

Algorithms for implementation of mechanical systems control

Author: Haroldo Levi Ibarra González

Supervisor: Sergej Čelikovský

Czech Technical University

Faculty of Electrical Engineering

Department of Control Engineering

June 2012

Contents

1	Introduction	1
1.1	Objectives of the thesis	1
1.2	Organization of the thesis	2
2	Mechanical systems	3
2.1	Simple underactuated mechanical systems	3
2.2	Human gait and robotics	4
2.3	Euler Lagrange Method	5
3	Modeling the 4-link	7
3.1	General model of Acrobot and 4-link	7
3.1.1	The first link	10
3.1.2	The second link	11
3.1.3	Kinetic Energy of the Acrobot	14
3.1.4	Potential Energy	15
3.1.5	Lagrangian	16
3.1.6	Extending the Acrobot to the 4-link	17
3.2	Real 4-link modeling	18
3.2.1	Elements of the real Actrobot	18
3.2.2	Center of mass and moment of inertia	19
4	Advanced Walking Control	22
4.1	Control of 4-link walker based on the method of virtual constraints	22
4.1.1	Control of 4-link model based on three virtual constraints	23
4.1.2	Control of 4-link model based on two virtual constraints	28
4.2	Defining the constrainting functions for the 4-link model	31
4.2.1	Regression approach	32
4.2.2	Lagrange Polynomial	37
4.2.3	Increased Lagrange Polynomial	43
4.3	Comparison of control algorithms for 4-link model	48
5	Conclusions	52

List of Figures

2.1	The Acrobot.	4
3.1	The 4-link.	8
3.2	Real Acrobot at CVUT.	9
3.3	Representation of the Acrobot for energy calculations.	10
4.1	4-link as an Acrobot.	26
4.2	Angular positions of 4-link controlled as an Acrobot.	27
4.3	Angular velocities of 4-link controlled as an Acrobot.	27
4.4	Animation of the 4-link as an Acrobot.	28
4.5	Stop by position control in Matlab.	31
4.6	Dependance of angle q_2 on q_1	32
4.7	Dependance of angle q_3 on q_1	33
4.8	Step with regression approach, \dot{q}_1 of 1.5 rad/sec and 0.115 time simulation.	34
4.9	Velocities with regression approach, \dot{q}_1 of 1.5 rad/sec and 0.115 time simulation.	35
4.10	Angle trajectory with regression approach, \dot{q}_1 of 1.5 rad/sec and 0.115 time simulation.	35
4.11	4-link animation sequence with regression approach.	36
4.12	New trajectory of q_2 on q_1	38
4.13	New trajectory of q_3 on q_1	38
4.14	New trajectory of q_4 on q_1	39
4.15	Example of a longer step.	40
4.16	Example of a shorter step.	41
4.17	Example of a step of proper lenght but with misalignment.	41
4.18	Example of velocities going to infinity.	42
4.19	4-link animation sequence with Lagrange Polynomial approach.	42
4.20	4 Points Lagrangian Polynomial q_2 on q_1	44
4.21	4 Points Lagrangian Polynomial q_3 on q_1	44
4.22	4 Points Lagrangian Polynomial q_4 on q_1	45
4.23	4 Points Lagrangian Polynomial's best step.	46
4.24	4 Points Lagrangian Polynomial velocities for best step.	46
4.25	4 Points Lagrangian Polynomial angles for best step.	47
4.26	4-link animation sequence with Lagrange Polynomial of 4 points approach.	48
4.27	Comparison of q_1 and q_2 for Acrobot embedding method.	49
4.28	Comparison of q_3 and q_4 for Acrobot embedding method.	49

4.29	Comparison of dq_1 and dq_2 for embedded and three virtual constraints methods.	50
4.30	Comparison of dq_1 and dq_2 for embeded and virtual constrain. . .	51
4.31	Comparison of step for embedded (red) and virtual constraints methods (black).	51

Acknowledgements

I would like to thank the Consejo Nacional de Ciencia y Tecnología (CONACyT) of Mexico for their financial support during the study of my Master program. I owe sincere and earnest thankfulness to Professor Sergej Čelikovský who was my supervisor and who guided me through my thesis work, and I also would like to show my gratitude to Milan Anderle who helped me to through the development of this work.

Abstract

Underactuated mechanical systems have less actuators than degrees of freedom and have been of particular interest during last decades. There are many different and well know underactuated mechanical systems that have been used to apply new concepts in non-linear control. One of the most well known is the Acrobot. The Acrobot is one of the simplest underactuated mechanical systems and consists of two links with only one actuator between them; it is seen as the simplest model for walking robots. In this thesis the model of the Acrobot is extended to the 4-link in order to make a more realistic model for a legged robot and the control will be done using the concept of virtual constraints. It will be shown how the 4-link can be controlled to achieve a stable tracking of the stable target walking trajectory. A model will be developed based on a real 4-link testbed to further practical implementation of the achieved control.

Chapter 1

Introduction

Efficient biped walking robots are still a huge challenge for engineering today. Despite the improvements and development of several walking robots there are still many challenges to make this robots efficient in terms of energy consumption. Many fully actuated walking robots calculate every position in order to maintain balance which is an unnatural way of walking. Underactuated systems represent a more natural approach to emulate walking. Nevertheless, control of non-linear systems still require the development of new concepts since much of the work done does not apply to a big branch of underactuated mechanical systems [13].

Although there are many different ways to move a robot, like wheels or 4 legged robots, biped robots present some advantages, for example they are highly adaptable to different types of grounds. Humanoid robotics are of interest since one of the main reasons of robots is to replace humans on activities that are dangerous because of the hazardous environment. Even NASA has a robot called robonaut and showed intentions of sending a humanoid walking robot to the moon with the Project M. Nevertheless, the challenge of designing a walking robot is difficult despite it has received lot of attention during the last decades. Biped robots are also of particular interest in the category of legged robots because understanding their dynamics can help develop orthopedic devices to help humans with disabilities [14].

1.1 Objectives of the thesis

The main objective of the thesis is to explore the concept of virtual constraints applied to underactuated mechanical systems, in particular to the 4-link to make it perform a single step and develop a realistic model of this system based on a real one at Czech Technical University in Prague (CTU). Two types of virtual constraints methods will be compared to see how they can be used to restrict the number of degrees of freedom (DOF) in the system. Detailed mathematical explanations of how the virtual constraints are obtained and how the control is achieved will be shown. Complete explanation how to obtain a realistic model will be done as well. In summary, the present work can be divided into 3 main objectives. The first, is to develop a realistic mathematical model based on an already built laboratory 4-link system. The second is to apply a control law

based on the concept of virtual constraints, more precisely, using three virtual constraints. Finally, this method will be compared with other existing method based on two virtual constraints only.

1.2 Organization of the thesis

The thesis is organized as follows. In chapter 2 general information is presented about underactuated mechanical systems in order to give a background and to help to understand the present work. The first objective, the development of a realistic model of a 4-link will be explained in detail in Chapter 3 where the modeling of the Acrobot starts and it will be explained how this concept is extended to the model of the 4-link. Then it will be used to show the modeling of a real 4-link and all considerations that should be taken. Chapter 4 contains information about how constraints are achieved and different polynomial equations to constrain the 4-link will be compared. These equations will be tested to see the performance of the 4-link in a simulation. Finally, the best result regarding complexity of equations vs. performance will be used to compare it with other control methods. The final chapter presents the conclusions and discusses the possibility of continuing work with more applied and technical focus.

Chapter 2

Mechanical systems

Mechanical systems can be classified depending on the number of actuators that they have in relation to their degrees of freedom. Typical mechanical systems are fully actuated, this means that they have as many actuators as degrees of freedom. In this work the concept will be focused on underactuated mechanical systems. This term refers to the fact that these systems have less actuators than degrees of freedom. These kind of systems can be seen in walking robots, flexible robots, and when a manipulator is attached to a mobile platform, space platform or undersea vehicle among others [13].

One of the main studied cases of underactuated mechanical systems is the biped walking robot without feet, meaning that the ankle actuator is not present, making it underactuated. The requirement of feet for walking has been proved to be not necessary [7]. The simplest of these walking robots is probably the Acrobot. The Acrobot consists of two links with an actuator at the joint. It is an unrealistic humanoid walker since it has no knees or ankle joint and has point feet. Nevertheless, it is still a good model to start to analyze biped walking. The level of under actuation of walking robot Acrobot is 1 since it has 2 degrees of freedom (DOF) but only one actuator.

2.1 Simple underactuated mechanical systems

There are many different underactuated systems that have been explored, some examples of these are the inverted pendulum, the Cart-Pole system, the Pendubot, the Furuta pendulum and the Acrobot which will be explained in more detail later. Some of the main studied underactuated mechanical systems are listed here.

- The Inverted pendulum consists of a rod with an actuator at one of two ends. The idea is to stabilize this system in the upper position and keep it stable under the presence of perturbations.
- The CartPole system is similar to the Inverted pendulum but at one end is a mass and in the other end, instead of the actuator, there is just a pivot at the top of a cart. The upper equilibrium position is achieved by applying a force in the cart.

- The Pendubot is an extension of the Inverted pendulum but instead of only one rod there are two rods connected with a joint. Actuator is in the pivot point. As it can be deduced, the system is much more complex to keep it in the upper position.
- The Furuta pendulum could be seen as a Pendubot where the rods are in perpendicular planes bringing the second rod to the upper position by rotating the first rod around the vertical axis.
- The Acrobot is just like the pendubot with the difference that the actuator is between two rods. It is used not only to explore control systems for the stability of the inverted position but also as a walking system.

It is possible to see how the Acrobot can represent a simple biped walking system if we consider that the joint is in the hip and each rod is a leg with no knees. In figure 2.1 the elements required to model the Acrobot are shown, l is the length of the link, l_c is the distance to the center of mass, m is the mass of each link, q_1 is the angle of the first link to the vertical axis, q_2 is the angle between the link two and an axis that extends from link one.

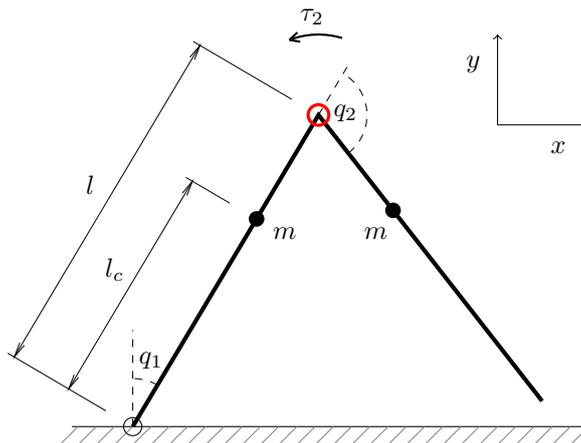


Figure 2.1: The Acrobot.

The advantage that the underactuated systems have for example in the case of the Acrobot is that the walking is more natural and similar to real human gait. That is why it is of interest not only for nonlinear control but this could represent a benefit in the design of walking robots because they could reduce energy consumption and increase efficiency [11].

2.2 Human gait and robotics

Human gait or human walk is the process of impelling the center of mass of the body forward, in a normal walk this is achieved by the sequence of movements of the legs. The important factor is that one of the legs is always touching the ground and for some period of time both legs are on the ground. Opposite is

the case of running in which at some moment both feet are off the ground or could be seen also as if at any period of time only one foot is in contact to the ground [12].

The first concepts developed for walking robots was *static walkers* meaning that the speed of the robot is so low that the Moment Of Inertia (MOI) of the legs is negligible [11]. A more advance approach is the so called dynamic walking which accounts for these MOI and the velocity of the robot is not necessary to be known. This concept gives a more human-like walking, nevertheless, it is not very efficient in terms of energy consumption [11].

The concepts described previously, namely, dynamic and static walking either account for the center of mass to be inside the support polygon (static walking) or have a Zero Moment Point in the support polygon (dynamic fully actuated walking) [11]. The phase when the robot has both feet on the ground is called double support. This phase can be instantaneous or non-instantaneous [12]. The cases described before are non-instantaneous. If we reduce the support polygon to be a point foot, it will not be theoretically stable, nevertheless, it can still have a stable gait. This gait will have instantaneous double support and will be dynamic since the transition of the center of mass from footprint to footprint will not be performed at infinitesimally short time [12].

Real successful applications of humanoid walking of robots are done by controlling every single articulation and computing every position of the robot. This is the case for fully actuated robots. Even one of the most advanced humanoid robots ASIMO from Honda uses the Zero Moment Point approach [11]. Although this approach has proved to be successful in achieving a stable walk it is not very efficient in terms of energy consumption because velocity and walking does not seem to be natural, as a matter of fact, it is stiff and slow.

In real human gait during each step one of legs is touching the ground. This leg is named stance leg while the other is swinging forward called swing leg because of the type of movement that it performs. There is a simplified biped walking with no feet and the whole movement is simplified to stance leg and swing leg and the actions of the feet in real human gait have been reduced to an instant contact with the ground [7]. This is the case analyzed in this work.

Other important simplification is the number of dimensions. Obviously human gait is performed in three dimensions but this is quite complicated to analyze. From this reason most of the walking robots experiments are simplified to two dimensional analysis by restricting mechanically with a support system not allowing the robot to fall to the side. In the 4-link case studied here this is done by putting the 4-link in the frame with wheels, letting the robotic legs to move forward only.

2.3 Euler Lagrange Method

The Euler-Lagrange equation also known as Lagrange equation is a differential equation whose solutions, because of the Hamiltonian principle of stationary action, describe the evolution of a physical system. It is equivalent to the Newtonian law of motion except that regarding the generalized coordinate system it takes the same form. The Lagrangian contains all physical information concerning the system and the forces acting on it. For many mechanical systems the scalar Lagrangian is found as the difference of the kinetic and potential energies.

This is expressed as follows

$$\mathcal{L}(q, \dot{q}) = K - U = \frac{1}{2} \dot{q}^T D(q) \dot{q} - V(q), \quad (2.1)$$

where K is the kinetic energy of the system and U is the potential energy of the system, q is a n -dimensional configuration vector and $D(q)$ is the inertia matrix. If there are external forces $u = (\tau_1, \tau_2, \dots, \tau_n)^T$ on the system the system is fully actuated and the Euler-Lagrange equation takes the following form

$$\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = \tau, \quad (2.2)$$

from which it is possible to derive the equations of motion. This is done as follows; once the Lagrangian L has been calculated it is necessary to obtain the derivative of the Lagrangian (L) with respect to \dot{q} and then derivate again with respect to time and subtract the derivative of the Lagrangian with respect to q , this will be equal to the torque τ . With this method it is possible to obtain the matrices that describe the system

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = Bu, \quad (2.3)$$

where $D(q)$ is the inertia matrix, $C(q, \dot{q})$ contains two types of terms depending on configuration variables $\dot{q}_i \dot{q}_j$ that are called Centrifugal terms ($i = j$) and Coriolis terms ($i \neq j$) and $G(q)$ contains the gravity terms. Matrix B maps the inputs into the desired coordinates.

Chapter 3

Modeling the 4-link

In this Chapter a description of how to model an ideal 4-link will be shown. This model will account for the potential and kinetic energies of each one of the four links to form the Lagrangian and then to obtain the equations of motion from it. It will be show as well how to model a real 4-link and what considerations are needed, it will be seen how simple assumptions in the ideal 4-link require many calculations for a real model.

3.1 General model of Acrobot and 4-link

The basic model of the 4-link consist of 4 rods connected linearly with articulated joints between them and having one of the ends of the first link set to the origin of the reference system and considered as an non actuated joint, called simply a pivot. Other joints are actuated. The 4 link is an extension of the Acrobot, nevertheless, the objective is not to find the equilibrium states of the 4 link balanced upside-down but rather to see the second joint as a hip and to use each pair of 2 links as legs with knees of a walking robot. In this case the knees would be the first and third joint. In figure 3.1 one can see the 4-link with four angles to describe position and torques expressed by τ .

The 4-link here studied is based on a real robotic mechanical system at CVUT. Figure 3.2 shows an image of the 4-link at CVUT.

The model uses the equations of motion obtained through the proper derivation and subtraction of the Lagrangian of the system. To calculate the Lagrangian it is necessary to obtain kinetic and potential energy of the system. For this sake, rods are considered to have the same mass and there is no mass at the joint itself, all the weight is considered to be at the center of links. This is pretty simplistic but helps to simplify the considerations when developing a model based on the real 4-link.

After obtaining the Lagrangian of the system it must be derived with respect to velocity and time and subtract from it the derivative with respect to position, with this the equation of motion will be obtained. For the case of the first joint we know it is zero since it is non-actuated.

Now that equations of motion have been obtained we can see that there are 4 variables. Virtual constraints will be created to restrict the movement of three of the angles to depend on the other one, in other words the control law in this

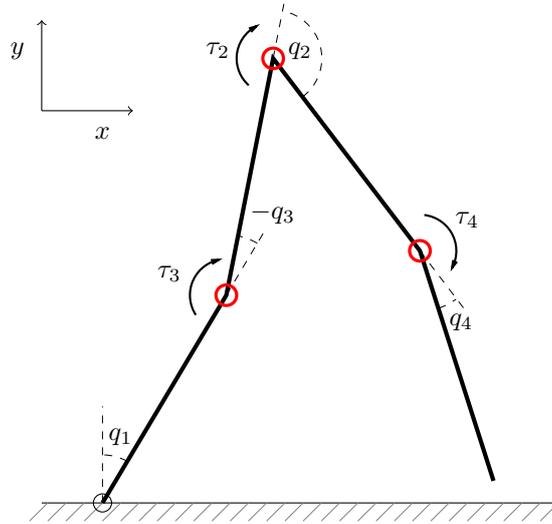


Figure 3.1: The 4-link.

case considers a predefined trajectory of the legs taking as reference the angle of the stance leg with the vertical axis. All of this will be explained in detail here and in following chapters.

To analyze the Acrobot and 4-link Lagrangian mechanics are used. The Acrobot and 4-link are both seen as having one leg in pivot with the ground and the positions of the center of mass of each leg is computed. This is done by considering the potential and kinetic energy of each leg. The weight of the actuator at the knee is considered as being part of the leg, having each leg half of the weight of the actuator and joints.

To derive equations of motion of a mechanical system it is necessary to start by calculating the Lagrangian (L) of the system, defined as kinetic energy (K) minus potential energy (U) expressed as

$$L = K - U. \quad (3.1)$$

Potential Energy is defined as

$$U = mgh, \quad (3.2)$$

where m is the mass of the system, g is the gravity and h is the height with respect to the origin of the system. Kinetic Energy is defined as

$$K = \frac{1}{2}mv^2, \quad (3.3)$$

where m is the mass and v is the velocity. Then it is necessary to use equation (2.2) to find equations of motion.

