Mechatronics by Analogy and Application to Legged Locomotion

by

Victor Ragusila

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Aerospace Science and Engineering
University of Toronto

© Copyright by Victor Ragusila 2016

Mechatronics by Analogy and Application to Legged Locomotion

Victor Ragusila

Doctor of Philosophy

Graduate Department of Aerospace Science and Engineering
University of Toronto

2016

Abstract

A new design methodology for mechatronic systems, dubbed as Mechatronics by Analogy (MbA), is introduced and applied to designing a leg mechanism. The new methodology argues that by establishing a similarity relation between a complex system and a number of simpler models it is possible to design the former using the analysis and synthesis means developed for the latter. The methodology provides a framework for concurrent engineering of complex systems while maintaining the transparency of the system behaviour through making formal analogies between the system and those with more tractable dynamics. The application of the MbA methodology to the design of a monopod robot leg, called the Linkage Leg, is also studied. A series of simulations show that the dynamic behaviour of the Linkage Leg is similar to that of a combination of a double pendulum and a spring-loaded inverted pendulum, based on which the system kinematic, dynamic, and control parameters can be designed concurrently.

The first stage of Mechatronics by Analogy is a method of extracting significant features of system dynamics through simpler models. The goal is to determine a set of simpler mechanisms with similar dynamic behaviour to that of the original system in various phases of its motion. A modular bond-graph representation of the system is determined, and subsequently simplified using two simplification algorithms. The first algorithm determines the relevant dynamic elements of the system for each phase of motion, and the second algorithm finds the simple mechanism described by the remaining dynamic elements.

In addition to greatly simplifying the controller for the system, using simpler mechanisms with similar behaviour provides a greater insight into the dynamics of the system. This is seen in the second stage of the new methodology, which concurrently optimizes the simpler mechanisms together with a control system based on their dynamics.

Once the optimal configuration of the simpler system is determined, the original mechanism is optimized such that its dynamic behaviour is analogous. It is shown that, if this analogy is achieved, the control system designed based on the simpler mechanisms can be directly implemented to the more complex system, and their dynamic behaviours are close enough for the system performance to be effectively the same.

Finally it is shown that, for the employed objective of fast legged locomotion, the proposed methodology achieves a better design than Reduction-by-Feedback, a competing methodology that uses control layers to simplify the dynamics of the system.

Acknowledgements

First and foremost I would like to thank my supervisor, Prof. M.R. Emami. His support in clarifying ideas, understanding how to approach them in a scientific and organized way and how to communicate them effectively was priceless. He was always supportive, pushing me and always providing very valuable feedback.

I also want to thank my Doctoral Examination Committee members, Prof. Gabriele D'Eleuterio and Prof. Tim Barfoot. Their advice was treasured in making this thesis a reality.

The Space Mechatronics group provided both emotional support and clear, concise and tough feedback, especially when rehearsing my unorganized presentations. Thank you for all your friendship and help, in the past seven years and in many more to come.

My parents stood by my side all the way here, and this would have not been possible without them, for which I will be always grateful.

Lastly, to Mimi, for putting up with me, and being amazing.

Contents

Abstract	ii
Acknowledgements	iv
List of Tables	vii
List of Figures	viii
1 Introduction	1
1.1 Motivation	1
1.2 Alternative mechatronics methodologies	1
1.3 Choosing legged robots as a test case for mechatronics methodologies	3
1.4 Contributions	3
1.5 Alternative approaches to the problem of legged robots	5
1.5.1 Static Stability	5
1.5.2 Zero Moment Point (ZMP)	5
1.5.3 Limit Cycle Walking (LCW)	6
1.5.4 Capture Point	7
1.5.5 Spring Loaded Inverted Pendulum (SLIP)	7
1.5.6 Swing leg control	8
2 Linkage Leg case study	10
2.1 Linkage Leg mechanism	10
2.2 Simulating the Linkage Leg	12
2.3 Verification of Linkage Leg simulation	16
3 Mechatronics by Analogy and application to legged robotics	19
3.1 Mechatronics by Analogy	19
3.1.1 Bond Graph Modeling	21
3.1.2 Analogous System Concurrent Optimization	21
3.1.3 Dimensional Analysis	23
3.2 Bond graph modeling stage applied to Linkage Leg	24
3.2.1 Bond graph model of the leg mechanism	24
3.2.2 Bond graph model simplification	29
3.2.3 Comparison between Linkage Leg and the simple mechanisms	37
3.2.4 Bond graph stage conclusions	38
3.3 Analogous System concurrent optimization	39

3.3.1	Robust trajectory control.....	48
3.4	Dimensional analysis stage	52
3.5	MbA Linkage Leg conclusions	57
4	Comparison between Mechatronics-by-Analogy and Reduction-by-Feedback	59
4.1	General description of Reduction-by-Feedback methodology.....	59
4.2	Reduction-by-Feedback applied to the Linkage Leg	60
4.3	Comparison between MbA Linkage Leg and RbF Linkage Leg	66
4.3.1	Disturbance rejection comparison between MbA and RbF Linkage Legs	67
5	Conclusion and further directions.....	71
	References	74
	Appendix: Dimensional Analysis	82

List of Tables

Table 3.1 Variables for the test cases studied for the bond graph simplification stage	27
Table 3.2 Geometric parameters for the linkage leg used in bond graph simulation	27
Table 3.3 Initial conditions for stance and swing phase bond graph simulations.....	27
Table 3.4 Results of the dynamic element simplification (DES) step.	32
Table 3.5 Analogous System Optimization variables.....	44
Table 3.6 Analogous System Optimization constraints.....	45
Table 4.1 Reduction-by-Feedback optimization variables	69
Table A.1 Variables for dimensional analysis optimization.....	82

List of Figures

Figure 1.1 Foot Place Estimator visualization	7
Figure 1.2 SLIP model illustration.....	8
Figure 1.3 Passive knee retraction using hip torque	9
Figure 1.4 Four bar linkage leg using passive knee retraction	9
Figure 2.1 Linkage Leg mechanism.....	10
Figure 2.2 Linkage Leg gait.....	12
Figure 2.3 Linkage Leg hybrid system, composed of two continuous motion phases and two transition events	13
Figure 2.4 Graphical representation of the coupling between the hip and toe velocities of the Linkage Leg.	16
Figure 2.6 Position and angle of the robot body comparison between the Linkage Leg model and Simulink model. The discontinuous model in Figure 2.3 is able to approximate the behaviour of the Linkage Leg.	18
Figure 2.5. Hip and knee angle comparison between the Linkage Leg model and the Simulink model. The transition functions in Eq. 2.5 are able to closely model the touchdown and takeoff events.	17
Figure 3.1 Mechatronics by Analogy methodology	19
Figure 3.2 Diagram of the Bond Graph Modelling phase	24
Figure 3.3 Bond graph representation of a rigid body dynamics.....	25
Figure 3.4 Bond graph representation of Linkage Leg, consisting of five solid body models from Fig 3.3	26
Figure 3.5 Trajectories of the knee and hip joint of the bond graph model.....	28
Figure 3.6 Swing phase joint angle results for DES step, Case 3. It can be seen that the last two simplification levels, 5 and 6, differ greatly from the original behaviour. As such, simplification level 4 is the one used in the IBS step.	33
Figure 3.7 Stance phase joint angle results for DES step, Case 3. It can be seen that all simplification levels have similar behaviour, so all the dynamic elements of the leg can be eliminated and the behaviour will be close to that of the original leg.	34
Figure 3.8 Swing phase, simplification level four	35
Figure 3.9 Swing phase, simplification level five.....	35

Figure 3.10 Stance phase, simplification level six.....	36
Figure 3.11 Comparison between Linkage Leg and double pendulum	37
Figure 3.12 Comparison between Linkage Leg and SLIP model	37
Figure 3.13 Diagram of the Analogous System Concurrent Optimization phase.....	39
Figure 3.14 Analogous System	40
Figure 3.15 Three steps of the optimal AS system	46
Figure 3.16 Comparison of the behaviours of the initial AS system (dotted line) and the optimized AS system (solid line) for one step.	47
Figure 3.17 AS two-layer control strategy.....	49
Figure 3.18 Diagram of Dimensional Analysis stage	52
Figure 3.19 Parallel Linkage Leg (left) and SLIP-like Linkage Leg (right).....	55
Figure 3.20 Gait of the optimally analogous Linkage Leg	56
Figure 3.21 Comparison between the behaviours of the optimal AS system and the ES (LL) optimally analogous to it.....	56
Figure 4.1 Reduction-by-Feedback Linkage Leg	63
Figure 4.2 Reduction-by-Feedback Linkage Leg gait	64
Figure 4.3 Reduction-by-Feedback Linkage Leg hip trajectory and joint torques	65
Figure 4.4 Reduction-by-Feedback Linkage Leg virtual joint constraints	65
Figure 4.5 MbA Linkage Leg compared with RbF Linkage Leg	66
Figure 4.6 MbA Linkage Leg compared with RbF Linkage Leg for a 0.1m step disturbance.....	68

1 Introduction

1.1 Motivation

The notion of mechatronics has evolved to a systematic design paradigm for creating synergy and providing catalytic impacts on discovering simpler solutions to traditionally complex problems. The physical artifacts of such a design philosophy, often referred to as mechatronic systems, demonstrate a seamless integration of mechanical, electrical and software constituents, in a sense that their characteristics are all specified concurrently during the design phase. Concurrent engineering of such multidisciplinary systems, however, is no trivial task, for it most likely results in a large number of objective and constraint functions that must be taken into account simultaneously with a great number of design variables. Should one follow a formal optimization approach, the multi-objective constrained optimization problem with large number of functions and variables can be quite challenging.

The motivation behind this work is to find an alternative solution to the problem of complex, multi-disciplinary systems, which results in a more intuitive and transparent system behaviour. The main goals are to find understand the system dynamics in a simple and intuitive way, which is useful for control design, and to be able to construct a system with desirable behaviour for a variety of situations. This thesis introduces a new concurrent design methodology, called Mechatronics by Analogy, that addresses the problem of system complexity in design by i) finding simpler mechanisms analogous to the original complex system, ii) optimizing such simpler mechanisms with the controller concurrently, and iii) designing the parameters of the original system such that it behaves similarly to the optimized simpler mechanisms with the same controller. The methodology offers qualitative and quantitative advantages over alternative methods. The qualitative advantage is that the simpler systems used for control design are real-life mechanisms, which capture non-linear effects while can be intuitively understood and studied, and additionally enjoy effective controllers already developed for them. The quantitative advantage is that the simplification of the system behaviour occurs at the design level, not at the control level. As such, a higher degree of synergy can be achieved between the different subsystems.

1.2 Alternative mechatronics methodologies

The demand for higher performance at lower cost has led to methodologies that consider concurrently the mechanical, electrical and control systems, and attempt to find synergies between

these subsystems, which has been demonstrated to generate better and previously unattainable performance [1]. The challenge is to consider the large number of variables and various objectives which belong to a number of different disciplines. A number of approaches exist to solve this problem, as will be discussed below.

One possibility is to develop better multidisciplinary optimization (MDO) algorithms that are able to deal with a large design space. Some involve evolutionary algorithms used for parallel robot arms [2] and genetic algorithms used for designing reconfigurable robots [3] [4], parallel manipulators [5] and mechatronic systems [6]. The design space can be simplified by employing a staged optimization procedure [7] or by lowering the dimensional space of the system [8] using approximations.

A number of concurrent engineering approaches have been suggested for mechatronic systems, which address the challenges of multi-objective constrained optimization problems while also considering subjective aspects of the design. For instance, a concurrent system evaluation model is proposed in [9], based on three indices (defined as intelligence, flexibility, and complexity [9]), and it is formulated using t-norm and mean operators. Another evaluation model is suggested in [10] based on Mechatronics Design Quotient (MDQ), where a nonlinear fuzzy integral is used for the aggregation of design criteria. An alternative concurrent design methodology is presented in [11], introducing the notion of Holistic Concurrent Design (HCD), where the subjective aspects of multidisciplinary systems design are captured through fuzzy operators, along with the quantitative system performances using a holistic system modeling based on bond graphs. Another notable effort is the Design for Control (DFC) methodology [12], which prescribes that control parameters be designed concurrently with the structure parameters, so that by simplifying the underlying system dynamics simple controllers can be employed successfully. In addition, a number of *ad hoc* techniques for mechatronics have been reported in the literature, which are innovative design schemes for specific applications such as robotics (e.g., see [13]). Method of Imprecision (MoI) [14] is a notable preliminary design methodology is able to consider imprecision in design and is used to define a set of design preferences regarding the design space optimization tradeoffs [15].

Methods exist to simplify the closed loop dynamics of the robots using a control layer. The method presented in [16] requires a fully actuated leg achieve forces upon the body similar to those generated by any desirable mechanism. The forces in the joints of the robot are obtained using the

reverse Jacobian and the forces generated by the virtual model. This also requires actuators able to achieve force or torque control of the joints, which is a difficult to achieve from a mechanical construction point of view [17].

Another method is presented that achieves desirable closed loop behaviour using a control layer [18]. There are examples shown for achieving similarity to simpler mechanism for a simple test case, but not for the full robot. This approach also requires fully actuated robotic legs, but it does not require the complex force feedback capable joints.

The main issue with these methods is that they require fully actuated robot legs and power to achieve the required close-loop dynamics because they attempt to cancel the natural dynamics of the leg instead of utilizing them. Moreover, the range of mechanisms they are able to emulate is restricted [18], and require extra actuator power and, in the case of [16], actuators able to achieve force feedback control.

1.3 Choosing legged robots as a test case for mechatronics methodologies

The Mechatronics by Analogy methodology is driven by the needs in the legged robots research community. Legged robots offer an interesting challenge for mechatronic methodologies. In order to simplify the control of robotic legs and make the behaviour more tractable and intuitive, the physical model of a leg mechanism must be simple. Simple models, such as the inverted and double pendulums are used to approximate the legs of walking robots [19], and the Spring Linear Inverted Pendulum (SLIP) model is used to approximate the behaviour of running and hopping robots [20]. The behaviour of such models, as well as the controllers developed for them, have been extensively studied both in the field of robotics and biological walkers and hoppers [21]. However, the control design can become far too complex, thus challenging, if the mechanical design of the legs goes beyond such simple models [22], and consequently it will be difficult to take advantage of the wealth of information already available for the simpler models.

1.4 Contributions

The thesis has a number of important contributions. First, a new mechatronics methodology, dubbed as Mechatronics by Analogy (MbA), is proposed and developed. This methodology is first introduced in [23]. The MbA methodology offers a number of advantages compared with the existing methodologies:

- Quantitative advantages:

- The methodology does not require additional control layers, making the control simpler and faster than alternative, control intensive methods, as the one in Chapter 4.
- No need for complex force-feedback capable actuators.
- Better performance can be achieved by exploiting synergies at the design phase.
- Qualitative advantages
 - Control is based on well-understood, real-life mechanisms.
 - The simplified systems capture the non-linear effects of the original system.
 - Simplification occurs at the design level, not at the control level. This results in a more intuitive understanding of the system.

The MbA methodology is detailed through the design of a legged robot. A number of important contributions are made:

- A simplification procedure based on bond graphs is developed, and it is shown to successfully simplify a robot leg for different motion phases [24].
- The simpler system is optimized using a concurrent optimization approach that avoids complex issues, such as impulses due to contact between the robot and the ground [25].
- A leg is developed using a dimensional analysis method, which achieves similar dynamics to those of a desired set of simpler systems [26].

The final contribution of the presented work is a comparison between the MbA methodology and a competing proven methodology. This comparison is made by designing a legged robot using both methodologies, with the aim of achieving fast legged locomotion. The MbA designed robot is able to achieve a higher top speed using less torque, while the competing robot is able to deal with disturbances slightly better.

Publications:

- V. Ragusila and M. R. Emami, "A mechatronics approach to legged locomotion," in *Advanced Intelligent Mechatronics*, Montreal, 2010.
- V. Ragusila and M. R. Emami, "A novel robotic leg design with hybrid dynamics," *Advanced Robotics*, vol. 27, no. 12, pp. 919-931, 2013.
- V. Ragusila and M. R. Emami, "Modelling of a robotic leg using bond graphs," *Simulation Modelling Practice and Theory*, vol. 40, pp. 132-143, 2014.

- V. Ragusila and M. R. Emami, "Mechatronics by Analogy and Application to Legged Locomotion," *Mechatronics*, accepted, July 2015.

V. Ragusila and M. R. Emami, "Reduction by Design vs. Reduction by Feedback: A Benchmark Study for Legged Robotics," *Mechatronics*, submitted September 2015.

1.5 Alternative approaches to the problem of legged robots

To understand where the current methodology fits comparatively with existing methods that address the problem of walking robots a review of existing approaches is presented below. Although the field of legged robots is vast and a lot of robots are being used in the industry for traveling inside pipes, window washing or heavy transport, this review will deal mainly with monopod and biped robots used in academia to study the basic features of legged locomotion. For a review of other legged robots, [27] is a very good reference.

The field of legged robots is new, with the first experiments being conducted at Waseda University in the 60s, there are many approaches defining how legged locomotion can be achieved. These paradigms represent the basic idea behind the design of the robot [28] and some have been defined and given a name after robots have been built based on them. There are two major types of bipedal or monopod robots, planar and 3D. Planar robots are restricted to movement in the sagittal plane and cannot move laterally, roll or yaw [20]. They are used mainly to test new theories in low cost and simple robots, while 3D robots do not have such restrictions.

1.5.1 Static Stability

First used in 60s, it was the first approach which allowed bipedal robots to walk. The static stability approach states that the Center of Mass (COM) must be over the support polygon at all times, and must move slowly enough such that dynamic effects are not important [28]. Most of the control is open loop trajectory tracking, using PD controllers for each individual joint. The trajectory is programmed in such that it keeps the COM over the support polygon. This approach is rarely used in active research, since Zero Moment Point (ZMP) has largely replaced it, but it is still used in hobby bipeds and in soccer playing bipeds [29], which are very cheap and simple to build, allowing amateur builders to design their own bipeds.

1.5.2 Zero Moment Point (ZMP)

The Zero-Moment-Point (ZMP) locomotion paradigm is one of the most popular, being used in such robots as the ASIMO [30], QRIO [31], HRP series of humanoids [32] and many others

[27] [33]. The ZMP is defined as the center of pressure of the foot. If this point is not on the edge of the support polygon, defined by the legs in touch with the ground, the robot can apply a moment on the ground and its trajectory can be controlled using classical robotic manipulator approaches [33]. This means the robot must have a foot flat on the ground at all times, increasing the complexity of trajectory calculation and severely limiting its stride length, especially preventing running gaits with aerial phases especially preventing running gaits with aerial phases without significant changes to the control algorithm [34].

1.5.3 Limit Cycle Walking (LCW)

Limit Cycle Walking was first introduced in [28] as a sequence of steps that is locally unstable but stable over a number of steps. However robots based on the LCW approach has been developed in some form since the first hopping robots developed by Raibert [35] and the passive dynamic walkers made by McGeer [36]. As the definition is quite wide, there are quite a number of robots designed and built which fall in this definition.

The first robots that achieved LCW are the passive dynamic walkers, which consist of a series of pendulums and can walk down a slope. There is no control and no actuation for these robots, but they have been proven to be stable, and to be able to deal with small disturbances [28] [36]. State of the art passive dynamic robots have an upper body and knees and are no longer planar [36] [37].

Recently, actuation has been added to passive dynamic robots keeping the control system if very simple. The Cornell Walker creates a small impulse to the ankle of the stance leg when the swing leg hits the ground [38] and Denise has a small moment applied to the hip joint when the next leg hits the ground [19] [39]. These robots are tuned to walk at a predefined speed and they can deal with small disturbances, without the need of actively controlled joints or any sensors beside foot switches. These robots are very energy efficient, achieving a cost of transportation similar to humans [19].

RunBot [40] has 4 servo actuators in the hip and knee joints and a fifth one in its torso, and is the fastest walking biped for its size. Its control is based on a neural network, which adjusts the signals to the 5 actuators based on data from foot switches and joint position sensors. The robot achieves stable cyclic walking at a desired speed and can learn to walk up a slope.

A method is proposed by Hyon that achieves stable hopping motion for a monopod based on the property of Hamiltonian systems of achieving a stable orbit if energy is not changed in the

system [41]. He proposes a monopod with a Spring Loaded Inverted Pendulum (SLIP) leg with impedance controllable hip and leg joints and which has three levels of control which can adapt to any starting conditions to achieve passive-dynamic running.

1.5.4 Capture Point

The Capture Point approach allows a robot to regain stability after being subjected to a disturbance. This concept can be developed into a walking gait by considering each step a disturbance, and not recovering completely from each previous step.

A Capture Point (CP) is the foot placement position that allows the robot to come at a stop without any more control being applied. Capture Point is used in [42], whereas a similar concept called Foot Place Estimator (FPE) is introduced in [43], and it is shown in Figure 1.1.

Figure 1.1 Foot Place Estimator visualization [43]

The stability margin for the CP approach is the number of steps it takes to reach a Capture Point. A three step margin means that the capture point is 3 steps away, and the robot cannot stop in less than three steps [42]. If the legs of the robot are not fast enough, a biped might fall down by tripping its swing foot.

