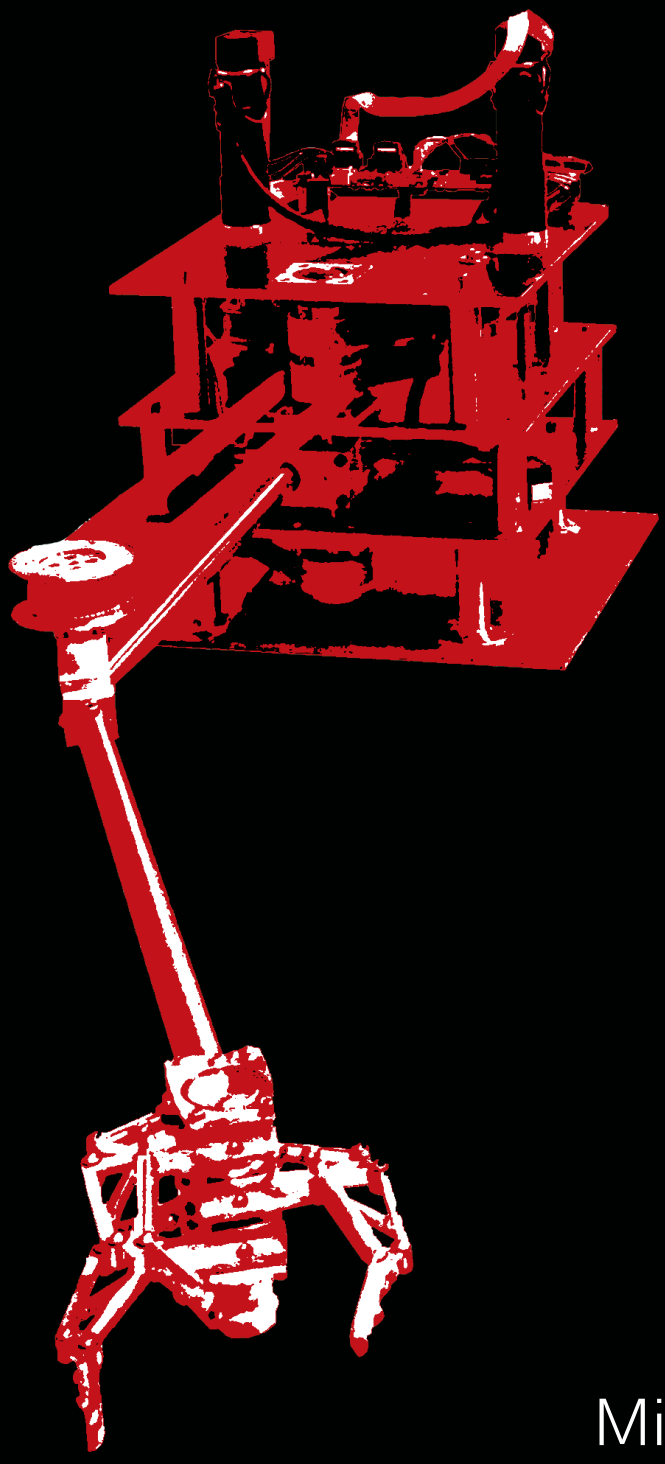
Exploiting dynamics

in robotic arms with repetitive tasks



Michiel Plooij

Exploiting dynamics

in robotic arms with repetitive tasks

M.C. Plooij

Exploiting dynamics

in robotic arms with repetitive tasks

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus, prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties
in het openbaar te verdedigen op
woensdag 9 december 2015 om 15:00 uur

door

Michaël Christiaan PLOOIJ
Werktuigkundig ingenieur
Technische Universiteit Delft, Nederland
geboren te Zoetermeer

This dissertation has been approved by the
promotors: Prof. dr. F.C.T. van der Helm en Prof. dr. ir. M. Wisse

Composition of the doctoral committee:

Rector Magnificus	chairman
Prof. dr. F.C.T. van der Helm	Delft University of Technology
Prof. dr. ir. M. Wisse	Delft University of Technology

Independent members:

Prof. dr. L. Birglen	Polytechnique Montréal, Canada
Prof. dr. ir. J.L. Herder	Delft University of Technology
Prof. dr. ir. J.M.A. Scherpen	Rijksuniversiteit Groningen
Prof. dr. ir. J. De Schutter	Katholieke Universiteit Leuven, Belgium
Prof. dr. ir. B. Vanderborght	Vrije Universiteit Brussel, Belgium
Prof. dr. ir. P. Breedveld	Delft University of Technology, reserve member

Ir. W.J. Wolfslag has contributed greatly to the preparation of this dissertation.

This research was financially supported by the Dutch Technology Foundation STW (project number 11282).

Enabling new technology

ISBN 978-94-6186-576-2
A digital copy of this thesis can be downloaded from http://repository.tudelft.nl.

Contents

Summary

Exploiting dynamics

in robotic arms with repetitive tasks

Since the industrial revolution, machines have taken over many tasks from humans, increasing labor productivity and prosperity. In the 20th century, the introduction of robots created a second wave of automation, increasing the labor productivity even further. In order to create a third wave of automation, it is necessary to develop a new generation of robots that is able to act in unknown, unstructured environments, such as households, space and factories in which humans and robots collaborate. Two of the main aspects of robots that have to be improved in order to be successful are their energy consumption and their reliability. This thesis is split into two parts. The first part focuses on reducing the energy consumption of robots by using elasticity. The second part focuses on increasing the reliability by using feedforward control. Throughout this thesis, we focus on one type of robot: robotic arms with repetitive tasks, such as pick-and-place tasks.

Part I: Elasticity in robots

One of the most effective techniques to obtain a low energy consumption of robots is the efficient recapture of negative work. There are multiple options to store the energy recaptured from the robot (e.g. electrical, chemical, potential, etc.), of which potential energy is the most promising in terms of efficiency. Compared to other potential energy storages, springs in parallel to the motor are relatively compact and therefore preferred. The problem of using parallel springs is that the timing of energy storage and release is not independently controllable from the position of the joints. This lack of control of the energy storage limits the versatility of robots. The most promising class of actuators that solve this problem is identified in this thesis: clutched elastic actuators (CEAs) in parallel with the motor. CEAs consist of at least one spring and one locking mechanism and possibly one or multiple gears or differentials.

The first part of this thesis addresses three issues in CEAs. First, the use of CEAs introduces a trade-off between the versatility of the robot and its energy consumption. A CEA design with many clutches and springs leads to a high versatility, while at the same time the locking mechanisms consume energy and the complexity of CEAs is likely to increase friction, weight and size. Therefore, the question is: what is the *best mechanism to reduce the energy consumption of pick-and-place robotic arms*? Secondly, one of the most important components of CEAs are the locking mechanisms. However, it is still unclear what the *best locking mechanism for CEAs*

is. And thirdly, there is a trend towards complex CEAs with many springs and many locking mechanisms. However, it is unclear *how these complex CEAs should be analyzed*.

The *best available mechanism to reduce the energy consumption of pick-and-place robotic arms* is the Bi-directional Clutched Parallel Elastic Actuator (BIC-PEA), that is introduced in this thesis. In pick-and-place tasks, it is crucial to be able to vary the pick position and the place position on the fly. The BIC-PEA is the only type of CEA that allows for such versatility. Results show that implementation of the BIC-PEA led to a reduction of the energy consumption up to 65%.

The *best available locking mechanism for CEAs* is the Statically Balanced Brake (SBB), that is introduced in this thesis. The three most important properties for a locking mechanism to be applicable in CEAs, are the ability to unlock under load, have a low energy consumption and lock at many positions. SBBs are the only locking mechanism with these properties. SBBs are friction based locking mechanisms and thus they unlock well under load and have an infinite number of locking positions. In comparison to other friction based locking mechanisms, the actuation force of SBBs is 95-97% lower, meaning that their energy consumption is negligible.

Finally, this part introduces a method to *analyze complex CEAs*. We propose a taxonomy to analyze the functionalities of CEAs. We argue that functionality can be expressed in terms of a stiffness matrix, a constraint matrix and a combination of a diagonal clutch matrix and an incidence matrix. Using this description, the set of possible resulting stiffnesses and equilibria can be found. Furthermore, it can lead to new CEA designs in which the number of resulting stiffnesses and equilibria grows exponentially with the number of springs and clutches.

Part II: Feedforward control in robots

The second aspect of robots that has to be improved for future applications is their reliability. This means their components should be robust, but also that robots should be able to deal with failure of components. One set of components that is sensitive to failure are the sensors. Sensor failure can either occur in the sensor itself or in the wiring. Without sensory information, the controller is bound to feedforward control, also called open loop control. Therefore, the second part of this thesis focuses on the question: what is still possible when no feedback is available?

When applying feedforward control, there are two main issues that have to be addressed: *model inaccuracies* and *disturbances*. Model inaccuracies cause a mismatch between the predicted and actual responses to different inputs. Disturbances are unpredictable in general and cause the system to deviate from its intended trajectory.

The results show that both model inaccuracies and disturbances can be handled in robots with solely feedforward control. With only *model inaccuracies*, we show that feedforward controllers can be optimized such that the sensitivity to inaccuracies in the friction model is minimized. This sensitivity can even be eliminated. With only *disturbances*, we show that repetitive motions can be optimized using limit cycle theory. Small disturbances diminish over time when the motion that the arm performs is a stable limit cycle. Feedforward control becomes more difficult when both *model inaccuracies* and *disturbances* are present. However, we show that it is possible for the robot to learn to perform cycles that are open loop stable, even when the model of the arm is inaccurate. The results show that the maximum position errors at the pick position and place position are 1-2.5 cm, which is accurate enough for coarse pick-and-place tasks.

