 \tag{2.11}$$

for arbitrary $\mathbf{x}$. Since Eq. 2.6 relates to Eq. 2.9, and similarly Eq. 2.8 relates to Eq. 2.10, it is possible to rewrite the multi body variational equations of motions in a form with Lagrange multipliers:

$$[\mathbf{M}\ddot{\mathbf{q}} - \mathbf{Q}]^\intercal \delta\mathbf{q} + \boldsymbol{\lambda}^\intercal \boldsymbol{\Phi}_\mathbf{q} \delta\mathbf{q} = 0 \tag{2.12}$$

Rewrite:

$$\left[\mathbf{M}\ddot{\mathbf{q}} + \boldsymbol{\Phi}_\mathbf{q}^\intercal \boldsymbol{\lambda} - \mathbf{Q}\right]^\intercal \delta\mathbf{q} = 0 \tag{2.13}$$

Therefore the coefficient of $\delta\mathbf{q}$ must be zero, and:

$$\mathbf{M}\ddot{\mathbf{q}} + \boldsymbol{\Phi}_\mathbf{q}^\intercal \boldsymbol{\lambda} = \mathbf{Q} \tag{2.14}$$

Since Eq. 2.7 must hold for all t, its derivative can be used to calculate the velocity $\dot{\boldsymbol{q}}$. Applying the chain rule for partial differentiation[1] gives:

$$\begin{gathered} \left(\frac{\partial \boldsymbol{\Phi}(\mathbf{q}, t)}{\partial t}\right) = \left[\frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{q}}\right]\left(\frac{\partial \mathbf{q}}{\partial t}\right) + \left(\frac{\partial \boldsymbol{\Phi}}{\partial t}\right) = \mathbf{0} \\ \boldsymbol{\Phi}_\mathbf{q}\dot{\mathbf{q}} = -\boldsymbol{\Phi}_t \end{gathered} \tag{2.15}$$

Differentiating both sides (again using the chain rule) obtains the acceleration $\ddot{\boldsymbol{q}}$ constraint equation:

$$\left(\frac{\partial^2 \boldsymbol{\Phi}(\mathbf{q}, t)}{\partial t^2}\right) = \left[\frac{\partial}{\partial \mathbf{q}}\left(\left[\frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{q}}\right]\left(\frac{\partial \mathbf{q}}{\partial t}\right)\right)\right]\left(\frac{\partial \mathbf{q}}{\partial t}\right) + 2\left[\frac{\partial^2 \boldsymbol{\Phi}}{\partial \mathbf{q}\partial t}\right]\left(\frac{\partial \mathbf{q}}{\partial t}\right) + \left[\frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{q}}\right]\left(\frac{\partial^2 \mathbf{q}}{\partial t^2}\right) + \left(\frac{\partial^2 \boldsymbol{\Phi}}{\partial t^2}\right) = \mathbf{0} \tag{2.16}$$

Rewritten and $\boldsymbol{\gamma}$ assigned to the right part of the equals sign:

$$\boldsymbol{\Phi}_\mathbf{q}\ddot{\mathbf{q}} = -(\boldsymbol{\Phi}_\mathbf{q}\dot{\mathbf{q}})_\mathbf{q}\dot{\mathbf{q}} - 2\boldsymbol{\Phi}_{\mathbf{q}t}\dot{\mathbf{q}} - \boldsymbol{\Phi}_{tt} \equiv \boldsymbol{\gamma} \tag{2.17}$$

γ can be calculated with the Jacobian and the velocity or as a function of $\mathbf{q}$. Combining the equations of 2.14 and 2.17 results in the total system of equations:

$$\begin{bmatrix} \boldsymbol{M} & \boldsymbol{\Phi}_\mathbf{q}^\intercal \\ \boldsymbol{\Phi}_\mathbf{q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{Q} \\ \boldsymbol{\gamma} \end{bmatrix} \tag{2.18}$$

With only $\ddot{\mathbf{q}}, \boldsymbol{\lambda}$ as unknowns, the dynamics of the system can be calculated.

[1] $\frac{\partial}{\partial t}[f(x(t), t)] = \frac{\partial f}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial f}{\partial t}$

3. Modelling

This chapter will cover the application of the method described in Section 2.5 to the system as described in Section 2.3.

3.1. Notation

To set the actuation and constraint equations for the system, the notation as seen in Figure 3.1 has been used. Every rigid single body is modelled via its centre of mass. Length has been taken as the distance between the end of the beam until its centre of mass. Every beam is labelled with a letter starting from A. The position of the bodies can be put in equations in a manner as:

$$\mathbf{\Phi}\left(\mathbf{q}\right) = \begin{bmatrix} x_A(t) - L_A\cos(\phi_A(t)) \\ y_A(t) - L_B\sin(\phi_A(t)) \\ x_B(t) - 2L_A\cos(\phi_A(t)) - L_B\cos(\phi_B(t)) \\ y_B(t) - 2L_B\sin(\phi_A(t)) - L_B\sin(\phi_B(t)) \end{bmatrix} = \mathbf{0} \tag{3.1}$$

which holds for al t. Since there are no driving constraints this is equivalent to Eq. 2.7. These equations give every solution for the position of the beams.

The denomination of the full system is shown in Figure 3.2, where beam R is appointed to a beam which represents the robot. Again, all beams are modelled at their centres of mass. For reasons of distinctness, lengths and masses are left out, their denomination is analogue to Figure 3.1.

3.2. Constraints

The positions of the beams can be put into constraints in an equal manner as in Eq. 3.1. The complete list of constraints is shown in Appendix A, as an example constraint equations for $x_{A_1}, y_{A_2}, x_{C_1}$ and y_F are shown in Eq. 3.2.

$$\mathbf{\Phi}(\mathbf{q}, t) = \begin{bmatrix} x_{A_1} - L_{A_1}\cos(\phi_A) \\ y_{A_2} - (2L_{A_1} + L_{A_2})\sin(\phi_A) \\ x_{C_1} - x_{A_2} - 2L_B\cos(\phi_B) \\ y_F - x_R - L_R\sin(\phi_R) - L_F\sin(\phi_F) \end{bmatrix} = \mathbf{0} \tag{3.2}$$

For simplicity it has been used that A_2, B, C_1, D_1 form a rhombus. Using this rhombus property, the notation simplifies to:

$$\phi_A \equiv \phi_{A_1} = \phi_{A_2} = \phi_{C_1} = \phi_{C_2} \qquad \text{and} \qquad \phi_B = \phi_{D_1} = \phi_{D_2} \tag{3.3}$$

Notice that the position of beams E and F is calculated via beam R instead of calculation via beam C_2 and respectively beam D_2. This was done because of the slider function of beams C_2 and D_2. The total system has 3 degrees of freedom ϕ_E, ϕ_F and ϕ_R. With 10 single bodies with each 3 degrees of freedom, 27 constraint equations are needed to describe the equations of motions. With x and y-positions constraint as shown in Eq 3.2,