There are 2 links to consider in the model of an Acrobot, this means 2 potential energies (first and second links) and two kinetic energies (first and second links). Within kinetic energy there are 4 elements to consider, translational and rotational energies of the first and second links. Translational expressed as T_1



Figure 3.2: Real Acrobot at CVUT.

and T_2 respectively for each link and rotational T_3 and T_4 respectively. Total kinetic energy is then expressed as

$$T = T_1 + T_2 + T_3 + T_4, \quad (3.4)$$

and substituting by definition of each element we obtain the following equation

$$T = \frac{1}{2}m_1(v_{c1}^0)^2 + \frac{1}{2}m_2(v_{c2}^0)^2 + \frac{1}{2}J_1(w_{0,1}^0)^2 + \frac{1}{2}J_2(w_{0,2}^0)^2, \quad (3.5)$$

where the subindices 1 and 2 represent the links 1 and 2 respectively, m_1 and m_2 are the masses, J_1 and J_2 are moments of inertia. v_{c1}^0 and v_{c2}^0 are the velocities of the center of mass of the first and the second links with respect to the main coordinate reference system and $w_{0,1}^0$ and $w_{0,2}^0$ are angular velocities of link 1 and 2 with respect to the main coordinate reference system.

The analysis of the Acrobot is simple but when extending the concept for the 4-link even when the computations are similar, the number of elements grows due to the number of rotation matrices to establish every element with respect to one common reference system. In figure 3.3 it's possible to see the elements that are necessary to consider when calculating potential and kinetic energies.

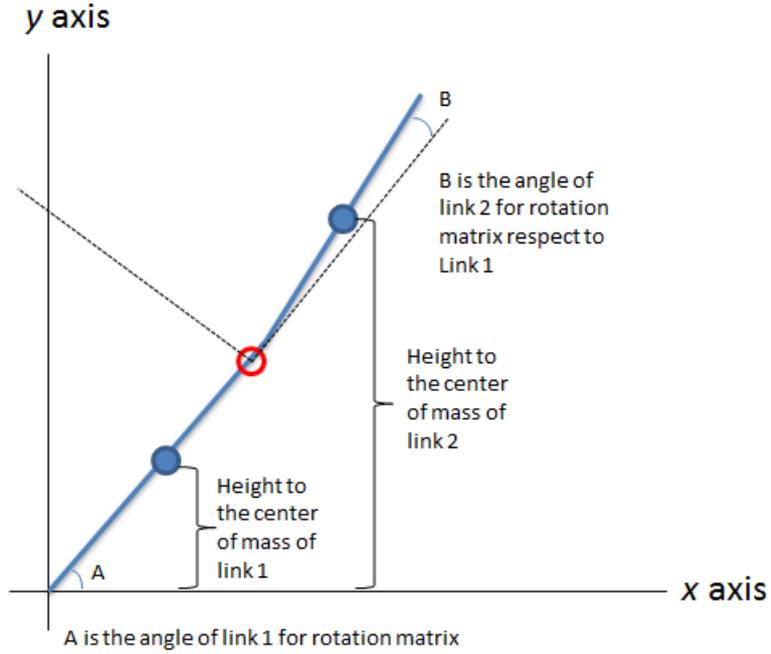


Figure 3.3: Representation of the Acrobot for energy calculations.

3.1.1 The first link

First of all the rotation matrix of the first link with respect to the main coordinate reference system is considered. This main coordinate reference system is established to have origin at the point of contact to the ground of the first link. X axis aligned horizontally and Y axis being positive in the upward direction. The rotation matrix of the first link with respect to the reference coordinate system is

$$R_1^0 = \begin{bmatrix} \cos q_1 & -\sin q_1 & 0 \\ \sin q_1 & \cos q_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (3.6)$$

while the rotation matrix of the second link with respect to the first link is

$$R_2^1 = \begin{bmatrix} \cos q_2 & -\sin q_2 & 0 \\ \sin q_2 & \cos q_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (3.7)$$

where q_1 and q_2 are the angles of link 1 and link 2, respectively, according to its own reference system. The angular velocity of the first link (\dot{q}_1), respect to the main coordiante reference system is defined as

$$w_{0,1}^0 = \begin{bmatrix} 0 \\ 0 \\ \dot{q}_1 \end{bmatrix}. \quad (3.8)$$

The angular velocity of the second link with respect to the main coordinate reference system is defined as

$$w_{0,2}^0 = w_{0,1}^0 + R_1^0 w_{1,2}^1 = \begin{bmatrix} 0 \\ 0 \\ \dot{q}_1 + \dot{q}_2 \end{bmatrix}, \quad (3.9)$$

where \dot{q}_2 is the angular velocity in scalar form and $w_{1,2}^1$ is the angular velocity of the second link with respect to the first link's reference system, it is rotated with respect to the main coordinate reference system by R_1^0 . These notations are according to [15]. The term $w_{1,2}^1$ denotes the angular velocity of frame 2 that corresponds to changing R_2^1 expressed relative to the coordinate system of link 1. The product $R_1^0 w_{1,2}^1$ expresses this angular velocity relative to the main coordinate reference system [15]. Now, the translational velocity of the center of mass of the first link is given by

$$v_{c1}^0 = w_{0,1}^0 \times r_{c1}^0, \quad (3.10)$$

where r_{c1}^0 is the position of the center of mass of the first link respect to the main coordinate reference system. The position of the center of mass of the first link with respect to its own reference system is given by

$$r_{c1}^1 = \begin{bmatrix} \frac{l_1}{2} \\ 0 \\ 0 \end{bmatrix}. \quad (3.11)$$

It is important to see that to simplify the analysis, the position of the center of mass is considered to be at the middle of the rod ($\frac{l_1}{2}$). To obtain this position with respect to the main coordinate reference system of the Acrobot it is necessary to transform it using the rotation matrix. Thus we obtain

$$r_{c1}^0 = R_1^0 r_{c1}^1 = \begin{bmatrix} \frac{l_1}{2} \cos q_1 \\ \frac{l_1}{2} \sin q_1 \\ 0 \end{bmatrix}. \quad (3.12)$$

Now the calculation of the velocity of the first center of mass as a vector is done as follows

$$v_{c1}^0 = \begin{bmatrix} 0 & -\dot{q}_1 & 0 \\ \dot{q}_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{l_1}{2} \cos q_1 \\ \frac{l_1}{2} \sin q_1 \\ 0 \end{bmatrix} = \begin{bmatrix} -\dot{q}_1 \frac{l_1}{2} \sin q_1 \\ \dot{q}_1 \frac{l_1}{2} \cos q_1 \\ 0 \end{bmatrix} \quad (3.13)$$

and calculation of the squared velocity is given by

$$(v_{c1}^0)^2 = \left(-\dot{q}_1 \frac{l_1}{2} \sin q_1 \right)^2 + \left(\dot{q}_1 \frac{l_1}{2} \cos q_1 \right)^2 = \left(\frac{l_1}{2} \right)^2 (\dot{q}_1)^2. \quad (3.14)$$

The velocity of the first link with respect to the main coordinate reference system has been calculated. This will be used to model the Acrobot.

3.1.2 The second link

To calculate the translational velocity of the second link it is necessary to transform its velocity with respect to the main coordinate reference system, this is

expressed as follows

$$v_{c2}^0 = v_2^0 + w_{0,2}^0 \times r_{c2}^0 = w_{0,1}^0 \times r_p^0 + w_{0,2}^0 \times r_{c2}^0, \quad (3.15)$$

where $w_{0,1}^0 \times r_p^0$ is used to calculate the translational velocity at the origin of the second link, this term expresses the angular velocity of the first link and the position of the end of the first link as a vector.

It is necessary to rotate the position of the end of the first link expressed as

$$r_p^1 = \begin{bmatrix} l_1 \\ 0 \\ 0 \end{bmatrix}, \quad (3.16)$$

respect to the main coordinate reference system, which leads to

$$r_p^0 = R_1^0 r_p^1 = \begin{bmatrix} l_1 \cos q_1 \\ l_1 \sin q_1 \\ 0 \end{bmatrix}, \quad (3.17)$$

now it is necessary to calculate the first part on the right hand side of the equation (3.15) for the velocity of the second link, which is expressed by

$$w_{0,1}^0 \times r_p^0 = \begin{bmatrix} 0 & -\dot{q}_1 & 0 \\ \dot{q}_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} l_1 \cos q_1 \\ l_1 \sin q_1 \\ 0 \end{bmatrix}, \quad (3.18)$$

which leads to

$$w_{0,1}^0 \times r_p^0 = \begin{bmatrix} -\dot{q}_1 l_1 \sin q_1 \\ \dot{q}_1 l_1 \cos q_1 \\ 0 \end{bmatrix}. \quad (3.19)$$

For the second part of the equation we need to know the position of the center of mass of the second link, again this will be considered to be ideally at the middle of the link and is expressed as

$$r_{c2}^2 = \begin{bmatrix} \frac{l_2}{2} \\ 0 \\ 0 \end{bmatrix} \quad (3.20)$$

and to rotate it to express it with respect to the main coordinate reference system, which is done as follows

$$r_{c2}^0 = R_1^0 R_2^1 r_{c2}^2 = \begin{bmatrix} \cos q_1 & -\sin q_1 & 0 \\ \sin q_1 & \cos q_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos q_2 & -\sin q_2 & 0 \\ \sin q_2 & \cos q_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{l_2}{2} \\ 0 \\ 0 \end{bmatrix}. \quad (3.21)$$

After some straight forward computations:

$$r_{c2}^0 = \begin{bmatrix} \frac{l_2}{2} (\cos q_1 \cos q_2 - \sin q_1 \sin q_2) \\ \frac{l_2}{2} (\cos q_2 \sin q_1 + \cos q_1 \sin q_2) \\ 0 \end{bmatrix}. \quad (3.22)$$

Now it is necessary to calculate the angular velocity of the second link times the distance to the center of mass of the second link from the own second link's reference coordinates, which is expressed as

$$w_{0,2}^0 \times r_{c2}^0 = \begin{bmatrix} 0 & -(\dot{q}_1 + \dot{q}_2) & 0 \\ (\dot{q}_1 + \dot{q}_2) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{l_2}{2}(\cos q_1 \cos q_2 - \sin q_1 \sin q_2) \\ \frac{l_2}{2}(\cos q_2 \sin q_1 + \cos q_1 \sin q_2) \\ 0 \end{bmatrix}. \quad (3.23)$$

Solving this results in

$$w_{0,2}^0 \times r_{c2}^0 = \begin{bmatrix} -(\dot{q}_1 + \dot{q}_2)\frac{l_2}{2}(\cos q_2 \sin q_1 + \cos q_1 \sin q_2) \\ (\dot{q}_1 + \dot{q}_2)\frac{l_2}{2}(\cos q_1 \cos q_2 - \sin q_1 \sin q_2) \\ 0 \end{bmatrix}. \quad (3.24)$$

Now it is possible to calculate the velocity of the second link as a vector

$$v_{c2}^0 = v_{p1}^0 + c_{c2,2}^0 = \begin{bmatrix} -\dot{q}_1 l_1 \sin q_1 - (\dot{q}_1 + \dot{q}_2)\frac{l_2}{2}(\cos q_2 \sin q_1 + \cos q_1 \sin q_2) \\ \dot{q}_1 l_1 \cos q_1 + (\dot{q}_1 + \dot{q}_2)\frac{l_2}{2}(\cos q_1 \cos q_2 - \sin q_1 \sin q_2) \\ 0 \end{bmatrix}, \quad (3.25)$$

note, that the expression can be reduced due to the identities

$$\cos q_2 \sin q_1 + \cos q_1 \sin q_2 = \sin(q_1 + q_2)$$

and

$$\cos q_1 \cos q_2 - \sin q_1 \sin q_2 = \cos(q_1 + q_2),$$

which reduce the term to

$$v_{c2}^0 = \begin{bmatrix} -\dot{q}_1 l_1 \sin q_1 - (\dot{q}_1 + \dot{q}_2)\frac{l_2}{2} \sin(q_1 + q_2) \\ \dot{q}_1 l_1 \cos q_1 + (\dot{q}_1 + \dot{q}_2)\frac{l_2}{2} \cos(q_1 + q_2) \\ 0 \end{bmatrix}. \quad (3.26)$$

Calculating the square of the velocity of the second link is expressed as follows

$$(v_{c2}^0)^2 = (-\dot{q}_1 l_1 \sin q_1 - (\dot{q}_1 + \dot{q}_2)\frac{l_2}{2} \sin(q_1 + q_2))^2 + (\dot{q}_1 l_1 \cos q_1 + (\dot{q}_1 + \dot{q}_2)\frac{l_2}{2} \cos(q_1 + q_2))^2 \quad (3.27)$$

and solved in the following form

$$\begin{aligned} (v_{c2}^0)^2 &= (-\dot{q}_1)^2 (l_1)^2 (\sin q_1)^2 \\ &+ 2(\dot{q}_1 l_1 \sin q_1)(\dot{q}_1 + \dot{q}_2)\frac{l_2}{2} \sin(q_1 + q_2) \\ &+ (q_1 + q_2)^2 + \left(\frac{l_2}{2}\right)^2 (\sin(q_1 + q_2))^2 \\ &+ (\dot{q}_1)^2 (l_1)^2 (\cos q_1)^2 \\ &+ 2(\dot{q}_1 l_1 \cos q_1)(\dot{q}_1 + \dot{q}_2)\frac{l_2}{2} \cos(q_1 + q_2) \\ &+ (\dot{q}_1 + \dot{q}_2)^2 \left(\frac{l_2}{2}\right)^2 (\cos(q_1 + q_2))^2. \end{aligned} \quad (3.28)$$

Eliminating common terms results in

$$\begin{aligned} (v_{c2}^0)^2 &= (q_1)^2 (l_1)^2 \\ &+ \dot{q}_1 l_1 l_2 (\dot{q}_1 + \dot{q}_2) (\sin q_1 \sin(q_1 + q_2) + \cos q_1 \cos(q_1 + q_2)) \\ &+ (\dot{q}_1 + \dot{q}_2)^2 \left(\frac{l_2}{2}\right)^2. \end{aligned} \quad (3.29)$$

It is seen that the expression $(\sin q_1 \sin(q_1 + q_2) + \cos q_1 \cos(q_1 + q_2))$ can be expanded to

$$\sin q_1 (\cos q_1 \sin q_2 + \cos q_2 \sin q_1) + \cos q_1 (\cos q_1 \cos q_2 - \sin q_1 \sin q_2)$$

and then simplified to $\cos q_2$ and thus the squared velocity of the second link with respect to the main coordinate reference system. Finally it results in the equation

$$(v_{c2}^0)^2 = (\dot{q}_1)^2 (l_1)^2 + \dot{q}_1 l_1 l_2 (\dot{q}_1 + \dot{q}_2) \cos q_2 + (\dot{q}_1 + \dot{q}_2)^2 \left(\frac{l_2}{2}\right)^2. \quad (3.30)$$

Now velocities of links have been expressed in terms of the main coordinate reference system. These velocities are needed to calculate kinetic energy of the Acrobot to model the system. These calculations will be done in detail in the following subsection.

3.1.3 Kinetic Energy of the Acrobot

Now, velocities of links with respect to the main coordinate reference system calculated previously are used to solve equation (3.5). It is known from equation (3.4) that there are four elements, first rotational energies of the first and second links and translational energies for the first and second links. Rotational energy of the first link is expressed as follows

$$T_3 = \frac{1}{2} J_1 (w_{0,1}^0)^2, \quad (3.31)$$

squaring the angular velocity results in

$$T_3 = \frac{1}{2} (w_{0,1}^0)^T J_1 w_{0,1}^0 = \frac{1}{2} I_{zz1} (\dot{q}_1)^2, \quad (3.32)$$

where the matrix of the moment of inertia is

$$J_1 = \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix}. \quad (3.33)$$

For the rotational energy of the second link the expression is

$$T_4 = \frac{1}{2} J_2 (w_{0,2}^0)^2, \quad (3.34)$$

and solving in the same way as T_3 results in the following equation

$$T_4 = \frac{1}{2} (w_{1,2}^1)^T (R_1^0)^T J_2 R_1^0 w_{1,2}^1 = \frac{1}{2} I_{zz2} (\dot{q}_1 + \dot{q}_2)^2, \quad (3.35)$$

where I_{zz2} is the moment of inertia of the second link. Finally, since the equations of velocity (3.14) and (3.30) were calculated before, the elements to substitute in T_1 and T_2 are known. Substituting velocity in

$$T_2 = \frac{1}{2} m_1 (v_{c2}^0)^2, \quad (3.36)$$

results in

$$T_2 = \frac{1}{2}m_2 \left[(\dot{q}_1)^2(l_1)^2 + \dot{q}_1 l_1 l_2 (\dot{q}_1 + \dot{q}_2) \cos q_2 + (\dot{q}_1 + \dot{q}_2)^2 \left(\frac{l_2}{2}\right)^2 \right] \quad (3.37)$$

and for T_1 the expression is

$$T_1 = \frac{1}{2}m_1(v_{c1}^0)^2 = \frac{1}{2}m_1(\dot{q}_1)^2 \left(\frac{l_2}{2}\right)^2. \quad (3.38)$$

The total kinetic energy for the first link including rotational and translational energies is expressed as

$$K_1 = T_1 + T_3 = \frac{1}{2} \left(I_{zz_1} + m_1 \left(\frac{l_1}{2}\right)^2 \right) (\dot{q}_1)^2. \quad (3.39)$$

The total kinetic energy for the second link including rotational and translational energies is expressed as

$$K_2 = T_2 + T_4, \quad (3.40)$$

which with the proper substitution results in

$$K_2 = \frac{1}{2}I_{zz_2}(\dot{q}_1 + \dot{q}_2)^2 + \frac{1}{2}m_2 \left[(\dot{q}_1)^2(l_1)^2 + \dot{q}_1 l_1 l_2 (\dot{q}_1 + \dot{q}_2) \cos q_2 + (\dot{q}_1 + \dot{q}_2)^2 \left(\frac{l_2}{2}\right)^2 \right], \quad (3.41)$$

and solving K_2 results in

$$K_2 = \frac{1}{2}I_{zz_2}(\dot{q}_1)^2 + I_{zz_2}\dot{q}_1\dot{q}_2 + \frac{1}{2}I_{zz_2}(\dot{q}_2)^2 + \frac{1}{2}m_2(\dot{q}_1)^2(l_1)^2 + \frac{1}{2}m_2 l_1 l_2 \cos q_2 (\dot{q}_1)^2 + \frac{1}{2}m_2 l_1 l_2 \cos q_2 \dot{q}_1 \dot{q}_2 + \frac{1}{2}m_2(\dot{q}_1)^2 \left(\frac{l_2}{2}\right)^2 + \frac{1}{2}m_2(\dot{q}_2)^2 \left(\frac{l_2}{2}\right)^2 + m_2 \dot{q}_1 \dot{q}_2 \left(\frac{l_2}{2}\right)^2. \quad (3.42)$$

Now, all the terms for the kinetic energy in the Acrobot are known. This represents the first part of calculations, since the potential energy is yet to be calculated as well as the Lagrangian. Nevertheless, the potential energy is more simple and the Lagrangian can be done using computational tools.