Finally, this part addresses the question whether feedforward techniques are still useful when a small amount of feedback is available?. We analyze the accuracy of a robotic arm as function of the feedback gain, while the motion is optimized for sensitivity to disturbances. The results show that for all gains, minimizing the sensitivity results in a better accuracy than maximizing the sensitivity. Therefore, pure feedforward techniques can still be useful when (partial) feedback is available.

Conclusion

Overall, the results of this thesis show that both the energy consumption and the reliability of robots can be improved significantly. The implementation of a clutched elastic actuator in parallel with the motor can reduce the energy consumption of robots with 65%. The setup with a differential and two locking mechanisms causes the versatility to remain high. Statically balanced brakes solve the problem of regular friction based locking mechanisms that a large actuation force is needed. The concept of statically balanced brakes allows for friction based locking mechanisms with an actuation force that is reduced with 95-97% in comparison to regular friction based locking mechanisms, while being relatively small. Finally, feedforward control on robotic arms is possible when there are disturbances, model inaccuracies or both.

Samenvatting

Dynamica uitbuiten
in robotarmen met repeterende taken

Sinds the industriële revolutie hebben machines veel taken van mensen overgenomen. Dit heeft de arbeidsproductiviteit en de welvaart verhoogd. In de twintigste eeuw heeft de introductie van robots een tweede golf van automatisering veroorzaakt, wat de arbeidsproductiviteit verder heeft verhoogd. Voor een derde golf van automatisering is het noodzakelijk om een nieuwe generatie robots te ontwikkelen die kunnen handelen in onbekende en ongestructureerde omgevingen, zoals het huishouden, de ruimte en fabrieken waarin robots samenwerken met mensen. Twee van de belangrijkste aspecten van robots die verbeterd moeten worden, zijn hun energieverbruik en betrouwbaarheid. Deze twee aspecten vormen de twee delen van dit proefschrift. Het eerste deel focust op het reduceren van het energieverbruik van robots door gebruik te maken van elasticiteit. Het tweede deel focust op het verhogen van de betrouwbaarheid door het gebruik van sensorloze aansturing. In het hele proefschrift wordt gefocust op één type robot, namelijk robotarmen met repeterende taken, zoals een pick-and-place-taak.

Deel I: Elasticiteit in robots

Een van de meest effectieve technieken om het energieverbruik van robots te verminderen, is het efficiënt terugwinnen van negatieve arbeid. Er zijn meerdere opties om de teruggewonnen energie op te slaan, bijvoorbeeld elektrisch, chemisch, potentieel, etc.. Potentiële energie is van deze het meest veelbelovend is qua efficiëntie. In vergelijking met andere vormen van opslag in potentiële energie zijn veren die parallel staan aan de motor relatief compact en hebben daarom de voorkeur. Het probleem met het gebruik van veren die parallel staan aan de motor is dat de timing van de energieopslag niet onafhankelijk aan te sturen is van de positie van de robot. Dit gebrek aan aansturing beperkt de veelzijdigheid van de robot. De meest veelbelovende klasse van actuatoren die dit probleem oplossen is geïdentificeerd in dit proefschrift: gekoppelde elastische actuatoren (GEA's) parallel aan de motor. GEA's bestaan uit tenminste één veer en één koppeling en mogelijk één of meerdere tandwieloverbrengingen of differentiëlen.

Het eerste gedeelte van dit proefschrift behandelt drie problemen met GEA's. Ten eerste, het gebruik van GEA's introduceert een afweging tussen de veelzijdigheid van de robot en zijn energieverbruik. Een GEA met veel koppelingen en veren leidt tot een grote veelzijdigheid, maar de koppelingen verbruiken energie en de complexiteit verhoogt de grootte, het gewicht en de wrijving. Daarom is het de vraag *wat het*

beste mechanisme is om het energieverbruik van robots met pick-and-place-taken te verlagen. Ten tweede, een van de belangrijkste componenten van GEA's zijn de koppelingen. Het is echter onduidelijk *wat de beste type koppeling is voor GEA's*. En ten derde is er een trend naar complexere GEA's met veel veren en koppelingen. Het is echter nog onduidelijk *hoe deze complexe GEA's geanalyseerd moeten worden*.

Het *beste mechanisme om het energieverbruik van robots met pick-and-place-taken te verlagen* is de Bi-directionele Gekoppelde Parallel Elastische Actuator (BIG-PEA), die geïntroduceerd wordt in dit proefschrift. In pick-and-place-taken is het cruciaal dat het mogelijk is om te pick-positie en place-positie te variëren tijdens de taak. De BIC-PEA is het enige type GEA dat voorziet in een dergelijke veelzijdigheid. Resultaten laten zien dat implementatie van de BIG-PEA heeft geleid tot een vermindering van het energieverbruik met 65%.

De *beste type koppeling voor GEA's* is de Statisch Gebalanceerde Rem (SGR), die geïntroduceerd wordt in dit proefschrift. De drie belangrijkste eigenschappen van koppelingen voor GEA's zijn dat ze kunnen ontkoppelen terwijl er kracht op staat, ze een laag energieverbruik hebben en ze kunnen koppelen op veel posities. De SGR is de enige koppeling met al deze eigenschappen. SGR's zijn gebaseerd op wrijving, dus ontkoppelen ze makkelijk terwijl er kracht op staat en hebben ze oneindig veel koppelingsposities. In vergelijking met andere koppelingen heeft de SGR 95-97% minder actuatiekracht nodig, wat betekent dat het energieverbruik verwaarloosbaar is.

Ten slotte introduceert dit gedeelte van het proefschrift een methode om *complexe GEA's te analyseren*. We stellen een taxonomie voor waarmee de functionaliteiten van GEA's geanalyseerd kunnen worden. Verder beargumenteren we dat functionaliteit uitgedrukt kan worden met behulp van een stijfheidsmatrix, een constraintmatrix en een combinatie van een diagonale koppelingsafhankelijke matrix en een incidentiematrix. Met deze beschrijving kunnen alle mogelijke resulterende stijfheden en evenwichtsposities gevonden worden. Daarnaast kan het leiden tot nieuwe GEA-ontwerpen waarin het aantal resulterende stijfheden en evenwichtsposities exponentieel groeit met het aantal veren en koppelingen.

Deel II: Sensorloze aansturing in robots

Het tweede aspect van robots dat verbeterd moet worden voor toekomstige toepassingen, is hun betrouwbaarheid. Dit betekent dat de componenten robuust moeten zijn, maar ook dat robots moeten blijven functioneren wanneer er een van de componenten kapot gaat. Componenten die bijvoorbeeld kapot kunnen gaan zijn sensoren. Zij kunnen ofwel zelf kapot kunnen gaan, of de bekabeling kan het begeven. Zonder

sensorinformatie is de aansturing gebonden aan zogenaamde sensorloze aansturing, ook wel open loop aansturing genoemd. Daarom focust het tweede gedeelte van dit proefschrift op de vraag: wat is nog mogelijk als er geen sensorinformatie beschikbaar is?

Er zijn twee problemen die zich voordoen wanneer sensorinformatie ontbreekt: *modelonnauwkeurigheden* en *verstoringen*. Model onnauwkeurigheden veroorzaken een discrepantie tussen de voorspelde en de feitelijke reacties op verschillende inputs. Verstoringen zijn in het geheel onvoorspelbaar en zorgen ervoor dat het systeem afwijkt van het gewenste traject.

De resultaten in dit proefschrift laten zien dat het mogelijk is om met zowel modelonnauwkeurigheden als verstoringen om te gaan. Voor *modelonnauwkeurigheden* blijkt dat sensorloze controllers geoptimaliseerd kunnen worden zodat de gevoeligheid voor onnauwkeurigheden in het wrijvingsmodel verwaarloosbaar is. Deze gevoeligheid kan zelfs tot nul worden gereduceerd. Voor *verstoringen* blijkt dat repeterende bewegingen kunnen worden geoptimaliseerd met behulp van theorie over limiet cycli. Kleine verstoringen verdwijnen over tijd als de beweging van de arm een stabiele limiet cyclus is. Sensorloze aansturing wordt moeilijker als er zowel *modelonnauwkeurigheden* als *verstoringen* zijn. Het blijkt echter dat het voor een robot mogelijk is om te leren cycli te maken die open loop stabiel zijn, zelfs als het model onnauwkeurig is. Dit heeft geresulteerd in maximale positiefouten op de pick-positie en place-positie van 1-2.5 cm, wat nauwkeurig genoeg is voor grove pick-and-place-taken.

Ten slotte behandelt dit deel van het proefschrift de vraag of sensorloze aansturingstechnieken nog steeds nuttig zijn als er een (beperkte) hoeveelheid sensorinformatie beschikbaar is. We analyseren de nauwkeurigheid van robots als functie van hun feedback-gain, terwijl we de bewegingen optimaliseren voor gevoeligheid voor verstoringen. De resultaten laten zien dat voor alle waarden van de feedback-gains, de robot nauwkeuriger is als die gevoeligheid geminimaliseerd is dan als deze gemaximaliseerd is. Daarom kunnen sensorloze aansturingstechnieken nog steeds nuttig zijn als er een (beperkte) hoeveelheid sensorinformatie beschikbaar is.

Conclusie

Concluderend, de resultaten in dit proefschrift laten zien dat zowel het energieverbruik als de betrouwbaarheid van robots significant verbeterd kunnen worden. De implementatie van gekoppelde elastische actuatoren parallel aan de motor kan het energieverbruik van robots met 65% verminderen. Het ontwerp hiervan met een differentieel en twee koppelingen zorgt ervoor dat de veelzijdigheid hoog blijft. Normale wrijvingskoppelingen vereisen een hoge actuatiekracht. Statisch gebalanceerde rem-

men lossen dit probleem op. Het concept van statisch gebalanceerde remmen leidt tot relatief compacte wrijvingskoppelingen met een actuatiekracht die 95-97% lager is dan in standaard wrijvingskoppelingen. Ten slotte laten de resultaten zien dat sensorloze aansturing mogelijk is, ook als er verstoringen en modelonnauwkeurigheden zijn.