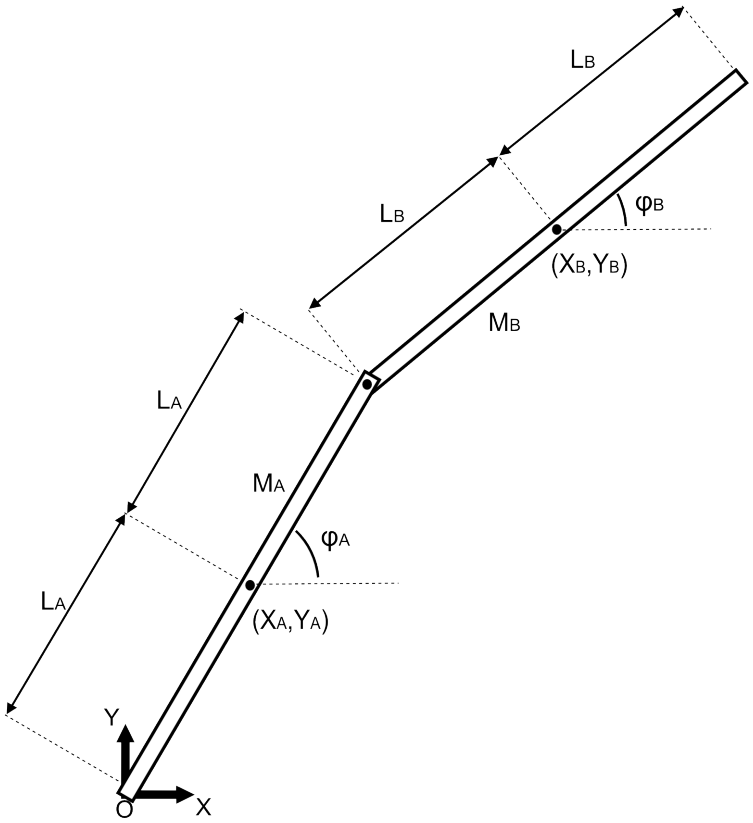


Figure 3.1.: Used notation for creating the constraint equations

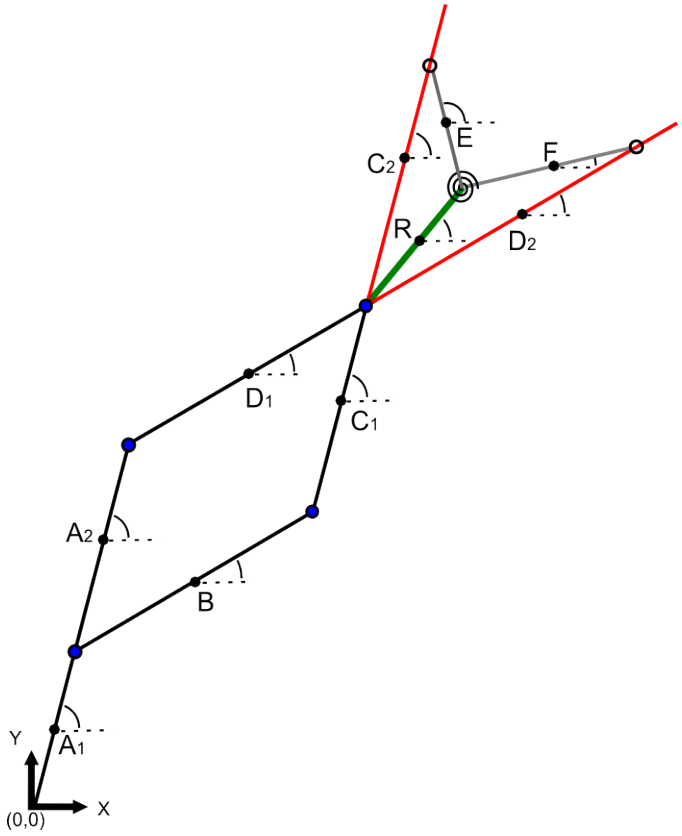


Figure 3.2.: Naming of the beams

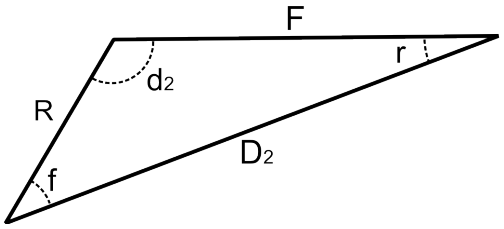


Figure 3.3.: Notation to assemble Eq 3.4

20 constraints have been taken care of. Equation 3.3 takes care of 5 more. In order to constrain the relations between ϕ_{C_2} and ϕ_E and analogue ϕ_{D_2} and ϕ_F the sine-rule has been used (see Figure 3.3), it states that:

$$\frac{D_2}{\sin d_2} = \frac{R}{\sin r} = \frac{F}{\sin F} \tag{3.4}$$

Take:

$$\begin{array}{rcl} f & = & \phi_R - \phi_{D_2} \\ r & = & \phi_{D_2} - \phi_F \\ F & = & L_F \\ R & = & L_R \end{array} \tag{3.5}$$

Which implies that:

$$L_R \sin(\phi_R - \phi_{D_2}) - L_F \sin(\phi_{D_2} - \phi_F) = 0 \tag{3.6}$$

Above equation relates angles ϕ_R and ϕ_{D_2} in order for beams R, D_2 and F to form a triangle. In a similar way for the triangle C_2, E, R:

$$L_R \sin(\phi_{C_2} - \phi_R) - L_E \sin(\phi_E - \phi_{C_2}) = 0 \tag{3.7}$$

The constraint vector $\mathbf{\Phi}(\mathbf{q}, t) = 0$ is set up. The Jacobian $\mathbf{\Phi_q}$ can be calculated. Take for vector $\mathbf{q}$ (length is 25, using Eq. 3.3) and vector $\mathbf{\Phi}(\mathbf{q}, t)$ (length is 22):

$$\mathbf{q} = \begin{bmatrix} x_{A_1} \\ y_{A_1} \\ x_{A_2} \\ y_{A_2} \\ x_B \\ \vdots \\ \phi_A \\ \phi_B \\ \phi_E \\ \phi_F \\ \phi_R \end{bmatrix}, \qquad \mathbf{\Phi}(\mathbf{q}, t) = \begin{bmatrix} x_{A_1} - L_{A_1} \cos(\phi_A) \\ y_{A_1} - L_{A_1} \sin(\phi_A) \\ x_{A_2} - (2L_{A_1} + L_{A_2}) \cos(\phi_A) \\ y_{A_2} - (2L_{A_1} + L_{A_2}) \sin(\phi_A) \\ x_B - 2L_{A_1} \cos(\phi_A) - L_B \cos(\phi_B) \\ \vdots \\ L_R \sin(\phi_R - \phi_B) - L_F \sin(\phi_B - \phi_F) \\ L_R \sin(\phi_A - \phi_R) - L_E \sin(\phi_E - \phi_A) \end{bmatrix} \tag{3.8}$$

Then the partial derivative can be calculated:

$$\mathbf{\Phi_q} = \begin{bmatrix} \frac{\partial \mathbf{\Phi}_1}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{\Phi}_1}{\partial \mathbf{q}_2} & \cdots & \frac{\partial \mathbf{\Phi}_1}{\partial \mathbf{q}_{25}} \\ \frac{\partial \mathbf{\Phi}_2}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{\Phi}_2}{\partial \mathbf{q}_2} & \cdots & \frac{\partial \mathbf{\Phi}_2}{\partial \mathbf{q}_{25}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{\Phi}_{22}}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{\Phi}_{22}}{\partial \mathbf{q}_2} & \cdots & \frac{\partial \mathbf{\Phi}_{22}}{\partial \mathbf{q}_{25}} \end{bmatrix} \tag{3.9}$$

The condition for kinematically admissible virtual displacements of Eq. 2.8 is:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \dots & L_{A_1}\sin(\phi_A) & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \dots & -L_{A_1}\cos(\phi_A) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & (2L_{A_1}+L_{A_2})\sin(\phi_A) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \dots & -(2L_{A_1}+L_{A_2})\cos(\phi_A) & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & L_E\cos(\phi_A-\phi_E)+L_R\cos(\phi_A-\phi_R) & 0 & -L_E\cos(\phi_A-\phi_E) & 0 & -L_R\cos(\phi_A-\phi_R) \end{bmatrix} \begin{bmatrix} \delta x_{A_1} \\ \delta y_{A_1} \\ \delta x_{A_2} \\ \delta y_{A_2} \\ \vdots \\ \delta\phi_A \\ \delta\phi_B \\ \delta\phi_E \\ \delta\phi_F \\ \delta\phi_R \end{bmatrix} = \mathbf{0} \tag{3.10}$$