1.5.5 Spring Loaded Inverted Pendulum (SLIP)

The SLIP model is a very popular model for describing dynamic locomotion, which might exhibit running gaits with aerial phases [20]. It consists of a point mass (the body of the robot) connected to the ground using a linear spring which is aligned with the leg axis. The SLIP model

has been used extensively in biology to simulate a running gait [44], and some argue that a good definition for running is of a motion which uses the SLIP model to store and release energy [45]. Raibert has used the SLIP first in robotics and a series of highly successful planar and 3D monopods, bipeds and quadrupeds have been created at the MIT Leg Lab using this model [20] [46].

The classic SLIP model is telescopic [47], but segmented legs with a knee or a knee and an ankle seem to give better stability and allow running at lower speeds [48] [49]. Construction is easier for a segmented leg, since no heavy prismatic joints need to be used [22]. Other legs are constructed using flexible bow-like structures [50] [51] or are approximated using linkages and springs [52].

Figure 1.2 SLIP model illustration [47]

Most SLIP control strategies are focused on the touchdown moment, when the leg touches the ground. The angle and angular velocity at which the leg hits the ground have been shown to be critical for the leg gait [41] [53]. Other methods have been able to use tuned springs in the hip joint and knee joint to obtain fully or partially passive running gaits. [54] [55]. A good overview of variations of the SLIP model and control strategies can be found in [56].

1.5.6 Swing leg control

The SLIP is a very popular model for the stance phase of a legged robot, and a lot of the current research is aimed at controlling and improving its behaviour. However, the dynamics of the leg during the swing phase are also very important, both for single legged and multi-legged robots. Most robotic legs are modelled as simple pendulums with rotation axis at the hip motion [39]. The period of these simple pendulum swing legs determines the period of stable gait for most

Figure 1.3 Passive knee retraction using hip torque [54]

Limit Cycle Walking robots [36]. Adding a spring around the hip joint can increase the range of swing speeds for these legs and also increase the robustness of the robot gait [57]. The behaviour of an actuated hip with passive knee is described in [58]. The hip describes a predetermined trajectory motion to bring the swing leg ahead of the stance leg in the required time, and the knee is left passive. Once the knee straightens, usually just before ground contact, it is mechanically locked into place to prevent bouncing or vibrations. This leads to both energy efficiency and better robustness against disturbances. A four bar linkage leg design with passive retraction, is described in [22], but it requires the joints to have zero impedance, which was impossible with the mechanical design of that robot.

Horse legs are theorized to have a passive catapult using elastic elements in the leg that store energy and use it to rapidly contract the swing leg [59]. The catapult system allows the horse to use 100 times more power in contracting and swinging the front legs than is available in the actual muscle, and thus lower the inertia of the swing leg.

Figure 1.4 Four bar linkage leg using passive knee retraction [22]

2 Linkage Leg case study

2.1 Linkage Leg mechanism

The Linkage Leg is a novel robotic leg design first proposed by the author in [26]. The goal of the design is to provide a structurally efficient, simple to control leg with tractable behaviour for both the swing and stance phases. To that end, the MbA methodology is used in the design and construction of the proposed leg. The proposed leg design, is composed of four links. The first link is the thigh, $\{O_0O_1O_4\}$ in Figure 2.1, which is connected to the body of the robot using the hip joint O_0 . The tibia $\{O_1O_2\}$, foot $\{O_2O_3O_5\}$ and tendon $\{O_4O_5\}$, together with the thigh, form a four-bar linkage. The lengths $a_1 \dots a_6$ together with the angles ϕ_1 and ϕ_3 determine the geometry of the leg.

Figure 2.1 Linkage Leg mechanism

The Linkage Leg has two degrees of freedom. For the swing phase, they are the angle between

coordinate frames $\{O_0\}$ and $\{O_1\}$, defined as the hip angle θ_1 and the angle between the coordinate frames $\{O_1\}$ and $\{O_2\}$, defined as the knee angle θ_2 , as shown in Figure 2.1. For the stance phase, they are the angle between coordinate frames $\{O_3\}$ and $\{O_2\}$ (θ_6), and the angle between the coordinate frames $\{O_2\}$ and $\{O_1\}$ (θ_7), as shown in Figure 3. Note that the Denavit-Hartenberg [60] notation is used for assigning the link coordinate frames of the linkage leg in both stance and swing phases.

As shown in Figure 3.1, a chain (1), starting from point A and ending at point O_5 , is wrapped around cogs at the joints O_1 and O_4 . Point A is chosen such that the length of the chain remains constant as the knee angle θ_2 changes. A motor is attached to the cog at joint O_4 and controls the length of the leg during the swing phase by actuating the chain. During the stance phase the cog at joint O_4 locks the chain preventing it from rotating around the cogs and forcing the elastic element (2) to extend and store energy.

The leg has two actuators. The hip actuator at joint O_0 controls the angle of leg relative to the body of the robot. The knee actuator at joint O_1 controls the length of the leg during swing, and it is also used in parallel to the elastic element (2) to control the height of each hop.

The Linkage Leg mechanism has a number of advantages over similar robotic legs. The leg employs only revolute joints, making it simpler to build and potentially lighter than legs with prismatic joints [50]. The tibia and tendon segments are loaded only in compression and tension, respectively, making them easier to design and lighter to build. Another significant advantage is that the elastic element (2) needs to act at tension only, allowing the use of rubber or latex that have a higher energy storing capacity per unit mass compared to steel springs. Further, the proposed Linkage Leg can change its knee angle at touchdown easily by changing the timing for locking the string with no need for mechanical adjustments.

The most significant advantage of the Linkage Leg is that its dynamics can be tuned to a SLIP model for the stance phase and to a specific double pendulum for the swing phase, using the approach that will be discussed in the sequel. This allows for a simple and intuitive method of understanding and defining the dynamics of the leg, as well as the use of existing control strategies for the SLIP and double pendulum [61] [20].

The complexity of achieving a desirable hopping gait is, therefore, shifted from the online control to the offline mechanical design of the leg. Although the mechanical design of the leg

Figure 2.2 Linkage Leg gait

requires more effort in assigning the physical parameters so that a desired dynamics is achieved, the online control will become simpler. This approach, which can be characterized as reduction-by-design [62], is in contrast with reduction-by-feedback approaches [61] [18], where the complexity lies in the online control. Unlike reduction-by-feedback approaches where the goal is to design a (complex) controller that makes the overall closed-loop system achieve a simpler dynamics, the proposed approach aims at designing a leg mechanism that achieves a simpler dynamics by optimizing its physical parameters.

2.2 Simulating the Linkage Leg

The Linkage Leg robot is designed to achieve a repeatable hopping gait, as shown in Figure 2.2. This gait starts at takeoff, when the leg breaks contact with the ground. The swing phase starts at takeoff and lasts until touchdown, when the leg contacts the ground. The stance phase lasts while the leg is in contact with the ground, after which the gait repeats itself. The dynamic equations of the Linkage Leg are show in Eq. (2.1)-(2.4):

$$\dot{x}_{sw} = f_{sw}(x_{sw}, \tau_{1,sw}, \tau_{2,sw}) \quad (2.1)$$

$$\dot{x}_{st} = f_{st}(x_{st}, \tau_{1,st}, f_{L,st}) \quad (2.2)$$

$$x_{sw} = [\dot{q}_{sw} \quad q_{sw}], \quad q_{sw} = [x_0 \quad y_0 \quad \theta_B \quad \theta_1 \quad \theta_2] \quad (2.3)$$

$$x_{st} = [\dot{q}_{st} \quad q_{st}], \quad q_{sw} = [\theta_L \quad L_{st} \quad \theta_H] \quad (2.4)$$

Figure 2.3 Linkage Leg hybrid system, composed of two continuous motion phases and two transition events

The gait is simulated using two functions: $\dot{x}_{sw} = f_{sw}(x_{sw}, \tau_{1,sw}, \tau_{2,sw})$ to simulate the swing phase and $\dot{x}_{st} = f_{st}(x_{st}, \tau_{1,st}, f_L)$ to simulate the stance phase, as shown in Figure 2.3. The inputs to these functions are the states of the system during swing (x_{sw}) and stance (x_{st}), together with the torques of the controlled degrees of freedom, τ_1 , τ_2 . The outputs of the two phase functions are the derivatives of the state space vectors.

The swing phase state vector, x_{st} is show in Eq. 2.3, where x_0 is the horizontal distance traveled by the hip joint from the previous takeoff location, y_0 and θ_B are the height and angle of the hip joint with respect to the ground and θ_1 and θ_2 are the controlled degrees of freedom of the Linkage Leg mechanism. The stance phase state space vector is shown in Eq. 2.4, where θ_L is the angle of the virtual leg (defined as the segment connecting the toe O_3 to the hip joint O_1) with respect to the ground, L_{st} is the length of the virtual leg and θ_H is the angle of the body with respect to the virtual leg.

The stance and swing functions are presented below in detail:

$$f_{sw} = \begin{bmatrix} M_{sw}^{-1} \begin{pmatrix} 0 \\ 0 \\ 0 \\ \tau_1 \\ \tau_2 \end{pmatrix} - C_{sw}(\dot{q}_{sw}, q_{sw})\dot{q}_{sw} - g_{sw}(q_{sw}) \\ \dot{q}_{sw} \end{bmatrix} \quad (2.5)$$

$$f_{st} = \begin{bmatrix} M_{st}^{-1} \begin{pmatrix} 0 \\ \tau_2 \\ \tau_1 \end{pmatrix} - C_{st}(\dot{q}_{st}, q_{st})\dot{q}_{st} - g_{st}(q_{st}) \\ \dot{q}_{sw} \end{bmatrix} \quad (2.6)$$

The matrices M_{sw} and M_{st} are the mass matrix of Linkage Leg and robot body mechanism, the matrix C_{sw} incorporates the Coriolis and centrifugal effects, and g_{sw} is the column vector expressing the effects of gravity. The values in the torque column vector are τ_1 , the torque in the hip joint, and τ_2 , the torque in the knee joint, which are determined by the control system as shown in Section 3.3.1.

Two transformation functions are used at the takeoff and touchdown moments to transition the system between the two motion phases, and thus represent the effects of impulses without requiring a detailed knowledge of the ground properties or the elasticity of the system. The initial state of the stance phase, $x_{st,i}$, is defined as a function of the final state of the swing phase, namely $x_{st,i} = \Delta_{sw \rightarrow st}(x_{sw,f})$. Similarly, the initial state of the swing phase, $x_{sw,i}$, is defined as a function of the final state of the stance phase, i.e., $x_{sw,i} = \Delta_{st \rightarrow sw}(x_{st,f})$ in Eq. (2.7).

$$\begin{aligned}
\Delta_{sw \rightarrow st} &= \begin{bmatrix} \tan^{-1}(y_{0,sw,f}/(x_{0,sw,f} - x_{toe,sw,f})) \\ \sqrt{(x_{0,sw,f} - x_{toe,sw,f})^2 + y_{0,sw,f}^2} \\ \pi - \phi_{offset} - \theta_{1,sw,f} \\ ((x_{0,sw,f} - x_{toe,sw,f})\dot{y}_{0,sw,f} - y_{0,sw,f}\dot{x}_{0,sw,f})/\sqrt{(x_{0,sw,f} - x_{toe,sw,f})^2 + y_{0,sw,f}^2} \\ ((x_{0,sw,f} - x_{toe,sw,f})\dot{x}_{0,sw,f} + y_{0,sw,f}\dot{y}_{0,sw,f})/\sqrt{(x_{0,sw,f} - x_{toe,sw,f})^2 + y_{0,sw,f}^2} \\ -\dot{\theta}_{1,sw,f} \end{bmatrix} \\
\Delta_{st \rightarrow sw} &= \begin{bmatrix} L_{st,f} \cos \theta_{L,st,f} \\ L_{st,f} \sin \theta_{L,st,f} \\ \theta_{L,st,f} + \theta_{H,st,f} \\ \pi - \phi_{offset} - \theta_{H,st,f} \\ (L_{st,f} - L_{st,i})C \\ \dot{L}_{st,f} \cos \theta_{L,st,f} - L_{st,f} \sin \theta_{L,st,f} \dot{\theta}_{L,st,f} \\ \dot{L}_{st,f} \sin \theta_{L,st,f} + L_{st,f} \cos \theta_{L,st,f} \dot{\theta}_{L,st,f} \\ \dot{\theta}_{L,st,f} + \dot{\theta}_{H,st,f} \\ -\dot{\theta}_{H,st,f} \\ \dot{L}_{st,f}C \end{bmatrix} \quad (2.7)
\end{aligned}$$

The parameter C in (7) represents the coupling between the rate of change of virtual leg length, $\dot{L}$, and the knee joint, $\dot{\theta}_2$, calculated from the geometry of the Linkage Leg.

The parameter C is calculated using the point G , defined as the intersection of segments O_1O_2 and O_4O_5 and shown in Figure 2.4. The point G becomes the instantaneous center of rotation of the solid body $\{O_2O_3O_5\}$. As such, the direction of the instantaneous velocity of the toe O_3 caused by a change in the knee angle is perpendicular to the segment O_2G , and its magnitude is $\dot{\theta}_2 \frac{a_2 O_3G}{O_2G}$.

Consequently, $C = \frac{a_2 O_3G}{O_2G}$.

Figure 2.4 Graphical representation of the coupling between the hip and toe velocities of the Linkage Leg.

2.3 Verification of Linkage Leg simulation

The swing and stance phase functions, f_{sw} and f_{sw} together with the transition functions $\Delta_{st \rightarrow sw}$ and $\Delta_{sw \rightarrow st}$ are able to simulate the behaviour of the Linkage Leg system. However, the assumption is made that the body velocity does not change at touchdown, but instead the velocities of the hip and knee degrees of freedom, $\dot{\theta}_1$ and $\dot{\theta}_2$ have an instantaneous change at touchdown. Of course a real system does not experience instantaneous changes of velocities. In order to validate the assumptions about the takeoff and touchdown transitions, a Simulink model is used to simulate the Linkage Leg and investigate the touchdown and takeoff moments.

SimMechanics is a solid body simulation package for Simulink. The Linkage Leg model is shown in Figure 2.5. It consists of five solid bodies, four for the Linkage Leg segments and one for the body of the robot, linked together by revolute joints. The toe point is connected to a ground model that calculates the force the ground would apply any time the point dips below the ground level. The function used for the ground model is described in [63] and uses the same parameters obtained there.

The torque profiles τ_1 and τ_2 used in the gait shown in Figure 2.2 are applied to the hip and knee joints of the Simulink model. The results are compared below with those obtained using the

functions described in section 2.2. At touchdown (time 0.6s) the Linkage Leg model transition function $\Delta_{sw \rightarrow st}$ changes the swing state space to the stance state space, and the Simulink model experiences ground contact. As it can be seen in Figure 2.5 the Linkage Leg model is discontinuous, using the matrices in Eq. (2.7) to approximate the impulses at touchdown and takeoff, whereas the Simulink model is continuous and both the hip angle θ_1 and θ_2 exhibit vibrations due to the contact with the ground, similarly to the touchdown vibrations shown in [63] [40]. The discontinuous Linkage Leg model was able to closely approximate the more realistic continuous Simulink model for the knee and joint trajectories, as well as for the behaviour of the robot body, shown in Figure 2.6.

As such, the transition functions in Eq. 2.5 are able to closely approximate the complex touchdown and takeoff events, and the simple, discontinuous model from Figure 2.3 is a suitable approximation of the realistic Linkage Leg behaviour. The discontinuous model will be used in the rest of the thesis to simulate the behaviour of the Linkage Leg.

Figure 2.5. Hip and knee angle comparison between the Linkage Leg model and the Simulink model. The transition functions in Eq. 2.5 are able to closely model the touchdown and takeoff events.

Figure 2.6 Position and angle of the robot body comparison between the Linkage Leg model and Simulink model. The discontinuous model in Figure 2.3 is able to approximate the behaviour of the Linkage Leg.

3 Mechatronics by Analogy and application to legged robotics

3.1 Mechatronics by Analogy

Mechatronics by Analogy (MbA) addresses the challenge of concurrent design by finding simple mechanisms analogous to a more complex mechanical system, optimizing the simpler

Figure 3.1 Mechatronics by Analogy methodology

mechanisms, and then designing the control system around these simpler mechanisms. The systems need not be equivalent, and a degree of analogy is introduced, similar to the imperfect analogy used in [64]. The closer the two systems are, the better the controller performance, but a perfect equivalence is neither needed nor possible in most cases. Furthermore, MbA offers techniques to improve the similarity between the simple mechanisms and the actual mechanical implementation of the system. The three stages of the MbA methodology are shown in Figure 3.1 and detailed in the following three sections.

The process begins with a desired concept of the original system, here referred to as Emulated System (ES). A bond graph model of the system is then developed, and the model is simplified based on the universal notion of energy to find a set of simpler mechanism, called the Analogous System (AS), which relevantly emulates the ES. Next, the AS is detail-designed concurrently with the control system to find the optimal system parameters for the desired behaviour, obtaining an optimal AS. The control system may be taken from the literature or determined using other methods, as explained in sections 3.1.2 and 3.3. Finally, the parameters of the ES are derived, through the dimensional analysis stage, such that the ES system behaves similarly to the optimum AS. If this similarity is achieved, then the controller developed for the AS system can be applied without modifications to the ES. Hence, the complex task of optimizing the physical and control parameters of the ES is replaced by the simpler tasks of optimizing the parameters for the AS together with the control system, and then finding the ES parameters that achieve similarity. As it will be demonstrated in Section 4, this approach is easier to achieve and leads to better results than tackling the ES optimization directly, for the case of a robotic linkage leg design.

MbA methodology is ideally suited for systems that have multiple phases. A phase is defined as a set of conditions or constraints that specify the system operation. The Linkage Leg robot case study has two motion phases, the stance phase, when the leg is in contact with the ground, and the swing phase, when the leg is not in contact with the ground. As will be shown in sections 3.2 and 3.3, the MbA methodology develops separate simpler mechanisms and controllers for each motion phase, making the AS concurrent optimization simpler. The dimensional analysis stage is then able to find an ES mechanism that is similar to each of the simpler mechanisms of the AS, combining the optimal behaviour of multiple mechanisms into one system, as seen in section 3.4. The concept of phases can be generalized to other mechatronic systems that need to function in different

conditions or achieve different goals at different times. If the system has only one phase, the MbA methodology becomes a simple simplification method, and is less beneficial.

The MbA methodology is explained in section 3.1 for a general system, and then applied for the case study of a legged robot in sections 3.2-3.4.

3.1.1 Bond Graph Modeling

The first stage of MbA involves analyzing the ES and finding the simplest mechanism that represents the dynamic behaviour of the original ES for each of the N motion phases of the ES. This is done through the development of the bond graph model of the ES.

Bond graphs represent the dynamics of a system by simulating the power exchange between its components [65]. The system variables (force, velocity, current, voltage, etc.) are unified into two generalize variables, the *flow* and *effort*. The power flow between components can be computed by multiplying the flow and effort of each element. The dynamic behaviour of the bond graph is computed by considering the relation between flow and effort in each element. Further, a more complex system can be represented by simpler bond graphs linked together in a modular fashion [66].

The MbA's Bond Graph Modeling stage consists of a two-step simplification. The first step, Dynamic Element Simplification (DES), eliminates the maximum number of dynamic elements while maintaining the system behaviour nearly unchanged [67]. For a bond graph representing a mechanism formed by solid bodies, the dynamic elements are masses, moments of inertia, springs and dampers. For each possible combination of the system dynamic elements the bond graph is simulated, and the behaviour of a set of representative test bonds is compared. The result of DES is the smallest set of dynamic elements that can still produce a close behaviour to the original system.

The second step, Individual Bond Simplification (IBS), is applied to the set of elements remaining after the DES step. The goal is to find the *essential* bonds in the bond graph described by the remaining dynamic elements. For this purpose, a metric related to the power flow of each bond is used to eliminate the bonds with zero or negligible power flow [66]. The two steps of the Bond Graph Modeling stage will be further illustrated through the case study in Chapter 3.

3.1.2 Analogous System Concurrent Optimization

The goal of the AS Concurrent Optimization stage is to obtain the best AS mechanism and controller for each respective motion phase. Since the AS is composed of different mechanisms,

each representing a valid analogy to the ES for its corresponding motion phase, it is possible to avoid difficult control problems, such as switching between different sets of constraints due to impact or contact events. This is achieved by designing controllers that are required to control each AS mechanism only during their respective motion phase, and by finding appropriate transition conditions between the motion phases.

Choosing the control strategy is the first step of the AS Concurrent Optimization stage. Since the AS is composed of realistic, nonlinear mechanisms (e.g., double pendulum, SLIP, etc.,) control strategies based on such mechanisms can be readily implemented. The selected control strategy is dependent on the mechanisms resulting from the Bond Graph Modeling stage, and the behaviour required from the final system. In the case of legged robots, for example, this task is simplified, because a variety of control strategies have been proposed for the group of simple mechanisms that are analogous to the Linkage Leg [20] [55] [68].