3.1.4 Potential Energy

Potential energy of the entire Acrobot is easier to calculate since it has fewer terms, it is expressed as

$$P = mgh, \quad (3.43)$$

where m denotes the mass, g the gravity and h is the height. The equation to model the Acrobot is formed of three parts, first potential energy of the first link at its center of mass, then potential energy of the origin of the second link and potential energy of the second link at its center of mass with respect to the origin of the second link, this is as follows

$$P = m_1 g l_{c1} \sin q_1 + m_2 g l_1 \sin q_1 + m_2 l_{c2} \sin(q_1 + q_2), \quad (3.44)$$

which is simplified as

$$P = g(m_1 l_{c1} \sin q_1 + m_2(l_1 \sin q_1 + m_2 l_{c2} \sin(q_1 + q_2))), \quad (3.45)$$

where terms m_1 and m_2 stand for the masses, q_1 and q_2 for the angles, l_1 and l_2 for the length of links, l_{c1} and l_{c2} for the distance to the center of mass of links, all of them of its corresponding link indicated by the subindices 1 and 2, respectively, and g is the gravity constant. Now, all terms to calculate potential energy for the model of the Acrobot have been obtained together with the kinetic energy. They will be used to obtain the Lagrangian and then equations of motion.

3.1.5 Lagrangian

Now, kinetic and potential energies have being expressed in equations (3.39), (3.42) and (3.45), these terms are used to be substituted into the Lagrangian expressed by

$$L = K - P, \quad (3.46)$$

where K and P are total kinetic and potential energies of the Acrobot, respectively, thus from the substitution the following expression is obtained

$$\begin{aligned} L = & \frac{1}{2} \left(I_{zz1} + m_1 \left(\frac{l_1}{2} \right)^2 \right) (\dot{q}_1)^2 + \frac{1}{2} I_{zz2} (\dot{q}_1)^2 + I_{zz2} \dot{q}_1 \dot{q}_2 + \frac{1}{2} I_{zz2} (\dot{q}_2)^2 \\ & + \frac{1}{2} m_2 (\dot{q}_1)^2 (l_1)^2 + \frac{1}{2} m_2 l_1 l_2 \cos q_2 (\dot{q}_1)^2 + \frac{1}{2} m_2 l_1 l_2 \cos q_2 \dot{q}_1 \dot{q}_2 \\ & + \frac{1}{2} m_2 (\dot{q}_1)^2 \left(\frac{l_2}{2} \right)^2 + \frac{1}{2} m_2 (\dot{q}_2)^2 \left(\frac{l_2}{2} \right)^2 + m_2 \dot{q}_1 \dot{q}_2 \left(\frac{l_2}{2} \right)^2 \\ & - m_1 g l_{c1} \sin q_1 - m_2 g l_1 \sin q_1 - m_2 l_{c2} \sin(q_1 + q_2). \end{aligned} \quad (3.47)$$

Now, that the Lagrangian has been expressed, it is necessary to use

$$\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = \tau \quad (3.48)$$

to calculate the equations of the motion, τ is torque applied at the joint. It is possible to see that it is necessary to calculate the derivative of the Lagrangian with respect to each angle and the time derivative with respect to each velocity, this leads to four important equations, which are

$$\begin{aligned} \frac{\partial L}{\partial \dot{q}_1} = & \left(m_1 (l_{c1})^2 + m_2 (l_1)^2 + I_{zz1} + m_2 (l_{c2})^2 + I_{zz2} + 2 \left(m_2 (l_{c2})^2 + I_{zz2} \right) \cos q_2 \right) \dot{q}_1 \\ & + \left(\left(m_2 (l_{c2})^2 + I_{zz2} + m_2 l_1 l_{c2} \right) \cos q_2 \right) \dot{q}_2, \end{aligned} \quad (3.49)$$

$$\frac{\partial L}{\partial q_1} = - (m_1 l_{c1} + m_2 l_1) g \cos q_1 - m_2 l_{c2} \cos(q_1 + q_2), \quad (3.50)$$

$$\frac{\partial L}{\partial \dot{q}_2} = \left(m_2 (l_{c2})^2 + I_{zz2} \right) \dot{q}_2 + \left(m_2 (l_{c2})^2 + I_{zz2} + m_2 l_1 l_{c2} \cos q_2 \right) \dot{q}_1, \quad (3.51)$$

$$\frac{\partial L}{\partial q_2} = - m_2 l_{c2} \sin q_2 (\dot{q}_1)^2 - \sin q_2 \dot{q}_1 \dot{q}_2 - m_2 l_{c2} g \cos(q_1 + q_2). \quad (3.52)$$

Now, terms are substituted in equation (3.48) respectively for each τ , terms will be rearranged into the Lagrangian formulation of the dynamics to obtain the following equation

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau, \quad (3.53)$$

where $D(q)$ is the inertia matrix and is straight forward to calculate it. The term $C(q, \dot{q})$ represents the Coriolis forces and is dependent of q and \dot{q} and at the same time is multiplied by q . This may look like a difficulty when obtaining the matrix C . Nevertheless this term always appears multiplied by 2, for this each term will be split and put one on each column according to their dependence.

For example, if a term appears as $2q_1q_2$, it will be split into two terms q_1q_2 , one will be put on the first column and one in second column. We must remember that τ_1 is equal to zero. After calculating the equations of motion for the 4-link three matrices are obtained that will be the base to our computational model. The matrix $G(q)$ represents the gravity forces and it is a column vector that is easy to obtain.

Now, all elements concerning the dynamics of the Acrobot have been calculated. First kinetic and potential energies to obtain the Lagrangian and then it is used to obtain the Lagrangian formulation of the dynamics of the Acrobot, expressed in equation (3.53). The concept of how to obtain the model of the 4-link is shown in the subsection (3.1.6).

3.1.6 Extending the Acrobot to the 4-link

In the previous section, a detailed explanation was done on how to model the Acrobot, now it is intended to obtain the model for the 4-link. The procedure is exactly the same. It is necessary to obtain potential and kinetic energy of the third and the fourth link as well. For the case of the ideal or simple 4-link with masses at the center, then the MOI are the same and the center of mass remains at the middle just like in other links. Since the mass is the same for each link and it is located at the middle of the link, then for the potential energy it is only necessary to apply the same rotation matrices that were used to transform from the links to the main coordinate reference system. Kinetic Energy of links 3 and 4 is computed as follows

$$T_3 = \left(\frac{1}{2}\right) m_3 (v_{c,3}^0)^2 + \left(\frac{1}{2}\right) J_3 (w_{0,3}^0)^2 \quad (3.54)$$

and

$$T_4 = \left(\frac{1}{2}\right) m_4 (v_{c,4}^0)^2 + \left(\frac{1}{2}\right) J_4 (w_{0,4}^0)^2, \quad (3.55)$$

where m_3 and m_4 are the masses of link 3 and 4 respectively. $v_{c,3}^0$ and $v_{c,4}^0$ are linear velocities of links 3 and 4 with respect to the zero frame. J_3 and J_4 are moments of inertia and $w_{0,3}^0$ and $w_{0,4}^0$ are angular with velocities respect to the main coordinate reference system. Angular velocities of links 3 and 4 are calculated as follows

$$w_{0,3}^0 = w_{0,1}^0 + R_0^1 w_{1,2}^1 + R_0^1 R_1^2 w_{2,3}^2 \quad (3.56)$$

and

$$w_{0,4}^0 = w_{0,1}^0 + R_0^1 w_{1,2}^1 + R_0^1 R_1^2 w_{2,3}^2 + R_0^1 R_1^2 R_2^3 w_{3,4}^3, \quad (3.57)$$

where the term $w_{2,3}^2$ represents the velocity of link 3 with respect to the reference system of link 2, and $w_{3,4}^3$ is the velocity of link 4 with respect to the reference system of link 3. The new rotation matrices R_1^2 and R_2^3 represent the rotation from the coordinate system of link 2 to 1 and from link 3 to 2 respectively.

The linear velocity for the third link is expressed as

$$v_{c3}^0 = v_3^0 + w_{0,3}^0 \times r_{c3}^0 = w_{0,1}^0 \times r_p^0 + w_{0,2}^0 \times r_{p2}^0 + w_{0,3}^0 \times r_{c3}^0, \quad (3.58)$$

where v_3^0 expresses the velocity at the origin of the third link, calculated by $w_{0,1}^0 \times r_p^0 + w_{0,2}^0 \times r_{p2}^0$ which accounts for velocities of previous links that should be added. And $w_{0,3}^0$ is the velocity of the third link respect to the main coordinate reference system times the distance to the center of mass of the third link from the origin of third link's reference system.

In the same way the angular velocity of the fourth link is defined by

$$v_{c4}^0 = v_4^0 + w_{0,4}^0 \times r_{c4}^0 = w_{0,1}^0 \times r_p^0 + w_{0,2}^0 \times r_{p2}^0 + w_{0,3}^0 \times r_{p3}^0 + w_{0,4}^0 \times r_{c4}^0, \quad (3.59)$$

where $w_{0,4}^0$ is the angular velocity of the fourth link expressed to the main reference system, r_{c4}^0 is the distance to the center of mass of link 4 according to the coordinate system of link 4 and r_{p3}^0 is the length of the third link. J_3 and J_4 are the moments of inertia of the links 3 and 4 respectively. Now the procedure to calculate potential and kinetic energies is the same as with the Acrobot. These energies are used to calculate the Lagrangian and equations of motion, but now four equations of motion (τ) will be obtained instead of two.

3.2 Real 4-link modeling

In previous subsections, an explanation of how to model the Acrobot was done, the 4-link modeling was mentioned just as an extension of the Acrobot. Nevertheless, this model considers just links as elements and masses at the middle of the rod. In reality it is necessary to consider many other elements that make the real 4-link, all elements must be considered in order to have a better calculation of the center of mass in each link.

3.2.1 Elements of the real Actrobot

The real model consists not only of rods for the links but also of actuators at joints, and mechanical parts to create the joints themselves, also it is necessary to put point-feet. A description of all the elements is presented here.

- The actuator in the hip is located at the second joint. There are two of them in order to move each of the rods that represent the upper legs. The weight of each one is of 0,151kg.
- The desk with electronic is an electronic board with sensors and other electronic elements to measure angles. There are 4 of them, one on every link and the weight of each one is 0,054kg. The dimensions of the desk with electronics are 0,176m x 0,039m.
- The rod is a metallic bar which represents the links. Each one weights 0,056kg and its length is 0,25m.
- The actuator at the knee is applies torque at the knees. There are 2 of them, one for each knee and each one weights 0,129kg.

- The joints are the mechanical elements at the knee and the hip that help hold together rods and the actuators in knees and hip. There are four of them and each one weighs 0,15kg.
- The bar in the hip is used to have a physical real axis for two legs since in the real model legs are parallel to each other but shifted in an axis that is not considered in the mathematical model. Its total weight is 0,082kg.
- The triangular feet are necessary to give the legs the property of point-feet that is considered in the mathematical model. Each foot is 0,012 m long and weighs 0,015 kg.

It is worth mentioning that the actuator at the hip and knee used to be the same. But the ones at the hip were replaced, this is mentioned because for purposes of technical work they are also referred as old actuators (for the knees) and new actuators (for the hip). All these values can be seen more clearly in the table 3.1.

Table 3.1: Parts of the real 4-link model.

	Part of the 4-link	Weight in kg.	Dimensions
1	Hip actuator	0,151	NA
2	Electronic desk	0,054	0,176m x 0,039m
3	Rod or bar	0,056	0,25 m.
4	Knee actuator	0,129	NA
5	Joints	0,15	NA
6	Hip bar	0,082	NA
7	Foot	0,015	0,012 m.

3.2.2 Center of mass and moment of inertia

Calculations of matrices are now done based on the real model. This means that it should be taken into account the distribution of the weight to calculate the real center of mass of each link.

Due to the fact a real model is trying to be developed it is shown how the MOI of the first link was done taking into account many considerations. Calculation of the center of mass of link 1 has origin at the feet, masses of long elements are considered at their centers. It can be decomposed as follows

$$TotalMassLink1 = FeetMass + BarMass + DeskMass + \frac{JointMass}{2} + \frac{OldActMass}{2},$$

now the position of the center of mass is required, this is calculated as follows

$$CenterMassLink1 = \frac{1}{TotalMassLink1} \left(FeetPtnL1 \cdot FeetMass + (BarMass + DeskMass) \cdot BarPtnL1 + \left(\frac{JointMass}{2} + \frac{OldActMass}{2} \right) \cdot KneePtnL1 \right),$$

where the expressions $FeetPtnL1$, $BarPtnL1$ and $KneePtnL1$ indicate the position of feet, bar or knee of link 1 respectively and the term Act indicates

actuator. At the same time it is necessary to calculate the distance from the center of mass to the end of the rod to account for the masses at that extreme, this is calculated as follows

$$OppositeDirectionCenterMassLink1 = LengthL1 - CenterMassLink1.$$

The same calculations are required for the second link, the total mass of the second links is calculated as follows

$$TotalMassLink2 = BarMass + DeskMass + \frac{JointMass}{2} + \frac{OldActMass}{2} + HipActMass + \frac{HipBarMass}{2} + JointMass,$$

and the position of the center of mass of the second link is expressed with the following equation

$$CenterMassLink2 = \frac{1}{TotalMassLink2} \left((BarMass + DeskMass) \cdot RodPtn2 + \left(\frac{JointMass}{2} + \frac{OldActMass}{2} \right) \cdot KneePtnL2 + HipActMass \cdot HipActPtnL2 + \left(JointMass + \frac{HipBarMass}{2} \right) \cdot HipActPtnL2 \right).$$

At the same time the distance required to calculate the MOI of the second link to account for masses at the hip is expressed in the following equation

$$OppositeDirectionCenterMassLink2 = LengthL2 - CenterMassLink2.$$

The resulting centers of mass are:

$$CenterofMassofLink1 = 20.1497cm$$

and

$$CenterofMassofLink2 = 16.7794cm.$$

Now, the calculation of the Moment of Inertia (MOI) is done as follows, the long elements (rod and desk) are assumed as moment of inertia of a bar and expressed respectively as follows

$$MOIlinkbar1 = \frac{BarMass \cdot BarLength^2}{12}$$

and

$$MOIdesk = \frac{DeskMass \cdot DeskLength^2}{12}.$$

Now, this considers that the rotation is around the center of the bar. It must be corrected and recalculated to the real axis. For this the parallel axis theorem is used which establishes that the new MOI (I_{new}) equals the original MOI (I_{com}) plus the mass of the bar times the squared distance to the real axis. This is expressed as follows

$$I_{new} = I_{com} + mr^2.$$

Equations to calculate the distance to the new axis for bar 1, the new moment of inertia of the bar 1 and the new moment of inertia of the desk 1, are expressed respectively by the following equations

$$NewAxisDist1 = CenterMassLink1 - RodPtn1,$$

$$NewMOIlinkbar1 = MOIlinkbar1 + BarMass \cdot NewAxisDist1^2$$

and

$$NewMOIdesk = MOIdesk + DeskMass \cdot NewAxisDist1^2.$$

Similarly for the second link the distance to the new axis, the new moment of inertia of the bar 2 and the new moment of inertia of the desk 2 are expressed respectively by the following equations

$$NewAxisDist2 = CenterMassLink2 - BarPtnL2,$$

$$NewMOIlinkbar2 = MOIbar2 + BarMass \cdot NewAxisDist2^2,$$

and

$$NewMOIdesk2 = MOIdesk + DeskMass \cdot NewAxisDist2^2.$$

For the calculation of the MOI of remaining elements masses are considered as point masses at a distance of the center of mass of the system for their respective link. For the feet element its center of mass was considered at a distance of 1.2 cm from the origin due to the equation that gives the center of mass of a triangle. Finally, the addition of all moments of inertia due to the superposition principle which establishes the moment of inertia of a system around a particular axis is equivalent to the sum of the moment of inertia of every element around that axis.

Similarly to this procedure, all the other MOI of the second link were calculated considering the origin of the system at the beginning of the second link or viewed from other point, the end of the first link. Since the second link is equivalent to the third one in measures and weights and the first to the fourth one, there is no need to calculate third and fourth. It is deduced that the MOI are equivalent, thus we have

$$MOIofLink1 = MOIofLink4 = 19.6306$$

and

$$MOIofLink2 = MOIofLink3 = 68.7128.$$

This equivalence of link 1 to link 4 and link 2 to link 3 also applies for the case of the Center of Mass but mirrored. The Center on mass for links 3 and 4 would be exactly the opposite distance of links 2 and 1 respectively. This is calculated by subtracting from the total length of each link accordingly, thus results in

$$\begin{aligned} OppositeDirectiontotheCenterofMassofLink1 &= \\ 6.6503cm &= CenterofMassofLink4 \end{aligned}$$

and

$$\begin{aligned} OppositeDirectiontotheCenterofMassofLink2 &= \\ 8.2206cm &= CenterofMassofLink3. \end{aligned}$$

Now all elements have been calculated to obtain equations of motion of a real Acrobot. Real matrices $D(q)$ which represents the Inertia matrix, $C(q, \dot{q})$ for the Coriolis forces and $G(q)$ for gravity terms can be calculated.

Chapter 4

Advanced Walking Control

The control of a biped robot is particularly hard to achieve due to the instability of the robots, some approaches simplify the dynamics to steady state force balance [7]. This approach is unrealistic for the real dynamics of human walking capabilities.

Due to this concern of the lack of new control concepts the RABBIT was created by several French laboratories as a testbed to make experiments with the control of biped robots. One of the main achievements of this project is the development of the concept of virtual constraints which reduces the number of degrees of freedom of the system [7]. Virtual constraints set dependence between links of the system through the feedback.

This is the concept chosen to control the 4-link in this work. The concept is open to choose any dependence between angles of the system. In this work all the angles will depend on the evolution of angle q_1 which is the angle between the stance leg and the vertical axis, hence achieving a controlled step that can be performed periodically.

It is important to say that this concept of virtual constraints can be applied not only to walking robots but many mechanical systems with a level of underactuation equal to one [7]. Thus it is worth studying the concept as a way to control underactuated system not to achieve biped walking motion only.

Of course, in the field of control of biped robots exist another ways of control of biped robots. One of them is based on the feedback control of biped robots, see series of work [8, 4, 3]. Using this concept it is possible to obtain similar results like using method of virtual constraints. In contrast to defining constraining functions, it is necessary to define here a reference system to create reference trajectory to be tracked. Using method of feedback control it is possible to minimize an additional error which can be caused by e.g. an impact. However, the feedback control method is more complicated to implement in a low-level hardware.

4.1 Control of 4-link walker based on the method of virtual constraints

One of the main intentions of this work is to show the application of the concept of virtual constraints. Here a complete explanation of this concept will be

done. The concept of virtual constrains is simple to understand, just as a real mechanical system can be physically constrained by adding a physical restriction to limit the system and reducing its degree of freedom. This is achieved in the 4-link through the control feedback by creating a dependence of some angles with respect to one of the angles of the system, so these angles will not be free to move since its position now depends on the position of a reference angle. In this section this concept will be shown in the case of imposing two and three virtual constraints in the 4-link. Also the different constraining functions will be analyzed for the case of the three constraints in the 4-link. The method of virtual constraints strongly depends on equation of motion (3.53). Let me to remind the form of this equation

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = u. \quad (4.1)$$

The inner structure of $D(q)$, $C(q, \dot{q})$ and $G(q)$ is very important for the computation of virtual constraints method later on:

$$D(q) = \begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & d_{34} \\ d_{41} & d_{42} & d_{43} & d_{44} \end{bmatrix}, C(q, \dot{q}) = \begin{bmatrix} C^1 \\ C^2 \\ C^3 \\ C^4 \end{bmatrix}, G(q) = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{bmatrix}, \quad (4.2)$$

where d_{xx} are scalar entries of the matrix D , C^x are row vectors and G_x are scalar entries of the column vector G .