3.3. Assembly of the analytical model

The diagonal matrix $\mathbf{M}$ in Eq. 2.18 stands for the inertia with respect to the state vector $\boldsymbol{q}$. In x and y-direction it is just the mass of the corresponding beam. For rotations it is a bit more complex. Since we accumulated $\phi_{A_1}, \phi_{A_2}, \phi_{C_1}, \phi_{C_2}$ to ϕ_A, the sum of the individual inertias corresponds to the inertia of a changing ϕ_A. Thus making matrix $\mathbf{M}$[1].

$$\mathbf{M} = \begin{bmatrix} m_{A_1} & 0 & 0 & \dots & 0 & \dots & 0 \\ 0 & m_{A_1} & 0 & \dots & 0 & \dots & 0 \\ 0 & 0 & m_B & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \frac{1}{3}(m_{A_1}L_{A_1}^2 + m_{A_2}L_{A_2}^2 + m_{C_1}L_{C_1}^2 + m_{C_2}L_{C_2}^2) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & K_{RF}\frac{1}{3}m_R L_R^2 \end{bmatrix} \tag{3.11}$$

Where K_{RF} is a rotation factor which corrects for the robot whose shape is not a beam and in all probability has a higher rotational inertia. External force vector $\mathbf{Q}$ is constructed with gravity in y-directions of every beam and a torque at beam E and F applied by spring K:

$$\mathbf{Q} = \begin{bmatrix} 0 \\ -m_{A_1}g \\ 0 \\ 0 \\ -m_{A_2}g \\ \vdots \\ -K_s(\phi_E - \phi_F + \theta_0) \\ K_s(\phi_E - \phi_F + \theta_0) \\ 0 \end{bmatrix} \tag{3.12}$$

[1] For full matrix, see Appendix A.

Where K_s is the spring constant and θ_0 the initial rotation of the spring. For setting up the equations of motions like in Eq. 2.18 only $\boldsymbol{\gamma}$ needs yet to be calculated. Eq 2.17 can be simplified because the partial derivative to the time is zero using the time independent actuation of the spring, resulting in:

$$\boldsymbol{\gamma} = -(\boldsymbol{\Phi}_{\mathbf{q}}\dot{\mathbf{q}})_{\mathbf{q}}\dot{\mathbf{q}} \tag{3.13}$$

Applied to the system:

$$\boldsymbol{\gamma} = \begin{bmatrix} -L_{A_1}\cos(\phi_A)\dot{\phi}_A^2 \\ -L_{A_1}\sin(\phi_A)\dot{\phi}_A^2 \\ -(2L_{A_1}+L_{A_2})\cos(\phi_A)\dot{\phi}_A^2 \\ -(2L_{A_1}+L_{A_2})\sin(\phi_A)\dot{\phi}_A^2 \\ \vdots \\ L_E\left(\dot{\phi}_A-\dot{\phi}_E\right)^2\sin(\phi_A-\phi_E)+L_R\left(\dot{\phi}_A-\dot{\phi}_R\right)^2\sin(\phi_A-\phi_R) \end{bmatrix} \tag{3.14}$$

Filling of Eq. 2.18 results in a theoretically solvable system of 47 equations with 47 unknowns.

3.4. Numerical model

Although the model as described in section 3.3 describes the motion of the beams, it has to be solved in order to validate its behaviour. Since algebraically solving a system with 47 equations can be quite difficult (even with the appropriate software) a numerical validation has been created. The program 20-sim has been used, more specifically the *constraint*-function. Using this function it is possible to calculate a parameter using a specific boundary condition. Instead of combining Eq. 2.14 and Eq. 2.17 into a system, it is less complex if $\boldsymbol{\lambda}$ can be calculated without the acceleration $\dot{\boldsymbol{q}}$ meaning the systems algebraic loops can be eliminated. Resulting in a far less complicated calculation. This is numerically done when the velocity equation (Eq. 2.15) is the input as a constraint for $\boldsymbol{\lambda}$:

$$\begin{aligned} \boldsymbol{v} &= \boldsymbol{\Phi}_{\mathbf{q}}\dot{\mathbf{q}} + \boldsymbol{\Phi}_t = \mathbf{0} \\ \boldsymbol{\lambda} &= constraint(\boldsymbol{v}) \end{aligned} \tag{3.15}$$

Which embeds Eq. 2.15 into $\boldsymbol{\lambda}$. Using bond graphs it can be easily seen that using the *constraint*-function simplifies calculation, as seen in Fig. 3.4 there are no loops.

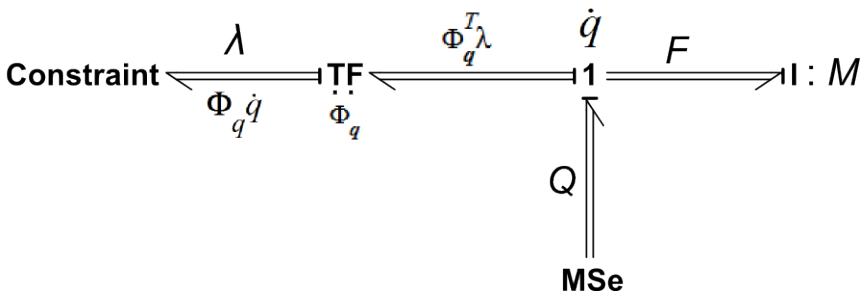
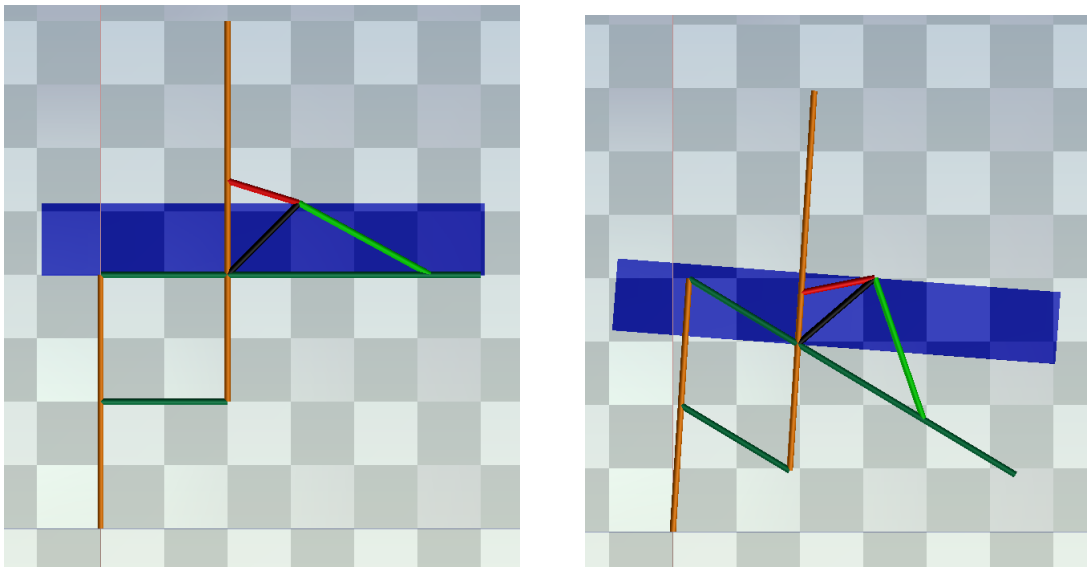


Figure 3.4.: Bond graph model with constraint function



(a) Configuration at $t = 0$ (b) Configuration at $t = 0.6$

Figure 3.5.: Leg without a spring

3.5. Implementation of the constraint function

For successful implementation, the system has to be rewritten in order to fit the method of assigning variables in 20-sim. The system of Eq. 2.15 can be seen as:

$$\begin{aligned} \mathbf{F} &= -\mathbf{\Phi}_{\mathbf{q}}^{\mathrm{T}}\boldsymbol{\lambda} + \mathbf{Q}; \\ \ddot{\mathbf{q}} &= \mathbf{M}^{-1}\mathbf{F} \end{aligned} \tag{3.16}$$

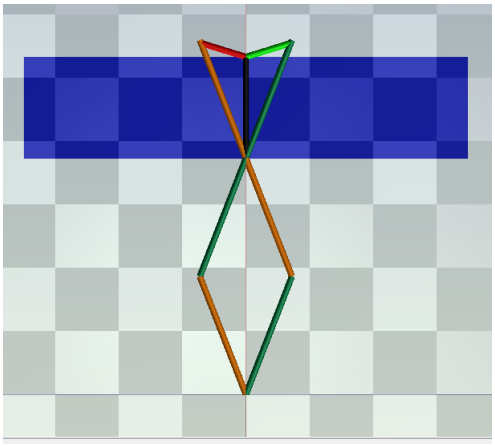
Since $\mathbf{\Phi}_t$ is zero for constraints and actuation independent of time Eq. 3.16 together with Eq. 3.15 describes the bond graph as in Figure 3.4. This is the method used in 20-sim (detailed preview of the modelling method can be seen in Appendix B).