1

Introduction

1.1 Motivation

Since the industrial revolution, machines have taken over many tasks from humans, increasing labor productivity and prosperity. In the 20th century, the introduction of robots created a second wave of automation, increasing the labor productivity even further. A well-known example of an industry in which robots perform many tasks, is the automotive industry (see Fig. 1.1a). In order to create a third wave of automation, it is necessary to develop a new generation of robots that is able to act in unknown, unstructured environments.

The current performance of robots has proven to be sufficient for utilization in predefined, ground-fixed, structured environments such as the car factory in Fig. 1.1a. However, the field of robotics is on the verge of moving into other, less structured environments. A first example is mobile robots (see Fig. 1.1b), such as household robots, servant robots and self-driving cars. Another example is space robots, such as the mars rover depicted in Fig. 1.1c. Such unstructured, unknown environments demand a better performance of the new generation of robots.

In order to be successful, it is crucial that the new generation of robots outperforms the previous on eight aspects:

1. Since many robots will be carrying their own energy supply, they have to have a *low energy consumption* in order to have a large uptime.
2. They should be *affordable* in order to apply them in low cost applications or to make them even disposable.
3. In order to manipulate all sorts of environments, the hardware should allow them to be *versatile*.
4. They should be *reliable*, because there will be no engineers close by to repair or reprogram them. This also means that they should be able to deal with failure of components.
5. Since they act in unknown environments, their *sensing* should be improved in order to perceive that environment.
6. In order to exploit their hardware to manipulate all sorts of environments, they should be *dexterous*.
7. In order to be able to interact with humans, they should be *safe to work with*.
8. They should have an *intuitive and attractive interface* to ensure that humans are able to work with the robots.

When considering the required performance improvements of robots, it is inspiring to look at the performance of the human body. Humans perform numerous tasks in an energy efficient and elegant way. Furthermore, the human body has good perception in comparison to many robots and is relatively safe to work with for other humans. In fact, the human body has many properties that are desired for the new generation of robots. This is not surprising since most environments those robots have to work in, are designed for humans. Therefore, the approach in this thesis is to use inspiration from the human body to solve the challenges that the new generation of robots faces.

This thesis focuses on the first and fourth issue mentioned above: energy consumption and reliability. Those issues were selected based on our previous experience with low energy consuming and low feedback requiring walking robots. To lower the energy consumption, we are inspired by the use of elasticity in the human body [6, 97, 98, 111]. Elasticity has multiple functions, ranging from shock absorption to storing energy. In this thesis, we will mainly use elasticity as a mechanical energy buffer to lower the energy consumption. To increase the reliability, we are inspired by the extensive use of feedforward control in the human body [47, 81]. While humans mainly use feedforward control because of their slow feedback loops [43, 215], robots could use it to deal with the situation where sensory feedback fails.[1] Therefore, this thesis is split into two parts: *elasticity in robots* and *feedforward control in robots*.

Throughout this thesis, we will focus on one type of robot: robotic arms. Specifically, we consider robotic arms with repetitive tasks, such as pick-and-place tasks. We chose this task, because it is well-defined by four phases: moving towards an object, grasping the object, moving towards the goal position and placing the object. Although pick-and-place robots do not belong to the new generation of robots envisioned above, they are very suitable for studying solely the energy consumption and reliability. Furthermore, the techniques developed in this thesis are widely applicable. For instance, in wheeled robots, elasticity can also be used to store energy when decelerating. When this energy is used to accelerate later in time, the overall energy consumption is reduced. Moreover, similar techniques can be used to increase the performance of all moving devices including cars, trains and busses, but also prostheses and exoskeletons.

[1] Here we mention reliability as the main reason to study feedforward control. Throughout the project, we have strugled to find a good application for pure feedforward controllers. We only started to consider reliability as a motivation at the end of the project. Therefore, the motivations of the indivisual chapters on feedforward control do not focus on reliability.

a) b) c)

Figure 1.1: a) Currently, robots are mostly applied in pre-defined ground-fixed, structured environments, such as factories. (image source: Wikimedia Commons) b) One of the future applications of robots: domestic robots. (image source: Wikimedia Commons) c) A challenging environment for robots: space. (image source: NASA)

1.2 Elasticity in robots

The first part of this thesis focuses on using elasticity in robots to reduce their energy consumption. Actuators that include a compliant element are called elastic actuators (EAs). This section gives an overview of the research field of EAs, identifies current problems and defines the goals for the first part of this thesis.

1.2.1 Overview of the research field

EAs can be found in robots using various configurations (see Fig. 1.2): parallel elastic actuators (PEAs), clutched elastic actuators (CEAs), series elastic actuators (SEAs) and variable stiffness actuators (VSAs). The oldest of those concepts is the PEA. In order to identify the most interesting configuration, we will now discuss those concepts.

PEAs determine the relationship between the position and the potential energy. There are two options for this relationship. First, the potential energy can be independent of the position. Such systems are statically balanced. Secondly, the potential energy can be position-dependent such that the springs provide part of the torque that is required to perform the intended task. The first option is the oldest and its history is well described in the PhD thesis of Herder [83]. This approach focuses at reducing the static torques on the motor caused by gravity, allowing for smaller motors and lower energy consumption. The second option started to be applied in the 1980s with the first application in robotic arms described by Babitsky and Shipilov [13]. Since the springs provide part of the torques required to accelerate and decelerate the robot, this approach has the potential to reduce the energy consumption even more than with static balancing. However, since the position-torque relationship is usually set, this approach often results in a decrease of the versatility. This decrease in versatility is less of a problem in systems with repetitive tasks, such

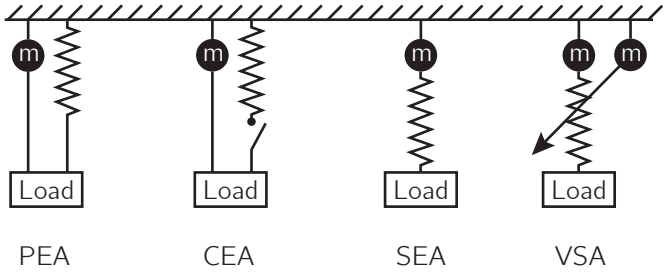


Figure 1.2: Four different types of elastic actuators: parallel elastic actuators (PEA), clutched elastic actuators (CEA), series elastic actuators (SEA) and variable stiffness actuators (VSA).

as walking robots or pick-and-place robotic arms. Therefore, it is no surprise that PEAs have mainly been applied in walking robots [39], prostheses [11, 214], orthoses [49, 177] and robotic arms [27, 171].

Around the same time that PEAs started to be used for other than balancing purposes, clutches started to be introduced [173]. This lead to a new elastic actuation concept, called clutched elastic actuation (CEA). Although CEAs has existed for decades, this thesis is the first to identify the class of CEAs. Again, multiple early designs are described by Babitsky and Shipilov [13]. CEAs have been used since in various forms and applications. Fig. 1.2 shows a CEA where a locking mechanism is used to lock a parallel spring to a load. Examples of other CEA designs include one where the locking mechanism locks the load to the ground [13, 38, 184] and one where the motion of a series spring is locked [129]. Recent CEA designs focus on using multiple clutches [40, 116, 170, 226] and multiple springs [130]. Due to the extra modes induced by the clutches, CEAs allow for more versatility than PEAs.

In 1995, Pratt and Williamson introduced the idea of series elastic actuators [175]. In such actuators, the interface between the actuator and the joint is elastic, which has several advantages. First, the elasticity low-pass filters the shock load. Secondly, it transforms the problem of force control into an easier position control problem. Thirdly, it introduces the possibility of mechanical energy storage. And finally, it decouples the motor inertia from the joint inertia, possibly making the system safer. Since the introduction of SEAs, they have been extensively used, especially in walking devices [78, 176]. SEAs do not decrease the versatility of robots. However, their ability to reduce the energy consumption is limited because the force on the joint also passes through the motor.

At the beginning of the 21st century, various types of variable stiffness actuators (VSAs) started to be developed [96, 228]. As the name suggests, the stiffnesses

of those devices can be controlled. VSAs have mainly been used in a configuration where the elasticity is placed in series with an actuator, like in a SEA. The advantage of VSAs is that the stiffness can be adjusted depending on the task, or even during a task. For instance, for a phase in which precision is required, the stiffness can be high, while in the next phase where a decoupling between the inertias of motor and joint is required, the stiffness can be low. Again, the main applications are walking devices and robotic arms. The parallel version of a VSA has also been proposed [65]. In theory, such a parallel spring in combination with an infinitely variable transmission (IVT) could reduce the energy consumption of robots significantly. However there are no studies showing the effectiveness of such an IVT in robotics.

Generally speaking, EAs can be divided into EAs in parallel with the motor and EAs in series with the motor. Logically, PEAs belong to the first group and SEAs belong to the second group. Interestingly, the vast majority of VSAs belong to the second group, while the vast majority of CEAs belong to the first group. From all EA concepts, EAs in parallel have the largest potential to reduce the energy consumption, because in those EAs, the elasticity takes over a large part of the task from the actuator. Furthermore, in order to preserve the versatility of the robot, the first part of this thesis focusses on CEAs in parallel with the motor.

1.2.2 Problem statement

There are three problems that need to be addressed in the field of CEAs in parallel with the motor. First, the challenge in any EA concept in parallel with the motor is to maintain a level of versatility that suits the task of the robot. This challenge introduces a trade-off between the versatility of the robot and its energy consumption. A CEA design with many clutches and springs leads to a high versatility, while at the same time the locking mechanisms consume energy and its complexity is likely to increase friction, weight and size. Secondly, one of the most important components of CEAs are the locking mechanisms. However, it is still unclear what the best locking mechanism for CEAs is. And thirdly, there is a trend towards complex CEAs with many springs and many locking mechanisms. However, it is unclear how these complex CEAs should be analyzed.