4.1.1 Control of 4-link model based on three virtual constraints

Using the method of virtual constraints with three constraining functions, all of them depending on q_1 , the original 4-link model with 4 DOF is restricted to system with 1 DOF only. This system may be naturally regarded as a certain kind of an unactuated 1 DOF generalized inverted pendulum.

The method of virtual constraints is based on definition of some functions of angles q_2 , q_3 and q_4 with dependence on angle q_1 . In other words, angle q_2 is given by function $q_2 = \Phi_2(q_1)$, angle q_3 is given by function $q_3 = \Phi_3(q_1)$ and angle q_4 is given by function $q_4 = \Phi_4(q_1)$. The exact mathematical definition of functions $\Phi_2(q_1)$, $\Phi_3(q_1)$, $\Phi_4(q_1)$ will be discussed later on. For shortness, we skip the dependence on angle q_1 in functions $\Phi_{2,3,4}(q_1)$. The first and second derivatives of angles q are expressed as follows

$$\begin{aligned} \dot{q}_2 &= \Phi_2' \dot{q}_1, & \ddot{q}_2 &= \Phi_2'' \dot{q}_1^2 + \Phi_2' \ddot{q}_1, \\ \dot{q}_3 &= \Phi_3' \dot{q}_1, & \ddot{q}_3 &= \Phi_3'' \dot{q}_1^2 + \Phi_3' \ddot{q}_1, \\ \dot{q}_4 &= \Phi_4' \dot{q}_1, & \ddot{q}_4 &= \Phi_4'' \dot{q}_1^2 + \Phi_4' \ddot{q}_1. \end{aligned}$$

The error \bar{q} between desired angular positions given by functions Φ and the real angular positions q and the error $\dot{\bar{q}}$ between desired angular velocities and the real angular velocities are defined as follows

$$\begin{aligned} \bar{q}_2 &= q_2 - \Phi_2, & \dot{\bar{q}}_2 &= \dot{q}_2 - \Phi_2' \dot{q}_1, \\ \bar{q}_3 &= q_3 - \Phi_3, & \dot{\bar{q}}_3 &= \dot{q}_3 - \Phi_3' \dot{q}_1, \\ \bar{q}_4 &= q_4 - \Phi_4, & \dot{\bar{q}}_4 &= \dot{q}_4 - \Phi_4' \dot{q}_1. \end{aligned}$$

Using the exact state feedback linearization and subsequent linear error dynamics stabilizer the following controller is designed

$$\begin{aligned}\ddot{\bar{q}}_2 &= \bar{\tau}_2 = -k_1^2 \bar{q}_2 - k_2^2 \dot{\bar{q}}_2, \\ \ddot{\bar{q}}_3 &= \bar{\tau}_3 = -k_1^3 \bar{q}_3 - k_2^3 \dot{\bar{q}}_3, \\ \ddot{\bar{q}}_4 &= \bar{\tau}_4 = -k_1^4 \bar{q}_4 - k_2^4 \dot{\bar{q}}_4.\end{aligned}\quad (4.3)$$

This feedback controller guarantees that errors tend exponentially to zero, $\bar{q}_{2,3,4} \rightarrow 0$, $\dot{\bar{q}}_{2,3,4} \rightarrow 0$. In other words, desired angular positions and velocities correspond to the real angular positions and velocities. Further, it means that desired angular accelerations correspond to the real angular accelerations $\ddot{\bar{q}}_{2,3,4} = \ddot{q}_{2,3,4}$. Using this assumption it is possible to express following equations

$$\begin{aligned}\bar{\tau}_2 &= \ddot{q}_2 - \Phi_2'' \dot{q}_1^2 - \Phi_2' \ddot{q}_1, \\ \bar{\tau}_3 &= \ddot{q}_3 - \Phi_3'' \dot{q}_1^2 - \Phi_3' \ddot{q}_1, \\ \bar{\tau}_4 &= \ddot{q}_4 - \Phi_4'' \dot{q}_1^2 - \Phi_4' \ddot{q}_1.\end{aligned}\quad (4.4)$$

Substituting feedback controller (4.3) in (4.4) the expression for vector of angular acceleration $[\ddot{q}_2 \ \ddot{q}_3 \ \ddot{q}_4]^T$ is obtained in the following form

$$\begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \\ \ddot{q}_4 \end{bmatrix} = \begin{bmatrix} -k_1^2 \bar{q}_2 - k_2^2 \dot{\bar{q}}_2 \\ -k_1^3 \bar{q}_3 - k_2^3 \dot{\bar{q}}_3 \\ -k_1^4 \bar{q}_4 - k_2^4 \dot{\bar{q}}_4 \end{bmatrix} + \begin{bmatrix} \Phi_2'' \\ \Phi_3'' \\ \Phi_4'' \end{bmatrix} \dot{q}_1^2 + \begin{bmatrix} \Phi_2' \\ \Phi_3' \\ \Phi_4' \end{bmatrix} \ddot{q}_1.\quad (4.5)$$

The vector motion equation (4.1) actually consists of four scalar equations, the first of them is expressed as

$$d_{11} \ddot{q}_1 + [d_{12} \ d_{13} \ d_{14}] \begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \\ \ddot{q}_4 \end{bmatrix} + C^1(q, \dot{q}) \dot{q} + G_1(q) = 0, \quad (4.6)$$

where $d_{11,12,13,14}$ are given by (4.2).

The remaining scalar equations in (4.1) are expressed as follows

$$\begin{bmatrix} \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} = \begin{bmatrix} d_{21} \\ d_{31} \\ d_{41} \end{bmatrix} \ddot{q}_1 + D^{22} \begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \\ \ddot{q}_4 \end{bmatrix} + \begin{bmatrix} C^2(q, \dot{q}) \\ C^3(q, \dot{q}) \\ C^4(q, \dot{q}) \end{bmatrix} \dot{q} + \begin{bmatrix} G_2(q) \\ G_3(q) \\ G_4(q) \end{bmatrix}, \quad (4.7)$$

where D^{22} is submatrix of matrix D of the following form

$$D^{22} = \begin{bmatrix} d_{22} & d_{23} & d_{24} \\ d_{32} & d_{33} & d_{34} \\ d_{42} & d_{43} & d_{44} \end{bmatrix},$$

$C^{2,3,4}$ are appropriate row vectors of matrix C and $G_{2,3,4}$ are appropriate scalar entries of matrix G according to (4.2).

When substituting for the vector of angular acceleration $[\ddot{q}_2 \ \ddot{q}_3 \ \ddot{q}_4]^T$ in (4.7) from (4.5), the target equation for applied actuators torques is as follows

$$\begin{aligned}\begin{bmatrix} \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} &= \begin{bmatrix} d_{21} \\ d_{31} \\ d_{41} \end{bmatrix} \ddot{q}_1 + D^{22} \begin{bmatrix} -k_1^2 \bar{q}_2 - k_2^2 \dot{\bar{q}}_2 \\ -k_1^3 \bar{q}_3 - k_2^3 \dot{\bar{q}}_3 \\ -k_1^4 \bar{q}_4 - k_2^4 \dot{\bar{q}}_4 \end{bmatrix} + D^{22} \begin{bmatrix} \Phi_2'' \\ \Phi_3'' \\ \Phi_4'' \end{bmatrix} \dot{q}_1^2 + \\ &D^{22} \begin{bmatrix} \Phi_2' \\ \Phi_3' \\ \Phi_4' \end{bmatrix} \ddot{q}_1 + \begin{bmatrix} C^2(q, \dot{q}) \\ C^3(q, \dot{q}) \\ C^4(q, \dot{q}) \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} + \begin{bmatrix} G_2(q) \\ G_3(q) \\ G_4(q) \end{bmatrix},\end{aligned}\quad (4.8)$$

where angular acceleration \ddot{q}_1 is substituted by expression for \ddot{q}_1 (4.6) after substitution of angular acceleration vector $[\ddot{q}_2 \ \ddot{q}_3 \ \ddot{q}_4]^T$ by expression (4.5). For the sake of completeness, the expression for \ddot{q}_1 is

$$\ddot{q}_1 = -\frac{1}{d_{11} + \begin{bmatrix} d_{12} \\ d_{13} \\ d_{14} \end{bmatrix}^T \begin{bmatrix} \Phi_2' \\ \Phi_3' \\ \Phi_4' \end{bmatrix}} \left[\begin{bmatrix} d_{12} \\ d_{13} \\ d_{14} \end{bmatrix}^T \begin{bmatrix} \Phi_2'' \\ \Phi_3'' \\ \Phi_4'' \end{bmatrix} \dot{q}_1^2 + \begin{bmatrix} d_{12} \\ d_{13} \\ d_{14} \end{bmatrix}^T \begin{bmatrix} -k_1^2 \ddot{q}_2 - k_2^2 \ddot{q}_2 \\ -k_1^3 \ddot{q}_3 - k_2^3 \ddot{q}_3 \\ -k_1^4 \ddot{q}_4 - k_2^4 \ddot{q}_4 \end{bmatrix} + C^1 \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} + G_1(q) \right]. \quad (4.9)$$

Using these quite detailed computations, the final expressions (4.8), (4.9) were obtained to control the 4-link model using virtual constraints. The resulting control crucially depends on the functions Φ_2 , Φ_3 , Φ_4 selection and on the initial angular velocity \dot{q}_1 . To control of 4-link model in a way that resembling walking, the functions Φ_3 and Φ_4 should represent bending and stretching knees while function Φ_2 should be monotonous. The initial angular velocity \dot{q}_1 should be high enough to ensure monotonous behaviour of the angle q_1 with respect to time. On the other hand, the maximal initial velocity \dot{q}_1 is limited by stability conditions during more steps, see [16, 6, 10].

Three virtual constraints applied to simplified 4-link

At this point degree of freedom of the 4-link has been reduced since \ddot{q}_1 is expressed in terms of the other variables. In other words a control for \ddot{q}_1 has been found.

Now this control is applied to the 4-link. To prove in a simplified way that it is working the 4-link will be treated as an Acrobot. From this reason, angles q_3 and q_4 will be set to zero, hence their derivatives, in this sense the 4-link will behave as having no knees.

The trajectory of the angle q_2 has been set to a simple straight line, the initial position has been defined in a simple way. It was decided that the initial angle between links 2 and 3 should be approximately 30 degrees and the initial angle of q_1 respect to the vertical should be 15 degrees.

The 4-link, treated as an Acrobot here, starts the step with a predefined initial position. It was decided that for this initial position q_2 should have a value of 3.5 radians and q_1 of -0.2 radians. The only dependence in this system would be q_2 on q_1 . Knowing that q_1 will move from -0.2 to $+0.2$ radians and that q_2 starts at 3.5 radians and ends at 2.7 radians a simple Lagrange polynomial was defined considering only initial and final positions. A complete explanation about how to obtain the Lagrange polynomial will be done in subsection 4.2.2. The obtained equation is

$$Phi_2 = q_{2i} \frac{q_1 - q_{1f}}{q_{1i} - q_{1f}} + q_{2f} \frac{q_1 - q_{1i}}{q_{1f} - q_{1i}}, \quad (4.10)$$

where q_{2i} and q_{2f} stand for the initial and final values of q_2 and q_{1i} q_{1f} stand for the initial and final values of q_1 . The trajectory of the angle Phi_2 is then

defined by the equation

$$Phi_2 = 3.5 \frac{q_1 - 0.2}{-0.4} + 2.7 \frac{q_1 + 0.2}{0.4}. \quad (4.11)$$

A simulation in Matlab was used to see the performance. This simulation requires initial data and after the computations shows plots of velocities and angles, it also displays an animation of the 4-link, a more detailed explanation about this simulation will be done in section 4.2. Initial velocities were selected manually until the initial velocity \dot{q}_2 was high enough to complete the step and do not make the 4-link fall down. The initial velocity \dot{q}_1 is 1 rad/sec and \dot{q}_2 is -3 rad/sec. Other angles q_3 , q_4 and the velocities \dot{q}_3 and \dot{q}_4 are zero.

By these results it is possible to conclude that the concept of the virtual constraints works fine. It was tested in a simple system with simple trajectory. Now, it is appropriate to try this concept in a more advanced system considering knees of the 4-link. The first approach was not realistic but was good enough to see the application of the concept in a simplified model.

In figure 4.1 it is possible to see how initial and final positions of the 4-link match with the value of feedback gains (4.3) equal to 150.

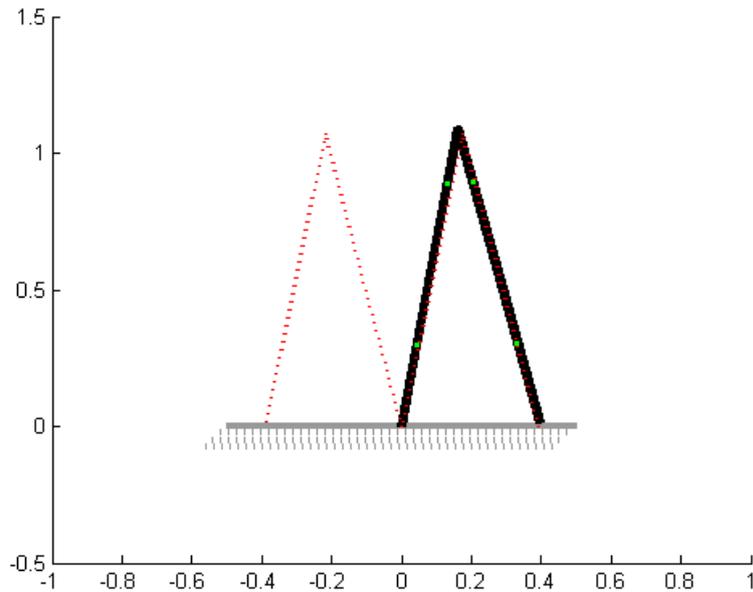


Figure 4.1: 4-link as an Acrobot.

In figures 4.2 and 4.3 it is possible to see the trajectory of the angles and velocities respectively.

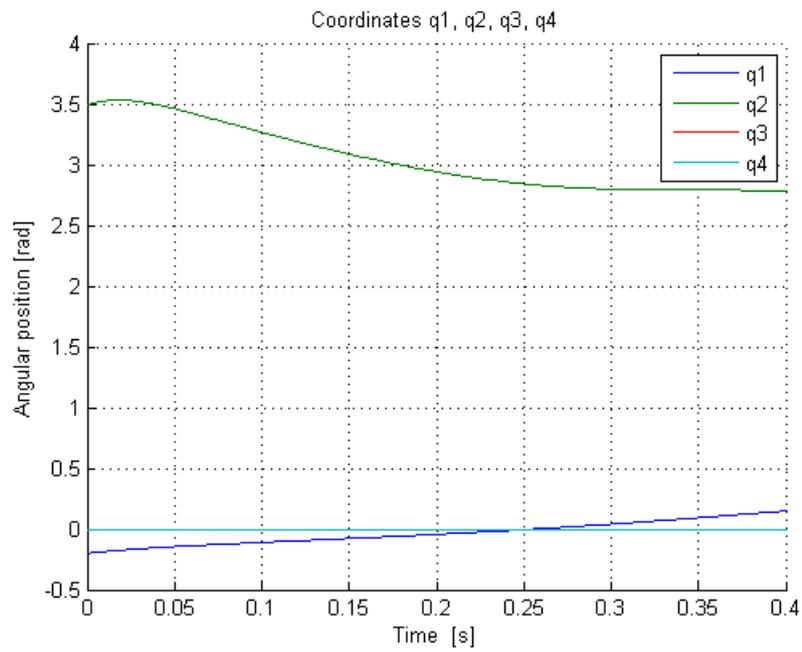


Figure 4.2: Angular positions of 4-link controlled as an Acrobot.

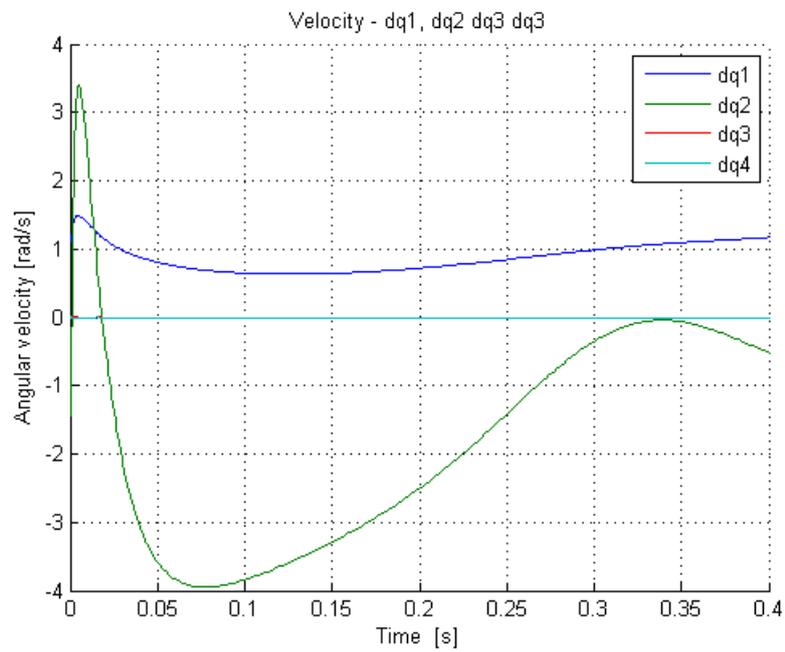


Figure 4.3: Angular velocities of 4-link controlled as an Acrobot.

In figure 4.4 a sequence of a single step for the 4-link restricted to walk like

an Acrobot is shown, this image is obtained from the best result which has very high gain.

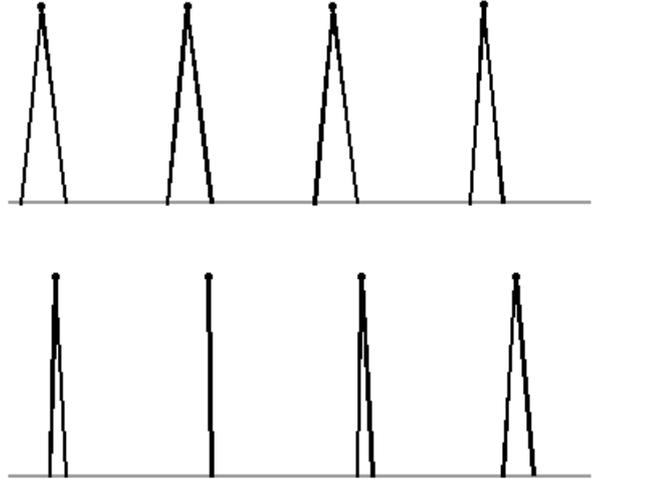


Figure 4.4: Animation of the 4-link as an Acrobot.

4.1.2 Control of 4-link model based on two virtual constraints

In the end of chapter 4 a comparison of the developed way of control of the 4-link walking robot based on method of virtual constraint with the previously developed feedback control in [9] will be made.

The main idea in [9] is based on embedding of the Acrobot into the 4-link walker model. In other words, in approach from [9] only two constraining functions are used, i.e. the original 4-link model with 4 DOF is restricted to a system with 2 DOF only, referred in [9] as the so-called generalized Acrobot.