3.6. Validation

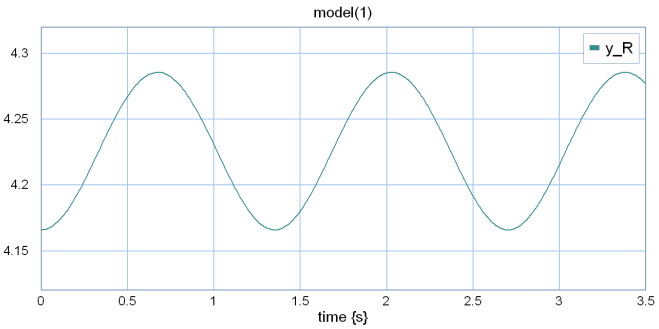
The model with the structure as in Section 3.5, can be checked for correctness with a 3D-Animation. In Figure 3.5 the configuration is shown at $t = 0$ and at $t = 0.6$. The blue object represents the robot, which is attached to the black beam. Because the spring is eliminated, the trajectory is highly predictable; it will just fall towards the ground (x-axis). Furthermore, it can be seen that the actuation beams E and F slide in a correct way over the beams of the leg.

When a spring is attached it can be seen that it is possible to direct the robot upwards using the spring. L_{A_1} has been set to zero so it can also be checked that the forces in x-direction sum up to zero. The configuration at $t = 0$ and the response are shown in Figure 3.6. Other observations which indicates the models correctness are:

- The robot has an initial acceleration of -9.81 with spring stiffness K_S and/or initial spring rotation θ_0 set at zero
- Graphically it has been checked that in the 3D Animation the beams stay attached, indicating the correct constraint relations.



(a) Configuration for testing spring response



(b) Response for the height of the robot in meters

Figure 3.6.: Simplified configuration with a spring

4. Evaluation

This chapter will use the model as explained in Chapter 3, to analyse the characteristics of the proposed mechanism. In Sections 4.1 and 4.2, the possibility is considered to actuate the leg in such a way that it will follow a chosen trajectory. In Section 4.5, the maximum payload of the leg is analysed.

4.1. Stance period

For measurement tasks such as mapping soil layers, maintaining the height of the robot during the walk cycle can be an advantage. For that reason, the proposed mechanism has been analysed to be used as such an actuator that the robot will travel along a chosen horizontal path during stance period.

The chosen path, and its corresponding begin and end configuration, are shown in Figure 4.1a. Where the angle of the robot (ϕ_R) has been chosen as constant. Or, the centre of mass of the robot maintains both its height and its angle with respect to the ground. Thus, the system has one degree of freedom, which can be expressed in ϕ_A, ϕ_B, ϕ_E or ϕ_F. The angles of beams E and F for the chosen trajectory (and initial conditions) are plotted in Figure 4.1b.

With relative ease, the model, as explained in Chapter 3, can be inverted[1]. This model can be used to analyse the time varying spring stiffness, required for the robot to travel the desired trajectory. Although the system has one degree of freedom, two actuators have to be used, since both ϕ_A and ϕ_B has to change with a different relation than the single linear spring. To account for that second actuator, it is proposed that the spring attached to beams E and F should be split into two springs. One spring attached to the robot and beam E, and one spring attached to the robot and beam F. These springs have K_{sE} and K_{sF} as their spring stiffness.

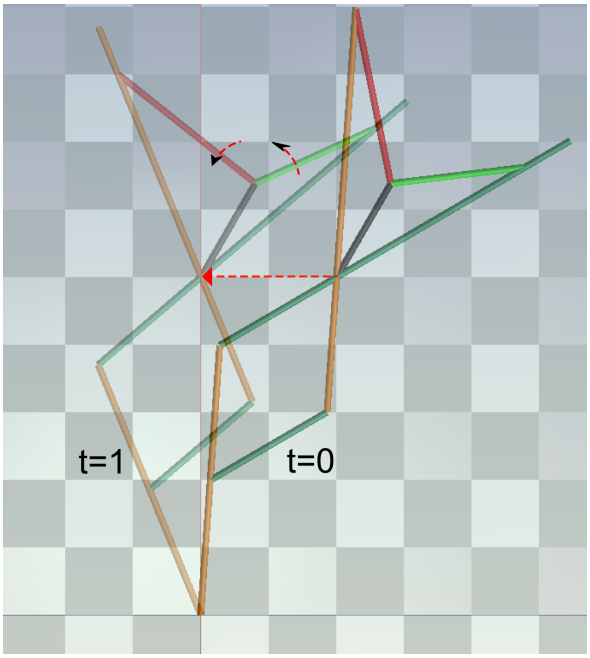
The output of the inverted model results in the relation between the spring stiffness as a function of time, in order for the robot to behave as in Figure 4.1a. This result is shown in Figure 4.2. For $t = 0.693$, the spring stiffness of both of the springs is the same. This means that at $t = 0.693$, it is possible to use only one spring, to actuate the system. However, for other t, it is not possible to use a single spring (with this configuration) since K_{sE} varies heavily in time. With the fixed lengths of beams E and F the proposed split of the spring thus is necessary to be able to travel the desired trajectory.

In order to use a single spring, the proposed mechanism could reconfigure to account for the time varying spring stiffness. In general, it can be concluded that for the robot (with a single energy storage system) to be able to travel a desired (randomly chosen) trajectory, continuous reconfiguration of the system is essential.

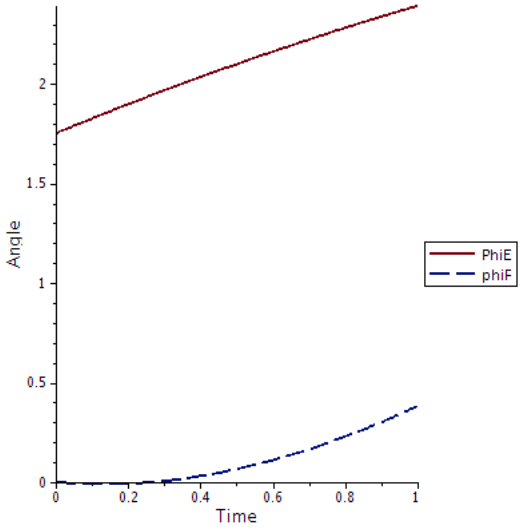
4.2. Continuous reconfiguration

Continuous reconfiguration of the system is possible when actuators are used to change the lengths and angles of the actuation beams. Actuators for elongation and rotation