1.2.3 Goal

The goal of the first part of this thesis is to answer the following questions:

1. What is the best mechanism to reduce the energy consumption of pick-and-place robotic arms?

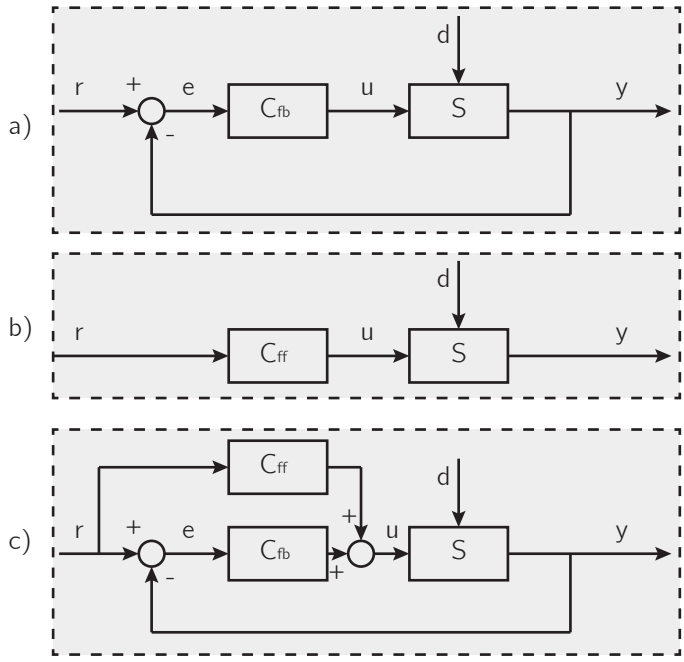


Figure 1.3: Three control designs. a) A feedback controller. b) A feedforward controller. c) A feedforward-feedback controller.

2. What is the best locking mechanism for clutched elastic actuators?[2]

3. How can the functionality of complex clutched elastic actuators be analyzed?

1.3 Feedforward control in robots

The second part of this thesis focuses on using feedforward control in robots to increase their reliability. This section gives an overview of the research field of the use of feedforward control, identifies current problems and defines the goals for the second part of this thesis.

[2]Note that without any further constraints, the first two questions can never be answered. It can never be proven that a certain mechanism is the *best* mechanism there ever will be. Therefore, one constraint in answering those questions is that it must be an existing mechanism. We chose to formulate the questions as they are because of two reasons. First, to answer those questions, a comparison must be made between the proposed mechanisms and existing literature, leading to an increased academic value. Secondly, these questions challenge to optimize the designs, increasing the quality of the proposed mechanisms.

1.3.1 Overview of the research field

In order to identify open problems in the field of feedforward control, we will discuss existing literature on feedforward control. There are innumerable robots that use a feedforward term in their controllers, often in combination with a feedback controller. In feedback-feedforward controllers, the feedforward part uses a model to predict the required input to perform a certain task and the feedback part compensates for model inaccuracies and stabilizes the system. How much feedback and feedforward are used, depends on the expected accuracy of the available model and the accuracy of the available feedback [112].

Only few robots combine the feedforward controller with a very limited amount of feedback, let alone no feedback at all. A well known example of a robot with limited feedback is the juggling robot by Schaal and Atkeson [188]. Their robot stabilizes the maximum height of the ball, while the position of the ball was not an input for the controller. However, the state of the robotic arm itself was an input for the controller. A similar example follows from research to the swing leg retraction rate of running robots [195]. Shaping the swing leg retraction rate correctly, increases the stability of the running cycle, while this only requires state feedback from the swing leg.

The previous two examples show stabilizing position controlled motions in which not the whole state is used as an input. Becker and Bretl [18] go one step further by considering differential drive robots in which wheel velocities are the input and the wheel diameters are uncertain. They show that both stabilization and handling of uncertain dynamics are possible. However, since the velocity is still an input to the controller, state feedback is still required.

In order for robots to still function when sensory feedback fails, pure feedforward control should be used, meaning that no feedback is available. There is a small group of researchers that have focused on the extreme scenario of a robot without feedback. The most well known example is the passive dynamic walkers by McGeer [133]. Since these walkers do not have any motors, they do not use feedback to stabilize their walking motion. Instead, they exploit stable cyclic motions, called stable limit cycles. His results show that the resulting walking patterns are similar to those of humans, suggesting that humans also exploit cyclic stability in walking motions. Similar techniques were later used in combination with feedback in so-called limit cycle walkers [72, 89, 94]. Mombaur et al. [139, 141] used open loop controllers in such limit cycle walkers to let them walk and run on a flat surface. They optimized those controllers both for energy consumption and stability. These studies show that there are a few results in walking robots that indicate that pure

feedforward control is possible. However, it is unclear how model inaccuracies could be handled in such systems and how these results translate to other robots, such as robotic arms.

1.3.2 Problem statement

Although some studies have been performed on feedforward control, its full capabilities are still unknown. Especially since the results are limited to the field of walking robots. When applying feedforward control, there are two main issues that have to be addressed to be accurate: model inaccuracies and disturbances. Model inaccuracies cause a mismatch between the predicted and actual responses to different inputs. Disturbances are unpredictable in general and cause the system to deviate from its intended trajectory.

1.3.3 Goal

The goal of the second part of this thesis is to answer the following questions about robotic arms without feedback:

4. How can the effect of model inaccuracies be eliminated?
5. How can disturbances be rejected?
6. How can both model inaccuracies and disturbances be handled at the same time?
7. Are feedforward techniques are still useful when a small amount of feedback is available?

1.4 Approach

The approach taken in this thesis is to study relatively simple systems in simulation and hardware experiments. Relatively simple means one or two DOF robotic arms in the horizontal plane, eliminating the effect of gravity. The choice for these systems makes it easier to interpret the results and to understand the fundamental principles behind them. These systems will first be modeled and analyzed in simulation. The simulation studies often include an optimal control study. When fully understood, the simulation results will be varified in hardware experiments.

1.5 Thesis outline

This thesis consists of two parts: *elasticity in robots* and *feedforward control in robots*. The chapters of this thesis have some overlap and are independently readable because they are written as papers for scientific journals or conferences. The first part of the thesis answers the first three questions above and consists of five chapters:

1. *What is the best mechanism to reduce the energy consumption of pick-and-place robotic arms?*
   - **Chapter 2** presents a non-linear spring mechanism. Using a singular locking mechanism, the spring is locked at the pick and the place position.
   - **Chapter 5** presents the bi-directional clutched parallel elastic actuator, in which a differential mechanism and two clutches control the connection between the joint and the spring.
2. *What is the best locking mechanism for clutched elastic actuators?*
   - **Chapter 3** gives an overview of all locking mechanisms that are used in robotics and discusses them based on the properties of an ideal locking mechanism.
   - **Chapter 4** introduces a new type of locking mechanism: the statically balanced brake. This locking mechanism eliminates the need of friction based locking mechanism for a large actuation force.
3. *How can the functionality of complex clutched elastic actuators be analyzed?*
   - **Chapter 6** investigates functionalities of current and future CEAs. It provides and overview of existing CEA designs and introduces a mathematical framework to analyze the functionalities of all possible CEA designs.

The second part of the thesis answers the last four question above and consists of four chapters:

4. *How can the effect of model inaccuracies be eliminated?*
   - **Chapter 7** considers feedforward control with uncertainties in the part of the model that is most prone to errors: the friction model.
5. *How can disturbances be rejected?*

- **Chapter 8** investigates the application of limit cycle theory in the cyclic motions of robotic arms with repetitive tasks.

6. *How can both model inaccuracies and disturbances be handled at the same time?*

   - **Chapter 9** introduces a new method that consists of finding robustly open loop stable cycles in simulation and then learning to follow these cycles on the robotic arm. After learning, the task can be performed with an open loop controller.

7. *Are feedforward techniques still useful when a small amount of feedback is available?*

   - **Chapter 10** investigates the accuracy of robots while varying the feedback gain and feedforward strategy.

The thesis ends with a general discussion and conclusion in Chapter 11.

Part I

Elasticity in robots

2

A Novel Spring Mechanism to Reduce Energy Consumption of Robotic Arms

Michiel Plooij and Martijn Wisse,
Largely appeared as a paper in:
IEEE/RSJ International Conference on Intelligent Robots and Systems 2012
pp. 2901-2908.

Abstract

Most conventional robotic arms use motors to accelerate the manipulator. This leads to an unnecessary high energy consumption when performing repetitive tasks. This chapter presents an approach to reduce energy consumption in robotic arms by performing its repetitive tasks with the help of a parallel spring mechanism. A special non-linear spring characteristic has been achieved by attaching a spring to two connected pulleys. This parallel spring mechanism provides for the accelerations of the manipulator without compromising its ability to vary the task parameters (the time per stroke, the displacement per stroke the grasping time and the payload). The energy consumption of the arm with the spring mechanism is compared to that of the same arm without the spring mechanism. Optimal control studies show that the robotic arm uses 22% less energy due to the spring mechanism. On the 2 DOF prototype, we achieved an energy reduction of 20%. The difference was due to model simplifications. With a spring mechanism, there is an extra energetic cost, because potential energy has to be stored into the spring during startup. This cost is equal to the total energy savings of the 2 DOF arm during 8 strokes. Next, there could have been an energetic cost to position the manipulator outside the equilibrium position. We have designed the spring mechanism in such a way that this holding cost is negligible for a range of start- and end positions. The performed experiments showed that the implementation of the proposed spring mechanism results in a reduction of the energy consumption while the arm is still able to handle varying task parameters.

2.1 Introduction

There is a growing need for energy efficient robotic systems in the field of industrial robots as well as in the field of mobile robotic platforms. Industrial robots need to be energy efficient because of the high cost of energy and the demand for sustainable industrial processes. Mobile robotic platforms (e.g. household robots) need to be energy efficient because they have to carry an energy storage (e.g. battery) with them. The challenge is to reduce the energy consumption of robotic systems, without compromising their performance.