The next step is to identify the transition conditions between motion phases. At these transitions the AS system changes between one mechanism to another. The transition conditions depend on the exact nature of the AS obtained from the Bond Graph Modeling stage, however two rules should be followed: The first rule is that the potential energy of the system must not change at the transition between the different mechanisms forming the AS. As such, the total mass and position of the centre of mass must remain the same for each mechanism forming the AS. The justification for this rule is that it ensures the optimized ES through the Dimensional Analysis stage is not required to be analogous to an AS formed by mechanisms with different mass or moment of inertia. Otherwise, it would be nearly impossible to define a unique set of parameters for the ES so that it behaves similarly to all the multiple mechanisms of the AS, if the mass or location of the center of mass of these mechanisms differs greatly. The kinetic energy or angular momentum cannot be preserved at transitions, because impacts cause energy loss in the system. The second rule is that the rates of the degrees of freedom unaffected by the switch of constraint conditions imposed by the contact are kept constant, while the rates of the other degrees of freedom experience an instantaneous change according to the specific impact or contact. This will be illustrated in chapter 3 by analyzing the specific cases of the touchdown and takeoff transitions between the AS mechanisms.

Once the transition conditions have been identified, each AS mechanism and control parameters can be concurrently optimized. Such a multidisciplinary concurrent optimization can

be solved by a variety of algorithms, due to its simple structure and low number of variables [3] [4] [69], whereas the same algorithms would likely have difficulty tackling with the original ES, as it will be illustrated in the following sections.

3.1.3 Dimensional Analysis

The purpose of the Dimensional Analysis stage is to find those parameters of the ES which result in the closest behaviour to that of the previously optimized AS. This is achieved by employing the concept of *dynamic similarity* and requiring the ES and AS to be approximately dynamically similar, i.e., analogous. The motivation for this approach is that if a sufficiently close dynamic similarity is achieved between the ES and AS, the controller developed concurrently with the AS during the second stage of MbA can be applied to the ES, without modifications while maintaining the same synergistic performance. The general concept of dynamic similarity is explained in [70].

Dynamic similarity starts with an equation of the general form Eq. (3.1), which relates the d_i groups of basic variables that define the phenomena:

$$d_1 + d_2 + d_3 + \dots + d_n = 0 \quad (3.1)$$

By replacing the units of Eq. (3.1) with appropriate values the groups d_i are replaced by dimensionless groups d'_i , called Pi-factors, which represent the class of phenomena independently of the basic variables. If the constitutive Pi-factors are equal for a class of phenomena, then these phenomena are considered dynamically similar. To apply the concept of dynamic similarity to the design of the ES, the Lagrangian equation of the ES is used as Eq. (3.1). The Pi-factors are obtained by replacing the units of length, time and energy as shown in [71]. An optimization procedure is used to obtain the parameters of the ES such that its Pi-factors are equal to the corresponding Pi-factors of the AS. If a perfect match is not achieved between the sets of Pi-factors, the dynamic similarity will not be exact. It is possible, however, to achieve a desirable result with an approximate similarity, i.e., analogy, as it will be shown for the case study of the Linkage Leg.

In some cases the mechanisms forming the AS will have a different set of degrees of freedom, due to some external constraints being present in some motion phases but not in others. In such cases the Lagrangian of the ES will be transformed to the coordinates of respective AS mechanism. An example of such transformation will be shown for the stance phase of the Linkage Leg.

Figure 3.2 Diagram of the Bond Graph Modelling phase

3.2 Bond graph modeling stage applied to Linkage Leg

The presented work aims at finding simple models, using bond graphs, with approximately similar dynamic behaviour to that of a given leg mechanism for each motion. Bond graphs are domain-independent graphical descriptions of dynamical behaviour of physical systems. Bond graphs are based on the concept that energy exchange is a common notion to dynamic systems regardless of their physical domain [72]. Therefore, in a bond graph components are defined by their energetic behaviour; they can either supply or absorb, store or dissipate, and reversibly or irreversibly transform energy [73]. Bond graphs can be used to represent complex three dimensional mechanical systems [74] [75], as well as mechatronic systems [76] [77] [78] [65].

The first part of this section expresses dynamics of a leg mechanism using bond graphs in a way that it facilitates further simplifications. The second part of the paper, Section 3.1.2, applies a combination of various simplification techniques to the bond graph model to find simpler representation of the system in each motion phase. The behaviour of simpler models is also compared with that of the original model in Section 3.1.3. Some concluding remarks are made in Section 3.1.4.

3.2.1 Bond graph model of the leg mechanism

The goal of this section is to obtain a model for the linkage leg mechanism that can be simplified to obtain the simple mechanisms that will form the AS. Bond graphs represent the system dynamics by considering the power exchange between its components. The system variables such as force, velocity, current, voltage, etc., are unified into two groups, the *flow* and *effort*. By multiplying these generalized variables, the power flow between components can be

computed. The bond graph uses the dynamic equations of its components and initial conditions to calculate the behaviour of the system [65]. Furthermore, bond graphs can be linked together in a modular fashion to represent a more complex mechanism. For example, the rigid body in Figure 3.3 is represented using a bond graph composed of several modules.

The generalized flow and effort variables correspond to velocity and force, respectively. The velocity (flow) of point O_1 is specified using flow source components (C), and the force (effort) applied at point O_2 is represented by effort source components (D). Both velocity and force vectors are expressed in the local coordinate frame of the body. The vectors r_1 and r_2 define the position of points O_1 and O_2 with respect to the centre of mass of the body in the local coordinate frame using transformer blocks (B). The rotational dynamics of the body are represented using an inductor component (A), and the translational dynamics of the body are represented by two inertia components as inductors and two effort sources in the world coordinate frame (F). A coordinate transformation block (E) relates the velocities and forces of the body centre of mass expressed in the world and local coordinate frames.

Using the above-mentioned six blocks the planar dynamics of the rigid body can be calculated, together with the power flow to each component. By linking a number of rigid body modules together, the dynamics of a more complex system, such as the leg mechanism shown in Figure 2.3 can be derived.

Figure 3.3 Bond graph representation of a rigid body dynamics

The bond graph representation of the Linkage Leg is shown in Figure 3.4. The bond graph is composed of five rigid body modules, the thigh, tibia, foot, tendon and robot body, to which the leg is attached. Additionally, a ground model is used to simulate the toe (O_3) touching the ground.

The bond graph model of Linkage Leg was simulated in two different conditions: the stance phase and swing phase, and its behaviour was compared with the models of the same mechanism

Figure 3.4 Bond graph representation of Linkage Leg, consisting of five solid body models from Fig 3.3

developed using both MATLAB[®] SimMechanics physical modeling as well as the m-file script, as described in Chapter 2.

Table 3.1 Variables for the test cases studied for the bond graph simplification stage

Inertia variable	Case 1	Case 2	Case 3
m_1 (kg)	0.5	1	0.5
m_2 (kg)	0.3	0.5	0.5
m_3 (kg)	0.2	0.5	0.5
m_4 (kg)	0.1	0.1	0.2
I_1 (kg m ²)	0.02	0.03	0.02
I_2 (kg m ²)	0.015	0.02	0.02
I_3 (kg m ²)	0.01	0.02	0.02
I_4 (kg m ²)	0.007	0.007	0.02

Table 3.2 Geometric parameters for the linkage leg used in bond graph simulation

Geometric variable	All cases	Geometric variable	All cases
a_1 (m)	0.5	x_1 (m)	0.25
a_2 (m)	0.5	x_2 (m)	0.25
a_3 (m)	0.5	x_3 (m)	0.15
a_4 (m)	0.3	x_4 (m)	0.25
a_5 (m)	0.5	y_1 (m)	0.00
a_6 (m)	0.2	y_2 (m)	0.00
ϕ_2 (deg)	0.00	y_3 (m)	0.00
ϕ_3 (deg)	π	y_4 (m)	0.00

Table 3.3 Initial conditions for stance and swing phase bond graph simulations

	Vertical body velocity (m/s)	Horizontal body velocity (m/s)	θ_1 (deg)	$\dot{\theta}_1$ (deg/s)	θ_2 (deg)	$\dot{\theta}_2$ (deg/s)
Stance Phase	-3	3	-50	-151	-103	-320
Swing Phase	3	3	-74	0	-104	0

The goal of the bond graph model is to obtain the power flow in the bonds that is representative for the Linkage Leg mechanism. The final parameters of the Linkage Leg are not know at this point, so three test cases will be simulated. In order to simulate the leg using the bond graph shown in Figure 3.44 the geometric parameters in Table 3.1 are used, together with the masses and

moments of inertia from Table 3.2. The bond graph is initialized using the initial conditions in Table 3.3. During the swing phase the knee is free to rotate, and during the stance phase a 38.4Nm/deg spring is applied around the knee joint.

Figure 3.5 shows the trajectories of the hip and knee joints during one step. The takeoff and touchdown points are also indicated. The bond graph has been simulated for a representative step motion, with the robot running at 3m/s and reaching 1.5m apex height, similar to a human gait. The swing and phase models are simulated independently, causing the instantaneous change in velocity at touchdown. The ground model is used during the stance phase, with a 10000 N/m spring

Figure 3.5 Trajectories of the knee and hip joint of the bond graph model.

and 100 N/(m/s) damper holding the toe point locked to the ground. The spring and dampers are set to zero during the swing phase so the body is free to move according to the projectile motion. The stance and swing phases are simulated independently, so the more complicated ground mechanism [63] used in Chapter 2 is not necessary.

This section detailed the modelling of the Linkage Leg using bond graphs. The next section will explain how the bond graphs model is simplified to obtain the AS mechanisms.

3.2.2 Bond graph model simplification

3.2.2.1 Simplification algorithms

This section defines a method of determining the analogous simpler mechanisms for a given mechanism using its bond graph model, and applies it to the Linkage Leg. The proposed method is a two-step process. The first step determines the dynamic elements that are necessary for the behaviour to remain similar to that of the original system, and the second step eliminates the bonds that have insignificant power flow and determines what mechanism is formed by the remaining dynamic elements. The term “dynamic element” refers to the bond graph component associated with masses and moments of inertia, generally represented by induction components.

The first step, called Dynamic Element Simplification (DES), is an element-oriented simplification approach that eliminates the maximum number of dynamic elements while keeping the dynamic behaviour unchanged [67]. The algorithm requires a set of dynamic elements S to be simplified. If there are m dynamic elements in the bond graph (such as masses and moments of inertia) and simplification is to be done for n of them, then there are C_{m-n}^m possible combinations. For each possible combination, the bond graph is simulated, and the behaviour of a set of test bonds is compared, such as the bonds related to the controlled degrees of freedom of the system. The result of this first step is a set of elements that produce the closest behaviour to the original system for each level of simplification, as measured by Root Mean Square (RMS) error [67]. The level of simplification represents the number of removed dynamic elements (level zero for no simplification, level one for one element removed, level two for two elements removed, etc.). Beyond a certain simplification level, the system behaviour will start markedly deviating from the original behaviour, indicating that all the remaining elements are necessary. This effectively sparsifies the dynamic elements of the bond graph.

The second simplification step, the Individual Bond Simplification (IBS), can be applied to any of the simplification levels from the first step. The goal is to find the necessary bonds for the

system behaviour to remain relatively unchanged. First, a metric is found that represents the importance of each bond and component for the bond graph. Bonds with zero or small power flow get reduced. The metric used in this step is based on the correlation of the energy flow pattern in the bonds of the bond graph. The end result is a rank for each bond that represents the importance of that bond to the overall system behaviour. The algorithm, based on the approach presented in [66], is summarized below:

- a) Find energy trajectories for each bond (E_i) from Fig. 3.5, and form the E matrix $E = [E_1, E_2 \dots E_n]$. The algorithm can analyze the entire bond graph or only a selection of bonds. The energy trajectories are obtained by simulating the bond graph for the desired behaviour.
- b) Apply Singular Value Decomposition (SVD) to the E matrix, and calculate the U, S, V matrices such that $E = USV^{-1}$.
- c) Calculate vector $I = \sum_{i=1}^n \sigma_i^2 |\vec{v}_i|$, where n is the number of bonds analyzed, $\sigma_{i=1..n}$ are the elements of the diagonal matrix S , and $\vec{v}_i$ is the i^{th} column of vector V .
- d) Normalize the vector I and order the bonds in the order of decreasing relative importance.
- e) Choose a value r to be the cut-off threshold. Every time $\frac{I_{i+1}}{I_i} < r$, a new reduction stage is reached. The cutoff threshold is chosen iteratively to result in clearly determined stages that correspond to meaningful parts of the bond graph [66]. For this analysis $r = 1.5$.
- f) Finally, one can choose as many reduction stages as necessary. The first two or three reduction stages usually contain the meaningful dynamics.

The rationale for the proposed approach consisting of both the DES and IBS steps is as follows: most methods, such as those presented in [66], [79] and [80] which form the basis for the IBS, are based on the notion of eliminating the bonds with zero or small power flow. For some cases, these methods are able to find the underlying simple mechanisms [79]. However, these methods are not suitable as a complete solution for mechanisms with parallel links, for example the Linkage Leg with the tibia-tendon and thigh-foot pairs, as it will be discussed in sequel. Therefore, if only the second simplification step is applied to the Linkage Leg, all the links will be considered as important and consequently no mass will be eliminated in the simplification process. The DES step is necessary for dealing with not only the insignificant individual masses but also the parallel links of the Linkage Leg by finding the minimal combination of masses that maintains the original system dynamics, regardless of power flow. Once the minimal set of significant masses

is found, IBS is able to determine the bonds that are significant in transmitting the energy in the step-1 mechanism.

It should be noted that the proposed simplification approach is dependent on the system motion phase as well as the initial conditions for each phase. In other words, the simplification should be done locally for a certain system behaviour, and it does not cover the entire dynamics of the system, because the significance of different dynamic elements varies in different system behaviours. Therefore, one must first define the working condition of the system with its motion phases and initial conditions prior to performing model simplification. This will be further discussed in the following section.

3.2.2.2 Linkage Leg model simplification

To determine the working behaviour of the Linkage Leg two simulations were initially performed for the two motion phases. The trajectories of the hip and the knee joints were consequently obtained, and they are shown in Figure 3.6, for the stance phase, and Figure 3.7, for the swing phase. Also, the proper initial conditions for both phases were obtained, as shown in Table 3.3. The following model simplifications will be based on the system behaviour in stance and swing phases, which resulted from the obtained initial conditions. This local simplification approach would provide more effective results that are directly relevant to the interested working behaviour of the system. For many systems, a global approach of simplification that can result in a unique simpler system resembling the original one at all working conditions may be inefficient or infeasible, as it is argued in [79].

To simplify the Linkage Leg bond graph model, three cases are investigated. The test cases are meant to reflect different construction possibilities for the Linkage Leg, based on CAD models of possible configurations. These cases, listed in Table 3.2, were chosen to represent different construction methods and leg configurations. The first one represents a more natural leg, with the thigh (m1) and tibia (m2) heavier than foot (m3) and tendon (m4). The third case represents a more classic robot construction, with all the linkages of equal mass, with only the tendon being lighter. The second case is in between the first and third ones, with the thigh heavier than the tibia and foot, which are equal, and the tendon lighter. In all three cases the Linkage Leg geometry is kept constant as shown in Table 3.1 and the mass and moment of inertia parameters are changed, as listed in Table 3.3.

The first step of the simplification (DES) requires a set of dynamic elements to be simplified and the bonds used to test the similarity to the original system. Similar to most robotic legs, the Linkage Leg has two degrees of freedom, the hip (θ_1) and the knee (θ_2) joints. These two joints are also the actuated and controlled joints. Therefore, the bonds associated with the moments of inertia of these two joints are used to test the similarity of each simplified combination to the original system, as explained in [67]. The components associated with the other dynamic elements (masses m_1 , m_2 , m_3 , m_4 of the linkage leg bodies, the mass m_B of the robot body and moments of inertia I_3 and I_4) form the set of elements to be simplified.

The results of the first simplification step, presented in Table 3.4, show the combination of elements with the lowest cost (closest fit to the original data) for each simplification level of the Linkage Leg in figure 2.1. The cost function is the square difference between the trajectories of the test joints (O_1 and O_2) of the simplified simulation and those of the original case. These trajectories for case 3 are shown in Figure 3.6, for the swing phase, and in Figure 3.7, for the stance phase. The unmodified trajectory is the same as the one in Figure 3.4.

Table 3.4 Results of the dynamic element simplification (DES) step.

Simp.	SWING PHASE								STANCE PHASE							
Level	Case 1								Case 1							
	m_1	m_2	m_3	I_3	m_4	I_4	m_{body}	cost	m_1	m_2	m_3	I_3	m_4	I_4	m_{body}	cost
Level 0	0.5	0.3	0.2	0.01	0.1	0.007	10	0	0.5	0.3	0.2	0.01	0.1	0.007	10	0
Level 1	0.5	0.3	0.2	0.01		0.007	10	0.035	0.5	0.3	0.2		0.1	0.007	10	0.002
Level 2	0.5	0.3	0.2	0.01			10	0.061	0.5	0.3	0.2		0.1		10	0.008
Level 3	0.5		0.2		0.1		10	0.123	0.5	0.3	0.2				10	0.028
Level 4	0.5		0.2				10	0.083	0.5	0.3					10	0.065
Level 5		0.3					10	0.244	0.5						10	0.113
Level 6							10	0.489							10	0.261
	Case 2								Case 2							
	m_1	m_2	m_3	I_3	m_4	I_4	m_{body}	cost	m_1	m_2	m_3	I_3	m_4	I_4	m_{body}	cost
Level 0	1	0.5	0.5	0.02	0.1	0.007	10	0	1	0.5	0.5	0.02	0.1	0.007	10	0
Level 1	1	0.5	0.5	0.02		0.007	10	0.022	1	0.5	0.5		0.1	0.007	10	0.003
Level 2	1	0.5	0.5	0.02			10	0.033	1	0.5	0.5		0.1		10	0.008
Level 3	1	0.5	0.5				10	0.103	1	0.5	0.5				10	0.024
Level 4	1		0.5				10	0.128	1	0.5					10	0.118
Level 5		0.5					10	0.320	1						10	0.164
Level 6							10	0.449							10	0.423
	Case 3								Case 3							
	m_1	m_2	m_3	I_3	m_4	I_4	m_{body}	cost	m_1	m_2	m_3	I_3	m_4	I_4	m_{body}	cost
Level 0	0.5	0.5	0.5	0.02	0.2	0.02	10	0	0.5	0.5	0.5	0.02	0.2	0.02	10	0
Level 1	0.5	0.5	0.5	0.02		0.02	10	0.119	0.5	0.5	0.5		0.2	0.02	10	0.003
Level 2	0.5		0.5	0.02	0.2		10	0.084	0.5	0.5	0.5		0.2		10	0.023
Level 3	0.5		0.5	0.02			10	0.159	0.5	0.5	0.5				10	0.055
Level 4	0.5		0.5				10	0.231	0.5	0.5					10	0.142
Level 5		0.5					10	0.518	0.5						10	0.194
Level 6							10	0.823							10	0.338

Figure 3.6 Swing phase joint angle results for DES step, Case 3. It can be seen that the last two simplification levels, 5 and 6, differ greatly from the original behaviour. As such, simplification level 4 is the one used in the IBS step.

The simplification levels start from level zero, with no simplified elements, and go up to level six, where all the dynamic elements of the linkage leg are removed. The data shows that the mass of the robot body is the most important element in all cases, and was never simplified.

For the swing phase, the lower simplification levels are different between the test cases, indicating that the least important dynamic elements vary depending on the test case being considered. However, the higher simplification levels are the same for all test cases. If four elements are simplified and two are kept intact, m_1 and m_3 are determined to be the important elements, and if only one element is kept, m_2 is determined to be the most influential. In Figure 3.6 it can be seen that simplification levels five and six have markedly different trajectories for the joints on interest, O_0 and O_1 , while simplification level four shows a closer behaviour to the unmodified system. Consequently, the simplification levels four and five will be analyzed further.

It should be noted that to obtain a valid comparison of hip and joint trajectories between different simplification levels in Figure 3.6, the hip torque τ_{hip} applied during the swing phase must be proportional with the overall moment of inertia of the leg around the hip joint.

For the stance phase, Figure 3.7, the Linkage Leg behaviour remains relatively unchanged no matter how many dynamic elements in the leg mechanisms are simplified. This is explained by the much larger mass assigned to the body of the robot and the fact that the toe is in contact with the

Figure 3.7 Stance phase joint angle results for DES step, Case 3. It can be seen that all simplification levels have similar behaviour, so all the dynamic elements of the leg can be eliminated and the behaviour will be close to that of the original leg.

ground, which dominates the dynamic behaviour of the joints O_1 and O_2 . The case of all the elements being simplified is the one with the most different behaviour, as expected.

The next step is to understand what kind of mechanism is described by the remaining dynamic elements. For the stance phase, the combination resulting from the sixth simplification level is investigated (all dynamic elements of the leg, but not the robot body, are removed), whereas for the swing phase the combinations from resulting from the fourth and fifth simplification levels are investigated. Given that the results are very similar for all three initial conditions cases, for the sake of brevity they will be presented only for case 3.