[1] The method for inverting the model is shown in Appendix B



(a) The chosen path with its begin, and end configuration



(b) Angles of the actuation beams

Figure 4.1.: Step with same hight of the robot

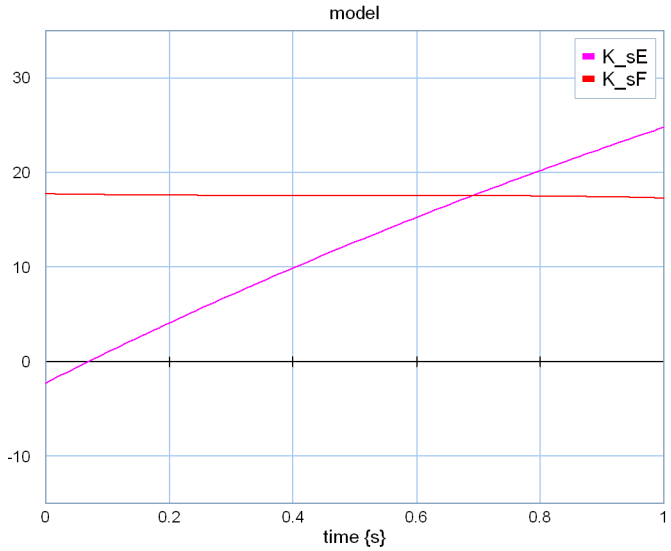


Figure 4.2.: Spring stiffness for K_{sE} and K_{sF} for a chosen horizontal trajectory

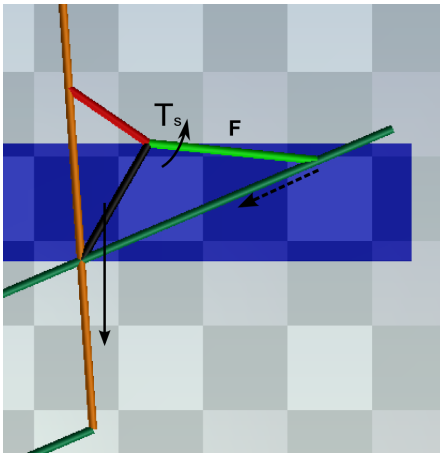


Figure 4.3.: Example where reconfiguration of beam F doesn't require work

of beams E and F are present in the conceptual design. It is the intention to use these actuators *only* to reconfigure beams E and F. It is *not* the intention to use them to actuate the legs of the robot. This could wind up the spring, adding energy to the system. That would counteract to the advantage of the rotational spring as an actuator. Thus, it is useful to evaluate continuous reconfiguration without the need to add energy into the system.

Both the angles and lengths of the actuation beams can only be adjusted when no work has to be put into the system. For instance, when the robot is falling down and a torque T_S is applied by the spring (see Figure 4.3), elongation of beam F does not require work since the spring is unwinding.

Let us go back to the straight trajectory as described in Section 4.1. It was concluded that, in order to travel the chosen horizontal path, the system had to reconfigure continuously. However, since K_{sF} is approximately constant, reconfiguration of beam F is not expected to be necessary and has been neglected for simplification. To evaluate if continuous reconfiguration of beam E can be done without adding energy to the system, one should look at the resulting torque T_R, at the robot, caused by the varying spring stiffness. This resulting torque should be the same for a time varying spring stiffness, as for a continuously reconfigured beam with a constant spring stiffness.

Recalling the torque T_E, applied by the spring to beam E:

$$T_E = K_{sE}(\phi_{E_{new}} - \phi_R + \theta_0) \tag{4.1}$$

Where *new* stands for the continuously reconfigured angle. It can be derived that the resulting torque T_R, depends on the torque of the spring, T_E, the length of beam E, L_E, and the distance between the end point of beam E and the robot, L_{RE}. Using the notation as in Figure 4.4 one can derive the relation:

$$T_R = \frac{L_E}{L_{RE}} \cos(\phi_E - \phi_A) T_E \tag{4.2}$$

Because L_{RE} is not a state variable, one should apply the sine rule[2] to obtain:

$$L_{RE} = \frac{L_E}{\sin(\phi_A - \phi_R)} \sin(\phi_E - \phi_R) \tag{4.3}$$

[2]See Figure 3.3

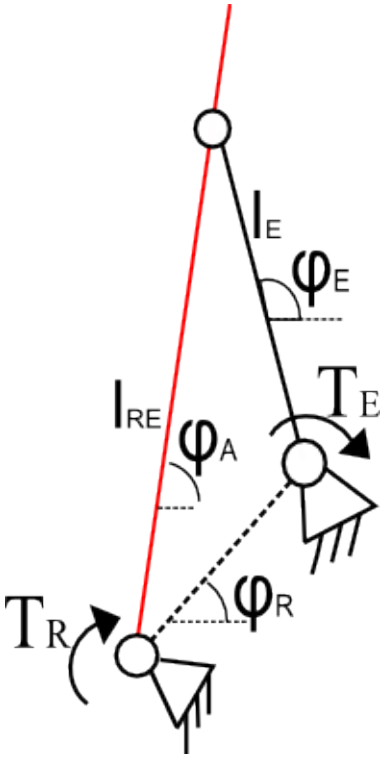


Figure 4.4.: Simplified actuation beam

Combining Eq. 4.1, Eq. 4.2 and Eq 4.3 and simplifying yields:

$$T_R = K_{sE}\frac{\sin(\phi_A - \phi_R)\cos(\phi_A - \phi_E)(\phi_E - \phi_R + \theta_0)}{\sin(\phi_E - \phi_R)} \tag{4.4}$$

In order to follow the chosen horizontal path, the resulting torque T_R should be the same for continuous reconfiguration ($\phi_{E_{new}}$, constant K_{sEnew}) as for the case as in Figure 4.2 ($\phi_{E_{old}}$, K_{sEold} varying with time).

$$K_{sEold}(t)\frac{\cos(\phi_A - \phi_{E_{old}})(\phi_{E_{old}} - \phi_R + \theta_0)}{\sin(\phi_{E_{old}} - \phi_R)} = K_{sEnew}\frac{\cos(\phi_A - \phi_{E_{new}})(\phi_{E_{new}} - \phi_R + \theta_0)}{\sin(\phi_{E_new} - \phi_R)} \tag{4.5}$$

Note that when $\phi_{E_{old}} \neq \phi_{E_{new}}$, $L_E(t)$ will either elongate or shrink. K_{sEnew} has been set to $\pm 17,5$, for the reason described in Section 4.1. Solving this relation for the only unknown; $\phi_{E_{new}}$, results in Figure 4.5. As can be seen, the behaviour of the rotation of beam E is completely different in order to account for the constant springs stiffness.

4.3. Existence of L_E

The angle of $\phi_{E_{new}}$ as seen in Figure 4.5, is not sufficient for conclusions without taken account for the relation of ϕ_E with the length of beam E. This length has been calculated using (again) the sine rule and is plotted in Figure 4.6a. The discontinuity at about $t = 0.82$ makes sense since:

$$L_E = \frac{L_R}{\sin(\phi_E - \phi_A)}\sin(\phi_A - \phi_R) \tag{4.6}$$

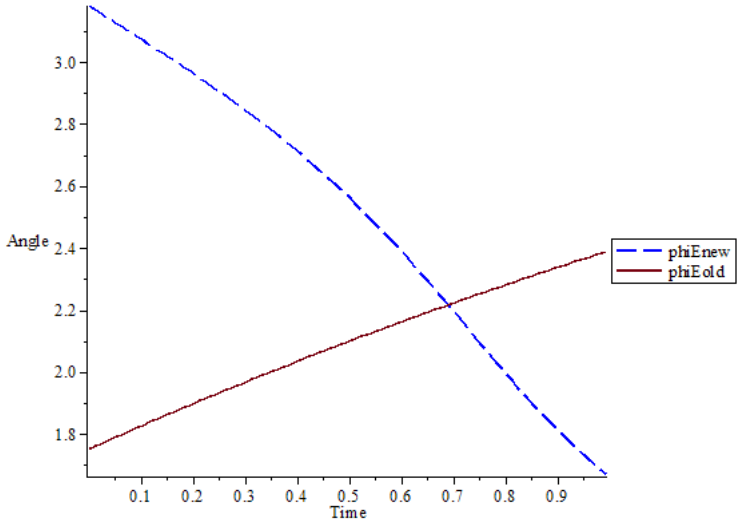


Figure 4.5.: $\phi_{E_{new}}$ plotted with $\phi_{E_{old}}$

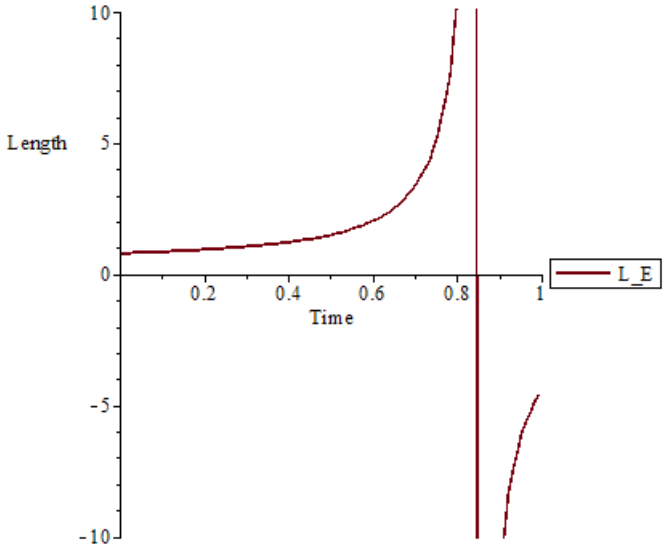
Eq 4.6 states that L_E will go to infinity if $\phi_E = \phi_A$, and $\phi_A \neq \phi_R$. For the chosen trajectory, it can be seen in Figure 4.6b that when, $\phi_{E_{new}}$ is equal to ϕ_A, L_E goes to infinity. Because beam E, in reality, can only elongate till a certain length, one can conclude that it is not possible to follow the entire chosen trajectory. There is a certain zone around $t = 0.82$ where the trajectory can not be travelled. In general, one can conclude that with the proposed mechanism, certain trajectories can not be travelled even with continuous reconfiguration.

4.4. Reconfiguration without adding energy to the system

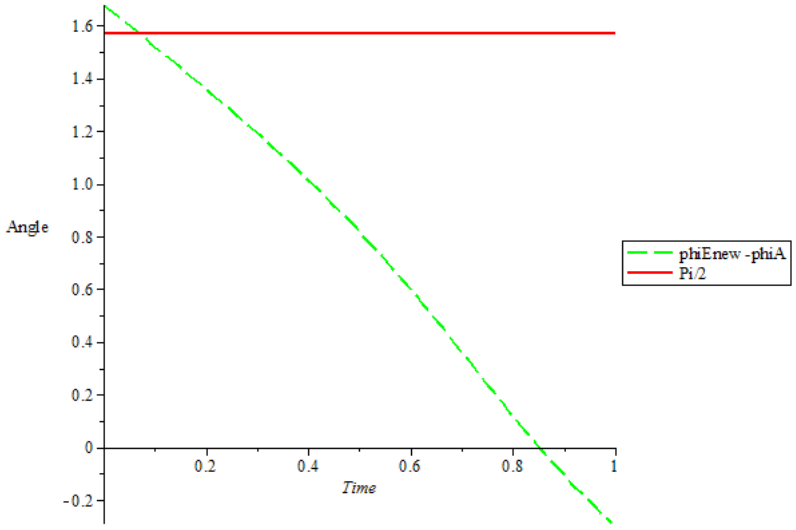
For the behaviour as described in the previous Section, one can also evaluate the possibility to travel the trajectory without adding energy to the system. In order to use energy from the spring to reconfigure, the spring should unwind. If the spring winds up, energy is added. As mentioned in Section 4.2, this counteracts the intention of the proposed design.

Since the spring has been split, the rotation is defined as: $\phi_E - \phi_R + \theta_0$. With both the initial angle θ_0, and the angle of the robot ϕ_R, set as constants[3], the change in potential energy of the spring depends only with ϕ_E. In Figure 4.5 it can be seen that $\phi_{E_{new}}$ only decreases, thus the rotation of the spring will only decrease in time. This means that for this specific trajectory, continuous reconfiguration is not limited by the energy supply. The energy, for reconfiguration, comes from the spring, which unwinds itself. However, with this trajectory inverted, the angle of beam E would be increasing adding energy to the system. One can conclude that the restriction of adding energy to the system during reconfiguration reduces the trajectories which can be travelled by the spring actuated robot.

[3]See Section 4.1

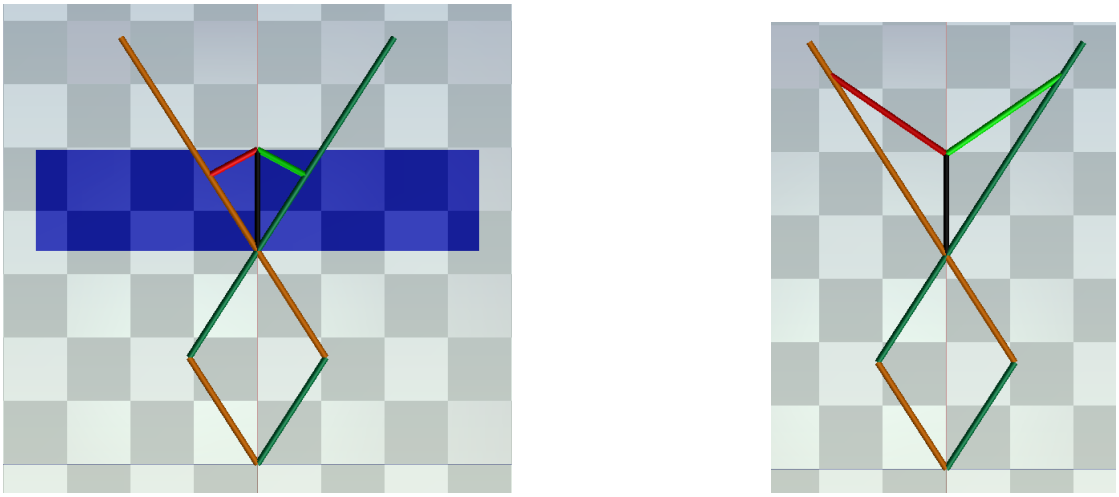


(a) The length of beam E



(b) Difference of $\phi_{E_{new}}$ and ϕ_A

Figure 4.6.: Existence of L_E



(a) (b)

Figure 4.7.: Simplified systems in equilibrium

4.5. Full use of the spring during walk cycles

Ideally, the system is useful in both stance and swing period. During stance period, the spring has to account for the mass which is carried by the leg; its payload. During swing period, the spring only has to actuate the leg with its smaller mass. In order to be useful at both of these stages of the walk cycle, the system has to reconfigure in order to account for this change in actuated weight. A simplified system worth investigating, is the system in Figure 4.7a, where the actuation beams E and F are chosen near their shortest position. This offset (instead of their shortest position) has been chosen for a reason. Recalling Eq. 4.2, the resulting torque at the robot (T_R) is zero for the singularity at: $\phi_E - \phi_A = \pm k\pi + \frac{\pi}{2}$. Or, in other words, the torque felt at the robot is zero as a result of the right angle between beam E and beam F.

The initial angle θ_0 and spring constant K_s are chosen such that the robot maintains its height; the system is in equilibrium. The mass, which is supported by this leg, is chosen to be 50 times the mass of the single leg. When the mass of the robot is removed, the system will have to reconfigure to a configuration such as shown in Figure 4.7b. Since the angle ϕ_E decreases, and the angle ϕ_F increases, the resulting torque at the robot will decrease, which is the desired behaviour. In the system, with the carried mass 50 times the mass of the leg, and the change in configuration exactly as in Figures 4.7a, the simulation reveals that, to be in equilibrium, $K_{s_{reconfigured}}$ is about 4% of the K_s required for the previous equilibrium. Thus in this specific case, it is not possible to use the stored energy in both the stance and swing period of the walk cycle because the spring stiffness should change to account for the change in payload.