One of the reasons why robotic manipulators use energy is the use of actuators to accelerate the manipulator. Most conventional robotic arms use motors as actuators. In repetitive tasks, the manipulator returns to the same state repetitively. An example

of such a task is a pick-and-place task, with the task parameters being the time per stroke, the distance per stroke, the grasping time and the payload. In practice, these task parameters vary per stroke. Theoretically, such repetitive tasks should only require the amount of energy equal to the potential energy added to the product, but this requires re-capturing energy when decelerating and a frictionless system. Recapturing energy only by means of using the motor as a generator is only efficient without gearbox and electrical losses, which is often not the case. We propose to apply a parallel spring mechanism, which stores energy during deceleration and releases it during acceleration.

Energy efficient repetitive motions have already been implemented in various applications. Akinfiev et. al. introduced the idea of using nontraditional drives in walking robots [5]. This led to the reduction in energy consumption of 65% in their robot. However, these nontraditional drives are fully determined so there is no freedom for varying the distance per stroke. Systems with repetitive motions that do allow for a variation of the distance per stroke are naturally oscillating mechanisms (e.g. mass-spring systems). These mechanisms have already successfully been used to reduce energy consumption in e.g. toothbrushes [74], compressors [145], shavers [107] and walking robots [39]. The idea of exploiting the natural motions of a system has also been applied on manipulators before. Williamson investigated control strategies for natural oscillating arms [239, 240, 241, 242]. However, he applied this on a robot that used a PD controller with low gains to create oscillating motions, instead of a mechanically oscillating device. Current research on mechanically oscillating mechanisms focusses on adaptive springs in series with the actuator [79, 229], which can introduce unwanted oscillations. Using springs in parallel with the actuator (as we propose in this chapter), does not introduce unwanted oscillations, but uses these oscillations to move energy efficiently.

The work most strongly related to this study is that by Akinfiev et. al. [3, 4, 13, 20, 193, 194, 232] who researched mechanically resonant robotic systems and designed interesting parallel spring mechanisms for those robots. The drawback of these mechanisms is that they lock into place at pre-determined positions, such that they are not able to vary the distance per stroke.

The state of the art spring mechanisms for robotic manipulators lack the ability to vary all the task parameters of pick-and-place tasks. Therefore, the key challenge is now to design a spring mechanism that reduces the energy consumption of robotic arms, while the arm is still able to handle a variation in the task parameters.

In this chapter, we introduce a novel parallel spring mechanism, we demonstrate its ability to handle varying task parameters, and we present the measured reduction of

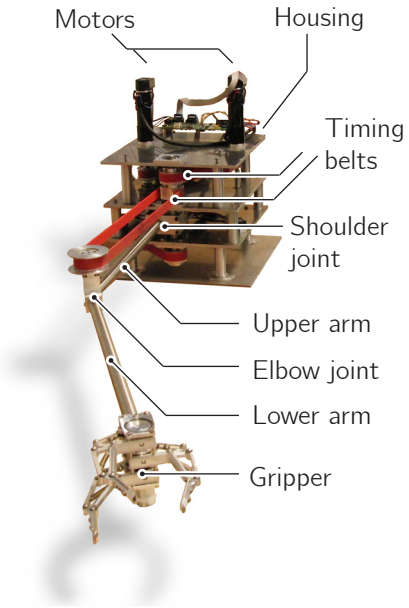


Figure 2.1: Prototype of the two DOF robotic arm.

energy consumption.[1]

The rest of this chapter is structured as follows. Section 2.2 explains the methods we used. Section 2.3 shows the working principle of the proposed spring mechanism. Section 2.4, 2.5 and 2.6 show the results from the optimal control study and the prototype experiments (see Fig. 2.1). Finally, the chapter ends with a discussion in section 2.7 and in section 2.8 we will conclude that the spring mechanism we implemented reduces the energy consumption while the arm is still able to vary the task parameters.

2.2 Methods

2.2.1 Studies

We studied the reduction in energy consumption by first optimally controlling the arm without the spring mechanism and then comparing its energy consumption with that

[1] In chapter 1, we mention that the proposed mechanism can be seen as a clutched elastic actuator with a singularity based locking mechanism, while we do not mention this througout this chapter. The reason is that this insight was obtained far after this chapter was published as a paper in 2012 and we chose to largely maintain the content of the original paper.

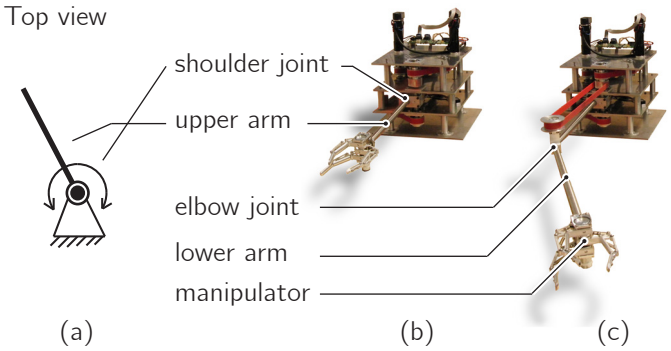


Figure 2.2: The three studies we performed, with a top view of the optimal control study and two 3D views of the prototype. a) An optimal control study of a realistic one DOF model with friction and copper losses. b) A one DOF prototype to confirm the results from the optimal control study. c) A two DOFs prototype to show that the principle can also be applied on a system with two DOFs. The second DOF is actuated by a motor at the base. The torques are transferred through a timing belt.

of the arm with the spring mechanism. We performed three studies[2]: a simulation model, a one DOF prototype and a two DOF prototype (see Fig. 2.2):

(a) **A one DOF optimal control study of the simulation model.** We obtained the optimal control strategy by applying optimal control theory [150] on the simulation models.

(b) **A one DOF prototype experiment.** The optimal control strategy was implemented on the prototype with one DOF (a rotation in the horizontal plane) by applying a feed-forward voltage.

(c) **A two DOF prototype experiment.** The same strategy as with one DOF was implemented on the prototype with two DOFs (two rotations in the horizontal plane), of which we will show preliminary results.

The DOFs in the three studies are all in the horizontal plane, which eliminates gravity. We did this because it was already shown in [231] that gravity can be eliminated by parallel springs.

Table 2.1: The task parameters and the variance in the pick/place areas

Parameter	Symbol	Value
Time per stroke	t_s	1 s
Distance between pick/place areas	d	0.5 m
Grasping time	t_g	0.5 s
Payload	m	1 kg
Width of pick/place areas	w	0.05 m

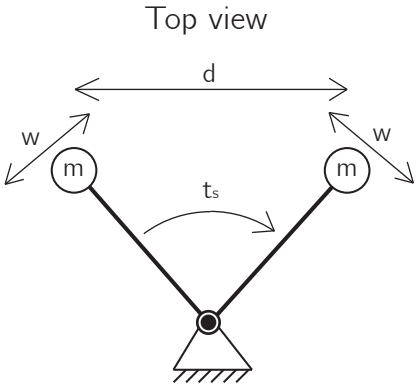


Figure 2.3: Visualization of the task parameters of a pick-and-place task that vary per stroke: the time per stroke (t_s), the distance per stroke (d) and the payload (m). The grasping time (not visualized) is the time the manipulator has to stand still at the pick/place areas. The width of the pick/place areas (w) represents the variance in the distance per stroke. The manipulator has to be able to stand still within these areas.

2.2.2 Task

A pick-and-place task is one of the most common repetitive tasks in industry. Such a task is mainly defined by four parameters: the time per stroke, the distance per stroke, the grasping time and the payload (including the gripper). These parameters are depicted in Fig. 2.3. There are no standard values for those parameters in industry, so the values listed in Table 2.1 are arbitrary. At the end of section 2.5 we will analyze how a variation in these parameters influences the energy consumption of the arm. We also defined the width of the pick/place areas, which represent the variance in the distance per stroke. We will need this parameter in the next paragraph.

[2]The original publication called the three studies *configurations*. We decided to change the terminology here, because study (a) and (b) study the same robotic configuration.

2.2.3 Measurements

We took three different measurements on the energy consumption of a spring loaded robotic arm:

- **The energy per stroke.** This is the energy that is needed to move from one pick/place area of the system to the other. We compared this value between the situations with and without the spring mechanism attached. In optimal control studies (eq. 1) and prototype experiments (eq. 2) we calculated this as follows:[3]

$$E = \int_{t_0}^{t_f} T_m \omega_m + R \left(\frac{T_m}{k_t} \right)^2 \mathrm{d}t \tag{2.1}$$

$$E = \int_{t_0}^{t_f} UI \mathrm{d}t \tag{2.2}$$

  Where ω_m is the angular speed of the motor, R is the terminal resistance of the motor, k_t is the torque constant of the motor, T_m is the torque of the motor, U is the voltage on the motor, I is the current through the motor and t_0 and t_f are the initial and final time respectively.[4]

- **The starting up energy.** This is the energy that is needed to move to the pick position at the start. The starting up energy increases when we attach a spring mechanism because the spring has to be stretched at the start. In both optimal control studies and prototype experiments, this is calculated by looking at the energy consumption while moving the system to the pick position.

  We also calculated the breakeven point, which is the number of strokes at which the cumulative energy saved due to the spring mechanism is equal to the starting up energy. This number is calculated by dividing the starting up energy by the net energy savings per stroke.

- **The standing still energy.** This is the energy needed when the motors are holding the system in place 0.06 rad outside an equilibrium position of the spring mechanism. This rotation corresponds with half of the width of the pick/place areas (w) as defined in Table 2.1. To make it comparable with the amount of energy consumed per stroke, we quantified this as the energy

[3] Note that these equations imply that we use brushed DC motors.