The results of the second simplification step are shown in Figure 3.8 and Figure 3.9 for the swing phase, and, in Figure 3.10 for the stance phase. The meaningful bonds are shown in black, and the reduced bonds are grayed out. For the fifth level of simplification, shown in Figure 3.9, the gravity and inertial effects of m_2 remain unchanged, together with the vector defining the location of the center of mass relative to O_1 . The tibia is linked to the robot body by the massless rod $\{O_0O_1\}$, represented in the bond graph by the remaining elements of the body $\{O_0O_1O_4\}$, forming a double pendulum system. For the simplification level four (Figure 3.8), body m_1 is left

Figure 3.10 Stance phase, simplification level six

intact, but the bonds associated with the mass m_2 are removed. Instead the mass m_3 acts at the point O_2 , effectively becoming part of the body $\{O_1O_2\}$. This also forms a double pendulum mechanism, this time however both links have mass and inertia.

This difference explains the change in behaviour between simplification levels four and five, indicating that a double pendulum where both links have mass is the simplest mechanism that has similar behaviour to the original linkage leg. The bonds associated with the position of the mass m_3 relative to point O_2 have an effect on the system behaviour, moving it away from a perfect double pendulum similarity. The effect is discussed below, in the comparison section.

The stance phase bond graph for simplification level six is shown in Figure 3.10. The robot body dynamics remain unchanged, and the other remaining elements are the ground contact model and the spring of the joint O_1 . The rest of the bonds are kinematic bonds, which describe the location of the body with respect to the ground contact point O_3 , as well as the orientation of the body velocity with respect to the ground.

The resulting mechanism is similar to the SLIP model, widely used for running and hopping robots. The main difference is that in the SLIP model the direction of the force due to the spring is coincident with the hip joint [20], whereas in the Linkage Leg this direction is determined by the instantaneous configuration of the four bar linkage system [81], and changes depending on the leg length.

3.2.3 Comparison between Linkage Leg and the simple mechanisms

The double pendulum is proposed as a simpler mechanism candidate for the swing phase and the SLIP model as a candidate for the stance phase. As indicated in the bond graph, the first rigid

Figure 3.11 Comparison between Linkage Leg and double pendulum

Figure 3.12 Comparison between Linkage Leg and SLIP model

body of the double pendulum is composed of the mass m_1 and rotational inertia I_1 , and the second rigid body is composed of the mass m_3 and rotational inertia I_2 . The comparison between the Linkage Leg and the double pendulum is shown in Figure 3.11, and it shows that the behaviour of the hip and knee trajectories is similar. The hip trajectory of the double pendulum has a 10% maximum difference compared with the linkage leg, while the knee is closer with a 5% maximum difference, as measured by RMS.

This difference between the double pendulum and linkage leg is explained by the effect of the moment arm bonds on the m_3 mass. This effect can be minimized depending on the location of mass m_3 relative to the point O_2 . A mass offset x_3 of 0.5 results in the mass m_3 coincident with O_2 , whereas an offset of 0.35 results in the mass m_3 in the center of the segment O_3O_5 . Both the Linkage Leg and the double pendulum trajectories end up at the same touchdown positions. The leg angles at touchdown are shown to have the greatest impact on the running gait in [20], further indicating suitability of the double pendulum for this particular application.

The SLIP model is composed of a massless spring between the body center of mass and the toe contact point [20]. Given that the SLIP model does not have the same hip and knee joints of the Linkage Leg, the two systems are compared based on the position of the center of mass of the robot body. The comparison is shown in Figure 3.12. Both the horizontal and the vertical position of the body are very similar between the Linkage Leg and the SLIP model, indicating that the offset between the hip and the spring force present in the Linkage Leg has little influence on the final behaviour. The maximum difference of 0.02m between body positions is similar to the difference found between the models shown in [66].

3.2.4 Bond graph stage conclusions

A method was presented that finds simpler mechanisms with a dynamic behaviour similar to that of a more complicated robotic leg mechanism for each specific motion phase. Such simpler mechanisms will be good representatives of the original mechanism while having the advantages of simplicity and tractability. The original leg mechanism was first modeled using a bond graph approach, and then a two-step simplification process was used to determine simpler mechanisms with the same significant dynamic characteristics. The first step finds the dynamic elements that are necessary for the system behaviour to remain unchanged for each motion phase, and the second step simplifies the bond graph formed by such elements and constructs the resulting mechanism expressed by the simplified graph.

For the case study of a Linkage Leg two motion phases were analyzed, i.e., swing and stance. During the swing phase, two masses elements were found to be required for generating the same behaviour as the original system, resulting in a double pendulum mechanism. And, for the stance phase the resulting simplified mechanism was found to be the classic SLIP model that has been used for modeling robotic and biologic leg dynamics.

Having such simpler mechanisms will enable the designer to fine-tune the kinematic, dynamic and control parameters of the original mechanism in a more intuitive way using methods such as the one discussed in the following section.

3.3 Analogous System concurrent optimization

The goal of the AS Concurrent Optimization stage is to obtain the best AS mechanism and controller for each respective motion phase. Since the AS is composed of different mechanisms, each representing a valid analogy to the ES for its corresponding motion phase controllers are designed for each AS mechanism only during their respective motion phase. Once the transition conditions have been identified and control strategy chosen, each AS mechanism and control parameters can be concurrently optimized.

The first step in the AS Concurrent Optimization stage, Figure 3.13, is to define the control strategy for the AS depicted in Figure 3.14. The swing state has two controllable degrees of freedom. They are chosen to be the angle of the leg with respect to the ground, θ_L , controlled by the torque at the hip (τ_1), and the angle of the knee, θ_2 , controlled by the torque at the knee (τ_2). The stance phase has one controllable degree of freedom, chosen to be the angle of the body with respect to the ground, θ_B , controlled by the torque at the hip. Furthermore, during the stance phase

Figure 3.13 Diagram of the Analogous System Concurrent Optimization phase

a constant force (f_L) is applied to the leg joint when the leg is extending, helping to offset any energy loss during the gait [20].

Figure 3.14 Analogous System

The selected control strategy consists of a desired trajectory defined by a Bezier curve [25] for each controllable degree of freedom, and a PD controller that attempts to follow the desired trajectory. Bezier curves were chosen as they allow for a simple way to define beginning and end points and guarantee smoothness. The formula for the Bezier curve is:

$$\theta_j^d(s) = \sum_{k=0}^{M_j} (P_j)_k \cdot \frac{M_j!}{k!(M_j-k)!} s^k (1-s)^{M_j-k}, \quad j \in \{\theta_L, \theta_2, \theta_B\}, \quad s \in [0,1] \quad (3.1)$$

where the superscript d denotes the desired values. Each Bezier curve in (3.1) is defined by the number of control points, M_j , and by the vector of control points, P_j .

For the hip joint during swing and stance $M_{\theta_L} = 4$ and $M_{\theta_B} = 4$, respectively. One of the useful properties of Bezier curves is that the first and last control points represent the initial and final trajectory values, and the second and second-last control points determine the initial and final slope of the trajectory. Consequently, the hip joint the control point vectors are:

$$P_{\theta_L} = [\theta_{L,sw,i} \quad \theta_{L,sw,i} + \dot{\theta}_{L,sw,i} \quad -\dot{\theta}_{L,sw,f}^d + \theta_{L,sw,f}^d \quad \theta_{L,sw,f}^d] \quad (3.2)$$

$$P_{\theta_B} = [\theta_{B,st,i} \quad \theta_{B,st,i} + \dot{\theta}_{B,st,i} \quad -\dot{\theta}_{B,st,f}^d + \theta_{B,st,f}^d \quad \theta_{B,st,f}^d] \quad (3.3)$$

The subscript sw denotes the swing phase, the subscript st denotes the stance phase, and the subscripts i and f refer to the initial and final values of the variables in their respective motion phase.

The knee trajectory for the swing phase has one intermediate control points, so the knee can be retracted. The initial and final position of the knee trajectory are the same.

$$P_{\theta_2} = [\theta_{2,sw,i} \quad \theta_{2,sw,i} + \dot{\theta}_{2,sw,i} \quad \theta_{2,inter}^d \quad -\dot{\theta}_{2,sw,f}^d + \theta_{2,sw,i}^d \quad \theta_{2,sw,i}^d] \quad (3.4)$$

The vector of desired parameters $\alpha = [\theta_{L,sw,f}^d, \dot{\theta}_{L,sw,f}^d, \dot{\theta}_{2,sw,f}^d, \theta_{2,inter}^d, \theta_{B,st,f}^d, \dot{\theta}_{B,st,f}^d, f_L]^T$ defines the seven tunnable parameters of this control system. The desired touchdown angle and angular velocities of the leg are defined as $\theta_{L,st,f}^d$ and $\dot{\theta}_{L,st,f}^d$, which affects the hip trajectory during swing. The knee trajectory during the swing phase is controlled by the knee touchdown velocity, $\dot{\theta}_{2,st,f}^d$ and the intermediate Bezier control point $\dot{\theta}_{2,st,f}^d$. During the stance phase the hip trajectory is controlled by the desired body angle and angular velocity at takeoff ($\theta_{B,st,f}^d$ and $\dot{\theta}_{B,st,f}^d$) and the force in the leg, f_L . The rest of the Bezier control points from Eq. 3.2-3.5 cannot be directly specified, but are given by the actual configuration of the robot at the start of the respective motion phase.

The PD controller is defined by the proportional gain K^P and differential gain K^D . A servo control strategy was chosen because of its simplicity, given that the goal of the present work is not to determine the best control strategy for a running robot, but to show that, given a selected control strategy, MbA offers a better alternative to the traditional design optimization procedures. During the stance phase the knee joint is actuated only when the leg is extending, using a constant force, $f_{L,st}$, similar to the strategy used in [20].

The next step of the AS Concurrent Optimization stage is to understand the transition conditions between the phases. The configuration space for the swing phase is parameterized using the Cartesian position of the hip joint (x_0, y_0) , the angle of the robot's body with respect to the ground (θ_B) , as well as the angles of the hip (θ_1) and knee (θ_2) , i.e., $q_{sw} = [x_{0,sw} \ y_{0,sw} \ \theta_{B,sw} \ \theta_{1,sw} \ \theta_{2,sw}]^T$, $x_{sw} = [q_{sw} \ \dot{q}_{sw}]^T$. The parameterization for the stance phase is done using the degrees of freedom of the SLIP model: the angle of the leg with respect to the ground (θ_L) , the length of the leg (l) , and the hip angle (θ_H) , as shown in Figure 3.14, i.e., $q_{st} = [\theta_{L,st} \ l_{st} \ \theta_{H,st}]^T$, $x_{st} = [q_{st} \ \dot{q}_{st}]^T$. At the touchdown and takeoff transitions the parameterization must be changed from the swing to the stance phase, and from stance to swing, respectively. The initial state of the stance phase, $x_{st,i}$, is defined as a function of the final state of the swing phase, namely $x_{st,i} = \Delta_{sw \rightarrow st}(x_{sw,f})$. Similarly, the initial state of the swing phase, $x_{sw,i}$, is defined as a function of the final state of the stance phase, i.e., $x_{sw,i} = \Delta_{st \rightarrow sw}(x_{st,f})$.

$$\Delta_{sw \rightarrow st} = \begin{bmatrix} \tan^{-1} \left(y_{0,sw,f} / (x_{0,sw,f} - x_{toe,sw,f}) \right) \\ \sqrt{(x_{0,sw,f} - x_{toe,sw,f})^2 + y_{0,sw,f}^2} \\ \pi - \phi_{offset} - \theta_{1,sw,f} \\ \left((x_{0,sw,f} - x_{toe,sw,f}) \dot{y}_{0,sw,f} - y_{0,sw,f} \dot{x}_{0,sw,f} \right) / \left((x_{0,sw,f} - x_{toe,sw,f})^2 + y_{0,sw,f}^2 \right) \\ \left((x_{0,sw,f} - x_{toe,sw,f}) \dot{x}_{0,sw,f} + y_{0,sw,f} \dot{y}_{0,sw,f} \right) / \sqrt{(x_{0,sw,f} - x_{toe,sw,f})^2 + y_{0,sw,f}^2} \\ -\dot{\theta}_{1,sw,f} \end{bmatrix} \quad (3.5)$$

$$\Delta_{st \rightarrow sw} = \begin{bmatrix} l_{st,f} \cos \theta_{L,st,f} \\ l_{st,f} \sin \theta_{L,st,f} \\ \theta_{L,st,f} + \theta_{H,st,f} \\ \pi - \phi_{offset} - \theta_{H,st,f} \\ (l_{st,f} - l_{st,i})C \\ \dot{l}_{st,f} \cos \theta_{L,st,f} - l_{st,f} \sin \theta_{L,st,f} \dot{\theta}_{L,st,f} \\ \dot{l}_{st,f} \sin \theta_{L,st,f} + l_{st,f} \cos \theta_{L,st,f} \dot{\theta}_{L,st,f} \\ \dot{\theta}_{L,st,f} + \dot{\theta}_{H,st,f} \\ -\dot{\theta}_{H,st,f} \\ \dot{l}_{st,f}C \end{bmatrix} \quad (3.6)$$

The parameter C in (7) represents the coupling between the rate of change of leg length, $\dot{l}$, and the knee joint, $\dot{\theta}_2$, of a Linkage Leg with the same geometry as that defined by the double pendulum, as explained in section 3.3.

To ensure that the potential energy remains the same while transitioning between the AS mechanisms, the SLIP masses, moments of inertia and locations of the centre of mass must be equal to those of the double pendulum, such that the following equations are satisfied:

$$\begin{aligned}
m_s &= m_1 \\
m_U &= m_2 \\
I_s &= I_1 \\
I_U &= I_2 \\
x_s &= -((x_1 + a_1) \cos(-\theta_{\text{offset}}) - y_1 \sin(-\theta_{\text{offset}})) \\
y_s &= -((x_1 + a_1) \sin(-\theta_{\text{offset}}) + y_1 \cos(-\theta_{\text{offset}})) \\
x_U &= l_{sw,i} - (a_1 \cos(-\theta_{\text{offset}}) + (a_2 + x_2) \cos(-\theta_{\text{offset}} + \theta_{2,sw,i})) - \\
&\quad y_2 \sin(-\theta_{\text{offset}} + \theta_{2,sw,i}) \\
y_U &= -(a_1 \sin(-\theta_{\text{offset}}) + (a_2 + x_2) \sin(-\theta_{\text{offset}} + \theta_{2,sw,i})) + \\
&\quad y_2 \cos(-\theta_{\text{offset}} + \theta_{2,sw,i})
\end{aligned} \tag{3.7}$$

The third stage is to optimize the AS mechanical system, Bezier trajectories and controller gains concurrently for the stance and swing phases. The optimization is performed for one gait cycle, consisting of swing and stance phases, as shown in Figure 3.14.

The dynamics of the double pendulum and robot body during the swing phase are expressed in the equation (3.8):

$$\begin{aligned}
\dot{x}_{sw} &= f_{sw}(x_{sw}, \tau_1, \tau_2) \\
\dot{q}_{sw} &= \begin{bmatrix} \dot{q}_{sw} \\ \dot{q}_{sw} \end{bmatrix} \\
f_{sw} &= \begin{bmatrix} M_{sw}^{-1} \left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ \tau_1 \\ \tau_2 \end{bmatrix} - C_{sw}(\dot{q}_{sw}, q_{sw}) \dot{q}_{sw} - g_{sw}(q_{sw}) \right) \end{bmatrix}
\end{aligned} \tag{3.8}$$

The matrix M_{sw} is the mass matrix of the double pendulum and robot body mechanism, the matrix C_{sw} is related to the Coriolis and centrifugal effects, and g_{sw} is the column vector expressing the effects of gravity. The values in the torque column vector are τ_1 , the torque in the hip joint, and τ_2 , the torque in the knee joint.

The dynamics of the SLIP leg and robot body system during the stance phase are expressed in a similar fashion in the equation (3.9):

$$\dot{x}_{st} = f_{st}(x_{st}, \tau_H, f_L) \quad \dot{q}_{sw}$$

$$f_{st} = \left[M_{st}^{-1} \left(\begin{bmatrix} 0 \\ f_L \\ \tau_H \end{bmatrix} - C_{st}(\dot{q}_{st}, q_{st}) \dot{q}_{st} - g_{st}(q_{st}) \right) \right] \quad (3.9)$$

The equations (3.8) and (3.9) together with the transition functions (3.5) and (3.6) define all the equations necessary to simulate the behaviour of the AS system during a gait cycle.

The variables to be optimized are shown in Table 3.5, and they are divided into three categories: the AS mechanical design variables, defining the dimensions, mass and inertia parameters of the AS, the Bezier trajectory variables, defining the desired trajectories of θ_L and θ_2 for the swing phase and θ_B for the stance phase, the control gains, as well as the variables defining the takeoff state at which the simulation is initialized ($x_{sw,i}$).

Table 3.5 Analogous System Optimization variables

Physical Variables					
Body Parameters					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
m_B	kg	10.00	8.00	12.00	9.827
I_B	kg m ²	0.83	0.67	1.00	0.800
x_B	m	0.00	-0.50	0.50	0.005
y_B	m	0.00	-0.10	0.50	0.025
Double Pendulum Parameters					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
m_1	kg	0.500	0.300	1.000	0.473
I_1	kg m ²	0.011	0.003	0.040	0.014
x_1	m	-0.250	-0.500	0.000	-0.246
y_1	m	0.000	-0.100	0.100	-0.006
m_2	kg	0.500	-0.300	1.000	0.472
I_2	kg m ²	0.011	0.003	0.040	0.018
x_2	m	0.000	-0.500	0.000	-0.015
y_2	m	0.000	-0.100	0.100	-0.011
a_1	m	0.500	0.300	0.700	0.415
a_2	m	0.500	0.300	0.700	0.480
SLIP Parameters are required to be equal to the double pendulum parameters					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
m_S	kg	0.500	0.300	1.000	0.473
I_S	kg m ²	0.011	0.003	0.040	0.014
x_S	m	-0.220	-0.500	0.000	-0.153
y_S	m	-0.117	-0.100	0.100	-0.072
m_U	kg	0.500	-0.300	1.000	0.472
I_U	kg m ²	0.011	0.003	0.040	0.018
x_U	m	0.404	-0.500	0.000	0.518
y_U	m	0.241	-0.100	0.100	0.245
$l_{st,i}$	m	1.000	0.900	1.100	1.056

ϕ_{offset}	deg	-28.000	-50.000	0.000	-27.078
K_{SLIP}	kN/m	7.000	6.000	9.000	7.067
Desired Trajectory Parameters (α)					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
$\theta_{L,sw,f}^d$	deg	103	90	120	106
$\dot{\theta}_{L,sw,f}^d$	deg/s	-200	-300	-100	-202
$\theta_{2,sw,f}^d$	deg/s	0	-100	100	-3
$\theta_{2,inter}^d$	deg	200	100	300	202
$\theta_{B,st,f}$	deg	0	-100	100	-1
$\dot{\theta}_{B,st,f}$	deg/s	10	-57	172	6
f_L	N	10	-10	30	10
Initialization state ($x_{sw,i}$)					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
$\theta_{L,sw,i}^d$	deg	76	60	85	74
$\dot{\theta}_{L,sw,i}^d$	deg/s	-200	-300	-100	-238
$L_{sw,i}$	m	1	0.8	1.2	1.06
$\dot{L}_{sw,i}$	m/s	4	3	5	4.05
$\theta_{B,st,i}$	deg	0	-30	30	4
$\dot{\theta}_{B,st,i}$	deg/s	5	-100	100	62
Control Parameters					
K^P	Nm/deg	100	50	500	216
K^D	Nm s/deg	100	50	500	260

The optimization constraints, shown in Table 3.6, are required such that the behaviour of the system represents a proper running gait. The most important characteristic of a running gait is its repeatability [20]. This is represented by the equality constraints 1-6 which require the linear and rotational positions and velocities of the robot body to be consistent between the initial state of the swing phase and the final state of the stance phase.

Table 3.6 Analogous System Optimization constraints

Equality Optimization Constraints			#
$\theta_{B,sw,i} - \theta_{B,st,f} = 0$	Difference between the initial and final hip angle		1
$\dot{\theta}_{B,sw,i} - \dot{\theta}_{B,st,f} = 0$	Difference between the initial and final hip angle		2
$x_{0,sw,i} - x_{0,st,f} = 0$	Difference between the initial and final hip horizontal position		3
$y_{0,sw,i} - y_{0,st,f} = 0$	Difference between the initial and final hip height		4
$\dot{x}_{0,sw,i} - \dot{x}_{0,st,f} = 0$	Difference between the initial and final hip horizontal velocity		5
$\dot{y}_{0,sw,i} - \dot{y}_{0,st,f} = 0$	Difference between the initial and final hip vertical velocity		6
$(m_1 + m_2) - (m_U + m_S)$	Mass of AS mechanism is the same in both phases		7
$I_{sw} - I_{st} = 0$	Moment of inertia of the stance AS mechanism is equal to that of the swing AS mechanism		8
$x_{com,sw} - x_{com,st} = 0$	Difference between the horizontal location of the centre of mass of the AS mechanisms is zero		9

$y_{com,sw} - y_{com,st} = 0$	Difference between the vertical location of the centre of mass of the AS mechanisms is zero	10
Inequality Optimization Constraints		
$\dot{y}_{0,sw,i} > 0$	Vertical velocity at takeoff is positive	11
$\min(y_0) > 0$	Minimum height of the hip centre of mass is positive	12
$\min(\dot{x}_0) > 1$	Forward velocity is above 1m/s	13
$\max(\tau_1) < 50$	Maximum torque is below 50Nm	14
$\max(\tau_2) < 50$		
$\max(\tau_H) < 50$		

where:

$I_{sw} = m_1((a_1 + x_1)^2 + y_1^2) + m_2(a_1^2 + a_1y_2 + (a_2 + x_2)^2 + y_2^2 + 2a_1 \cos \theta_2 (a_2 + x_2) - 2a_1y_2 \sin \theta_2) + I_1 + I_2$
is the moment of inertia of the double pendulum (swing phase AS mechanism) around the O_1 joint, and:

$$I_{st} = I_U + I_S + m_S(x_S^2 + y_S^2) + m_U((l_{st} + x_U)^2 + y_U^2)$$

is the moment of inertia of the SLIP (stance phase AS mechanism).