Algebraically this result can be extended to a more general relation. The system of Equation 2.18 simplifies since $\ddot{\boldsymbol{q}}$ and $\boldsymbol{\gamma}$ are zero for equilibrium states. This simplified system can be solved for the constants as in the symmetric Figure 4.7a. Then, the algebraic relation is obtained for the length of the beams L_E/L_F (required for equilibrium) as a function of the carried mass during swing period[4]. The results are shown in Figure 4.8. The line crosses the y-axis at $L = \pm 34$. The initial length can be seen at the point where the mass m_R is 50. This is for $L = \pm 0.54$. Thus, in order for the system to reconfigure

[4]Since this relation is very long and complicated it is not algebraically shown in this report

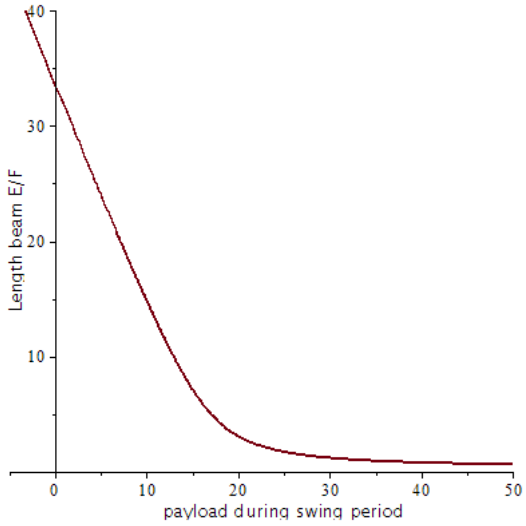


Figure 4.8.: Robotic mass for equilibrium as a function of the lengths of the actuation beams

to account for a carried mass of 50 times the mass of the leg, the elongation of beams E and F has to be 63 times their original length. This confirms the numerically derived indication that a 'normal' elongation of beams E and F as in Figure 4.7b, is not sufficient to account for a change in carried mass.

In order for the system to be useful in equilibria in both stages of the walk cycle the following relation must be solved:

$$T_{spring}(L_{E_{short}}, m_R) = T_{spring}(L_{E_{long}}) \tag{4.7}$$

Where the torque T_{spring} should be the same in both situations because the system only reconfigures without loosening or winding of the spring. The initial conditions are kept the same as in Figure 4.7a. Solving this relation results in the relation between the elongation of beams E and F as a function of the ratio Carried/Own mass of the leg. This relation can be seen in Figure 4.9. If, for instance, the maximum elongation of the beams is three times its original length (the dotted line), then the mass which can be carried in equilibrium is about 5 times the mass of the leg itself. This relation gives a major design criterion of keeping the robots mass low with respect to the mass of its legs.

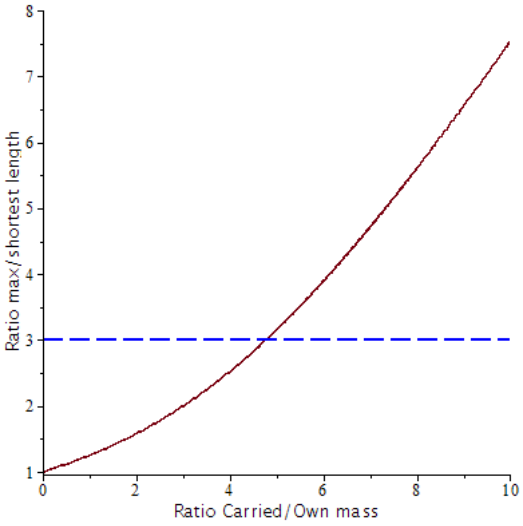


Figure 4.9.: Elongation necessary for equilibrium during swing period

5. Discussion

The energy storage system has been analysed, the torques needed for certain motions have been calculated and, inversely, the output when a certain torque is applied. The theoretical flexibility has been proven by the calculations. This can be concluded since continuous reconfiguration extends the possible trajectories. However, it is not possible to travel along any chosen path. Both the maximal elongation of the beams (Sections 4.3 and 4.5), and the restriction of adding energy to the system (Section 4.4, restrains the system to travel along any random trajectory. Thus the design is less flexible than initially thought.

In practise, this does not mean that the proposed mechanism can not create an energetic advantage during a walk cycle. In almost every real system (including a robot), friction occurs at (for example) the moving joints. Thus with every step, energy will be lost in heat. In order to keep a constant gait, actuators will always be necessary to account for friction losses. In practise, it may be possible to account for the friction losses with 'new' energy in one direction and use stored energy with the proposed system to actuate in other directions.

The flexibility of the system concerning the mass of the leg with respect to the carried mass also has its limitations. In practise this also could be less of a problem when friction losses are accounted. The robot can use its stored energy only when it has an energetic advantage, which could be during stance or swing period.

6. Conclusions and recommendations

With the commentary mentioned in the previous chapter, one can conclude that:

- It has been proven that it is possible to actuate a leg in the 2D-plane using the described energy storage system.
- Limitations for the available paths of the leg are due to; both the maximum elongation of the actuation beams, and the restriction of adding energy to the system during reconfiguration.
- To be able to use the full potential of the system, the mass of the robot should be not to large compared to the mass of the leg (see Figure 4.9).
- In practise the system could, with the remarks concerning the mass of the robot and continuous reconfiguration, still create a theoretical advantage when friction losses are implemented.

It is recommended that the research is extended in the direction of the following topics:

- Evaluate the possible paths using continuous reconfiguration without adding energy to the system.
- Investigate the possibility to lock the spring in the middle, thus giving the possibility to temporarily use one spring for each actuation beam.
- Investigate the possibility to use a non linear spring. The linear spring evaluated in this research could not deliver the large changes in necessary torque. With a non linear spring, it is possible to have higher change in torque when the rotation of the spring increases. This maybe useful in order to account for larger allowed difference in the carried weight and the mass of the leg itself.
- Investigate the possibility to wind up the spring as the only way to actuate the robot. There may be trajectories for which this is possible even with the limitations as mentioned above. Such trajectories could be very efficient since the actuator, for winding of the spring, could have a constant winding speed.
- Model the system with friction and actuators to measure the actual energetic advantage when, for instance, impact energy is reused.
- The theoretic advantage of the analytic method used for modelling, has not made full use of. Algebraically solving the equations of motions was not as easy as thought, even with the appropriate software. It is recommended that the full analytical solution is further evaluated.