[4] The definitions of the initial and the final time were added in this chapter in comparison to the original publication.

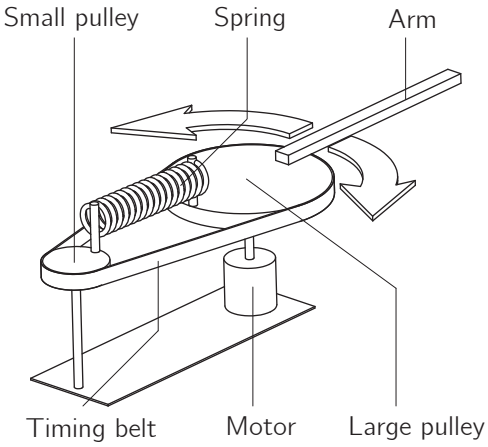


Figure 2.4: The concept of the spring mechanism. The arm is attached to the large pulley. This pulley is connected to the small pulley through a timing belt and a spring. The spring is stretched non-linearly with respect to the rotation of the arm due to the fact that the end points of the spring make rotational movements. The non-linear stretching of the spring leads to the characteristic of the spring mechanism.

consumed while standing still during the grasping time (t_g). In respectively optimal control studies and prototype experiments we calculate this as follows:

$$E = R\left(\frac{T_m}{k_t}\right)^2 t_g \tag{2.3}$$

$$E = UIt_g \tag{2.4}$$

Where t_g is the grasping time.

2.3 Novel spring Mechanism

A schematic drawing of the proposed novel spring mechanism is shown in Fig. 2.4. The key challenge in designing this mechanism was to reduce the energy consumption of the robotic arm, while the arm would still be able to handle a variation in the task parameters. We will now explain the requirements on the characteristic of this spring mechanism that led to the current design.

The characteristic of a spring mechanism can be expressed as the potential energy stored in the spring as a function of the displacement. In the robotic arm, this dis-

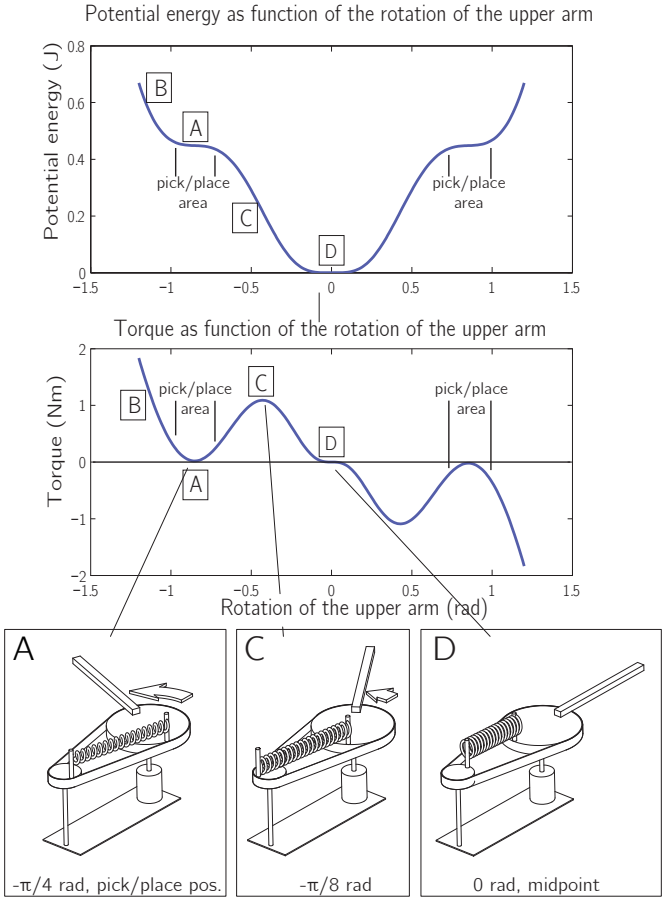


Figure 2.5: A visualization of the working principle of the proposed spring mechanism. The first plot shows the potential energy in the system as function of the rotation of the upper arm. The second plot shows the torque about the shoulder joint as function of the rotation of the upper arm. [A], [B] and [C] represent the requirements on the characteristic of the arm. [A]: At the pick/place areas, the derivative of the potential energy is equal to 0 J/rad. This means that there is no torque. [B]: Outside the pick/place areas, the potential energy increases. This means that there is a torque towards the pick/place areas. [C]: in between the pick/place areas, the potential energy decreases fast. This means that there is a torque towards the midpoint. [D]: At the midpoint, the potential energy has a minimum. The movement of the arm and the spring mechanism are visualized at the bottom. When the upper arm reaches an angle of $\pi/4$ rad, the small pulley has rotated for about 4.2 rad and the connection between the spring and the small pulley is moving towards the large pulley, with the same speed as the connection between the spring and the large pulley. This means that with a virtual small rotation of the arm, no extra energy is stored in the system, so the derivative of the potential energy graph is 0 J/rad.

placement is the rotation of the shoulder joint (Fig. 2.2b). The torque about the joint is equal to the derivative of the potential energy with respect to the displacement:

$$T = -\frac{\partial P}{\partial \theta} \tag{2.5}$$

Where P is the potential energy stored in the system and θ is the angular displacement.

We propose four requirements on the characteristic of a spring loaded robotic arm with a repetitive task. These requirements are based on ideas about how to support the pick-and-place task and have to be verified in future optimizations. The requirements are also indicated in Fig. 2.5.

A. The spring mechanism should not counteract the task. This means that when the system is at a pick/place area, the motor should not have to counteract the spring mechanism to keep the manipulator in place. There should be no net torque about the joint. This means that the derivative of the potential energy with respect to the rotation of the shoulder should be zero (or at least relatively low) at the pick/place areas.

B. The spring mechanism should always provide motions from one pick/place area to the other. This means that when the system is neither in the pick/place areas nor in between the pick/place areas, the spring mechanism has to provide a torque towards the pick/place areas. Therefore, the potential energy should increase outside the pick/place areas.

C. The characteristic between the pick/place areas should be such that the system can make fast motions. This means a high and fast drop in potential energy between the pick/place areas. Therefore, there is a torque towards the midpoint.

D. D. In between the pick/place areas, there should be a point where the potential energy reaches a minimum such that the kinetic energy reaches a maximum. This is called the midpoint.

Linear spring mechanisms do not meet requirement A. Therefore, we propose the spring mechanism as shown in Fig. 2.4, which has two equilibrium positions at the pick/place areas. This has the advantage of being energy efficient while still being able to vary all the task parameters. The time per stroke and the grasping time can be varied because the system has no eigenfrequency and can stand still at the pick/place

areas. The distance per stroke can be varied because the spring mechanism has low torques inside the pick/place areas. The payload can be varied because the working principle of the mechanism does not depend on the mass. At the end of section 2.5 we will analyze how varying these parameters influences the energy consumption of the arm.

The working principle of this mechanism is shown in Fig. 2.5. This mechanism is inspired by the work of Babitsky [13], who designed spring mechanisms with all kinds of characteristics. The potential energy E_P in the spring mechanism is equal to[5]:

$$E_P = \frac{1}{2}kx^2 + F_0 x \tag{2.6}$$

where

$$x = \sqrt{(a^2 + b^2)} - l_0 \tag{2.7}$$

with

$$a = r_2 \sin\frac{\theta r_1}{r_2} + r_1 \sin\theta \tag{2.8}$$

$$b = r_1 + l_0 + r_2 - r_1 \cos\theta - r_2 \cos\frac{\theta r_1}{r_2} \tag{2.9}$$

2.4 Optimal Control

In order to compare the system with and without the spring mechanism attached, the control strategies for both systems have to be optimal. A theoretical framework for this is given in the field of optimal control theory [150]. The pick-and-place task is an optimal control problem with fixed final time and fixed final state. We will now describe the optimal control problem for one DOF. First, we will describe the simulation model. Second, we will calculate the optimal control strategy.

[5] In the original publication, the parameters in these equations were not defined. They refer to the parameters in Table 2.2. The same table listed a parameter called *Transfer ratio*, which was unused and is therefore omitted in this chapter. Also note that the minimum distance between the two pulleys is equal to l_0, such that the spring is at rest when θ=0. Also note that θ is the angular displacement of the joint, meaning that θ is equal to the rotation of the large pulley.

Table 2.2: The parameters of the simulation model

Parameter	Symbol	Value
Length of arm	l	0.4 m
Inertia of the arm	I_{joint}	0.185 $kgm2$
Spring Stiffness	k	150 N/m
Pretension of the spring	F_0	6 N
Initial length of spring	l_0	10 cm
Radius of large pulley	r_1	10 cm
Radius of small pulley	r_2	2 cm
Coulomb friction	c_{cf}	0.48 Nm
Viscous friction	c_{vf}	0.00 Nms/rad
Torque dependent Coulomb friction	c_{tf}	13%

2.4.1 Simulation model

In the simulation model, we included three types of frictional losses: coulomb friction, viscous friction and torque dependent gearbox losses. These frictional losses were estimated during a system identification of the prototype. The parameters of the simulation model are listed in Table 2.2.

The equations of motion are[6]:

$$x = \begin{bmatrix} \theta \\ \omega \end{bmatrix} \tag{2.10}$$

$$\dot{x} = \begin{bmatrix} \omega \\ \left(\frac{T + T_s(\theta) - c_{vf}\omega - c_{cf} - c_{tf}|T|}{I_{joint}} \right) \end{bmatrix} \tag{2.11}$$

Where θ is the angle of the shoulder joint, ω is the speed of the shoulder joint, T is the toque exerted by the motor on the joint, T_s is the torque exerted by the spring mechanism on the joint, c_{vf} is the viscous friction coefficient, c_{cf} is the coulomb friction and I_{joint} is the mass moment of inertia about the joint.