The detailed description of method of the Acrobot embedding into to 4-link model will be given here, for details see [9]. The control of the Acrobot was based on the partial exact feedback linearization of order 3. As a matter of fact, this linearization is based only on the favorable mentioned property that the Acrobot has kinetic symmetry with respect to the non-actuated angle q_1 . It can be easily seen that so is the case of the generalized Acrobot. Therefore, the generalized Acrobot is order 3 exact feedback linearizable using the following

transformations

$$\begin{aligned}
\xi_1 &= q_1 + \int_0^{q_2} d_{11}^{-1}(s)d_{22}(s)ds, \\
\xi_2 &= d_{11}(q_2)\dot{q}_1 + d_{12}(q_2)\dot{q}_2, \\
\xi_3 &= -G_1(q), \\
\xi_4 &= \frac{\partial G_1}{\partial q_1}(q)\dot{q}_1 + \frac{\partial G_1}{\partial q_2}(q)\dot{q}_2, \\
w_2 &= \dot{q}^\top \frac{\partial \bar{G}^2}{\partial \bar{q}^2}(\bar{q})\dot{\bar{q}} + \left[\frac{\partial \bar{G}_1}{\partial \bar{q}_1}(\bar{q}), \frac{\partial \bar{G}_1}{\partial \bar{q}_2}(\bar{q}) \right] \ddot{q}_1 \\
&\quad \times \bar{D}(\bar{q})^{-1} \begin{bmatrix} 0 \\ \bar{\tau} \end{bmatrix} - \bar{C}(\bar{q}, \dot{\bar{q}}) - \bar{G}(\bar{q}).
\end{aligned} \tag{4.12}$$

As a matter of fact, $\dot{\xi}_1 = \bar{d}_{11}^{-1}(q_2)\xi_2$, further, using Euler-Lagrange formalism it can be seen that $\dot{\xi}_2 = -G_1 = \xi_3$. Finally, it is straightforward, that $\dot{\xi}_3 = \xi_4$, $\dot{\xi}_4 = w_2$. Summarizing, we have the transformed generalized Acrobot

$$\begin{aligned}
\dot{\xi}_1 &= \bar{d}_{11}^{-1}(\bar{q})\xi_2, & \dot{\xi}_2 &= \xi_3, \\
\dot{\xi}_3 &= \xi_4, & \dot{\xi}_4 &= w_2.
\end{aligned} \tag{4.13}$$

For this generalized Acrobot in partially linear coordinates it can be designed a reference walking trajectory along the ideas in [8] as follows. Fixing the initial and final postures during a single step corresponds, in particular, in fixing $\xi_3(0)$ and $\xi_3(T)$ where T denotes the duration of the step. Taking the reference virtual input $w_r = 0$ means, that $\xi_4(t)$ is constant and can be computed as $\xi_4(t) \equiv [\xi_3(T) - \xi_3(0)]/T$. Therefore, the only remaining reference step parameter to be computed is $\xi_2(0)$. This can be done by tuning in simulations in original \bar{q} coordinates, resulting in the so-called **pseudo-passive** reference walking trajectory.

Next, computing the approximate linearization along the above trajectory gives the following approximate tracking error dynamics

$$\begin{aligned}
\dot{e}_1 &= \mu_1(t)e_1 + \mu_2(t)e_2 + \mu_3(t)e_3 \\
\dot{e}_2 &= e_3, \quad \dot{e}_3 = e_4, \quad \dot{e}_4 = w_2.
\end{aligned} \tag{4.14}$$

This error dynamics can be stabilized by various ways, e.g based on LMI approach in [5], or time varying feedback in [9, 1].

The following transformations were used to control of angles in knees in the stance or swing leg using the method of virtual constraints

$$\xi_5 = q_3 - \bar{\Phi}_3(q_2), \quad \xi_6 = \dot{q}_3 - \bar{\Phi}'_3(q_2)\dot{q}_2, \tag{4.15}$$

$$\xi_7 = q_4 - \bar{\Phi}_4(q_2), \quad \xi_8 = \dot{q}_4 - \bar{\Phi}'_4(q_2)\dot{q}_2. \tag{4.16}$$

Transformations (4.15) and (4.16) have similar structure like (4.13). The error of constraints violation for the stance leg has the following dynamics

$$\dot{\xi}_5 = \xi_6, \quad \dot{\xi}_6 = w_3, \tag{4.17}$$

where w_3 is a state feedback controller for angle q_3 with following structure

$$w_3 = -K_3^1 \xi_5 - K_3^2 \xi_6. \tag{4.18}$$

The same dynamics was obtained for the second constraints violation error applied to the swing leg

$$\dot{\xi}_7 = \xi_8, \quad \dot{\xi}_8 = w_4, \quad (4.19)$$

where w_4 is a state feedback controller for angle q_4 with following structure

$$w_4 = -K_4^1 \xi_7 - K_4^2 \xi_8. \quad (4.20)$$

Definitions of functions $\bar{\Phi}_3(q_2)$ and $\bar{\Phi}_4(q_2)$ are given in [9] and will be discussed later. Obviously, the first of them should express the bending of the swing leg while the second one the stretching of the stance leg.

Using of the structure of the dynamics equation of motion (4.1) the following linearizing output was obtained

$$w = \begin{bmatrix} w_2 \\ w_3 \\ w_4 \end{bmatrix} = \beta(q) \begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \\ \ddot{q}_4 \end{bmatrix} + \alpha(q, \dot{q}), \quad (4.21)$$

where

$$\beta(q) = \begin{bmatrix} -\frac{\partial G_1}{\partial q_1} \frac{d_{12}}{d_{11}} + \frac{\partial G_1}{\partial q_2} & -\frac{\partial G_1}{\partial q_1} \frac{d_{13}}{d_{11}} + \frac{\partial G_1}{\partial q_3} & -\frac{\partial G_1}{\partial q_1} \frac{d_{14}}{d_{11}} + \frac{\partial G_1}{\partial q_4} \\ -\bar{\Phi}_3'(q_2) & 1 & 0 \\ -\bar{\Phi}_4'(q_2) & 0 & 1 \end{bmatrix}$$

and

$$\alpha(q, \dot{q}) = \begin{bmatrix} -\frac{\partial G_1}{\partial q_1} \left(\frac{C^1(q, \dot{q}) \dot{q}}{d_{11}} + \frac{G_1(q)}{d_{11}} \right) + \frac{\partial^2 G_1(q)}{\partial q^2} \dot{q}^2 \\ -\bar{\Phi}_3''(q_2) \dot{q}_2^2 \\ -\bar{\Phi}_4''(q_2) \dot{q}_2^2 \end{bmatrix}$$

terms $d_{11,12,13,14}$ are appropriate scalar entries in inertia matrix D , C^1 is the first line of matrix C and G_1 is the first scalar entry in vector G according to (4.2).

Substituting (4.6) in (4.7) it is possible to obtain primary equation (4.7) in following form

$$\bar{D}(q) \begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \\ \ddot{q}_4 \end{bmatrix} = \begin{bmatrix} \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} - \bar{C}(q, \dot{q}) \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{pmatrix} - \bar{G}(q), \quad (4.22)$$

where

$$\bar{D} = \begin{bmatrix} d_{22} - \frac{d_{21}d_{12}}{d_{11}} & d_{23} - \frac{d_{21}d_{13}}{d_{11}} & d_{24} - \frac{d_{21}d_{14}}{d_{11}} \\ d_{32} - \frac{d_{31}d_{12}}{d_{11}} & d_{33} - \frac{d_{31}d_{13}}{d_{11}} & d_{34} - \frac{d_{31}d_{14}}{d_{11}} \\ d_{42} - \frac{d_{41}d_{12}}{d_{11}} & d_{43} - \frac{d_{41}d_{13}}{d_{11}} & d_{44} - \frac{d_{41}d_{14}}{d_{11}} \end{bmatrix},$$

$$\bar{C} = \begin{bmatrix} C^2 - C^1 \frac{d_{21}}{d_{11}} \\ C^3 - C^1 \frac{d_{31}}{d_{11}} \\ C^4 - C^1 \frac{d_{41}}{d_{11}} \end{bmatrix}, \quad \bar{G} = \begin{bmatrix} G_2 - G_1 \frac{d_{21}}{d_{11}} \\ G_3 - G_1 \frac{d_{31}}{d_{11}} \\ G_4 - G_1 \frac{d_{41}}{d_{11}} \end{bmatrix}.$$

From equations (4.22) and (4.21) is possible to obtain final equation for torque which can be directly applied to actuators

$$\begin{bmatrix} \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} = \bar{D}(q) \beta^{-1}(q) (w - \alpha(q, \dot{q})) + \bar{C}(q, \dot{q}) \dot{q} + \bar{G}(q). \quad (4.23)$$

Vector $w = [w_2, w_3, w_4]^T$ contains particular feedback controllers for angles q_2 , q_3 and q_4 . The angle q_2 is controlled through the structure of embedded Acrobot and angles q_3 and q_4 are controlled using method of virtual constraint.

4.2 Defining the constraining functions for the 4-link model

The definition of constraining functions is very important for the virtual constraint method. One of the aims of this thesis is to compare the developed virtual constraint method with previously developed method of embedding Acrobot into 4-link system in the previous section. From this reason, trajectories of dependence q_2 , q_3 and q_4 on q_1 were taken as reference. In section 4.2.1 they will be approximated by regression method, in sections 4.2.2 and 4.2.3 they will be approximated using the method of Lagrange polynomial in order to find the simplest constraining functions to control 4-link model. The main idea is that the 4-link will start and finish in the same position but with shifted legs; this is because after a single step conditions can be switched and the 4-link can repeat the step to make a continuous walk.

A simulation of the Acrobot in Matlab will be used to see the performance of the constraining functions. The simulation requires predefined values for all the angles, q_1 , q_2 , q_3 and q_4 , also needs initial values for the velocities \dot{q}_2 , \dot{q}_3 and \dot{q}_4 . The initial velocity for \dot{q}_1 should also be defined, different values will be tried to see how the 4-link behaves under diverse initial velocities for q_1 . The simulation is designed to be stopped by position control, this happens when the foot is close enough to the ground to say that the step is finished. The time the simulation should run is another parameter to define, this time is not very important under these circumstances since the simulation is stopped by position. Nevertheless, while running some simulations the foot can be too close to the ground before the swing leg has finished its movement due to different factors, for example the leg is not lifted enough and reaches the limits of the stop by position too soon. In these cases it will be necessary to remove the stop by position control and set a limit to the time the simulation should run. In figure 4.5 an image of the stop by position control in Matlab can be seen.

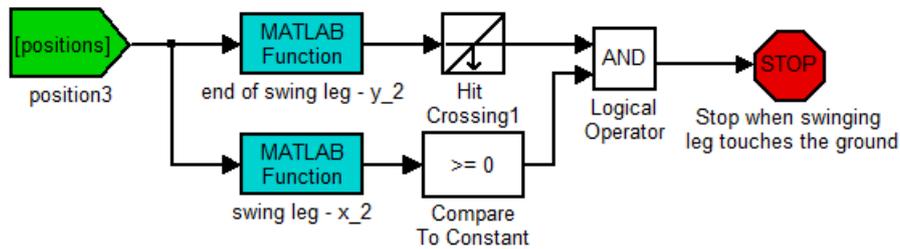


Figure 4.5: Stop by position control in Matlab.

If a simulation is stopped by position before reaching the final desired values then it is an indication that the simulation was not successful. Nevertheless,

it is useful to see the full performance to see how it can be improved. After removing the stop by position control there is the risk that the time set for the simulation is too long. It is possible for the velocities to increase at a very high rate during the computation of the simulation, like going to infinity. In this case the computation takes too long and has to be stopped and the time of the step has to be reduced.

4.2.1 Regresion approach

For this approach a regression of trajectories of q_2 q_3 and q_4 in section 4.1.2 was done using Matlab tools to fit equations that best describe trajectories that angles should follow. The originally created functions for purpose of embedding Acrobot into 4-link model had dependance on angle q_2 . But, it is not possible to fit the original dependance of functions on angle q_1 because they have not one-to-one correspondence, see figures 4.6 and 4.7.

In figure 4.6 it is possible to see in red color the trajectory that the angle q_2 should follow and the actual trajectory obtained after the regression is shown in blue. Figure 4.7 shows the same information but for the angle q_3 . For angle q_4 the trajectory has the same shape that the one for q_3 but with different values, it is shifted upward in position.

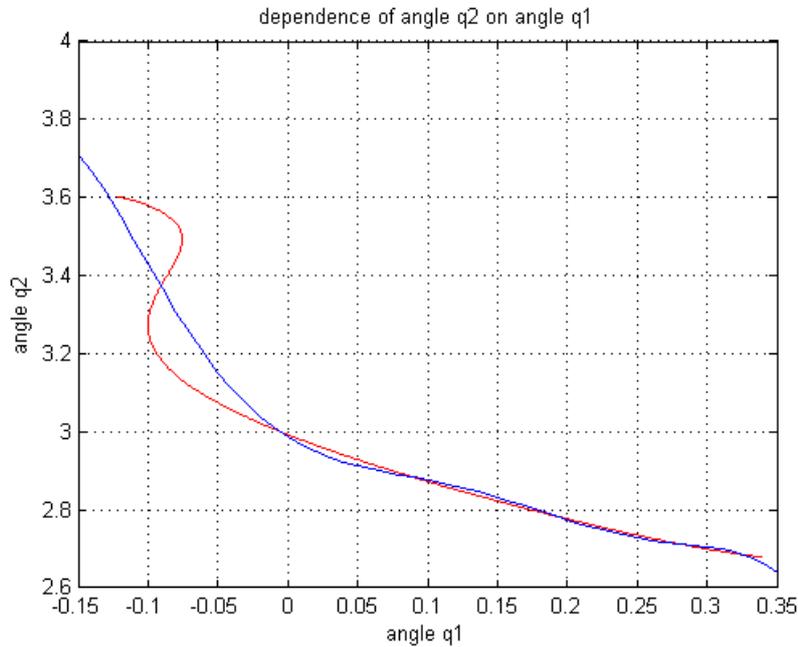


Figure 4.6: Dependance of angle q_2 on q_1 .

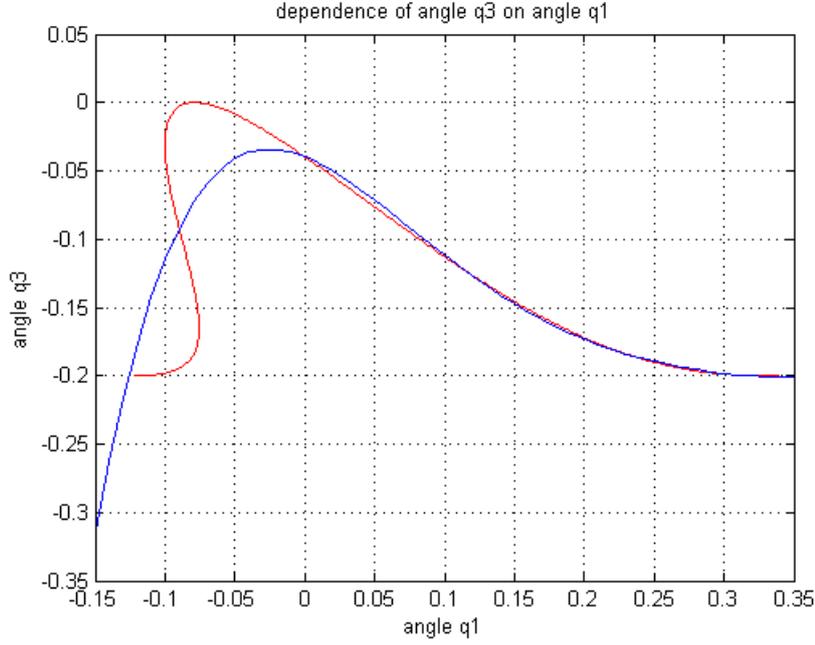


Figure 4.7: Dependence of angle q_3 on q_1 .

The values of q_1 run from -0.15 to 0.35 and the equation obtained from the regressions are

$$Phi_2 = -2733.1q_1^6 + 1837.2q_1^5 - 228.04q_1^4 - 72.828q_1^3 + 19.186q_1^2 - 2.2207q_1 + 2.9859, \quad (4.24)$$

$$Phi_3 = 78.943q_1^5 - 94.443q_1^4 + 41.297q_1^3 - 6.4298q_1^2 - 0.40561q_1 - 0.039581, \quad (4.25)$$

and

$$Phi_4 = 78.943q_1^5 - 94.443q_1^4 + 41.297q_1^3 - 6.4298q_1^2 - 0.40561q_1 + 0.36042. \quad (4.26)$$

The initial conditions are also taken from previous method, note that q_2 needs to have high velocity in order to avoid falling back. The initial condition are as follow

$$\begin{aligned} q_{10} &= -0.1213, \\ q_{20} &= 3.5612, \\ q_{30} &= -0.1812, \\ q_{40} &= 0.2188, \\ dq_{20} &= -7.9128, \\ dq_{30} &= 4.8934, \\ dq_{40} &= 4.8934, \end{aligned} \quad (4.27)$$

where the subindex 0 denotes that is an initial condition and each angle is expressed in radians and velocities in radians per second. The only values

allowed to be changed are the initial velocity q_1 and the time of the simulation. Different initial velocities were tried to see how the 4 link behaves.

For an initial velocity dq_{10} of 1.5 rad/sec the time of the simulation can be limited to 0.115 seconds to get a satisfactory result, previously there were small misalignments that were possible to correct without increasing the gain. The gain was set to 100 for all values and the simulation is stopped by position through control in simulink. The results can be seen in figures 4.8, 4.9 and 4.10. It is possible to see that the step has the correct length and the axis in red are within the limits of the width of the legs.

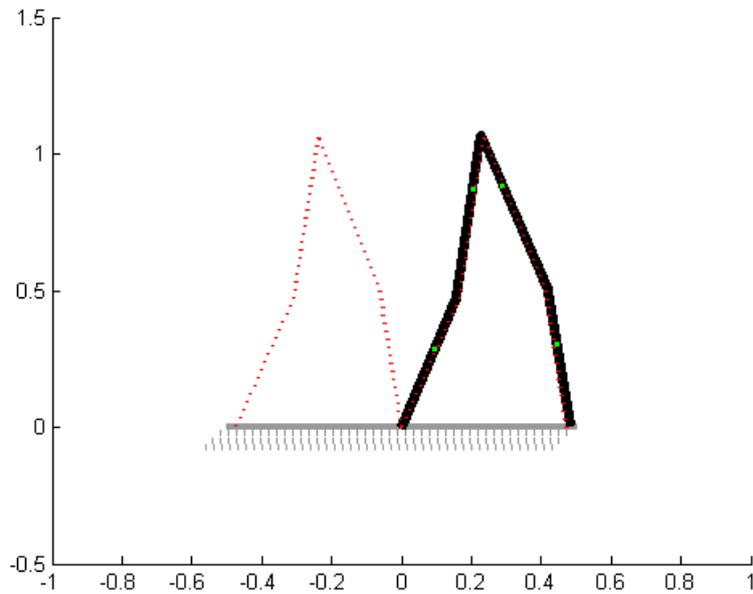


Figure 4.8: Step with regression approach, \dot{q}_1 of 1.5 rad/sec and 0.115 time simulation.