The inequality constraints 11-14 are used to make sure the gait is desirable for a legged robot. Constraint 11 requires that the vertical velocity at takeoff be positive, so that the robot can leave the ground. Constraint 12 requires that the velocity always stay above the desired value of 1m/s, and constraint 13 requires that the minimum height of the hip be positive. The last constraint requires that the maximum torque values of the hip and knee joints be below 50Nm, as a typical upper bound for motors used for legged robots [68].

The cost function of the AS optimization is $1/\max(\dot{x}_0)$. Minimizing this cost function results in the highest possible forward speed given the optimization constraints. Notably, the maximum torque constraint is limiting the achievable maximum speed.

Three gait cycles of the optimized AS are shown in Figure 3.15, and the behaviour of the initial and optimized AS are compared in Figure 3.16. The horizontal and vertical velocities at the beginning of the swing and the end of the stance are not equal, so the initial AS, shown with dotted lines, does not achieve a repeatable gait with the initial control vector α and initial configuration

Figure 3.15 Three steps of the optimal AS system

Figure 3.16 Comparison of the behaviours of the initial AS system (dotted line) and the optimized AS system (solid line) for one step.

$x_{sw,i}$. The optimized AS, shown using solid lines, achieves both a repeatable gait as well as a faster speed than the initial AS. The optimal hip and knee torque required by the optimal trajectories is much lower than the torque of the initial case.

The optimized behaviour is similar to the one shown by other single legged hopping robots [20]. The velocity of the hip is fairly constant during the swing phase, and decreases in the middle of the stance phase [55]. The touchdown angle of the leg was found to be the most important factor affecting the forward velocity of the robot [55], and the optimization procedure was shown to result in a trajectory of the hip, θ_L , which ensures a touchdown angle that maintains constant forward velocity. The hip trajectory during stance phase, θ_B , ensures that the body angle at the end

of stance is the same as the angle at the beginning of swing, so the gait cycle is repeatable as seen in Figure 3.15.

3.3.1 Robust trajectory control

The controller described previously follows desired trajectories that, if optimized, can lead to a repeatable gait, but is not able to deal with disturbances that affect the unactuated degrees of freedom. The angular displacement of the body and its centre of mass velocity during the swing phase are especially difficult to control, since the robot has no contact with the ground. Hence, a top-level control is required to help the robot deal with disturbances. Two approaches exist in the literature to achieve this goal. The first is to modify the simple PD controller so that it can account for exteroceptive errors [82], and the second approach is to modify the previously obtained optimal trajectories of subsequent gait steps to account for the errors in the current step. The first approach is a continuous-time control, and the second one is discrete time control, updating the controller only once for each step of the gait, and it is thus more desirable for lowering the required computational power of the system. This approach of controlling the joint trajectories based on the error of a Poincare point is common in the control of walking and running robots [18].

The top-level control consists of a control matrix K that relates the error in the states of the robot at a certain time in the gait to the change in the trajectory parameters in the subsequent step. The instant at which this controller is applied (Poincare point) was chosen to be the takeoff moment. The states of the robot at takeoff are defined by three variables, $x_{st,f} = [\theta_{L,st,f}, \dot{\theta}_{L,st,f}, l_{st,f}, \dot{l}_{st,f}, \theta_{B,st,f}, \dot{\theta}_{B,st,f}]^T$ and their derivatives, similarly to the way the stance states are defined. The top-level control is defined in Eq. (3.10):

$$\Delta\alpha = K \cdot (x_{st,f} - x_{st,f}^d) \quad (3.10)$$

The control gain K is a 7-by-6 matrix that calculates each of the desired parameters in α , depending on the errors between the takeoff states $x_{st,f}$ and the desired takeoff states $x_{st,f}^d$. The task now becomes to design the matrix K in such a way that the vector of desired parameters α do not deviate too much from that obtained in the AS concurrent optimization, but is also able to deal with external stimuli. Several of the takeoff states of the robot, such as the angular velocity $\dot{\theta}_{B,st,f}$ and the leg extension rate, $\dot{l}_{st,f}$, cannot deviate too much from their nominal values, or the subsequent step will fail with no chance for recovery.

In [20] it is assumed that the horizontal and vertical components of the velocity of the body centre of mass at takeoff can be decoupled, with the former being controlled by the leg touchdown angle ($\theta_{L,sw,f}^d$) and the later by the force in the leg during stance (f_L). If this strategy was used, the matrix K would have only two non-zero elements, (1,4) and (7,5). However, a better performance can be obtained if all the elements of α are controlled depending on the takeoff state, and thus have a full matrix K [18].

Figure 3.17 AS two-layer control strategy

Discrete-time Linear Quadratic Regulator (LQR) is a classic solution to the problem of designing the controller in Eq. (3.10). The LQR takes a discrete-time linear system $x_{k+1} = A x_k + B u_k$ and a performance index $J = \sum_{k=0}^{\infty} (x_k^T Q x_k + u_k^T R u_k)$, and calculates the optimal gain K that minimizes the deviations of the states and control inputs from their desired values [83]. Matrices Q and R define the weight associated with the states and the inputs, respectively. The designer must supply these matrices, which are often found by trial-and-error.

To obtain the A and B matrices describing the system, the system in Figure 3.14 must be linearized. A step of the AS is formed by a swing phase (f_{sw}), a swing-to-stance transition ($\Delta_{sw \rightarrow st}(x_{sw,f})$) and a stance phase (f_{st}). The Poincare return map function $x_{st,f,k+1} =$

$f_{AS}(x_{st,f,k}, \alpha_k)$ is constructed to represent this step, where $x_{st,f,k}$ represents the takeoff state at the beginning of the current step, α_k represents the desired trajectory parameters, and $x_{st,f,k+1}$ is the takeoff at the end of the current step. This function has no closed-form solution, but has to be numerically evaluated by simulating the step. The linear system needed for the LQR controller is achieved by taking the Jacobians of the f_{AS} function, as seen in Eq. (3.11).

$$\begin{aligned}
 x_{st,f,k+1} &= f_{AS}(x_{sw,i,k}, \alpha_k) \text{ nonlinear system} \\
 A &= \frac{\partial f_{AS}(x_{sw,i}, \alpha)}{\partial x_{sw,i,k}} \text{ Jacobian with respect to states} \\
 B &= \frac{\partial f_{AS}(x_{sw,i}, \alpha)}{\partial \alpha} \text{ Jacobian with respect to input} \\
 x_{sw,i,k+1} &= Ax_{sw,i,k} + B\alpha_k \text{ linearized system}
 \end{aligned} \tag{3.11}$$

For the values optimal AS shown in table 3.5, the A and B matrices are:

$$\begin{aligned}
 A = \frac{\partial f_{AS}(x_{sw,i}, \alpha)}{\partial x_{sw,i,k}} &= \begin{bmatrix} -0.130 & 0.100 & -0.025 & -0.019 & 0.103 & 0.057 \\ 2.768 & 1.010 & -0.034 & -0.021 & 0.019 & 0.104 \\ -0.006 & -0.004 & 0.001 & -0.002 & -0.013 & -0.007 \\ 4.954 & -0.059 & 0.096 & 0.934 & -0.204 & -0.123 \\ 0.002 & 0.001 & -0.001 & 0.000 & -0.002 & -0.001 \\ -0.706 & -0.094 & 0.209 & -0.079 & -0.462 & -0.184 \end{bmatrix} \\
 B = \frac{\partial f_{AS}(x_{sw,i}, \alpha)}{\partial \alpha} &= \begin{bmatrix} 0.811 & 0.004 & 0.002 & 0.011 & -0.075 & 0.001 & 0.000 \\ 3.418 & 0.018 & 0.004 & -0.005 & 0.106 & -0.094 & 0.020 \\ -0.002 & 0.000 & 0.001 & 0.000 & 0.013 & -0.001 & 0.003 \\ 4.922 & -0.003 & -0.020 & 0.012 & 0.126 & 0.014 & 0.067 \\ 0.003 & 0.000 & 0.000 & 0.000 & 1.001 & -0.001 & 0.000 \\ -0.555 & 0.008 & 0.002 & 0.106 & 0.606 & 0.971 & -0.002 \end{bmatrix}
 \end{aligned}$$

Finding the Q and R matrices usually involves trial-and-error until the values resulting in a stable AS gait over more than 30 steps. The values of the Q and R matrices, together with the values obtained for A and B for the AS described in Table 3.5 are shown in below.

$$K = \begin{bmatrix} 0.684 & 0.416 & -0.014 & -0.089 & 0.011 & 0.047 \\ -3.148 & 6.318 & -0.435 & -3.924 & 0.503 & 0.942 \\ -3.049 & 5.332 & -0.384 & -3.358 & 0.483 & 0.823 \\ -2.978 & -1.840 & 0.007 & 0.682 & 0.346 & -0.032 \\ 0.269 & 0.252 & -0.002 & -0.110 & -0.039 & 0.009 \\ -0.084 & 0.057 & 0.218 & -0.055 & -0.497 & -0.185 \\ 22.494 & -29.560 & 2.342 & 19.269 & -3.514 & -4.964 \end{bmatrix}$$

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix}, \quad R = 0.005 \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 \end{bmatrix}$$

It was found that the values corresponding to the differential of the state values in $x_{sw,i}$ (rows 1, 3 and 5 of matrix Q) needed to be much higher than the rest. For the last row of the matrix R, corresponding to the cost of the leg force, the value needed to be finely adjusted so that the gait does not exhibit a tall-short, double-step pattern similar to the one discussed in [20]. The matrix B is quite informative as to the behaviour of the AS mechanism. The forward velocity (row 2) and vertical velocity (row 4) at takeoff can be controlled very efficiently by a small change in the touchdown angle (column 1). The last column, representing the effect of the leg force, also affects these two states. The takeoff body angle and angular velocity can be directly controlled by adjusting $\theta_{B,st,f}^d$ and $\dot{\theta}_{B,st,f}^d$, respectively, since B(5,5) and B(6,6) are both very close to 1. However, the vertical and horizontal components of the body centre of mass cannot be controlled independently, so an LQR control is more suitable for this case than the method suggested in [20].

The overall control strategy employed for the AS can be summarized in Fig. 3.17. The low-level control is a PD continuous-time control which ensures that the desired trajectories are followed. The top-level control determines the desired trajectory parameters for the subsequent step based on the difference between the actual and desired takeoff states.

One of the advantages of the MbA methodology is that the controller is based on the simpler AS, so that the A and B matrices can be easily understood. This leads to a more transparent and intuitive control design.

3.4 Dimensional analysis stage

The goal of the Dimensional Analysis stage is to determine the parameters of the ES (the Linkage Leg) which provide the closest behaviour to that of the optimized AS (double pendulum and SLIP). To achieve this goal the concept of dynamic similarity as discussed in [70], is applied to the dimensionless Lagrangian equations of the ES and AS. If the similarity is achieved, it means that for any given state vector q_{LL} the potential and kinetic energies of the Linkage Leg system are equal to those of the AS in either motion phase. Because this similarity holds for any state space vector, it also means that the same change in state space vectors produces equal changes in the energies of the system. While similarity requires exact match between the two systems, in the dimensionless analysis step it was found that an approximate similarity (here called analogy) is sufficient to achieve the desired goals of the MbA methodology. The approximate conditions result in systems with close dynamic behaviours, as it will be shown in the results section.

The steps of the Dimensional Analysis stage are shown in Figure 3.18. First, the dimensionless Lagrangian equations of the mechanisms forming the AS are obtained, and the Pi-factors are calculated for the AS. Secondly, the Pi-factors are obtained from the dimensionless Lagrangian of the Linkage Leg. Finally, an optimization procedure is used to find the parameters of the Linkage Leg which result in the closest Pi-factors between the two systems. The subscript AS has been used in the following equations to differentiate the variables of the AS from those of the Linkage Leg, for which the subscript LL will be used.

Figure 3.18 Diagram of Dimensional Analysis stage

The dimensionless Lagrangian equation of the double pendulum leg attached to a robot body can be expressed as:

$$\mathcal{L}_{AS,sw} = \left(\frac{d(q_{AS,sw})}{dT_{AS}} \right)^T V_{AS,sw} \left(\frac{d(q_{AS,sw})}{dT_{AS}} \right) + G_{AS,sw} Q_{AS,sw} \quad (3.12)$$

The mass matrix $V_{AS,sw}$, pre- and post-multiplied by the time derivative of the state vector $q_{AS,sw}$, defines the kinetic energy of the system. The term $G_{AS,sw} Q_{AS,sw}$ defines the potential energy of the system, where $Q_{AS,sw}$ is a convenient combination of the cosine and sine of the elements of $q_{AS,sw}$. The $V_{AS,sw}$ and $G_{AS,sw}$ matrices are defined in the appendix. Notably, the time and energy units are changed to $T_{AS} = \sqrt{a_{AS,1}/g} t$ and $\mathcal{L}_{AS,sw} = L_{AS,sw} \left(I_{AS,2} + m_{AS,2} \left(y_{AS,2}^2 + (a_{AS,2} + x_{AS,2})^2 \right) \right) g/a_{AS,1}$, respectively, following the procedure outlined in [84], which allows the comparison of dimensionless equations of motion.

The Lagrangian equation for the SLIP model shown Figure 3.14 is:

$$\mathcal{L}_{AS,st} = \left(\frac{d(q_{AS,st})}{dT_{AS}} \right)^T V_{AS,st} \left(\frac{d(q_{AS,st})}{dT_{AS}} \right) + G_{AS,st} Q_{AS,st} + \frac{1}{2} (l_{st} - l_{st,i})^2 K_{AS} \quad (3.13)$$

where K_{AS} is the spring coefficient of the SLIP mechanism, and $l_{st,i}$ is the leg length at touchdown. The formulas for V_{AS} and G_{AS} matrices for both swing and stance phases are also included in the appendix at the end of the section.

The Lagrangian of the Linkage Leg during the swing phase is:

$$\mathcal{L}_{LL,sw} = \left(\frac{\delta(q_{LL,sw})}{\delta(T_{LL})} \right)^T V_{LL,sw} \left(\frac{\delta(q_{LL,sw})}{\delta(T_{LL})} \right) + G_{LL,sw} Q_{LL,sw} \quad (3.14)$$

The expressions for $V_{LL,sw}$ and $G_{LL,sw}$ can be found in the Appendix.

The stance phase adds two constraints to the dynamics of the Linkage Leg, namely that the vertical and horizontal positions of the toe be zero. To obtain the Lagrangian for the stance phase $q_{LL,sw}$ and $Q_{LL,sw}$ have to be converted to a new set of coordinates that incorporate these kinematic constraints. It is assumed that the toe does not slide on the ground, thus the contact can be represented as a revolute joint. This assumption greatly simplifies the dynamics of the stance phase, and it is validated in section 2.3. To facilitate the comparison between the Linkage Leg during the stance phase and the SLIP model, the degrees of freedom of the SLIP are chosen as the generalized coordinates to represent the stance Linkage Leg Lagrangian. As such, the matrices $V_{sw \rightarrow st}$ and $G_{sw \rightarrow st}$ are used to transform the two sets of degrees of freedom, and can be found in the appendix at the end of the section:

$$\begin{aligned} q_{LL,sw} &= V_{sw \rightarrow st} q_{LL,st} \\ Q_{LL,sw} &= G_{sw \rightarrow st} Q_{LL,st} \end{aligned} \quad (3.15)$$

Using these transformation matrices the new Lagrangian for the Linkage Leg during the stance phase is obtained:

$$\mathcal{L}_{LL,st} = \left(\frac{\delta(q_{LL,st})}{\delta(T_{LL,st})} \right)^T (V_{sw \rightarrow st})^T V_{LL,sw} (V_{sw \rightarrow st}) \left(\frac{\delta(q_{LL,st})}{\delta(T_{LL,st})} \right) + G_{LL,sw} G_{sw \rightarrow st} Q_{st} + L_K K_{LL} \quad (3.16)$$

where K_{LL} is the spring constant of the Linkage Leg, and L_K is the change in the length of the segment $O_1 O_5$ (seen in Fig. 5), which is a close approximation of the spring length.

The next step in the Dimensional Analysis stage is to obtain the Pi-factors from the dimensionless Lagrangians. This step is shown in [71] to compare double pendulums independently of size, and in [26] to compare the Linkage Leg and the double pendulum. The Pi-factors of the double pendulum system, the Pi-factors of the SLIP model and the Pi-factors of the Linkage Leg for each phase are shown in the appendix at the end of the section. The Pi-factors of the AS system mechanisms are independent of the state space vector (i.e. configuration of the robot) but the Pi-factors of the Linkage Leg are, however, dependent on the state vector, so they change for different knee angles θ_2 . As such, they are averaged for the range of θ_2 angles expected during the swing phase.

In order to achieve dynamic similarity between the Linkage Leg and the AS, the respective Pi-factors must be equal. In order to achieve physical analogy between the linkage leg and double pendulum, there should be a one-to-one correspondence between their Pi-factors. Therefore, those Pi-factors of the linkage leg that do not show in the dimensionless Lagrangian for the double pendulum must be zero, and the remaining Pi-factors must be equal to their double pendulum counterparts. Consequently, the following equalities are held for the Pi-factors:

$$\mathcal{M}_4 = 0, \mathcal{M}_5 = 0, \mathcal{M}_6 = 0, \mathcal{M}_7 = 0, \mathcal{M}_8 = 0, \mathcal{M}_9 = 0, \mathcal{M}_{14} = 0, \mathcal{M}_{15} = 0 \quad (3.17)$$

$$\mathcal{M}_M = \mathcal{N}_M, \mathcal{M}_1 = \mathcal{N}_1, \mathcal{M}_2 = \mathcal{M}_{10} = \mathcal{N}_2, \mathcal{M}_3 = \mathcal{M}_{11} = \mathcal{N}_3, \mathcal{M}_{12} = \mathcal{N}_4, \mathcal{M}_{13} = \mathcal{N}_5 \quad (3.18)$$

Note that if the linkage leg is parallel, the Pi-factors are constant, and all the equalities in Equation 3:17 are satisfied. Therefore, a parallel linkage leg that satisfies the equalities in Eq. (19) can behave exactly as a chosen double pendulum. For a generic linkage leg, however, the Pi-factors are not constant, but they vary with θ_2 . It was seen in section 3.1 that the Linkage Leg during stance is similar to a SLIP model.

The SLIP model optimized in section 3.2 has the prismatic joint aligned with the virtual leg (i.e. the line connecting the toe with the hip joint). A parallel Linkage Leg cannot approximate a SLIP model, because the direction of motion of the toe with respect to the hip, shown by the arrow in figure 3.19, is not aligned with the virtual leg. A Linkage Leg where the intersection of O_1O_2 and O_4O_5 is coincident with the ground plane when the virtual leg is vertical will have the same kinematics as a SLIP model. If the leg angle changes the alignment will be only approximate. As such, an optimization procedure is used to achieve the best balance between satisfying equations (3.17), (3.18) to achieve a close approximation to the double pendulum, and satisfying the dynamic and kinematic constraints imposed by the SLIP model.

The parameters to optimize are shown in Table A.1 in the Appendix, together with the range used and with the final optimized values. The geometry and the dynamic parameters of the leg is the same as the one used in section 3.1 bond graph stage. The lengths a_5 and a_6 are not used in the optimization, but they are instead calculated separately to ensure the point G in Figure 3.19 is on the ground level for the nominal leg length. A multigoal optimization using the *fgoalattain* algorithm in Matlab is used to achieve the optimal linkage leg shown in Table A.1. The goal was a vector formed by the equations (3.17), (3.18) and the difference between the respective Pi-factors of the SLIP model.

Figure 3.19 Parallel Linkage Leg (left) and SLIP-like Linkage Leg (right)

Figure 3.20 Gait of the optimally analogous Linkage Leg

Figure 3.21 Comparison between the behaviours of the optimal AS system and the ES (LL) optimally analogous to it.

The gait of the Linkage Leg optimally analogous to the AS obtained in section 3.2 is shown in Fig 3:20, and is compared with the gait of the optimized AS in Fig. 3.21. The desired trajectories and PD controllers for the controlled degrees of freedom (θ_L , θ_2 , and θ_B) are the same for the AS and the Linkage Leg. As it can be seen in Fig. 3:21, the behaviour of the two systems is reasonably

close, indicating that a good analogy is achieved by the dimensional analysis stage, and the optimal joint trajectories obtained for the AS can be directly applied to the Linkage Leg.