[6]The friction terms in these equations are adjusted in comparison to the original publication, such that the notation is more in line with the rest of this thesis.

2.4.2 Optimal control strategy

The cost function is equal to the energy consumed per stroke which we can rewrite as:

$$J = \int_{t_0}^{t_f} T\omega + R\left(\frac{T}{k_t n}\right)^2 \mathrm{d}t \tag{2.12}$$

Where n is the gearbox ratio. We can now write down the Hamiltonian:

$$\begin{aligned} \mathcal{H} &= T\omega + R\left(\frac{T}{k_t n}\right)^2 + \lambda_1 \omega \\ &\quad + \lambda_2 \left(\frac{T + T_s(\theta) - c_{vf}\omega - c_{cf} - c_{tf}|T|}{I_{joint}}\right) \end{aligned} \tag{2.13}$$

Using the necessary condition for optimality $\frac{\partial \mathcal{H}}{\partial T} = 0$ we find that the optimal control strategy for T has to suffice:

$$T = \frac{\left(I_{joint}\lambda_2(1 \mp c_{tf}) - \omega\right) k_t^2 n^2}{2 I_{joint} R} \tag{2.14}$$

Where the $\mp$ depends on whether the mechanical power $T\omega$ is positive or negative. Note that the sign of T determines the sign of the mechanical power. Therefore, both signs should be evaluated and if for both signs the results contradicts itself, the torque T should be zero. The differential equations of the co-state λ can be derived from the Hamiltonian and the necessary condition. These equations are:

$$\dot{\lambda}_1 = \frac{\lambda_2}{I_{joint}} \frac{\partial T_s}{\partial \theta} \tag{2.15}$$

$$\dot{\lambda}_2 = -\lambda_1 + \frac{k_t^2 n^2 \omega}{2R} + \lambda_2 \left(\frac{c_{vf}}{I_{joint}} + \frac{k_t^2 n^2}{2 I_{joint} R} + \frac{c_{tf} k_t^2 n^2 \mathrm{sign}(\lambda_2 + I_{joint}\omega \mp c_{tf}\lambda_2)}{2 I_{joint} R}\right) \tag{2.16}$$

The starting conditions of the state x are given by the task parameters (Table 2.1). The starting conditions of the co-state λ have to be chosen such that the state at final time t_f suffices the task parameters. We found the initial co-state by using the fminsearch function in MATLAB for a multi-start optimization. The evaluation function of the optimization returned the distance in state-space to the goal state at time t_f as function of the initial co-state. We found that the multi-start optimization

Table 2.3: Results of the one DOF optimal control study with and without the spring mechanism attached.

Measurement	With spring	Without spring
Energy per stroke (J)	1.02	1.31
Starting up energy (J)	2.24	0.60
Standing still energy (J)	0.00	0.00

gave only one solution for the system without the spring mechanism attached and one solution for the system with the spring mechanism attached. This suggests that the control strategy we found is optimal. The results from the optimization are shown in Fig. 2.6 and Table 2.3.

In Fig. 2.6b we can see that the optimal control torques for both the system with and without the spring mechanism attached, consist of three phases. When the spring mechanism is not attached, we first see a phase of linear decreasing torque, then a phase with zero torque and then again a phase of linear decreasing torque. When the spring mechanism is attached, we first see a phase with a non-linear torque profile, then a phase with zero torque and then a phase of linear decreasing torque.

We can conclude that implementing the spring mechanism leads to an energy reduction of 22% per stroke, the breakeven point is at 6 strokes and the standing still energy is 0.00 J.

2.4.3 Parameter variation

Table 2.1 shows the values of the task parameters. We now want to know if the system can handle a variation in the task parameters. Therefore, we evaluate the energy consumption of the arm when we decrease the time per stroke with 10%, increase the displacement per stroke with 0.06 rad (the width of a pick/place area as defined in Table 2.1) or increase the payload with 10%. These variations are arbitrary, but we expect them to be a good representation of the variation in a pick-and-place task. Table 2.4 shows the energy consumption of the arm with and without the spring mechanism attached, when the parameters are varied. From this we can conclude that the energy savings due to the spring mechanism only decrease max 4 percent points when we vary the task parameters. The system is most vulnerable to a variation in the displacement per stroke. When we decrease the time per stroke, the energy savings of the arm even increase.

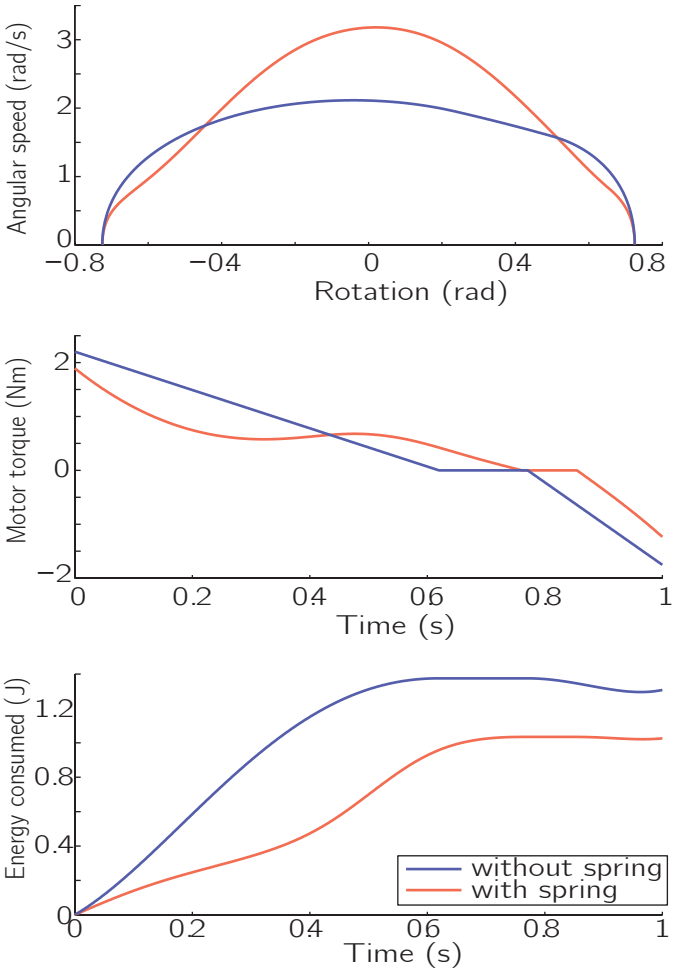


Figure 2.6: Results from the optimal control study. a) The movement of the arm visualized in state-space. b) The optimal control torque that is applied on the arm by the motor. c) The energy consumed during one stroke, while being optimally controlled. This graph shows that the system uses 22% less energy when the spring mechanism is attached. We can also see that without the spring mechanism attached, part of the energy consumed is recaptured at the end of the stroke by using the motor as a generator. The amount of energy recaptured is small because of electrical and frictional losses.

Table 2.4: The energy consumption per stroke when the time per stroke is decreased to 0.9 s, the angular displacement is increased to 1.6 rad and the payload is increased to 1.1 kg

Parameter set	Energy per stroke with spring (J)	Energy per stroke without spring (J)	Energy savings
Normal parameters	1.02	1.31	22%
Less time	1.11	1.45	23%
More displacement	1.13	1.38	18%
Additional payload	1.18	1.47	20%

Table 2.5: Design parameters of the spring loaded robotic arm and requirements on the stroke

Parameter	Symbol	Value
Length of arm	l	0.4 m
Additional payload	M	1 kg
Spring Stiffness	K	150 N/m
Initial length of spring	l_0	10 cm
Radius of large pulley	r_1	10 cm
Radius of small pulley	r_2	2 cm
Transfer ratio from small pulley to motor	r_3	1:1.8
Transfer ratio from motor to large pulley	r_4	1:3
Time per stroke	t	1 s
Rotation per stroke	θ	1.45 rad

2.5 Prototype experiments with one DOF

2.5.1 Dimensional Design one DOF

The one DOF implemented mechanism as shown in Fig. 2.7 is slightly different from the conceptual design in Fig. 2.4. A picture of the prototype (including the second DOF) can be seen in Fig. 2.1. The DOF is created by an 18x1.5mm stainless steel tube, connected with a joint. The motor is placed on a housing, which also contains the spring mechanism. AT3-gen III 16mm timing belts were used to transfer torques within the housing. The joint is actuated by a Maxon 60W RE30 motor with a gearbox ratio of 18:1. The timing belts provide an additional transfer ratio of 3:1. The design parameters are shown in Table 2.5. The measured characteristic of the spring mechanism is compared to the theoretical characteristic in Fig. 2.8.

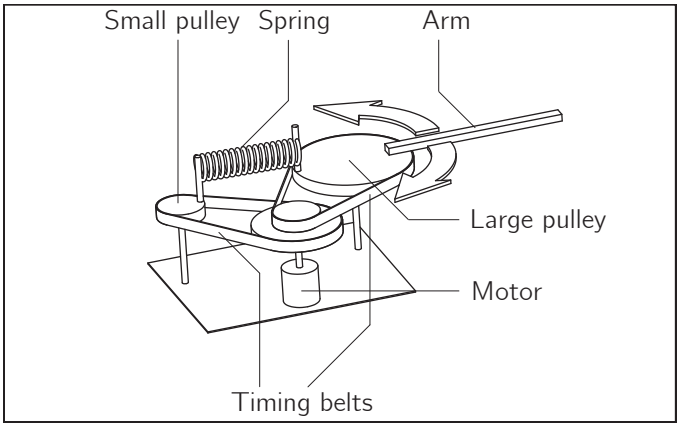


Figure 2.7: A schematic picture of the practical implementation of the spring mechanism in the one DOF prototype. In comparison to the concept, an extra timing belt and two extra pulleys were added because it was easier to drive the large pulley through a timing belt instead of directly connecting it to the motor and it was hard to get the right transfer ratio between the large and the small pulley.