The previous result is satisfactory, now the same conditions were used to test the 4 link at different initial velocities. For an initial velocity of 2 rad/sec the 4-link performs well, having only small misalignments seen on links 2 and 3, links 1 and 4 have misalignments that were very small. The simulation is stopped by position around 0.110 seconds which means that the time of 0.115 seconds did not run completely. A new run was done removing the stop by position control which leads to a very long step, approximately 30% longer.

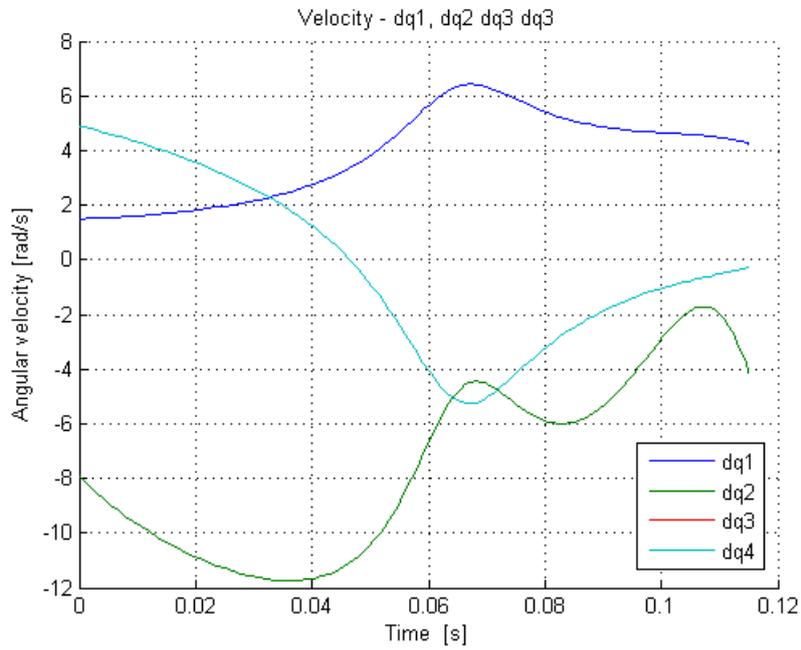


Figure 4.9: Velocities with regression approach, \dot{q}_1 of 1.5 rad/sec and 0.115 time simulation.

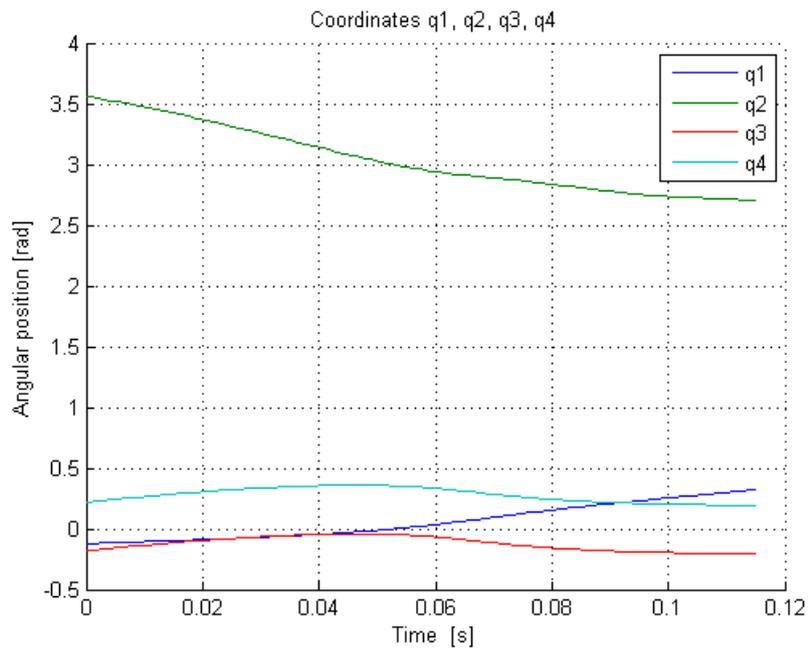


Figure 4.10: Angle trajectory with regression approach, \dot{q}_1 of 1.5 rad/sec and 0.115 time simulation.

For an initial velocity of 1 rad/sec and time limit of 0.115 seconds with stop by position control the step is unfinished, it seems obvious that at lower velocities it will be required longer time to finish the step, but since 0.115 seconds was too much for initial velocity of 2 rad/sec this time was not modified at first. New simulations were done increasing the time to improve the results, finding a good step at 0.130 seconds, it is seen that it is possible achieve good results for the 4-link without modifying the gain due to the time step factor.

Setting time back to 0.115 seconds to try slower initial velocity of 0.5 rad/sec, to compare with initial assumptions, results in an unfinished step. This was expected since the initial velocity is too low to achieve the desired final positions. The time is then increased to 0.140 seconds for which the step was almost finished but with big misalignments. With time of 0.150 seconds the step was a little longer. It was difficult to achieve satisfactory result at the intermediate points between 0.140 and 0.150 seconds, for example at 0.145 seconds link 2 was outside the axis of the final position it should follow, it was not possible to get good results with the change of time. From this reason, it was decided to increase the gain. After trying gains of 200, 300 and 1000 it was not possible to achieve good performance. It was then obvious the relevance of the initial velocity. It was not possible to achieve desired results at very low initial velocities even with very high gain.

In figure 4.11 it is possible to see a sequence of movements that the 4-link performs with an initial velocity for q_1 of 1.5 red/sec, the simulation stopped by position control around 0.115 seconds.

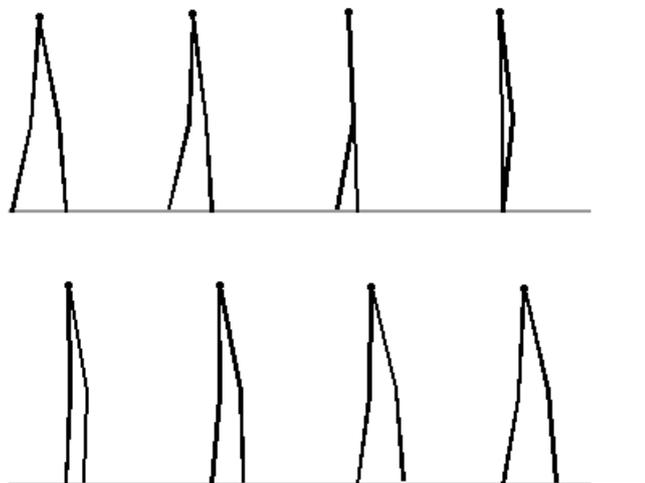


Figure 4.11: 4-link animation sequence with regression approach.

4.2.2 Lagrange Polynomial

Even when an approximation for a given trajectory was being made, it is not so necessary for it to be exactly the same. The main characteristic of the trajectory is to have well established initial and final position so the 4-link can “finish where it started” with shifted legs and start the cycle again, making a cyclic walking. It was decided to use Lagrange polynomials to approach the trajectory of the angles. The Lagrange polynomial uses two or more x and y pairs to make a polynomial that passes through all the specified points. In this case this points are expressed by q_1 for x and q_2, q_3 or q_4 for y respectively. For Phi_2 a polynomial with two points was calculated having degree 1. For Phi_3 , three points were defined to make the polynomial of degree 2 and for Phi_4 , four points were used to make a degree 3 Lagrange polynomial. All equations take into account initial and final values and if more points are required they take intermediate values in the desired trajectory. Resulting equations are as follow

$$Phi_2 = q_{2i} \frac{(q_1 - q_{1f})}{(q_{1i} - q_{1f})} + q_{2f} \frac{(q_1 - q_{1i})}{(q_{1f} - q_{1i})}, \quad (4.28)$$

where initial velocity for angle q_1 is q_{1i} with a real value of -0.1213 radians, q_{1f} is the final value of q_1 with a real value of 0.3390 radians. The initial value for q_2 is 3.6025 radians denoted by q_{2i} , q_{2f} is the final q_2 with real value of 2.6787 radians.

For Phi_3 the Lagrange polynomial equation calculated is

$$Phi_3 = q_{3i} \frac{(q_1 - q_{1m3})(q_1 - q_{1f})}{(q_{1i} - q_{1m3})(q_{1i} - q_{1f})} + q_{3m3} \frac{(q_1 - q_{1i})(q_1 - q_{1f})}{(q_{1m3} - q_{1i})(q_{1m3} - q_{1f})} + q_{3f} \frac{(q_1 - q_{1i})(q_1 - q_{1m3})}{(q_{1f} - q_{1i})(q_{1f} - q_{1m3})}, \quad (4.29)$$

where the first point is given by (q_{1i}, q_{3i}) with real value of $(-0.1213, -0.2000)$. The final point (q_{1f}, q_{3f}) has real value of $(0.3390, -0.2000)$. The medium point is expressed by (q_{1m3}, q_{3m3}) with real value of $(0, -0.0475)$. All the values are in radians. Finally Phi_4 is expressed by

$$Phi_4 = q_{4i} \frac{(q_1 - q_{1m4})(q_1 - q_{1n4})(q_1 - q_{1f})}{(q_{1i} - q_{1m4})(q_{1i} - q_{1n4})(q_{1i} - q_{1f})} + q_{4m} \frac{(q_1 - q_{1i})(q_1 - q_{1n4})(q_1 - q_{1f})}{(q_{1m4} - q_{1i})(q_{1m4} - q_{1n4})(q_{1m4} - q_{1f})} + q_{4n} \frac{(q_1 - q_{1i})(q_1 - q_{1m4})(q_1 - q_{1f})}{(q_{1n4} - q_{1i})(q_{1n4} - q_{1m4})(q_{1n4} - q_{1f})} + q_{4f} \frac{(q_1 - q_{1i})(q_1 - q_{1n4})(q_1 - q_{1m4})}{(q_{1f} - q_{1i})(q_{1f} - q_{1n4})(q_{1f} - q_{1m4})}, \quad (4.30)$$

values are as follow, initial q_1 and q_4 are denoted by q_{1i} and q_{4i} with real values of -0.1213 and 0.2000 radians respectively. The final point is given by q_{1f} with value of 0.3390 radians and q_{4f} with value of 0.2000 radians. Phi_4 requires 4 points to be made, the other extra are represented by (q_{1m4}, q_{4m}) , with value of $(0, 0.3525)$, and q_{1n4}, q_{4n} with value of $(0.15, 0.25)$ radians.

Names of points in equations have many sub-indices to avoid confusion, for example q_{1m4} represents angle q_1 in a medium point denoted by m and belongs to the Phi_4 denoted by the number 4. Position of the foot of the swing leg between the initial and final value is chosen within an approximation of a real trajectory. In figures 4.12, 4.13 and 4.14 pictures it is possible to see the new trajectories of Phi_2 , Phi_3 and Phi_4 .

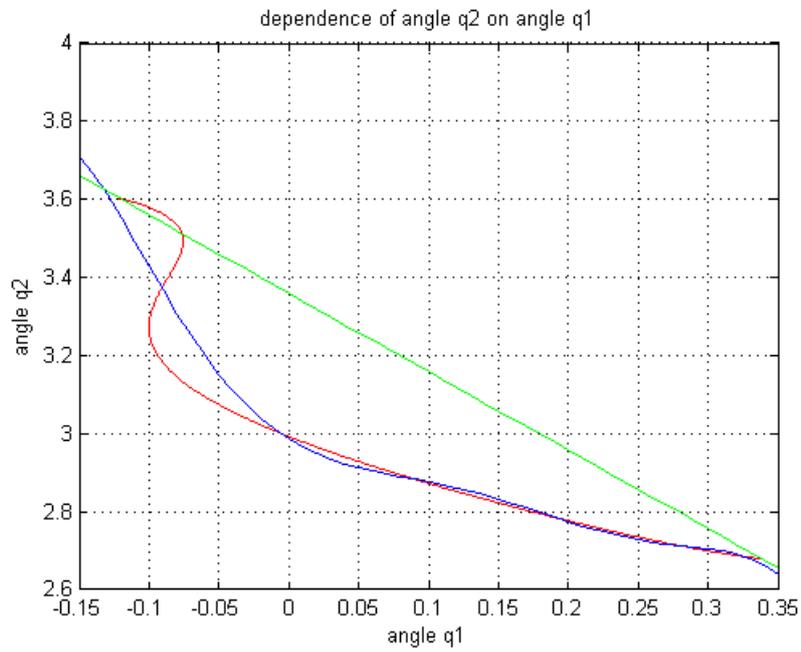


Figure 4.12: New trajectory of q_2 on q_1 .

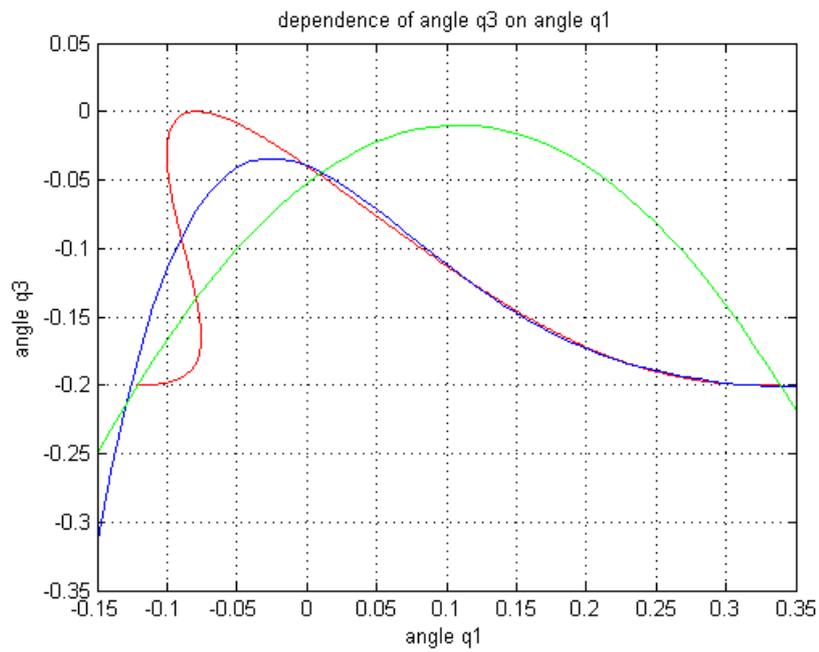


Figure 4.13: New trajectory of q_3 on q_1 .

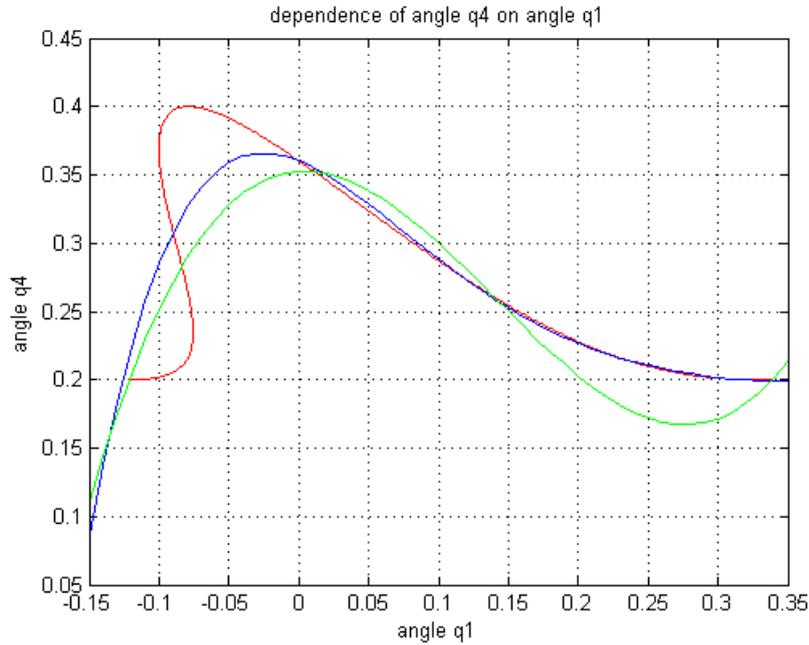


Figure 4.14: New trajectory of q_4 on q_1 .

Following similar conditions as tested on the regression method for better comparison of results of these equations were tested at different velocities. First, for an initial velocity q_1 of 1.5 rad/sec and a time 0.250 seconds and gain 100, simulation was run and stopped around 0.090 seconds at the middle of the step when both feet are overlapped. Because of this, the stop by position control was removed to see full performance. Velocities start to increase at very high rate around 0.130 seconds and the computation takes too long. That is why a limit was set to 0.120 seconds. As a result, the step was improved but still shorter and there is misalignment in all links. Time was set to 0.125 seconds which gave an already too long step. Trying shorter time again, 0.121 seconds, the performance was improved the misalignment is small and small changes in time lead to large errors that it was decided to try with the gain increased to 1000 directly to see best results. This leads to better results but yet not satisfactory.

Due to the difficulty to achieve good results with this first velocity, small changes in the velocity were tried. First with 100 gain and limit of 0.125 seconds, 1.4 rad/sec was set as initial velocity. This leads to a final correct position of feet but link 3 and 4 are not aligned in the axis as they should. Reduced velocity for same conditions gives a slightly shorter step. It was difficult to find satisfactory results.

For a high initial velocity 2 rad/sec the simulation time has to be reduced to less than 0.110 seconds. Otherwise, velocities would increase at very fast rate, but still 0.115 seconds as limit in time gives a long step, 0.105 seconds as limit gives a shorter step than required even with these small changes in time the difference in the step is big.

For initial velocity of 1 rad/sec the best step is given at 0.139 seconds but

there are misalignments. Since changes in time will affect the length of the step, the gain was set to 1000 to try to improve performance. Nevertheless, unsatisfactory results were obtained, the step is longer than required.

For velocity of 0.5 rad/sec and gain of 100 the best step was achieved at time 0.161 seconds. The length is good but there is misalignment in links 2, 3 and 4. In figures 4.15, 4.16 and 4.17 it is possible to see some examples of unsatisfactory results for longer and shorter steps also for a step of required length but with misalignment. In figure 4.18 it is possible to see the velocities increasing rapidly if the simulation is let to run longer periods of time which means that the simulation never stops computing.

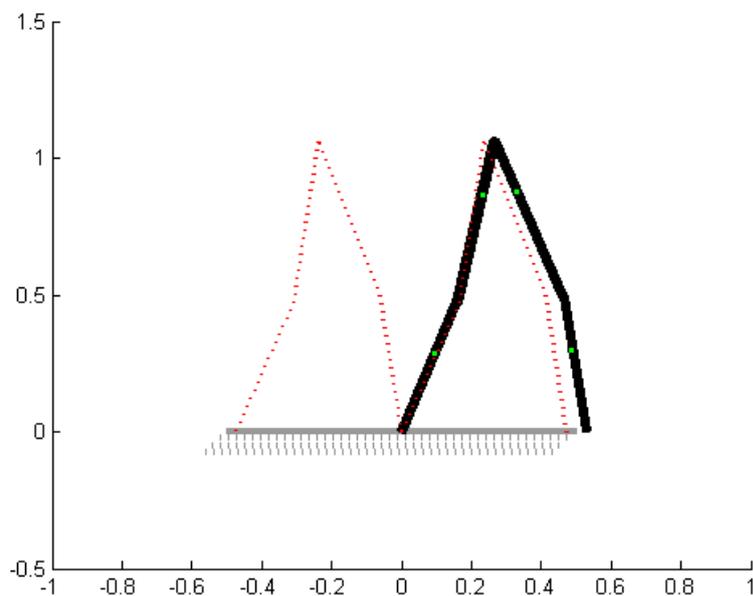


Figure 4.15: Example of a longer step.