A. Constraint equations of the system (Maple)

$$\mathbf{\Phi}(\mathbf{q},t) = \begin{bmatrix} x_{A_1} - L_{A_1}\cos(\phi_A) \\ y_{A_1} - L_{A_1}\sin(\phi_A) \\ x_{A_2} - (2L_{A_1} + L_{A_2})\cos(\phi_A) \\ y_{A_2} - (2L_{A_1} + L_{A_2})\sin(\phi_A) \\ x_B - 2L_{A_1}\cos(\phi_A) - L_B\cos(\phi_B) \\ y_B - 2L_{A_1}\sin(\phi_A) - L_B\sin(\phi_B) \\ x_{C_1} - (2L_{A_1} + L_{A_2})\cos(\phi_A) - 2L_B\cos(\phi_B) \\ y_{C_1} - (2L_{A_1} + L_{A_2})\sin(\phi_A) - 2L_B\sin(\phi_B) \\ x_{C_2} - (2L_{A_1} + 2L_{A_2} + L_{C_2})\cos(\phi_A) - 2L_B\cos(\phi_B) \\ y_{C_2} - (2L_{A_1} + 2L_{A_2} + L_{C_2})\sin(\phi_A) - 2L_B\sin(\phi_B) \\ x_{D_1} - (2L_{A_1} + 2L_{A_2})\cos(\phi_A) - L_B\cos(\phi_B) \\ y_{D_1} - (2L_{A_1} + 2L_{A_2})\sin(\phi_A) - L_B\sin(\phi_B) \\ x_{D_2} - (2L_{A_1} + 2L_{A_2})\cos(\phi_A) - (2L_B + L_{D_2})\cos(\phi_B) \\ y_{D_2} - (2L_{A_1} + 2L_{A_2})\sin(\phi_A) - (2L_B + L_{D_2})\sin(\phi_B) \\ x_E - (2L_{A_1} + 2L_{A_2})\cos(\phi_A) - L_B\cos(\phi_B) - L_E\cos(\phi_E) - 2L_R\cos(\phi_R) \\ y_E - (2L_{A_1} + 2L_{A_2})\sin(\phi_A) - L_B\sin(\phi_B) - L_E\sin(\phi_E) - 2L_R\sin(\phi_R) \\ x_F - (2L_{A_1} + 2L_{A_2})\cos(\phi_A) - L_B\cos(\phi_B) - L_F\cos(\phi_F) - 2L_R\cos(\phi_R) \\ y_F - (2L_{A_1} + 2L_{A_2})\sin(\phi_A) - L_B\sin(\phi_B) - L_F\sin(\phi_F) - 2L_R\sin(\phi_R) \\ x_R - (2L_{A_1} + 2L_{A_2})\cos(\phi_A) - L_B\cos(\phi_B) - L_R\cos(\phi_R) \\ y_R - (2L_{A_1} + 2L_{A_2})\sin(\phi_A) - L_B\sin(\phi_B) - L_R\sin(\phi_R) \\ L_R\sin(\phi_R - \phi_B) - L_F\sin(\phi_B - \phi_F) \\ L_R\sin(\phi_A - \phi_R) - L_E\sin(\phi_E - \phi_A) \end{bmatrix} \tag{A.1}$$

B. 20-sim models

Since 20-sim has problems with the calculation of large vectors in combination with the constraint function, the model for calculating the response when a certain torque is applied has de following structure (Mm is the inertia matrix.)

```
            //v=PHI_q*q_dot+ PHI_t
v[1]        =q_dot[1]+L[1]*sin(q[21])*(q_dot[21]);
v[2]        =q_dot[2]-L[1]*cos(q[21])*(q_dot[21]);
v[3]        =q_dot[3]+(2*L[1]+L[2])*sin(q[21])*(q_dot[21]);
.
.
v[22]       =(-L[9]*cos(-q[24]+q[22])+
              L[10]*cos(-q[25]+q[22]))*(q_dot[22])
             -L[10]*cos(-q[25]+q[22])*(q_dot[25])
             +L[9]*cos(-q[24]+q[22])*(q_dot[24])                    ;

lambda=constraint(v);

            //F =-transpose(PHI_q)*lambda + Q;
F[1]        =-lambda[1];
F[2]        =-lambda[2]-M[1]*g;
F[3]        =-lambda[3];
.
.
F[25]       =   -2*L[10]*sin(q[25])*lambda[15]
                          +2*L[10]*cos(q[25])*lambda[16]
                          -2*L[10]*sin(q[25])*lambda[17]
                          +2*L[10]*cos(q[25])*lambda[18]
                            -L[10]*sin(q[25])*lambda[19]
                            +L[10]*cos(q[25])*lambda[20]
                    +L[10]*cos(q[21]-q[25])*lambda[21]
                -L[10]*cos(-q[25]+q[22])*lambda[22]                 ;

q_dotdot=inverse(Mm)*F;
q_dot=int(q_dotdot);
q=int(q_dot,q_0);
```

The structure of the inverted model has the structure as below. Where the input of the system ϕ_A, ϕ_B, ϕ_E, ϕ_F, and its derivatives, have been algebraically calculated using maple. This was done because second derivatives cause numeric problems. After that, the calculation is straightforward resulting in the desired Torques and spring stiffness which vary with the time.

```
t=time;

phiA=atan2(....expression with variable t..)
phiB=atan2(....expression with variable t..)
phiE=atan2(....expression with variable t..)
phiF=atan2(....expression with variable t..)
phiR              = 1;

q        =  [L[1]*cos(phiA)   ;          L[1]*sin(phiA)   ;
(2*L[1]+L[2])*cos(phiA)   ;(2*L[1]+L[2])*sin(phiA)   ;
                                        .
                                        .
                                        .
phiA ;   phiB    ;       phiE ;  phiF ; phiR ];

phiA_dot =....expression with variable t.. ;
phiB_dot =....expression with variable t.. ;
phiE_dot =....expression with variable t.. ;
phiE_dot =....expression with variable t.. ;
phiR_dot = 0;

-q_dot = [L[1]*sin(q[21])*(phiA_dot);
                  -L[1]*cos(q[21])*(phiA_dot);
  (2*L[1]+L[2])*sin(q[21])*(phiA_dot);
-(2*L[1]+L[2])*cos(q[21])*(phiA_dot);
                                                              .
                                                              .
                                                              .
-phiA_dot; -phiB_dot; -phiE_dot; -phiF_dot; -phiR_dot];

q_dotdot=ddt(q_dot,q_dotdot_0);\

F = Mm*q_dotdot;

lambda[1] = -F[1]
;
lambda[2] = -F[2]                -M[1]*g                    ;
lambda[3] = -F[3]
;
lambda[4] = -F[4]                -M[2]*g                    ;
.
.
lambda[20] = -F[20]              -M[10]*g                          ;
lambda[21] =(expression varies with q[21..25],F[21] and lambda[1..20];
```

```
lambda[22] =(expression varies with q[21..25],F[22] and lambda[1..20];

F_K_E = -F[23] -L[8]*sin(q[23])*lambda[15]
                                  +L[8]*cos(q[23])*lambda[16]
                            +L[8]*cos(q[21]-q[23])*lambda[21];
F_K_F=  F[24] +L[9]*sin(q[24])*lambda[17]
                           -L[9]*cos(q[24])*lambda[18]
         +L[9]*cos(-q[24]+q[22])*lambda[22];

K_sE    =F_K_E/(q[23]-q[25]+theta_0);
K_sF    =F_K_F/(-q[24]+q[25]+theta_0);
```

Bibliography

[1] W. ten Brinke and B. Bannink, "Risico's in bedijkte termen, een thematische evaluatie van het nederlandse veiligheidsbeleid tegen overstromen," p. 242, 2004.

[2] C. Mai Van, P. van Gelder, and J. Vrijling, "Failure mechanisms of sea dikes: inventory and sensitivity analysis," 2007.

[3] F. Iida, Y. Minekawa, J. Rummel, and A. Seyfarth, "Toward a human-like biped robot with compliant legs," *Robotics and Autonomous Systems*, vol. 57, no. 2, pp. 139 – 144, 2009, <ce:title>Selected papers from 9th International Conference on Intelligent Autonomous Systems (IAS-9)</ce:title> <xocs:full-name>9th International Conference on Intelligent Autonomous Systems</xocs:full-name>. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889007001765

[4] E. Haug. Boston: Allyn and Bacon, 1989.

[5] D. Hobbelen and M. Wisse, "Limit cycle walking," 2007.

[6] S. Stramigioli and M. D. van, "Energy conservative limit cycle oscillations," in *Proceedings of the 17th World Congress The International Federation of Automatic Control*. Korea: IFAC, July 2008, pp. 15 666–15 671. [Online]. Available: http://doc.utwente.nl/64882/

[7] J. Nishii, "An analytical estimation of the energy cost for legged locomotion," *Journal of theoretical biology*, vol. 238, no. 3, pp. 636–645, 2006, cited By (since 1996): 7.