The ES is able to achieve a repeatable gait around the same Poincare point, optimal trajectories and low- and top-level controllers. The similarity in the dynamic behaviour of the Linkage Leg to that of the AS means that the control system developed based on the AS can be applied to a Linkage Leg without changes. This is an important result, as it indicates that the mechanical implementation of the leg can be different than the model used in the control system. The dynamic model can be optimized to make the control of the running gait simple and intuitive, and then the mechanical implementation of the leg can be optimized for simple construction, light weight, or other such criteria. Furthermore, the analogy between the mechanical implementation and control model can hold for multiple different models. This means that each phase of motion can have a different optimal model which follows a different behaviour, and the Dimensional Optimization procedure outlined is able to find a single mechanical system that achieves a behaviour optimally similar to each of these mechanisms.

3.5 MbA Linkage Leg conclusions

The goal outlined by the MbA methodology is achieved for the case study of the Linkage Leg. The Bond Graph stage was able to find much simpler mechanisms that have very similar behaviour to the Linkage Leg for each of the motion phases. The SLIP model is a good approximation during the stance phase, and the double pendulum is a good approximation during the swing phase. This simplification was achievable because the behaviour was considered only during the specific motion phase, so generality was not a required property of the analogy.

The mechanical parameters and the control strategy are simultaneously optimized in the subsequent Analogous System Optimization stage. The system is simulated as if the robot leg consists of a SLIP model during stance and of a double pendulum during the swing phase. Two transition functions switch the system between the two motion phases, making the touchdown and takeoff transitions simple to simulate and easy to understand intuitively, and are validated against a more realistic Simulink model with ground contact in chapter 2. The Analogous System is able to achieve a speed of 5m/s using less than 50Nm peak torque and showed a stable repeatable gait.

The goal of the third stage, the Dimensional Optimization stage is to achieve a mechanical design of the Linkage Leg such that it behaves like a SLIP model while on the ground as similar to a double pendulum while in the air. To achieve this goal the theory of dynamic similarity is

used. The Lagrangians of the SLIP model, double pendulum and Linkage Leg are expressed in dimensionless forms, and Pi-factors are calculated. If the corresponding Pi-factors of the Linkage Leg and each of the mechanisms forming the Analogous System are equal, then the Lagrangians will be equal independent of the system state, and as such will result in similar equations of motion.

The requirements imposed by the double pendulum and the SLIP analogies are not able to be simultaneously exactly achieved. An optimization procedure is used to find the Linkage Leg that results in the closest behaviour to both the simpler mechanisms, and the resulting Linkage Leg gait is similar to that of the Analogous System when the control strategy and controller gains obtained based on the AS are applied to both systems.

In the following chapter the Linkage Leg obtained using the MbA methodology is compared with the Linkage Leg obtained by a competing strategy, Reduction-by-Feedback.

4 Comparison between Mechatronics-by-Analogy and Reduction-by-Feedback

Mechatronics by Analogy is able to reduce the design space for a legged robot, and achieve a good performance when applied to the case study of the Linkage Leg presented in the previous section. Legged robots are a good test case for mechatronic design methodologies, as they require a close interaction between the mechanical design and the control strategy [21]. A lot of work has been done to find simple models that are able to represent the behaviour of walking machines, either robots, humans or animals [21], [27]. As such, methods that simplify the dynamic behaviour of real mechanisms are very useful in the design of legged robots. Mechatronics by Analogy was already shown in Chapter 3 to achieve this goal, and in this section it will be compared with a competing methodology, which uses a control layer to force the leg to have the desired dynamic behaviour. A top level control layer then controls the gait of the robot assuming it has the dynamics of the simpler system. This methodology is referred to as Reduction-by-Feedback (RbF) here and by its authors [18].

4.1 General description of Reduction-by-Feedback methodology

Reduction-by-feedback is a control methodology used that creates an embedded simpler mechanism which achieves the desired gait. This is achieved using kinematic couplings between the joints in the robot (dependent degrees of freedom) and the unactuated degrees of freedom (independent degrees of freedom). This effectively creates a desirable mechanism whose hybrid zero-dynamics have a desirable gait pattern and achieve passive running. Of course, actuator inputs are necessary to achieve the kinematic couplings, so the system is still closed loop. Once the mechanism with desirable hybrid zero dynamics is designed, the kinematic couplings between the degrees of freedom can be finely tuned to deal with outside disturbances, but the nature of the kinematic couplings should allow for better disturbance rejection than traditionally controlled joints [18]. There are two control loops to the reduction-by-feedback methodology: the inner-loop control, which achieves the desired kinematic couplings between the dependent degrees of freedom and the independent degrees of freedom, and the outer loop control, which changes the couplings depending on external disturbances.

The main benefits of the reduction-by-feedback approach is the ability to construct the kinematics of the zero dynamics mechanism any way the designer desires. This allows for an

intuitive understanding of the mechanism behaviour, and it allows the designer to build up the theoretical mechanism in stages. Furthermore, if an independent degree of freedom suffers an unexpected change of position or velocity due to an outside disturbance, the system reacts instantaneously because of the kinematic couplings between that independent degree of freedom and the dependent degrees of freedom of the robot.

The main issues with the reduction-by-feedback approach is that the designer has no way of defining the dynamics of the system. The mechanism is defined by the kinematic couplings between joints, and the designer cannot influence the dynamics directly, in an intuitive fashion.

4.2 Reduction-by-Feedback applied to the Linkage Leg

The Reduction-by-Feedback (RbF) method is applied to the Linkage Leg presented above, following the same steps as presented in [18], where it was applied the control of Thumper, a single legged running robot. The control in [18] is based on the idea of embedding a simple mechanism by designating virtual mechanical constraints between the controlled degrees of freedom and the uncontrolled degrees of freedom, and thus reduce the behaviour of Thumper to that of a simpler mechanism.

The RbF Linkage Leg is simulated just like the Linkage Leg used to test the Mechatronics by Analogy (MbA) methodology, presented in chapter 3 and in Figure 3.1. The superscript RbF was added to the parameters of the RbF Linkage Leg to differentiate them from the ones used previously. For the MbA Linkage Leg, the swing and stance phases are simulated independently, using the f_{sw} and f_{st} functions, shown in Chapter 2. The inputs to these functions are the current state of the robot and the joint torques, and the output is the derivative of the robot state. For the RbF methodology, two virtual constraints are imposed to the positions of the controlled degrees of freedom. During the swing phase $\theta_{1,sw} = \theta_1^{RbF}(x_0^{RbF})$, $\theta_{2,sw} = \theta_2^{RbF}(x_0^{RbF})$ where x_0^{RbF} is the distance traveled by the hip from the previous takeoff position, as seen in figure 4.1. For the stance phase $\theta_B = \theta_B^{RbF}(\theta_L^{RbF})$, where θ_L^{RbF} is the angle of virtual leg (line connecting the toe point and the hip joint) of the Linkage Leg, show in figure 4.1. The x_0^{RbF} and θ_L^{RbF} independent variables were chosen similarly to [18], as the robot presented there is also a single legged running robot.

The θ_1^{RbF} and θ_B^{RbF} functions are chosen to be fourth order Bezier function:

$$\theta_1^{RbF}(x_0^{RbF}) = \sum_{k=0}^3 \left(P_{\theta_1^{RbF}} \right)_k \frac{3!}{k!(3-k)!} \left(\frac{x_0^{RbF}}{x_{0,sw,f}^{RbF}} \right)^k \left(1 - \left(\frac{x_0^{RbF}}{x_{0,sw,f}^{RbF}} \right) \right)^{3-k}$$

$$\theta_B^{RbF}(\theta_L^{RbF}) = \sum_{k=0}^3 \left(P_{\theta_B^{RbF}} \right)_k \frac{3!}{k!(3-k)!} \left(\frac{\theta_L^{RbF} - \theta_{L,st,i}^{RbF}}{\theta_{L,st,f}^{RbF} - \theta_{L,st,i}^{RbF}} \right)^k \left(1 - \left(\frac{\theta_L^{RbF} - \theta_{L,st,i}^{RbF}}{\theta_{L,st,f}^{RbF} - \theta_{L,st,i}^{RbF}} \right) \right)^{3-k} \quad (4.1)$$

where $x_{0,sw,f}^{RbF}$ is the expected distance traveled by the hip during the swing phase, and it is based on the estimate of the time spent in the air and the speed at takeoff.

$$x_{0,sw,f}^{RbF} = \dot{x}_{0,sw,i}^{RbF} \underbrace{\left(\frac{2 \dot{y}_{0,sw,i}^{RbF}}{g} \right)}_{\text{estimate swing time}}$$

For the stance phase $\theta_{L,st,i}^{RbF}$ is the virtual leg angle at touchdown, and $\theta_{L,st,f}^{RbF}$ is the estimate of the virtual leg takeoff angle, based on the stiffness of the virtual leg, K_{vr} .

$$\theta_{L,st,f}^{RbF} = \dot{\theta}_{L,st,i}^{RbF} \underbrace{\left(\pi / \sqrt{\frac{K_{vr}}{m_B + m_1 + \frac{(m_2 + m_4)}{2}}} \right)}_{\text{estimate stance time}}$$

The vectors $P_{\theta_1^{RbF}}$ and $P_{\theta_B^{RbF}}$ are the Bezier parameters vectors [85]. For a fourth degree Bezier equation the initial position, initial velocity, final position and final velocity can be specified independently. As such, $P_{\theta_1^{RbF}} = [\theta_{1,sw,i} \quad \theta_{1,sw,i} + \dot{\theta}_{1,sw,i} \quad -\dot{\theta}_{1,sw,f}^d + \theta_{1,sw,f}^d \quad \theta_{1,sw,f}^d]$ and $P_{\theta_B^{RbF}} = [\theta_{B,st,i} \quad \theta_{B,st,i} + \dot{\theta}_{B,st,i} \quad -\dot{\theta}_{B,st,f}^d + \theta_{B,st,f}^d \quad \theta_{B,st,f}^d]$, where the superscript d indicates the desired values at the end of the respective phases of motion.

The knee constraint during the swing phase is a fifth order Bezier function, with the parameters $P_{\theta_2^{RbF}} = [\theta_{2,sw,i} \quad \theta_{2,sw,i} + \dot{\theta}_{2,sw,i} \quad \theta_{2,sw,inter}^d \quad -\dot{\theta}_{2,sw,f}^d + \theta_{2,sw,i} \quad \theta_{2,sw,i}]$:

$$\theta_2^{RbF}(x_0^{RbF}) = \sum_{k=0}^4 \left(P_{\theta_2^{RbF}} \right)_k \cdot \frac{4!}{k!(4-k)!} \left(\frac{x_0^{RbF}}{x_{0,sw,f}^{RbF}} \right)^k \left(1 - \left(\frac{x_0^{RbF}}{x_{0,sw,f}^{RbF}} \right) \right)^{4-k} \quad (4.2)$$

The extra parameter $\theta_{2,sw,inter}^d$ is necessary to give the knee the possibility to retract during swing, and to prevent the toe from impacting the ground. Note that the final position of the knee at the end of the swing is the same as the initial position at the beginning of the swing.

The virtual constraints $\theta_1^{RbF}(x_0^{RbF})$, $\theta_2^{RbF}(x_0^{RbF})$ and $\theta_B^{RbF}(\theta_L^{RbF})$ are thus defined by the desired values at the end of the respective phases of motion, and by the knee intermediate position, gathered in the vector α^{RbF} :

$$\alpha^{RbF} = [\theta_{1,sw,f}^d \quad \dot{\theta}_{1,sw,f}^d \quad \dot{\theta}_{2,sw,i}^d \quad \theta_{2,sw,inter}^d \quad \theta_{B,st,f}^d \quad \dot{\theta}_{B,st,f}^d \quad \tau_{\theta_2}]$$

The last parameter in α_{RbF} is the constant torque applied to the knee joint during the extension part of the stance phase, similar to the knee torque applied in Chapter 3. The construction of the

Linkage Leg, described in Chapter 2, is not able to control the position of the knee during the stance phase, but only apply a constant torque.

Given the virtual constraints the swing and stance functions for the RbF Linkage Leg become

$$\begin{aligned}\dot{x}_{sw}^{RbF} &= f_{sw}^{RbF}(x_0^{RbF}, \alpha^{RbF}, x_{sw,i}) \\ \dot{x}_{st}^{RbF} &= f_{st}^{RbF}(\theta_L^{RbF}, \alpha^{RbF}, x_{st,i})\end{aligned}$$

where $x_{sw,i}$ is the initial state of the system of the swing phase, and $x_{st,i}$ is the initial state for the stance system.

The transformations $\Delta_{st \rightarrow sw}$ and $\Delta_{sw \rightarrow st}$ between the swing and stance phases are similar to the ones used in the MbA Linkage Leg simulation and are shown in below at the end of this chapter.

RbF states that an embedded mechanism, defined by the holonomic constraint equations (4.1) and (4.2) is simpler than the original Linkage Leg and it is defined by the parameters α^{RbF} [18]. These parameters need to be chosen such that the embedded mechanism results in a gait that is desirable. To obtain a good comparison with the Linkage Leg obtained in chapter 3 the same optimization procedure is used. The constraints and cost function are show in Table 3.6. The optimization variables are shown in Table 4.1 at the end of this section, with the initial and final optimized values. The initial Linkage Leg parameters are the same ones from Table 3.3, which were used to obtain the bond graphs shown in section 3.1.1. This ensures that the comparison between the MbA and the RbF is correct, as both methods start from the same initial Linkage Leg.

Once the constraint parameters α^{RbF} are optimized the second step of the RbF methodology designs an outer control system that tunes the parameters α^{RbF} based on the error in each step. This is necessary because the holonomic constraints that form the embedded mechanism are able to achieve the desired gait for a specific $x_{sw,i}$ starting configuration. If the robot deviates from the specific starting configuration the embedded mechanism will not be able to achieve the desired step unless the embedded mechanism is modified to account for it.

The method chosen in [18], as well as in section 3.2.1, is to design a Linear Quadratic Regulator (LQR) which changes the parameters α^{RbF} dependent on the initial state of the system at the beginning of each step. The procedure used to determine the LQR gain K_{LQR}^{RbF} for the RbF Linkage Leg is very similar to the one previously described for the Analogous System in 3.2. It involves calculating the Jacobian of the system equation with respect the control parameter vector α^{RbF} and to the initial state vector $x_{sw,i}$.

Figure 4.1 Reduction-by-Feedback Linkage Leg

The resulting gait of the RbF Linkage Leg is show for three steps in Figure 4.2. The gait obtained is similar to the one obtained using the RbF method in [18]. Figure 4.3 shows the behaviour of the Linkage Leg starting from the first step. The first three rows show the horizontal position of the hip joint, vertical position of the hip joint, body angle with respect to the ground in

the left column, and their derivatives in the right side column. The bottom row shows the torques of the hip (left) and the knee joint (right). It can be seen that the top level control adjusts the α^{RbF} constraint parameters to account for the errors in the initial optimization, likely due to the constraints in table 3.6 not being perfectly satisfied.

Figure 4.4 shows the virtual kinematic constraints (Equations 4.1 and 4.2) for the gait obtained using RbF methodology. As it can be seen, the hip and knee torques are able to keep the virtual constraints satisfied. The constraint $\theta_B^{RbF}(\theta_L^{RbF})$ has to be adjusted in the range of $\theta_L^{RbF} \in [82^\circ, 85^\circ]$ because this is beyond the expected maximum virtual leg angle $\theta_{L,st,f}^{RbF}$, likely due to the stance time being longer than the estimate.

The gait velocity is limited by the maximum torque during the stance phase. A faster gait generally results in a shorter stance phase, as shown in [20] and [53], which means that the stance phase virtual constraint experiences a higher virtual reaction torque (that must be supplied by the hip actuator). The constraints in table 3.6 come into effect for torques higher than 50Nm meaning the optimization must reduce the top speed to keep the torque from becoming too high. This is especially significant for the hip torque, which must rotate the robot body, the part with the highest moment of inertia) back to its takeoff position. Furthermore, the optimized RbF Linkage Leg does not exhibit the uncoupled property of the MbA Linkage Leg, explained in Chapter 3. This means that the force exerted by the leg on the body is not aligned with the hip – toe virtual leg, and thus creates a torque around the hip. This torque must be counteracted by the hip actuator torque. Higher speeds generate higher forces in the leg [53], hence higher torque that must be counteracted, and

Figure 4.2 Reduction-by-Feedback Linkage Leg gait

Figure 4.3 Reduction-by-Feedback Linkage Leg hip trajectory and joint torques

the optimization algorithm used (*fmincon* in Matlab) was not able to find a decoupled leg architecture.

Figure 4.4 Reduction-by-Feedback Linkage Leg virtual joint constraints

4.3 Comparison between MbA Linkage Leg and RbF Linkage Leg

The Linkage Leg obtained using the RbF methodology is compared with the Linkage Leg obtained using the MbA methodology for three steady state steps, and it is shown in Figure 4.5. The three steps of the different legs have different durations, and have been aligned to start at the zero time mark. As it can be seen, MbA results in a higher speed and a lower torque, both during the swing and the stance phases. The optimization method used to obtain the RbF and MbA Linkage Legs was able to obtain a faster gait for the simpler Analogous System used in the MbA than for the more complex Linkage Leg mechanism and the RbF embedded mechanism. Specifically, the higher hip torque required by the uncoupled RbF linkage leg limits its achievable top speed. The Knee torque is also much higher for the RbF Linkage Leg, indicating that the useful retraction mechanisms presented by double pendulum dynamics are not present to help keep the torque low. The much lower torque requirements during stance phase of the MbA Linkage Leg are explained by the presence of the user-defined constraint that the Linkage Leg is decoupled, that is,

Figure 4.5 MbA Linkage Leg compared with RbF Linkage Leg

the force due to the leg spring does not generate a torque at the hip, as explained in .Figure 3.19. The decoupling property has shown beneficial properties for other SLIP-like robots [20]. The RbF optimization does not have a mechanism to insert such constraints, and is not able to find a decoupled configuration. This results in the hip joint requiring a high torque to counteract the moment due to the force of the leg spring. In conclusion, the ability of the MbA methodology to extract meaningful and transparent mechanisms from the Linkage Leg concept, and then give the designer the ability to refine the design based on prior knowledge about these simpler mechanisms resulted in a lower torque requirement and thus higher possible speeds.

4.3.1 Disturbance rejection comparison between MbA and RbF Linkage Legs

One of the main advantages of the RbF methodology is that its embedded mechanism relates the independent degrees of freedom to the controlled degrees of freedom using virtual constraints. This causes an instantaneous change in the velocity of the controlled degree of freedom (to the limits of the actuator torque) when the independent degree of freedom experiences an instantaneous change in velocity, such as due to an external disturbance. The trajectories chosen for the MbA methodology are time dependent, and as such will follow the pre-programmed desired trajectory independent of the external disturbances. Once the step is complete, the trajectories are recalculated based on the disturbances encountered, as explained in section 3.2.1. However, some disturbances might be too great to allow for the step to be completed, and the gait will fail.

The MbA and RbF Linkage Legs are compared below for a terrain disturbance that simulates a step 0.1m on the ground. The disturbance happens at touchdown, so both robots must deal with the stance phase before they have a chance to adjust their gaits at takeoff. The gaits of the RbF and MbA Linkage Legs show that the robots take a few steps to recover from the disturbance. The RbF robot shows less deviation from the original gait pattern, especially for the angle of the body θ_B and height of the hip joint y_0 , but both robots take around 5 steps to return close to their original gait. This indicates that the RbF is to some extent better than the MbA at dealing with disturbances, as expected.

In conclusion, the Reduction-by-Feedback approach was applied to the design of a Linkage Leg similar to the one obtained using the MbA methodology. The RbF Linkage Leg displayed slightly less deviation from its optimal gait than the MbA Linkage Leg if disturbed, but at the cost of a greatly reduced performance. The MbA Linkage Leg was able to achieve 5m/s using less than

50Nm of maximum actuator torque, whereas the RbF Linkage Leg achieved 2.3m/s using a maximum of 150Nm of torque.