From this results it can be concluded that a simple Lagrange polynomial is an easy way to find the trajectories that the angles for the 4-link should follow, just by having initial and final positions and one or two other intermediate points. Nevertheless, even when some almost satisfactory results were obtained, the simulation has to be stopped by time or the velocities would go to infinity. It was very difficult to find conditions for good performance and what is more important, as can be seen in figure 4.19, the 4-link seems to have an intermediate position where the foot of the swing leg goes under the ground which would be impossible in reality. This leads to conclude that a polynomial of higher degree is required.

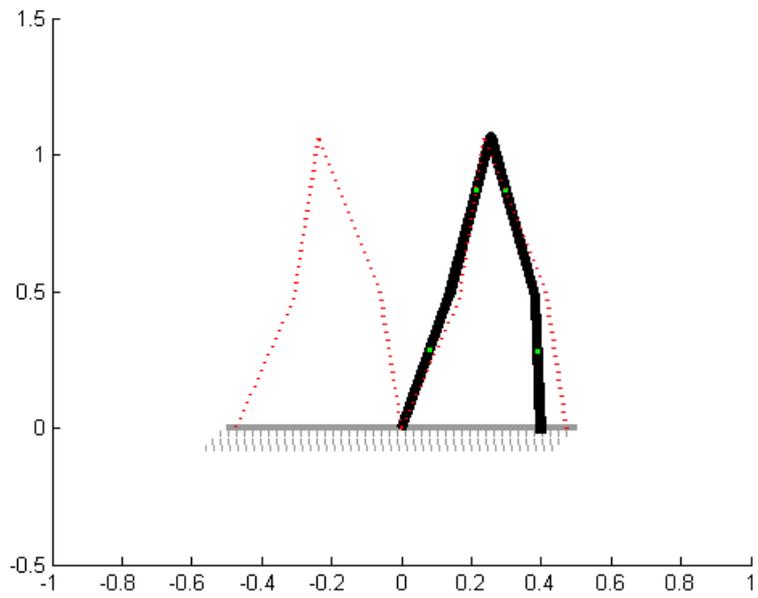


Figure 4.16: Example of a shorter step.

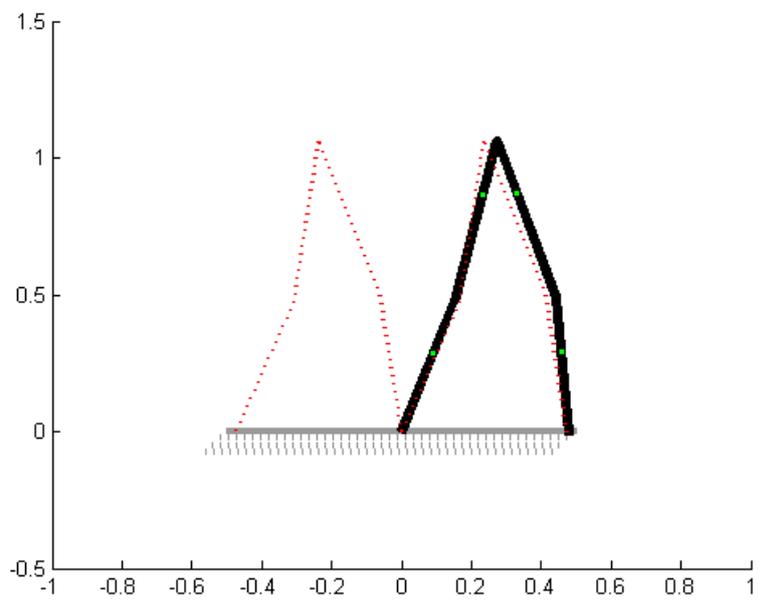


Figure 4.17: Example of a step of proper length but with misalignment.

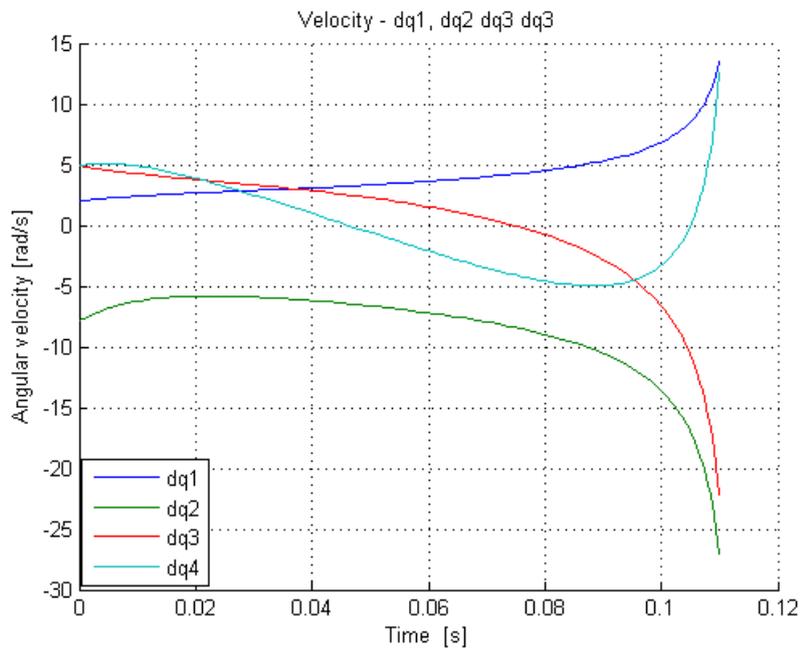


Figure 4.18: Example of velocities going to infinity.

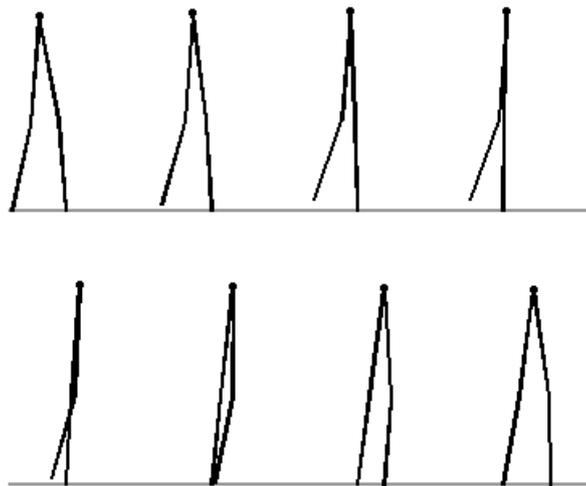


Figure 4.19: 4-link animation sequence with Lagrange Polynomial approach.

4.2.3 Increased Lagrange Polynomial

The previous approach was too simple to achieve good performance at different velocities. It is possible that this is due to the big differences in trajectories of q_2 and q_3 of the simple Lagrange polynomial compared with trajectories in section 4.1.2. Even more, there is risk that the 4-link actually goes under the ground. Because of this it was decided to try a Lagrange polynomial with 4 points in all the angle's trajectories. This means polynomials of degree 3. The new polynomials are

$$\begin{aligned}
Phi_2 = & q_{2i} \frac{(q_1 - q_{1m2})(q_1 - q_{1n2})(q_1 - q_{1f})}{(q_{1i} - q_{1m2})(q_{1i} - q_{1n2})(q_{1i} - q_{1f})} + \\
& q_{2m2} \frac{(q_1 - q_{1i})(q_1 - q_{1n2})(q_1 - q_{1f})}{(q_{1m2} - q_{1i})(q_{1m2} - q_{1n2})(q_{1m2} - q_{1f})} + \\
& q_{2n2} \frac{(q_1 - q_{1i})(q_1 - q_{1m2})(q_1 - q_{1f})}{(q_{1n2} - q_{1i})(q_{1n2} - q_{1m2})(q_{1n2} - q_{1f})} + \\
& q_{2f} \frac{(q_1 - q_{1i})(q_1 - q_{1n2})(q_1 - q_{1m2})}{(q_{1f} - q_{1i})(q_{1f} - q_{1n2})(q_{1f} - q_{1m2})},
\end{aligned} \tag{4.31}$$

where q_{1i} denotes initial q_1 with value of -0.1213 and q_{1f} is the final q_1 with value of 0.3390 . The initial and final points for q_2 are 3.6025 and 2.6787 respectively denoted by q_{2i} and q_{2f} . There are other two intermediate points $(0.0, 3.0)$ and $(0.2, 2.8)$ denoted by q_{1m2} , q_{2m2} , q_{1n2} and q_{2n2} respectively. Phi_3 is expressed by

$$\begin{aligned}
Phi_3 = & q_{3i} \frac{(q_1 - q_{1m3})(q_1 - q_{1n3})(q_1 - q_{1f})}{(q_{1i} - q_{1m3})(q_{1i} - q_{1n3})(q_{1i} - q_{1f})} + \\
& q_{3m3} \frac{(q_1 - q_{1i})(q_1 - q_{1n3})(q_1 - q_{1f})}{(q_{1m3} - q_{1i})(q_{1m3} - q_{1n3})(q_{1m3} - q_{1f})} + \\
& q_{3n3} \frac{(q_1 - q_{1i})(q_1 - q_{1m3})(q_1 - q_{1f})}{(q_{1n3} - q_{1i})(q_{1n3} - q_{1m3})(q_{1n3} - q_{1f})} + \\
& q_{3f} \frac{(q_1 - q_{1i})(q_1 - q_{1n3})(q_1 - q_{1m3})}{(q_{1f} - q_{1i})(q_{1f} - q_{1n3})(q_{1f} - q_{1m3})},
\end{aligned} \tag{4.32}$$

where the initial q_1 is denoted by q_{1i} with value -0.1213 radians and final by q_{1f} with value 0.3390 radians. Initial q_3 is represented by q_{3i} with value of -0.2000 radians and final is represented by q_{3f} also with value -0.2000 radians. Intermediate points are $(0.0000, -0.0475)$ and $(0.1500, -0.1500)$ in radians and represented by q_{1m3} , q_{3m3} , q_{1n3} and q_{3n3} respectively. Phi_4 is exactly the same as in subsection 4.2.2.

In figures 4.20, 4.21 and 4.22 it is possible to see the comparison between the new polynomial (in dotted blue), the regression (continuous blue) and original trajectory from which the regression was done (in red). It is possible to see how new polynomials are similar to ones obtained by the regression even when they are of lower order.

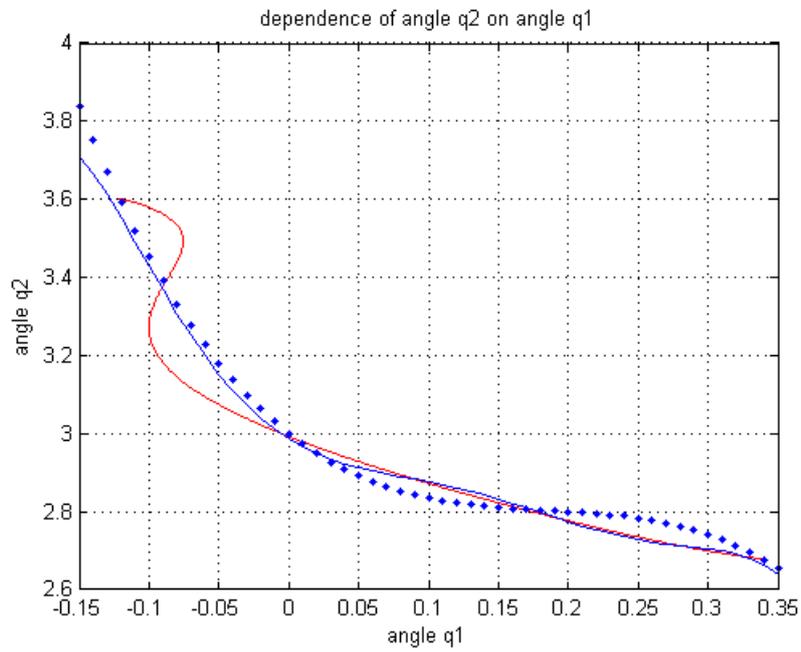


Figure 4.20: 4 Points Lagrangian Polynomial q_2 on q_1 .

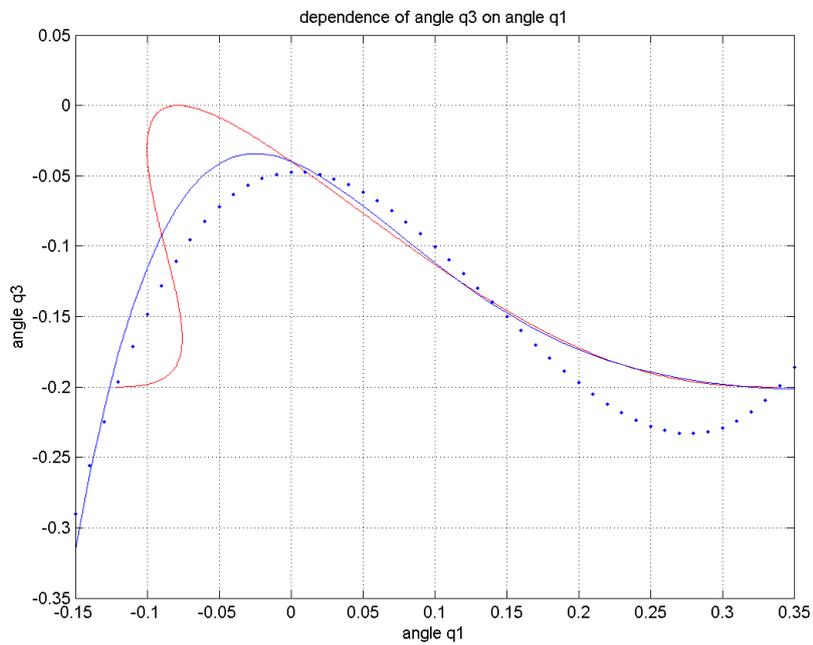


Figure 4.21: 4 Points Lagrangian Polynomial q_3 on q_1 .

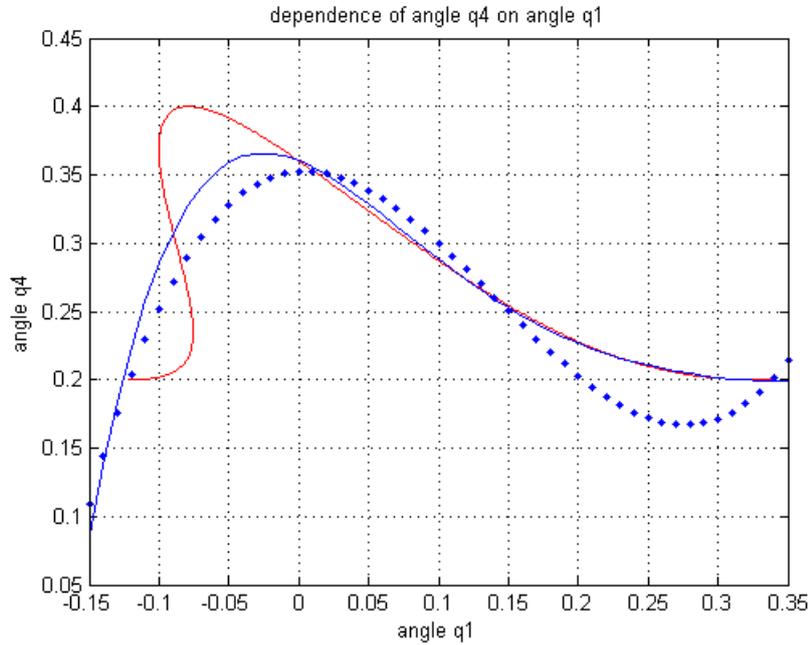


Figure 4.22: 4 Points Lagrangian Polynomial q_4 on q_1 .

Initial values for angles and velocities are the same as in the regression, except for q_1 for which different initial velocities will be tried. For an initial velocity of 1.5 rad/sec, gain of 100, and the stop by position control connected, time was set to 0.250 seconds. For these conditions the step is good, but it is possible to see that the simulation stops around 0.110 seconds which means that the stop by position control makes the step slightly shorter. Due to the fact that it is stopped by position increasing time would not modify the situation. It was decided to modify the gain increasing it to 1000 to test this approach. This leads to a slightly larger step. The gain does not play so much role as the initial velocity, higher velocity makes a shorter step in size and time.

For a lower velocity of 1 rad/sec step is unfinished. Because of this, time was increased to 0.250 seconds, which leads to a good step that stops by position around 0.130 seconds. The best step is achieved at 1.100 seconds, the results can be seen in figure 4.23, 4.24 and 4.25

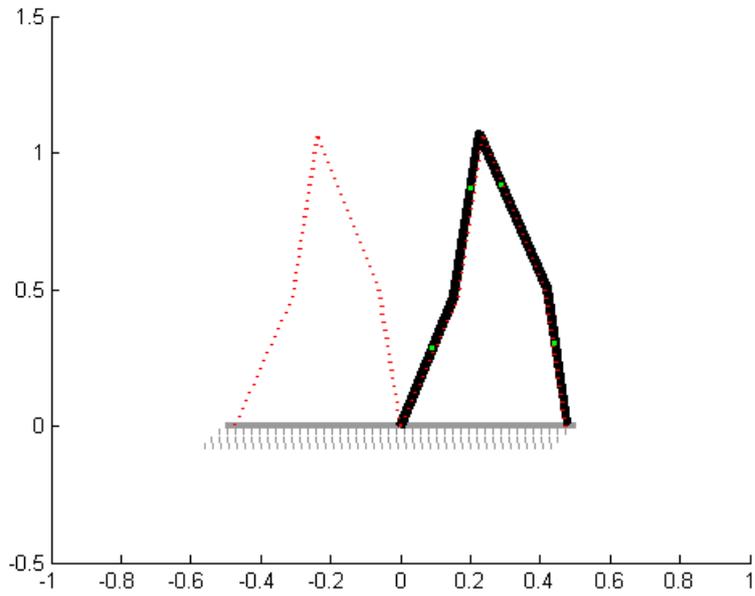


Figure 4.23: 4 Points Lagrangian Polynomial's best step.

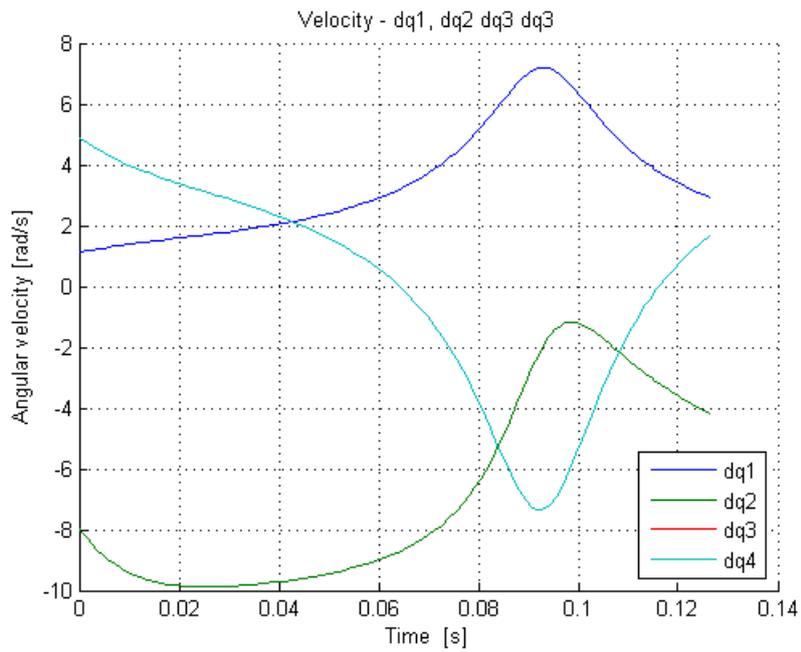


Figure 4.24: 4 Points Lagrangian Polynomial velocities for best step.

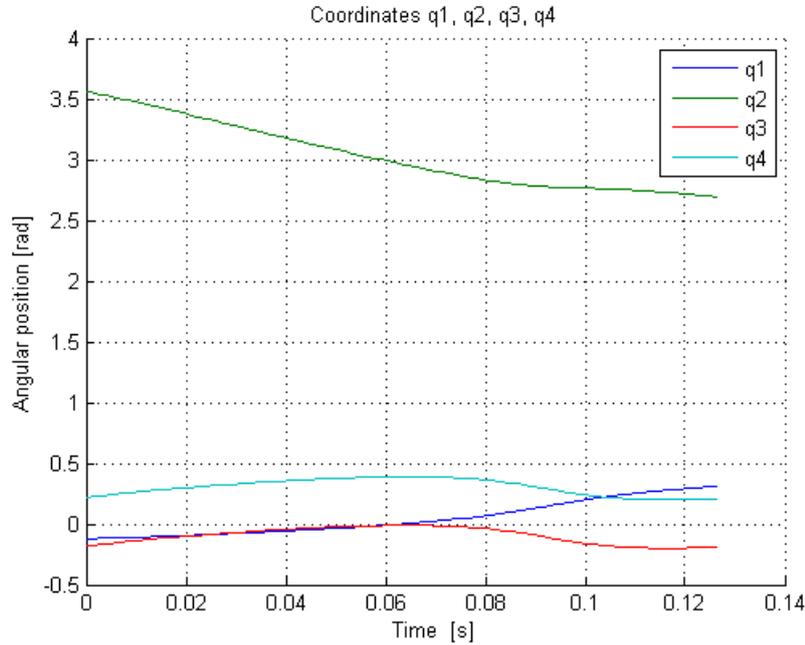


Figure 4.25: 4 Points Lagrangian Polynomial angles for best step.

For lower initial velocity of 0.5 rad/sec the step is longer and stops around 0.155 seconds. Because of this, time limit was decreased to 0.145 seconds which results in a good step with proper length but with misalignment of link 2 and 3. The 4-link is very sensitive to small changes in time. It leads to longer or shorter steps. It was decided to try with increased gain. A gain of 1000 results in a step of almost proper length, little longer and with small misalignment of link 2 mainly. A small improvement was done at very large gain cost. In figure 4.26 the sequence of steps for the Lagrange Polynomial of 4 points is shown. It corresponds to an initial velocity of q_1 of 1.5 rad/sec with stop by position control.

What can be seen now from a comparison of different approaches like regression and Lagrange polynomials of different degrees, is that even when good results can be achieved by the 3 methods. The regression needs a previous set of values. For the Lagrange Polynomials it is only necessary to have initial and final values for angles and a few intermediate values. The first Lagrange polynomial approach was too simple and the 4-link was difficult to tune. Especially, it had to be stopped by time. The second approach with increased degree for the polynomial was better to achieve good results, especially because the simulation can run completely and be stopped by position. It has a good performance around the values of main interest this is 1.1 and 1.5 rad/sec approximately.

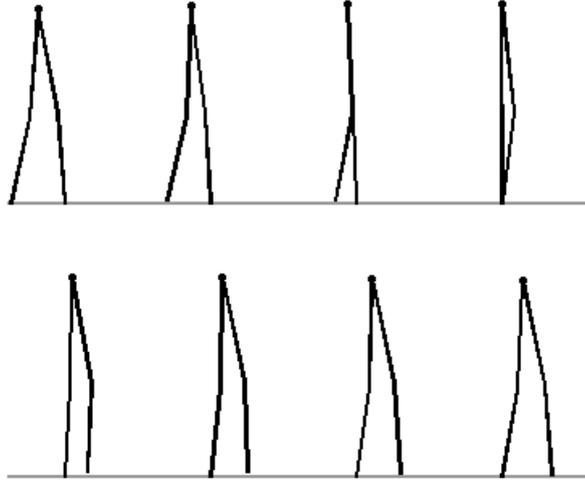


Figure 4.26: 4-link animation sequence with Lagrange Polynomial of 4 points approach.

4.3 Comparison of control algorithms for 4-link model

Since the best approach was given with the Lagrange polynomial of degree 3, this was compared with the approach of embedding Acrobot in subsection 4.1.2. Comparison of angular position and velocities was done, the result for angular positions can be seen in figures 4.27 and 4.28. It can be seen that for the angular position for angles q_1 and q_2 there is not much difference since this was calculated to be similar with the Lagrange polynomial and for angles q_3 and q_4 positions are not equal through time but it follows a similar path.

Nevertheless, for velocities it can be seen that they are completely different through time, shown in figures 4.29 and 4.30. For accelerations of the three virtual constraints method it is seen that they are higher at the very beginning of the step as seen in the slopes of velocities dq_1 and dq_2 to then decrease the velocity almost to zero and then accelerate again at the end of the step. While the embedded approach accelerates more slowly to higher velocity than the three virtual constraints method at the beginning of the step and then decelerates smoothly till the end of the step.

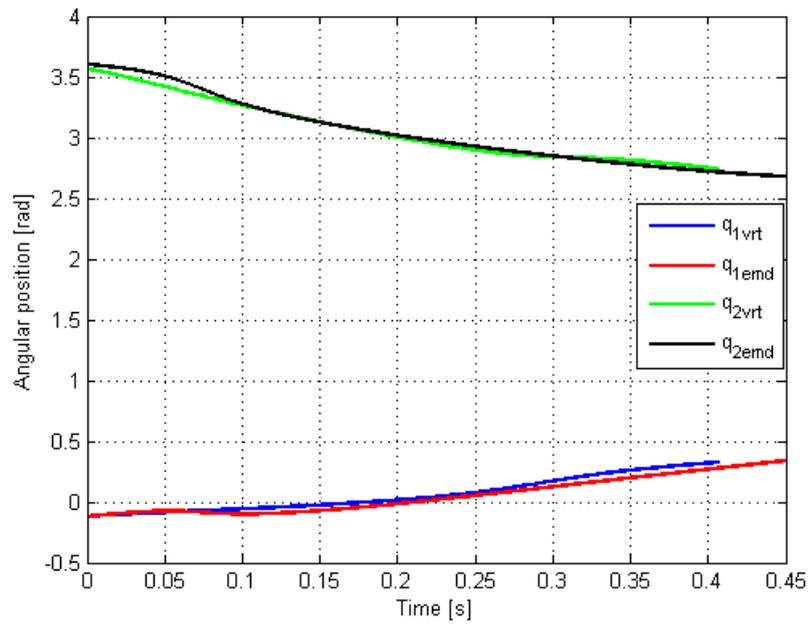


Figure 4.27: Comparison of q_1 and q_2 for Acrobot embedding method.

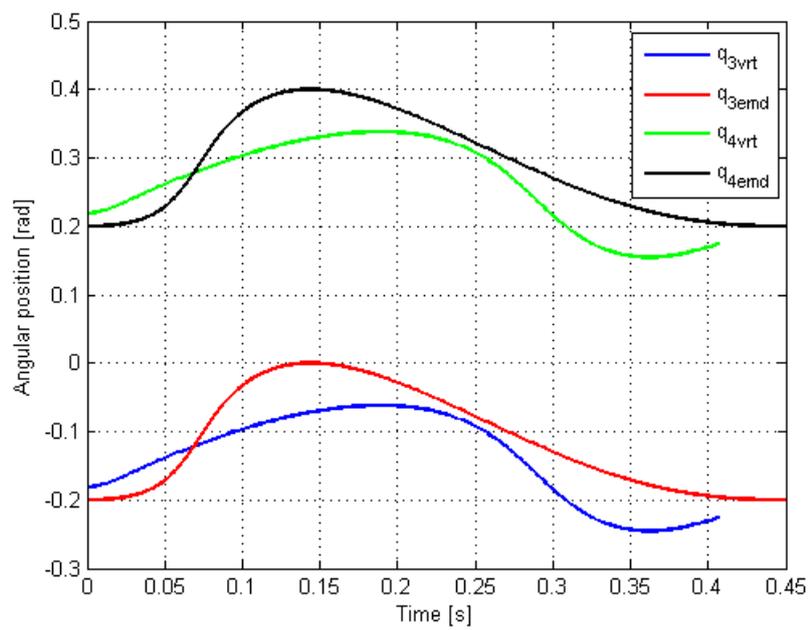


Figure 4.28: Comparison of q_3 and q_4 for Acrobot embedding method.

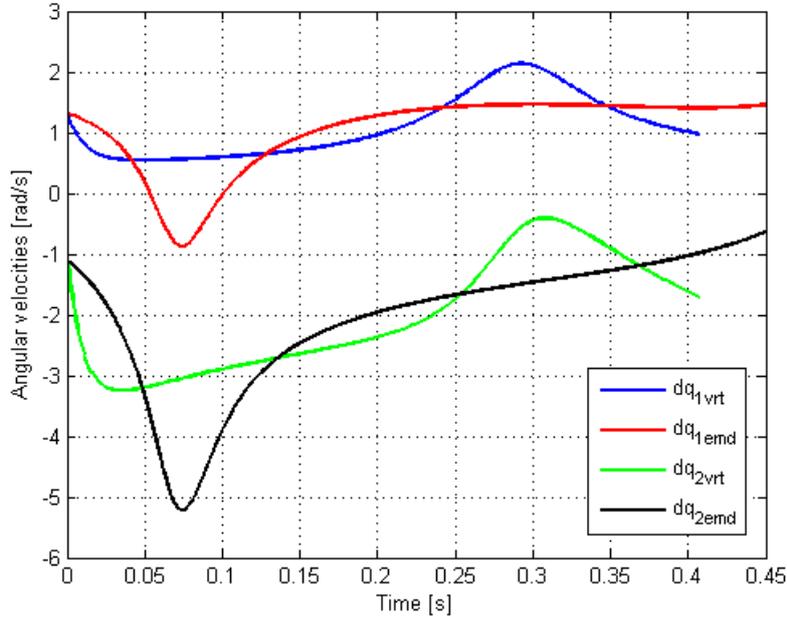


Figure 4.29: Comparison of dq_1 and dq_2 for embedded and three virtual constraints methods.

For the velocity dq_4 even when the comparison of plots in figure 4.30 seem to be different they are alike in the sense that both accelerate at the beginning of the step, then go down to negative values and then decelerate again. Similar to other velocities, the approach of embedding Acrobot first goes to very high value to then have a smoother curve, while the velocities of the three virtual constraints seems to have the bigger effort at the end of the step. The difference is due to the control approach. The velocity dq_3 is overlapped with dq_4 of its corresponding method. In figure 4.31 it is possible to see a comparison of the steps for both methods.

Two virtual constraints method is shown in red and three virtual constraints in black. It is possible to see that the high acceleration of the three virtual constraints method makes not so much difference at the beginning of the step, but then it decelerates while the two virtual constraints method keeps increasing velocity, there is where the two virtual constraints method goes ahead in the step, shown in red. But at the end the bigger effort is done by the three virtual constraints method that, as seen from figure 4.30, has a second peak in velocity (negative) much higher than the smooth transition of the two virtual constraints method.

It is possible to conclude from the comparison of the methods that both work fine. The transition of velocities is smoother in the embedded or two virtual constraints method. Nevertheless, constraining functions of low degree are easy to obtain for the three virtual constraints approach.

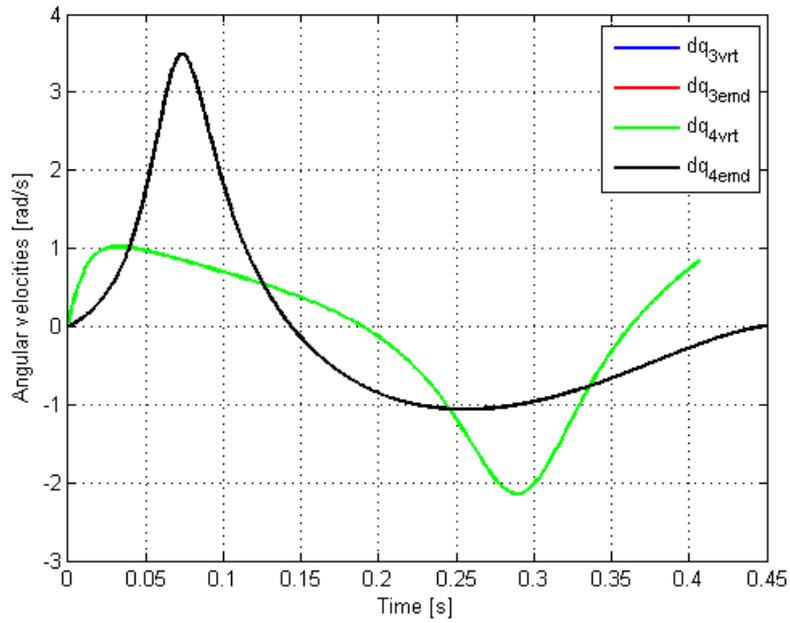


Figure 4.30: Comparison of dq_1 and dq_2 for embedded and virtual constrain.

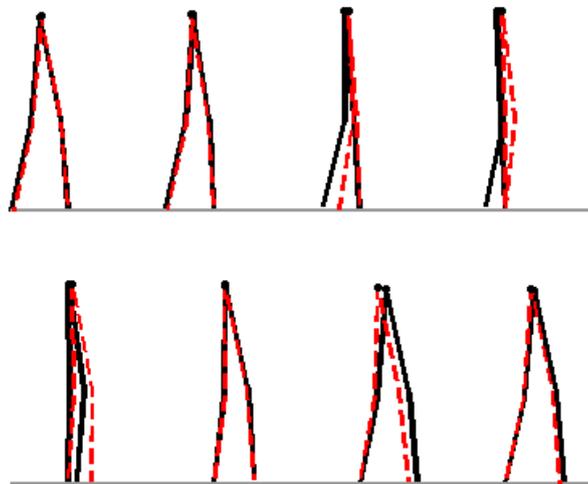


Figure 4.31: Comparison of step for embedded (red) and virtual constraints methods (black).

Chapter 5

Conclusions

In the present work a detailed explanation of the modeling of one of the most studied underactuated mechanical systems was done. This system named 4-link consists of 4 rods with articulated joints between them. The modeling was also done for a real 4-link model placed in a laboratory at Czech Technical University. Underactuated mechanical systems are of interest because new concepts for non-linear control can be studied. In this work the concept of virtual constraints was used and compared in different ways of applying it. The main reason to study the 4-link as underactuated mechanical systems is not only because of the underactuation but because it is one of the most simple systems that can represent a walking robot. These types of robots are of interest because of the multiple applications they have. Developing walking robots that are underactuated represent an improvement over the fully actuated walking robots since they are more efficient in terms of energy and more agile. Walking is more natural and human like. Nevertheless, the concept of virtual constraints can be applied to other underactuated mechanical systems not only to walking robots. That is why is even more worth studying.

The ongoing research is focused on work with the real 4-link model. It would be of interest to test the model and any other control approach developed on a real robot.

Bibliography

- [1] M. Anderle and S. Čelikovský. Analytical design of the acrobot exponential tracking with application to its walking. In *Proceedings of the IEEE International Conference on Control and Automation, ICCA 2009*, pages 163–168, 2009.
- [2] M. Anderle and S. Čelikovský. Position feedback tracking of the acrobot walking-like trajectory based on the reduced velocity observer. In *Preprints of the 8th IFAC Symposium on Nonlinear Control Systems*, pages 1011–1016, Bologna, Italy, 2010. IFAC.
- [3] M. Anderle and S. Čelikovský. Feedback design for the acrobot walking-like trajectory tracking and computational test of its exponential stability. In *Proceedings of the IEEE Multi-Conference on Systems and Control*, Denver, Colorado, 2011.
- [4] M. Anderle, S. Celikovsky, D. Henrion, and J. Zikmund. LMI based design for the Acrobot walking. In *Preprints of the 9th International Symposium on Robot Control(SYROCO'09)*, Gifu, Japan, 2009. IFAC.
- [5] M. Anderle, S. Celikovsky, D. Henrion, and J. Zikmund. Advanced lmi based analysis and design for acrobot walking. *International Journal of Control*, 83(8):1641–1652, 2010.
- [6] Ch. Chevallereau, G. Bessonnet, G. Abba and Y. Aoustin. Bipedal Robots: Modeling, Design and Walking Synthesis. *Wiley-ISTE*, 2009.
- [7] C Chevallereau, Gabriel Abba, Yannick Aoustin, Franck Plestan, E.R. Westervelt, Carlos Cabudas-de-Wit, and J.W. Grizzle. RABBIT: A Testbed for Advanced Control Theory, Designing building, and controlling an experimental platform for the study of walking. 2003. In *IEEE Control Systems Magazine*, pages 57–79, 2003.
- [8] S. Čelikovský, J. Zikmund, and C. Moog. Partial exact linearization design for the acrobot walking. In *Preprints of the American Control Conference, 2008*, pages 874–879, Seattle, USA, 2008.
- [9] S. Čelikovský, M. Anderle, and C. Moog. Embedding the acrobot into a general underactuated n-link with application to novel walking design. 2012. Submitted to *the 51st IEEE Conference on Decision and Control, 2012*, Maui, Hawaii, USA.

- [10] J.W. Grizzle, Ch. Chevallereau, A.D. Ames, R.W. Sinnet. 3D Bipedal Robotic Walking: Models, Feedback Control, and Open Problems. in *Preprints of the 8th IFAC Symposium on Nonlinear Control Systems, NOLCOS 2010*, pages 505–532, Bologna, Italy, 2010.
- [11] Tuomas Haarnoja. Dynamic Modeling and Velocity Control for Limit Cycle Walking. Luleå University of Technology. 2010.
- [12] Marek Peca, Michal Sojka, Zdeněk Hanzálek. SPEJBL-The biped walking robot. 7th IFAC International Conference on Fieldbuses and Networks in Industrial and Embedded Systems, 2007.
- [13] Mark W. Spong. Underactuated Mechanical Systems. Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL.
- [14] Mark Spong and Masayuki Fujita. Control in Robotics. The Impact of Control, T.Samad and A.M. Annaswamy (eds.) IEEE Control Systems Society, 2011.
- [15] Mark W. Spong, Seth Hutchinson, and M Vidyasagar. Robot Modelling and Control. *JHON WILEY & SONS, INC.*, 2006.
- [16] E. Westervelt, J. Grizzle, Ch. Chevallereau, J. Choi and B. Morris. Feedback control of dynamic bipedal robot locomotion. *CRC Press*, 2007.