Figure 4.6 MbA Linkage Leg compared with RbF Linkage Leg for a 0.1m step disturbance

Table 4.1 Reduction-by-Feedback optimization variables

Physical Variables					
Body Parameters					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
m_B^{RbF}	kg	10.00	8.00	12.00	9.9713
I_B^{RbF}	kg m ²	0.83	0.67	1.00	0.8247
x_B^{RbF}	m	0.00	-0.50	0.50	0.004
y_B^{RbF}	m	0.00	-0.10	0.50	-0.001
Linkage Leg Parameters					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
m_1^{RbF}	kg	0.500	0.300	1.000	0.563
I_1^{RbF}	kg m ²	0.011	0.003	0.040	0.017
x_1^{RbF}	m	-0.250	-0.500	0.000	-0.142
y_1^{RbF}	m	0.000	-0.100	0.100	0.012
m_2^{RbF}	kg	0.500	0.300	1.000	0.500
I_2^{RbF}	kg m ²	0.011	0.003	0.040	0.010
x_2^{RbF}	m	0.000	-0.500	0.000	-0.140
y_2^{RbF}	m	0.000	-0.100	0.100	-0.030
m_3^{RbF}	kg	0.500	0.300	1.000	0.559
I_3^{RbF}	kg m ²	0.011	0.003	0.040	0.017
x_3^{RbF}	m	-0.250	-0.500	0.000	-0.280
y_3^{RbF}	m	0.000	-0.100	0.100	-0.060
m_4^{RbF}	kg	0.500	0.100	1.000	0.125
I_4^{RbF}	kg m ²	0.011	0.003	0.040	0.002
x_4^{RbF}	m	0.000	-0.500	0.000	-0.223
y_4^{RbF}	m	0.000	-0.100	0.100	-0.002
a_1^{RbF}	m	0.500	0.300	0.700	0.396
a_2^{RbF}	m	0.500	0.300	0.700	0.389
a_3^{RbF}	m	0.500	0.300	0.700	0.509
a_4^{RbF}	m	0.200	0.100	0.300	0.201
a_5^{RbF}	m	0.500	0.300	0.700	0.473
a_6^{RbF}	m	0.300	0.200	0.400	0.313
ϕ_1^{RbF}	rad	0.000	-0.785	0.785	0.047
ϕ_3^{RbF}	rad	3.142	2.618	3.665	3.134
K_{spring}	kN/mm	0.2	0.3	0.9	0.375

Desired Trajectory Parameters (α^{RbF})					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
$\theta_{1,sw,f}^d$	deg	355	270	360	106
$\dot{\theta}_{1,sw,f}^d$	deg/s	-121	-200	-20	-202
$\dot{\theta}_{2,sw,f}^d$	deg/s	-160	-360	0	-3
$\theta_{2,inter}^d$	deg	200	100	300	202
$\theta_{B,st,f}^d$	deg	0	-90	90	-1
$\dot{\theta}_{B,st,f}^d$	deg/s	15	-120	120	6
τ_{θ_2}	Nm	10	-10	30	10
Initialization state ($x_{sw,i}$)					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
$\theta_{1,sw,i}^d$	deg	295	250	320	74
$\dot{\theta}_{1,sw,i}^d$	deg/s	-250	-350	-100	-238
$\theta_{2,sw,i}^d$	deg	260	240	280	74
$\dot{\theta}_{2,sw,i}^d$	deg/s	-340	200	460	-238
$\theta_{B,st,i}$	deg	0	-90	90	4
$\dot{\theta}_{B,st,i}$	deg/s	15	-120	120	62
Control Parameters					
K^P	Nm/deg	500	250	2500	216
K^D	Nm s/deg	500	250	2500	260

5 Conclusion and further directions

Mechatronics by Analogy is a concurrent design methodology that replaces the challenging task of optimizing a complex system, including all the kinematic, dynamic and control parameters, with the tasks of first finding simpler models that can represent the complex system for all motion phases, then optimizing the simpler models concurrently with their control system, and finally finding the parameters of complex system so that it can behave analogously to the simpler models.

Simple leg mechanisms have been studied in the field of legged locomotion, and each offers significant advantages in different circumstances and motion phases. A number of methods are proposed in the literature that are able to make a robot leg mechanism appear to the controller as a simpler system, and the most relevant ones are discussed in Chapter 1. Unlike existing methods, the presented methodology addresses the problem of designing robotic legs from the direction of tuning the natural dynamics and kinematics of the mechanism, such that it behaves similar to the desired simpler models without the need for additional control layers.

The MbA methodology was applied to the design case of a robotic leg called Linkage Leg and consisting of four segments connected by revolute joints. The presented leg design, described in Chapter 2 has a number of potential advantages relative to existing mechanisms, such as lower mass due to the absence of a prismatic joint, desirable natural dynamics that help the control system achieve the desired gait, and the ability to precisely specify the desired kinematics of the toe point relative to the body of the robot.

The MbA methodology is not only applicable to legged robots, but is also useful in any mechatronic field where there is a need to design mechanisms with intuitive and simple dynamics. The methodology consists of 3 stages:

First a method based on bond graphs was presented that finds simpler mechanisms with a dynamic behaviour similar to that of a more complicated robotic leg mechanism for each specific motion phase. Such simpler mechanisms will be good representatives of the original mechanism while having the advantages of simplicity and tractability. The original leg mechanism was first modeled using a bond graph approach, and then a two-step simplification process was used to determine simpler mechanisms with the same significant dynamic characteristics. The first step found the dynamic elements that are necessary for the system behaviour to remain unchanged for each motion phase, and the second step simplified the bond graph formed by such elements and constructs the resulting mechanism expressed by the simplified graph.

For the case study of a Linkage Leg two motion phases were analyzed, i.e., swing and stance. During the swing phase, two masses elements were found to be required for generating the same behaviour as the original system, resulting in a double pendulum mechanism. And, for the stance phase the resulting simplified mechanism was found to be the classic SLIP model that has been used for modeling robotic and biologic leg dynamics.

Having such simpler mechanisms enables the designer to fine-tune the kinematic, dynamic and control parameters of the original mechanism in a more intuitive and effective way. This is shown in the second part of the MbA methodology, the Analogous System optimization phase. The SLIP model and the double pendulum model, together with controllers derived from their dynamic characteristics, were concurrently optimized. The result was satisfactory, achieving a fast running locomotion without very high torque requirements in the actuators.

The controller design and the mechanism design were greatly simplified because each was considered only during their phase of the motion. Rules for transitioning between the motion phases were proposed, and their behaviour was compared against a more realistic Simulink model of the leg that uses a ground interaction model. It was found that the simple and intuitive transition functions closely predict the contact between the robot and the ground.

In the third stage of MbA, the Dimensional Analysis stage, a Linkage Leg mechanism is designed to mimic the dynamics of the SLIP model during the stance phase and the dynamics of the double pendulum during the swing phase. A design approach able to quantify the degree of similarity is also introduced, based on the dynamic analogy between dimensionless Lagrangians. The geometric and dynamic parameters of the Linkage Leg were optimized such that a set of cost functions was minimized to find the best physical analogy.

It is shown that the proposed Linkage Leg can be physically analogous to both the SLIP and the double pendulum models. Consequently, controllers developed for the two simpler mechanisms were used to control the Linkage Leg and these controllers resulted in equivalent performance between the systems. The control torque and actuator power required to achieve the desired behaviour were similar to those required by a robot equipped with the virtual hybrid SLIP and double pendulum leg.

To understand the effectiveness of the MbA approach, it was compared with a competing approach, Reduction-by-Feedback (RbF). This approach designs virtual kinematic constraints

between the controlled degrees of freedom and the uncontrolled degrees of freedom, reducing the complexity of the system.

A second Linkage Leg was designed using the RbF methodology. And was compared with the one obtained using the MbA methodology. The RbF methodology is designed to achieve a behaviour robust to external disturbances, and this effect was indeed observed. It took the RbF Linkage Leg less steps to recover after stepping on an obstacle, compared with the MbA Linkage Leg. However, the performance achieved by the MbA Linkage Leg was much better than the performance of the RbF Linkage Leg, in terms of maximum speed, indicating that the simpler Analogous System was easier to optimize than the original Linkage Leg together with the virtual constraints. Moreover, the MbA methodology offers significant advantages in shifting the complexity of the control task from the online control to the offline design of the leg.

The MbA methodology relies on generating dimensionless Lagrangians and obtaining the Pi-factors, which is a tedious process. A future advance of the MbA methodology is to develop an automated method to obtain the Pi-factors, and to compare the dynamics of different systems.

The ability of the MbA methodology to find simpler mechanisms embedded in existing complex systems and then design a mechanism that achieves the same dynamics offers another intriguing avenue of research when applied to systems where desired dynamics are already available, such as prosthetics. In this application, the dynamics of an existing leg would be analyzed and an Analogous System obtained. The optimization step would be overlooked because the existing leg cannot be optimized, and the Dimensional Analysis stage would be used to generate a robotic leg that has very close dynamics to the existing template.

References

- [1] J. Hewit, "Mechatronics design - the key to performance enhancement," *Robot Auton Syst*, no. 19, pp. 135-42, 1996.
- [2] O. Chocron, "Evolutionary design on modular robotic arms," *Robotica*, no. no. 26, pp. 323-30, 2008.
- [3] K. JY., "Task based kinematic design of a two dof manipulator with a parallelogram five-bar link mechanism," *Mechatronics*, no. no 16, pp. 323-9, 2006.
- [4] Z. Affi, B. El-Keribi and L. Romdhane, "Advanced mechatronic design using a multi-objective genetic algorithm optimization of a motor-driven four-bar system," *Mechatronics*, vol. no 17, pp. 489-500, 2007.
- [5] Z. Bi and W. Zhang, "Concurrent optimal design of modular robotic configuration," *Journal of robotic systems*, vol. 18, no. 2, pp. 77-87, 2001.
- [6] L. Gamage, C. de Silva and R. Campos, "Design evolution of mechatronic systems through modeling, on-line monitoring, and evolutionary optimization," *Mechatronics*, vol. 22, pp. 83-94, 2012.
- [7] C. Paredis, "An agent-based approach to the design of rapidly deployable fault tolerant manipulators, PhD Thesis," Departament of Electrical and Computer Engineering, Carnegie Mellon Univeristy, Pittsburgh, 1996.
- [8] A. K. Dhingra, S. Rao and H. Miura, "Multi-objective decision making in a fuzzy environment with application to helicopter design," *AIAA Journal*, vol. 4, no. 28, pp. 703-10, 1990.
- [9] V. C. Moulianitis, N. A. Aspagathos and A. J. Dentsoras, "An application to mechatronic design of robot gripper," *Mechatronics*, vol. 14, no. 6, pp. 599-622, 2004.
- [10] S. Begbahani and C. de Silva, "Mechatronic design quotient as the basis of the new multicriteria mechatronic design methodology," *EEE/ASME Transactions on Mechatronic*, vol. 12, no. 2, pp. 227-232, 2007.

- [11] R. Chhabra and M. R. Emami, "A holistic concurrent design approach to robotics using hardware-in-the-loop simulation," *Mechatronics*, vol. 23, no. 3, pp. 335-345, 2013.
- [12] Q. Li and F. X. Wu, "Control performance improvement of a parallel robot via the design for control approach," *Mechatronics*, vol. 14, no. 8, pp. 947-964, October 2004.
- [13] S. Lohmeier, T. Buschmann and H. Ulbrich, "System design and control of anthropomorphic walking robot LOLA," *IEEE/ASME Transactions on Mechantronics*, vol. 14, no. 6, pp. 658-666, 2009.
- [14] K. N. Otto and E. K. Antonsson, "Imprecision in engineering desgin," *ASME J Mech Des*, no. 117, pp. 25-32, 1995.
- [15] K. L. Wood and E. K. Antonsson, "Engineering design calculations with fuzzy parameters," *Fuzzy Set Syst*, no. 52, pp. 1-20, 1992.
- [16] J. E. Pratt, C.-M. Chew, A. Torres, P. Dilworth and G. Pratt, "Virtual Model Control: An Intuitive Approach for Bipedal Locomotion.," *International Journal of Robotics Research*, vol. 20, no. 2, pp. 129-143, February 2001.
- [17] J. Pratt, P. Dilworth and G. Pratt, " Virtual model control of a bipedal walking robot," in *Proceedings of International Conference on Robotics and Automation*, Albuquerque, NM, USA, 1997.
- [18] I. Poulakakis, " Stabilizing Monopedal Robot Running: Reduction-by-Feedback and Compliant Hybrid Zero Dynamics," Ann Arbor, MI, U.S.A, 2009.
- [19] S. Collins, A. Ruina, R. Tedrake and M. Wisse, "Efficient Bipedal Robots Based on Passive Dynamic Walkers," *Science Magazine*, pp. 1082-1085, 18 February 2005.
- [20] M. Raibert, "Legged Robots," *Communications of the ACM*, pp. 499-514, 1986.
- [21] H. Witte, H. Hoffmann and R. Hackert, "Biomimetic robotics should be based on functional morphology," *Journal of Anatomy*, pp. 331-342, 2004.
- [22] S.-H. Hyon, "Development of a Biologically Inspired Robot "Kenken"," in *Proceedings 2002 IEEE International Conference on Robotics and Automation*, Washington DC, USA, 2002.

- [23] V. Ragusila and M. R. Emami, "A mechatronics approach to legged locomotion," in *Advanced Intelligent Mechatronics* , Montreal, 2010.
- [24] V. Ragusila and R. M. Emami, "Modelling of a robotic leg using bond graphs," *Simulation Modelling Practice and Theory*, vol. 40, pp. 132-143, 2014.
- [25] V. Ragusila and M. R. Emami, "Mechatronics by Analogy and Application to Legged Locomotion," *Mechatronics*, submitted 2015.
- [26] V. Ragusila and M. R. Emami, "A novel robotic leg design with hybrid dynamics," *Advanced Robotics*, vol. 27, no. 12, pp. 919-931, 2013.
- [27] M. F. Silva, "A historical perspective of legged robots," *Journal of Vibration and Control*, pp. 1447-1486, 2007.
- [28] D. Hobbelen and M. Wisse, "Limit Cycle Walking," in *Humanoid Robots, Human-like Machines* , Vienna Austria, Itech, 2007, p. 277.
- [29] S. Behnke, J. Muller and M. Schreiber, "Toni: A Soccer Playing Humanoid Robot," in *RoboCup 2005: Robot Soccer World Cup IX*, Berlin, Springer, 2005, pp. 59-70.
- [30] K. Hirai, M. Hirose, Y. Haikawa and T. Takenaka, "The development of Honda Humanoid Robot," in *IEEE International Conference on Robotics and Automation* , 1998.
- [31] T. Ishida, "Development of a small biped entertainment robot QRIO," in *Proceedings of the 2004 IEEE International Symposium on Micro-Nanomechantronics and Human Science*, 2004.
- [32] K. Kaneko, F. Kanehiro, M. Morisawa, K. Akachi, G. Miyamori, A. Hayashi and N. Kanehira, "Humanoid robot HRP-4 — Humanoid robotics platform with lightweight and slim body," in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2011)*, San Francisco, 2011.
- [33] M. Vukobratovic and B. Borovac, "Zero-moment point - Thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 1, pp. 157-173, 2004.

- [34] Y. Ogura and K. Shimomura, "Human-like walking with knee stretched, heel-contact and toe-off motion by a humanoid robot," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing China, 2006.
- [35] C. Thompson and R. Marc, "Passive Dynamic Running," in *The First International Symposium on Experimental Robotics I*, London U.K., 1989.
- [36] T. McGeer, "Passive walking with knees," in *1990 IEEE International Conference on Robotics and Automation*, Cincinnati USA, 1990.
- [37] M. Wisse, A. Schwab and F. van der Helm, "Passive Dynamic Walking Model with Upper Body," *Robotica*, pp. 681-688, 2004.
- [38] D. Hobbelen and M. Wisse, "Ankle Actuation for Limit Cycle Walkers," *International Journal of Robotics Research*, pp. 709-735, 2008.
- [39] M. Wisse, A. Schwab and R. van der Linde, "How to keep from falling forward: elementary swing leg action for passive dynamic walkers," *IEEE Transactions on Robotics*, pp. 393-401, 2005.
- [40] P. Manoonpong, T. Geng and B. Porr, "The RunBot architecture for adaptive, fast, dynamic walking," in *2007 IEEE International Symposium on Circuits and Systems*, New Orleans, 2007.
- [41] S.-H. Hyon, "Energy-preserving control of a passive one-legged running," *Advanced robotics*, pp. 357-81, 2004.
- [42] J. Pratt and R. Tedrake, *Velocity-Based Stability Margins for Fast Bipedal Walking*, Heidelberg, 2005.
- [43] D. Wight, *A Foot Placement Strategy for Robust Bipedal Gait Control*, Waterloo: University of Waterloo, 2008.
- [44] R. Johnston and A. Bekoff, "Energetics of bipedal running. II. Limb design and running mechanics," *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, pp. 169-184, 1996.
- [45] A. Biewener, "Patterns of Mechanical Energy Change in Tetrapod: pendula, springs and work," *Journal of Experimental Zoology Part A: Comparative Experimental Biology*, pp. 899-911, 2006.

- [46] M. Raibert, "Running with symmetry," *International Journal of Robotics Research*, pp. 3-19, 1986.
- [47] S. A. and M. Buehler, "A planar hopping robot with one actuator: design, simulation, and experimental results," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan, 2004.
- [48] A. Seyfarth and J. Rummel, "Stable running with segmented legs," *International Journal of Robotics Research*, pp. 919-934, 2008.
- [49] A. Seyfarth, H. Geyer and R. Blickhan, "Running and walking with compliant legs," in *Fast Motions in Biomechanics and Robotics*, Berlin, Germany, Springer, 2006, pp. 282-401.
- [50] B. Brown and G. Zeqlin, "The bow leg hopping robot," in *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, 1998.
- [51] U. Saranli, M. Buehler and D. E. Koditschek, "RHex: A Simple and Highly Mobile Hexapod Robot," *The International Journal of Robotics Research*, vol. 20, no. 7, pp. 616-631, 2001.
- [52] A. Ramezani, J. W. Hurst, K. A. Hamed and J. W. Grizzle, "Performance Analysis and Feedback Control of ATRIAS, A Three-Dimensional Bipedal Robot," *J. Dyn. Sys., Meas., Control*, vol. 136, no. 2, 2013.
- [53] J. Koechling and M. Raibert, "How fast can a legged robot run," *American Society of Mechanical Engineers, Dynamic Systems and Control Division (Publication) DSC*, vol. 11, no. 1988, pp. 241-249, November 1988.
- [54] M. Ahmadi and M. Buehler, "Stable control of a simulated one-legged running robot with hip and leg compliance," *IEEE Transactions on Robotics and Automation*, pp. 96-104, 1997.
- [55] M. Ahmadi and M. Buehler, "Controlled passive dynamic running experiments with the ARL-monopod II," *IEEE Transactions on Robotics*, pp. 974-86, 2006.
- [56] A. Sayyad, B. Seth and P. Seshu, "Single-legged hopping robotics research-A review," *Robotica*, vol. 25, pp. 587-613, 2007.

- [57] A. Kuo, "energetics of actively powered locomotion using the simplest walking model," *Journal of biomechanical engineering*, pp. 113-120, 2002.
- [58] J. Pratt, *Exploiting Inherent Robustness and Natural Dynamics in*, MIT, 2000.
- [59] G. Lichtwark, "A catapult action for rapid limb protraction," *Nature*, pp. 35-36, 2003.
- [60] J. Denavit and R. S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," in *American Society of Mechanical Engineers -- Meeting A-34*, New York, NY, United States, 1954.
- [61] J. Pratt, *"Exploiting Inherent Robustness and Natural Dynamics in the Control of Bipedal Walking Robots,"*, MIT.
- [62] M. Wisse and R. Q. v. d. Linde, Delft Pneumatic Biped, B. Siciliano, O. Khatib and F. Groen, Eds., Berlin: Springer-Verlag, 2007.
- [63] D. Wight, *"A Foot Placement Strategy for Robust Bipedal Gait Control,"*, Waterloo: University of Waterloo.
- [64] A. Goswami, "Kinematic and dynamic analogies between planar biped robots and the reaction mass pendulum (RMP) model," in *Humanoids 2008*, Daejeon, Korea, 2008.
- [65] D. C. Karnopp, D. L. Margolis and C. Rosenberg, *System dynamics : modeling and simulation of mechatronic systems*, 4th ed. ed., Hoboken: John Wiley & Sons, 2006.
- [66] T. Ersal, F. K. Hosam and J. L. Stein, "Realization-Preserving Structure and Order Reduction of Nonlinear Energetic System Models Using Energy Trajectory Correlations," *Journal of Dyanmic Systems, Measurement and Control*, vol. 131, no. 3, p. 031004(8 pp.), May 2009.
- [67] Y. H. Hung, C. H. Wu and C. W. Hong, "An element-oriented model simplification algorithm based on dynamic similarity," *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering*, pp. 56-69, 2012.

- [68] C. Chevallereau, E. R. Westervelt and J. W. Grizzle, "Asymptotically stable running for a five-link, four actuator, planar biped robot," *Int. J. Rob. Res.*, no. 24, pp. 431-464, 2005.
- [69] K. Mombaur, "Open-loop stable solutions of periodic optimal control problems," *Zeitschrift für Angewandte Mathematik und Mechanik*, pp. 499-515, 2005.
- [70] A. A. Gukhman, Introduction to the theory of similarity, R. D. Cess, Ed., New York: Academic Press, 1965.
- [71] A. Ohlhoff and P. H. Richter, "Forces in the double pendulum," *Journal of Applied Mathematics and Mechanics*, vol. 80, no. 8, pp. 517-534, July 2000.
- [72] H. M. Paynter, Analysis and Design of Engineering Systems, Cambridge: M.I.T. Press, 1961.
- [73] W. Borutzky, "Bond Graph Modelling and Simulation of Mechatronic Systems: an Introduction into the Methodology," in *20th European Conference on Modelling and Simulation ECMS 2006. Modelling Methodologies and Simulation Key Technologies in Academia and Industry*, Sankt Augustin, 2006.
- [74] M. J. Tierneho and A. M. Bos, "Modeling the Dynamics and Kinematics of Mechanical Systems with Multibond Graphs," *Journal of the Franklin Institute*, vol. 319, no. 1-2, pp. 37-50, Jan-Feb 1985.
- [75] J. Jang and C. Han, "Proposition of a Modeling Method for Constrained Mechanical Systems Based on the Vector Bond Graph," *Journal of the Franklin Institute*, vol. 335B, no. 3, pp. 451-69, April 1998.
- [76] P. J. Gawthrop, "Bond Graphs: A Representation for Mechatronic Systems," *Mechatronics*, vol. 1, no. 2, pp. 127-56, 1991.
- [77] R. Fotsu-Ngwompo, S. Scavarda and D. Thomasset, "Bond Graph Methodology for the Design of an Actuating System: Application to a Two-link Manipulator," in *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, Orlando, 1997.
- [78] J. van Amerongen, "Mechatronic Design," *Mechatronics*, vol. 13, no. 10, pp. 1045-66, Dec. 2003.

- [79] T. Ersal, H. K. Fathy and J. L. Stein, "Structural simplification of modular bond-graph models based on junction inactivity," *Simulation Modelling Practice and Theory*, vol. 17, no. 1, pp. 175-96, Jan. 2009.
- [80] D. G. Rideout, J. L. Stein and L. S. Louca, "Systematic Identification of decoupling in dynamic system models," *Transactions of the ASME. Journal of Dynamic Systems, Measurement and Control*, pp. 503-513, 2007.
- [81] W. L. Cleghorn, *Mechanics of Machines*, 1 ed., Oxford University Press, 2005.
- [82] H. Ashrafiuon and R. S. Erwin, "Sliding control approach to underactuated multibody systems," in *American Control Conference, Proceedings of the.*, Boston, 2004.
- [83] H. Kwakernaak and R. Sivan, *Linear Optimal Control Systems*, Chichester, UK: Wiley Interscience, 1972.
- [84] A. Ohlhoff and P. H. Richter, "Forces in the Double Pendulum," *Zeitschrift fur Angewandte Mathematik und Mechanik*, vol. 80, no. 8, pp. 517-34, 2000.
- [85] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. -H. Choi and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*, CRC Press, 2007.

Appendix: Dimensional Analysis

Table A.1 Variables for dimensional analysis optimization

Physical Variables					
Body Parameters					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
m_B	kg	Same as of the m_B of the optimized AS			9.827
I_B	kg m ²	Same as of the I_B of the optimized AS			0.8007
x_B	m	Same as of the x_B of the optimized AS			0.005
y_B	m	Same as of the y_B of the optimized AS			-0.002
Linkage Leg Parameters					
variable	units	Initial value	Lower bound	Upper bound	Optimized value
m_1	kg	0.2500	0.150	0.500	0.249
I_1	kg m ²	0.005	0.002	0.020	0.001
x_1	m	-0.250	-0.500	0.000	-0.216
y_1	m	0.000	-0.100	0.100	-0.015
m_2	kg	0.2500	0.150	0.500	0.150
I_2	kg m ²	0.005	0.002	0.020	0.021
x_2	m	0.000	-0.500	0.000	-0.100
y_2	m	0.000	-0.100	0.100	-0.093
m_3	kg	0.250	0.150	0.500	0.335
I_3	kg m ²	0.005	0.002	0.020	0.001
x_3	m	-0.250	-0.500	0.000	-0.474
y_3	m	0.000	-0.100	0.100	-0.0436
m_4	kg	0.100	0.025	0.250	0.152
I_4	kg m ²	0.03	0.001	0.010	0.010
x_4	m	0.000	-0.500	0.000	-0.347
y_4	m	0.000	-0.100	0.100	0.110
a_1	m	Same as of the a_1 of the optimized AS			0.415
a_2	m	Same as of the a_2 of the optimized AS			0.480
a_3	m	0.500	0.300	0.700	0.583
a_4	m	0.100	0.000	0.300	0.217
a_5	m	Calculated according to figure 3.19			0.533
a_6	m	Calculated according to figure 3.19			0.158
ϕ_1	rad	0.000	-0.785	0.785	0.299
ϕ_3	rad	3.242	2.618	3.665	3.100
K_{spring}	kN/mm	0.6	0.2	0.9	0.655

Lagrangian of the double pendulum

$$\mathcal{L}_{AS,sw} = \left(\frac{d(q_{AS,sw})}{dT_{AS}} \right)^T V_{AS,sw} \left(\frac{d(q_{AS,sw})}{dT_{AS}} \right) + G_{AS,sw} Q_{AS,sw}$$

$$V_{AS,sw} = \begin{bmatrix} \frac{1}{2}\mathcal{N}_M & 0 & \begin{pmatrix} -m_B(y_B CB + x_B SB)/\vartheta_2 + \\ \frac{-\mathcal{N}_2}{a_1}S12B + \frac{-\mathcal{N}_4}{a_1}S1B + \\ \frac{-\mathcal{N}_3}{a_1}C12B + \frac{-\mathcal{N}_5}{a_1}C1B \end{pmatrix} & \begin{pmatrix} \frac{-\mathcal{N}_2}{a_1}S12B + \frac{-\mathcal{N}_4}{a_1}S1B + \\ \frac{-\mathcal{N}_3^D}{a_1}C12B + \frac{-\mathcal{N}_5}{a_1}C1B \end{pmatrix} & \begin{pmatrix} \frac{-\mathcal{N}_2}{a_1}S12B + \\ \frac{-\mathcal{N}_3}{a_1}C12B \end{pmatrix} \\ 0 & \frac{1}{2}\mathcal{N}_M & \begin{pmatrix} m_B(x_B CB - y_B SB)/\vartheta_2^D + \\ \frac{\mathcal{N}_2}{a_1}C12B + \frac{\mathcal{N}_4}{a_1}C1B + \\ \frac{-\mathcal{N}_3}{a_1}S12B + \frac{-\mathcal{N}_5}{a_1}S1B \end{pmatrix} & \begin{pmatrix} \frac{\mathcal{N}_2}{a_1}C12B + \frac{\mathcal{N}_4}{a_1}C1B + \\ \frac{-\mathcal{N}_3}{a_1}S12B + \frac{-\mathcal{N}_5}{a_1}S1B \end{pmatrix} & \begin{pmatrix} \frac{\mathcal{N}_2}{a_1}C12B + \\ \frac{-\mathcal{N}_3}{a_1}S12B \end{pmatrix} \\ 0 & 0 & \begin{pmatrix} \frac{1}{2}\mathcal{N}_1 + \mathcal{N}_2 C2 - \mathcal{N}_3 S2 + \\ \left(\frac{1}{2}\vartheta_B\right)/\vartheta_2 \end{pmatrix} & \mathcal{N}_1 + 2\mathcal{N}_2 C2 - 2\mathcal{N}_3 S2 & 1 + \mathcal{N}_2 C2 - \mathcal{N}_3 S2 \\ 0 & 0 & 0 & \frac{1}{2}\mathcal{N}_1 + \mathcal{N}_2 C2 - \mathcal{N}_3 S2 & 1 + \mathcal{N}_2 C2 - \mathcal{N}_3 S2 \\ 0 & 0 & 0 & 0 & 1/2 \end{bmatrix}$$

$$G_{AS,sw} = [\mathcal{N}_3 \quad \mathcal{N}_2 \quad \mathcal{N}_5 \quad \mathcal{N}_4 \quad m_B a_1 y_B / \vartheta_2 \quad m_B a_1 x_B / \vartheta_2 \quad a_1 \mathcal{N}_M]$$

$$Q_{AS,sw} = \begin{bmatrix} C12B \\ S12B \\ C1B \\ S1B \\ CB \\ SB \\ y_0 \end{bmatrix}$$

Pi-factors of the double pendulum

$$\mathcal{N}_M = (m_1 + m_2 + m_B)/\vartheta_2$$

$$\mathcal{N}_1 = 1 + \vartheta_1 + a_1^2 m_2 / \vartheta_2$$

$$\mathcal{N}_2 = a_1(a_2 + x_2)m_2/\vartheta_2$$

$$\mathcal{N}_3 = a_1 m_2 y_2 / \vartheta_2$$

$$\mathcal{N}_4 = a_1((a_1 + x_1)m_1 + a_1 m_2)/\vartheta_2$$

$$\mathcal{N}_5 = a_1 m_1 y_1 / \vartheta_2$$

$$\vartheta_i = I_i + m_i((a_i + x_i)^2 + (y_i)^2); \quad i \in \{1, 2, B\}$$

The notation $C12B$ refers to $\cos(\theta_1 + \theta_2 + \theta_B)$

SLIP model dimensionless Lagrangian:

$V_{AS,st}$

$$= \begin{bmatrix} \frac{1}{2} \left(I_B + I_S + I_U + M_B (x_S^2 + y_S^2 + l^2 + 2l(x_B CH - y_B SH)) \right) \\ + M_S(x_S^2 + (y_S + l)^2) + M_U(x_U^2 + y_U^2) \\ \frac{(m_S y_S + M_B y_B CH + M_B x_B SH)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} \frac{(m_S y_S + M_B y_B CH + M_B x_B SH)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ \frac{1}{2} \frac{(M_B + m_S)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ 0 \end{bmatrix} + \begin{bmatrix} \frac{(M_B x_B^2 + M_B y_B^2 + M_B CH l x_B - M_B SH l y_B + I_B)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ - \frac{M_B (y_B CH + x_B SH)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ \frac{1}{2} \frac{M_B (x_B^2 + y_B^2) + I_B}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \end{bmatrix}$$

$$V_{AS,st} = \frac{a_1 [m_B y_B \quad m_B x_B \quad 0 \quad (m_B + m_S) \quad (m_S y_S + m_U y_U) \quad (m_S x_S + m_U x_U)]}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)}$$

$$Q_{AS,st} = \begin{bmatrix} \cos(\theta_H + \theta_L) \\ \sin(\theta_H + \theta_L) \\ \cos \theta_L l \\ \sin \theta_L l \\ \cos \theta_L \\ \sin \theta_L \end{bmatrix}$$

SLIP model Pi-factors:

$$\begin{aligned} \mathcal{N}_1^S &= \frac{\frac{1}{2} (I_B + I_S + I_U + M_B (x_S^2 + y_S^2 + l^2 + 2l(x_B \cos \theta_H - y_B \sin \theta_H)) + M_S(x_S^2 + (y_S + l)^2) + M_U(x_U^2 + y_U^2))}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ \mathcal{N}_2^S &= - \frac{(m_S y_S + M_B y_B \cos \theta_H + M_B x_B \sin \theta_H)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ \mathcal{N}_3^S &= \frac{(M_B x_B^2 + M_B y_B^2 + M_B \cos \theta_H l x_B - M_B \sin \theta_H l y_B + I_B)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ \mathcal{N}_4^S &= \frac{1}{2} \frac{(M_B + m_S)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ \mathcal{N}_5^S &= - \frac{M_B (y_B \cos \theta_H + x_B \sin \theta_H)}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \\ \mathcal{N}_6^S &= \frac{1}{2} \frac{M_B (x_B^2 + y_B^2) + I_B}{(\vartheta_2 + \vartheta_4 + a_2^2 m_3)} \end{aligned}$$

Linkage Leg dimensionless Lagrangian:

$$\begin{aligned}
V_{LL,sw} &= \begin{bmatrix} \frac{1}{2}\mathcal{M}_M & 0 & \begin{pmatrix} -\frac{m_B(y_B CB + x_B SB)}{(\vartheta_2 + \vartheta_5 + a_2^2 m_3)} + \\ -\frac{\mathcal{M}_{10}}{a_1} S12B + \frac{-\mathcal{M}_{12}}{a_1} S1B + \\ -\frac{\mathcal{M}_{11}}{a_1} C12B + \frac{-\mathcal{M}_{13}}{a_1} C1B \end{pmatrix} & \begin{pmatrix} -\frac{\mathcal{M}_{10}}{a_1} S12B + S1B + \\ -\frac{\mathcal{M}_{11}}{a_1} C12B + \frac{-\mathcal{M}_{13}}{a_1} C1B \end{pmatrix} & \begin{pmatrix} -\left(\frac{\mathcal{M}_{10}}{a_1} + \mathcal{M}_{14}\right) S12B + \frac{-\mathcal{M}_8}{a_2} S1B + \\ -\left(\frac{\mathcal{M}_{11}}{a_1} + \mathcal{M}_{15}\right) C12B + \frac{-\mathcal{M}_9}{a_2} C1B \end{pmatrix} \\ 0 & \frac{1}{2}\mathcal{M}_M & \begin{pmatrix} \frac{m_B(x_B CB - y_B SB)}{(\vartheta_2 + \vartheta_5 + a_2^2 m_3)} + \\ -\frac{\mathcal{M}_{11}}{a_1} S12B + \frac{-\mathcal{M}_{13}}{a_1} S1B + \\ \frac{\mathcal{M}_{10}}{a_1} C12B + \frac{\mathcal{M}_{12}}{a_1} C1B \end{pmatrix} & \begin{pmatrix} -\frac{\mathcal{M}_{11}}{a_1} S12B + \frac{-\mathcal{M}_{13}}{a_1} S1B + \\ \frac{\mathcal{M}_{10}}{a_1} C12B + \frac{\mathcal{M}_{12}}{a_1} C1B \end{pmatrix} & \begin{pmatrix} -\left(\frac{\mathcal{M}_{11}}{a_1} + \mathcal{M}_{15}\right) S12B + \frac{-\mathcal{M}_9}{a_2} S1B + \\ \left(\frac{\mathcal{M}_{10}}{a_1} + \mathcal{M}_{14}\right) C12B + \frac{\mathcal{M}_8}{a_2} C1B \end{pmatrix} \\ 0 & 0 & \begin{pmatrix} \frac{1}{2}\mathcal{M}_1 + \mathcal{M}_2 C2 - \mathcal{M}_3 C2 + \\ (\vartheta_B) \\ (\vartheta_2 + \vartheta_5 + a_2^2 m_3) \end{pmatrix} & \mathcal{M}_1 + 2\mathcal{M}_2 C2 - 2\mathcal{M}_3 C2 & \begin{pmatrix} 1 + \mathcal{M}_6 + (\mathcal{M}_2 + \mathcal{M}_4 + \mathcal{M}_8) C2 \\ -(\mathcal{M}_3 + \mathcal{M}_5 + \mathcal{M}_9) S2 \end{pmatrix} \\ 0 & 0 & 0 & \frac{1}{2}\mathcal{M}_1 + \mathcal{M}_2 C2 - \mathcal{M}_3 C2 & \begin{pmatrix} 1 + \mathcal{M}_6 + (\mathcal{M}_2 + \mathcal{M}_4 + \mathcal{M}_8) C2 \\ -(\mathcal{M}_3 + \mathcal{M}_5 + \mathcal{M}_9) C2 \end{pmatrix} \\ 0 & 0 & 0 & 0 & \frac{1}{2} + \mathcal{M}_7/2 + \mathcal{M}_8 C2 + C2 \end{bmatrix} \\
G_{LL,sw} &= [\mathcal{M}_{11} \quad \mathcal{M}_{10} \quad \mathcal{M}_{13} \quad \mathcal{M}_{12} \quad m_B a_1 y_B / (\vartheta_2 + \vartheta_5 + a_2^2 m_3) \quad m_B a_1 x_B / (\vartheta_2 + \vartheta_5 + a_2^2 m_3) \quad a_1 \mathcal{M}_M] \\
Q_{AS,sw} &= [C12B \quad S12B \quad C1B \quad S1B \quad CB \quad SB \quad y_0]^T
\end{aligned}$$

Linkage Leg Pi-factors:

$$\begin{aligned}
\mathcal{M}_M &= (m_1 + m_2 + m_3 + m_4 + m_B) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_1 &= 1 + (\vartheta_1 + \vartheta_3 + a_1^2 m_2 + a_1^2 m_3 + a_2^2 m_4 + 2a_1(a_3 + x_3)m_3 C23 - 2a_1 y_3 m_3 S23) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_2 &= (a_1(a_2 + x_2)m_2 + a_1 a_2 m_3 + a_2(a_3 + x_3)m_3 C23 + a_4(a_5 + x_4)m_4 C5 - a_2 y_3 m_3 S23 - a_4 y_4 m_4 S5) \\
&\quad / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_3 &= (a_1 y_2 m_2 - a_2 y_3 m_3 C23 + a_4 y_4 m_5 C5 - a_2(a_3 + x_3)m_3 S23 + a_4(a_5 + x_4)m_4 S5) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_4 &= a_4 m_4 R_4((a_5 + x_4)C5 - y_4 S5) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_5 &= a_4 m_4 R_4(y_4 C5 + (a_5 + x_4)S5) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_6 &= (\vartheta_4 R_4 + R_3(\vartheta_3 + a_1(a_3 + x_3)m_3 C23 - a_1 y_3 m_3 S23)) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_7 &= (\vartheta_3 R_3^2 + 2\vartheta_4 R_4 + \vartheta_4 R_4^2) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_8 &= a_2 m_3 R_3((a_3 + x_3)C23 - y_3 S23) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_9 &= a_2 m_3 R_3(y_3 C23 + (a_3 + x_3)S23) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_{10} &= (a_1(a_2 + x_2)m_2 + a_1 a_2 m_3 + a_1(a_5 + x_4)m_4 C6 - a_1 y_4 m_4 S6) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_{11} &= (a_1 y_2 m_2 + a_1 y_4 m_4 C6 + a_1(a_5 + x_4)m_4 S6) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_{12} &= (a_1(a_1 + x_1)m_1 + a_1^2 m_2 + a_1^2 m_3 + a_1 a_4 m_4 C\phi_1 + a_1(a_3 + x_3)m_3 C23 - a_1 y_3 m_3 S23) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_{13} &= (a_1 y_1 m_1 + a_1 a_4 m_4 S\phi_1 + a_1 y_3 m_3 C23 + a_1(a_3 + x_3)m_3 S23) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_{14} &= m_4 R_4((a_5 + x_4)C6 - y_4 S6) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3) \\
\mathcal{M}_{15} &= m_4 R_4(y_4 C6 + (a_5 + x_4)S6) / (\vartheta_2 + \vartheta_4 + a_2^2 m_3)
\end{aligned}$$

where

$$\theta_5 = \theta_4 - \theta_2, \quad \theta_6 = \phi_1 + \theta_4 - \theta_2,$$

$$\vartheta_i = I_i + m_i((a_i + x_i)^2 + (y_i)^2); \quad i \in \{1,2,3,4,B\}$$

$$R_3 = a_2 \sin(\phi_1 + \theta_4 - \theta_2) / a_6 \sin(\theta_2 + \theta_3 + \phi_3 - \phi_1 - \theta_4)$$

$$R_4 = a_2 \sin(\phi_3 + \theta_3) / a_5 \sin(\theta_2 + \theta_3 + \phi_3 - \phi_1 - \theta_4) - 1$$

Linkage Leg Lagrangian transformation matrixes

$$T_V^{swing \rightarrow stance}$$

$$= \begin{bmatrix} -L^S \sin \theta_L & \cos \theta_L & 0 \\ L^S \cos \theta_L & \sin \theta_L & 0 \\ 1 & 0 & 1 \\ 0 & -\frac{a_2 \cos(\theta_1 + \theta_2 + \theta_H) + R_3 a_3 \cos(\theta_1 + \theta_2 + \theta_3 + \theta_H)}{a_1 a_2 \sin \theta_2 - a_2 a_3 \sin \theta_3 + R_3 a_1 a_3 \sin(\theta_2 + \theta_3) + R_3 a_2 a_3 \sin \theta_3} & -1 \\ 0 & \frac{a_3 \cos(\theta_1 + \theta_2 + \theta_3 + \theta_H) + a_1 \cos(\theta_2 + \theta_3) + a_2 \cos(\theta_1 + \theta_2 + \theta_H)}{a_1 a_2 \sin \theta_2 - a_2 a_3 \sin \theta_3 + R_3 a_1 a_3 \sin(\theta_2 + \theta_3) + R_3 a_2 a_3 \sin \theta_3} & 0 \end{bmatrix}$$

$$T_G^{swing \rightarrow stance}$$

$$= \begin{bmatrix} 0 & 0 & -\frac{1}{a_2} & 0 & \frac{a_1 \cos \phi_{off} + a_3 \cos(\phi_{off} - \theta_2 - \theta_3)}{a_2} & \frac{a_1 \sin \phi_{off} + a_3 \sin(\phi_{off} - \theta_2 - \theta_3)}{a_2} \\ 0 & 0 & 0 & -\frac{1}{a_2} & -\frac{a_1 \sin \phi_{off} + a_3 \sin(\phi_{off} - \theta_2 - \theta_3)}{a_2} & \frac{a_1 \cos \phi_{off} + a_3 \cos(\phi_{off} - \theta_2 - \theta_3)}{a_2} \\ 0 & 0 & 0 & 0 & -\cos \phi_{off} & -\sin \phi_{off} \\ 0 & 0 & 0 & 0 & \sin \phi_{off} & -\cos \phi_{off} \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$