2.5.2 Results

The optimal control strategy we found was implemented in the arm as a feed-forward voltage. The data of the movements of the prototype with one DOF is shown in Fig. 2.9. In Fig. 2.9a, we can see that the total angular displacement of the arm with the spring mechanism is equal to the total angular displacement of the arm without the spring mechanism. In Fig. 2.9b we can see that the current through the motor has about the same profile as the torque profile obtained in the optimal control studies (Fig. 2.6), although there are two main differences. The first main difference is the slow start-up effect, due to the fact that we cannot reach a current of about 2 A instantaneously[7]. The second main difference is that the current doesn't drop below zero as much as in the optimal control study. This is due to the fact that the friction caused more breaking torque than in simulation. In Fig. 2.9c we can see that the system with the spring mechanism uses less energy than the system without the spring mechanism. We can also see that in both cases, the energy consumption of the prototype is higher than in optimal control studies.

A comparison between the performances of the prototype with one DOF is shown in Table 2.6. We can conclude that with one DOF the system consumes 19% less

[7] In hindsight, we suspect that this slow response is mainly due to filtering. The current sensor on the control board we used for this chapter was not very precise and therefore, the current signal was filtered, leading to a slow response.

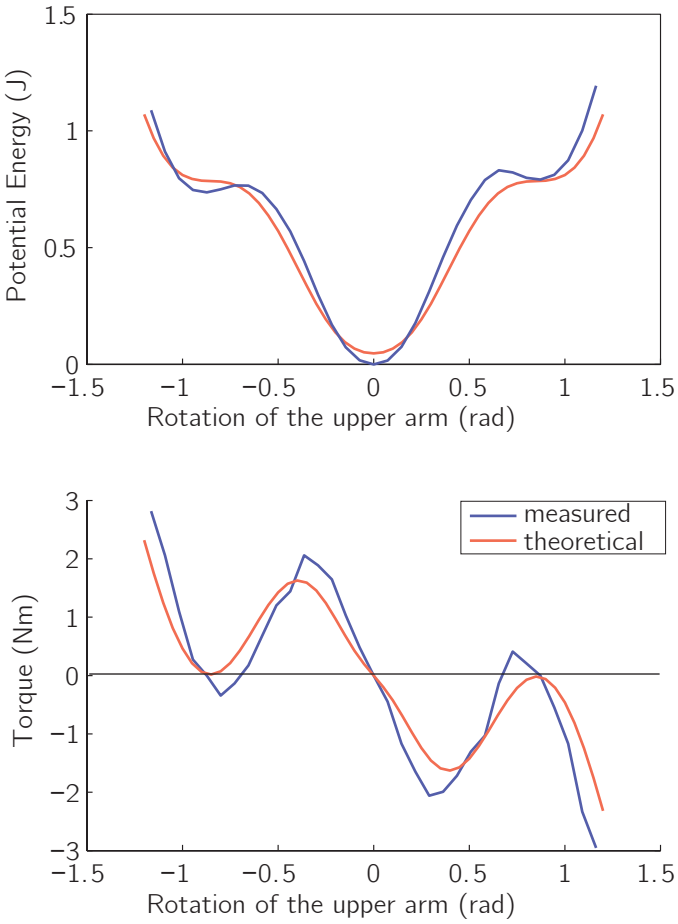


Figure 2.8: The characteristic of the spring mechanism. The solid line is obtained by measurements. The dotted line is the theoretical characteristic.

Table 2.6: Performance of the one DOF prototype with and without the spring mechanism attached. The values between the brackets represent the standard deviations.

Measurement	With spring	Without spring
Energy per stroke (J)	1.39 (± 0.02)	1.72 (± 0.01)
Starting up energy (J)	3.44 (± 0.03)	0.71 (± 0.01)
Standing still energy (J)	0.00 (± 0.00)	0.00 (± 0.00)

energy per stroke when the spring mechanism is attached, the breakeven point is at 9 strokes and the standing still energy is 0 J.

2.6 Preliminary Results for two DOFs

We add a second DOF to make the system more applicable. A picture of the two DOF prototype can be seen in Fig. 2.1. The second DOF is created by an 18x1.5mm stainless steel tube, connected with the elbow joint. This elbow joint is actuated by a Maxon 60W RE30 motor with a gearbox ratio of 66:1. The timing belts provide an additional transfer ratio of 3:1. The motor is placed in the housing and the torques are transmitted to the elbow through a timing belt, which creates a parallel mechanism.

The optimal control strategy we found for one DOF was implemented in the arm as a feed-forward voltage for the motor on the shoulder joint. The motor on the elbow joint was controlled by a PID controller to keep a constant angle of 0 rad. Due to the timing belt that functions as a parallel mechanism, 0 rad means that the lower arm constantly points to the same direction.

Fig. 2.10 shows the energy consumption of the prototype with two DOFs. The energy consumptions of motor 1 and motor 2 are added. A comparison between the performance of the prototype with two DOFs with and without the spring mechanism is shown in Table 2.7. From Table 2.7, we can conclude that with two DOFs the system consumes 20% less energy when the spring mechanism is attached, the breakeven point is at 8 strokes and the standing still energy is 0 J.

2.7 Discussion

In this study we showed that using a parallel spring mechanism in robotic systems with repetitive tasks, can lead to a reduction in energy consumption, while the performance of the system remains the same. The characteristic of the spring mechanism can be adjusted such that it fits the requirements of the repetitive tasks.

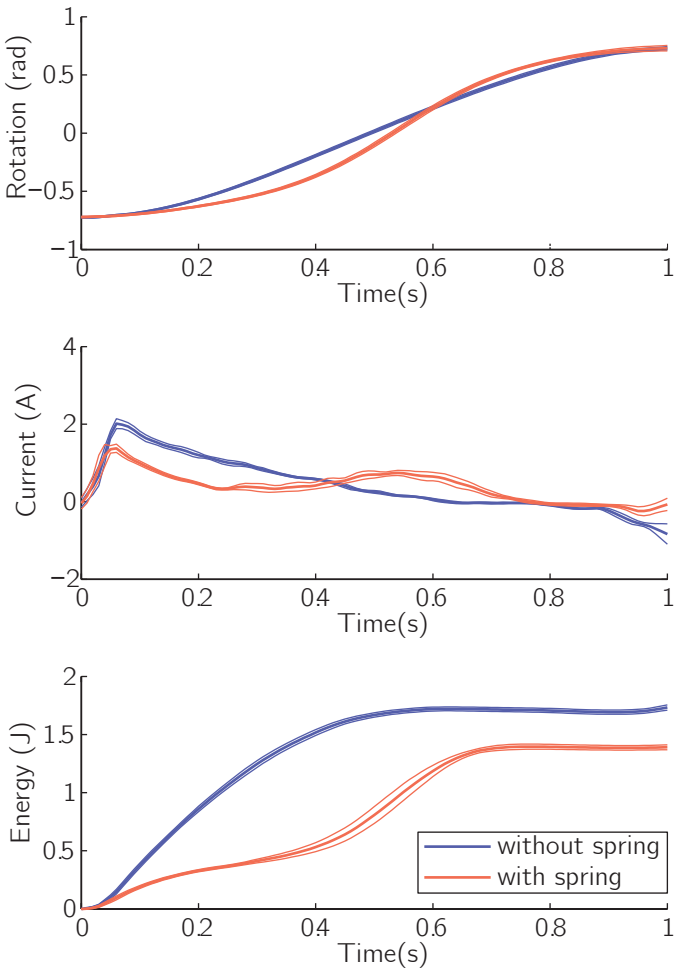


Figure 2.9: Results of the one DOF prototype experiments. The thick line shows the mean of the measurements, the thin lines show the standard deviation. The dotted lines show the data of the arm without the spring mechanism (n=18), the solid lines show the data of the arm with the spring mechanism (n=14). a) The movement of the arm, visualized as the angular displacement of the arm as function of the time. b) The current through the motor as function of the time. c) The energy consumed by the motor during one stroke. This graph shows that the arm uses 19% less energy when the spring mechanism is attached.

Table 2.7: Performance of the two DOF prototype with and without the spring mechanism attached. The values between the brackets represent the standard deviations.

Measurement	With spring	Without spring
Energy per stroke (J)	1.77 (± 0.05)	2.21 (± 0.04)
Starting up energy (J)	4.05 (± 0.04)	0.86 (± 0.03)
Standing still energy (J)	0.00 (± 0.00)	0.00 (± 0.00)

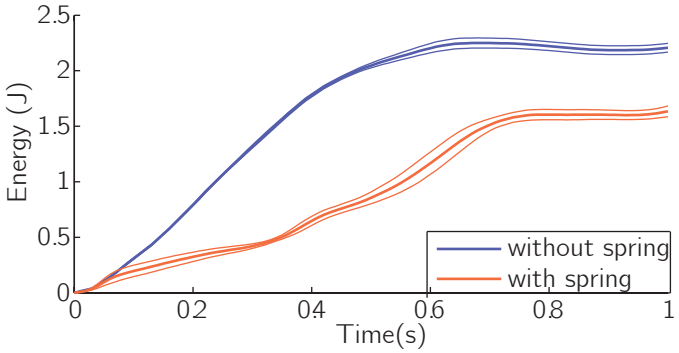


Figure 2.10: The accumulated energy consumption of the two DOF prototype with (n=18) and without (n=17) the spring. In this graph, the energy consumptions of motor 1 and motor 2 are added. The thick lines show the mean over different strokes, the thin lines show the standard deviations. This graph shows that the arm uses 20% less energy when the spring mechanism is attached.

The optimal control study showed that an energy reduction of 22% per stroke can be achieved. In prototype experiments, we achieved an energy reduction of 19% per stroke (for one DOF). The main difference between the model and the prototype is the electrical circuit, which caused two additional sources of energy losses. The first additional energy loss was the electrical resistance. The motor has a specified terminal resistance of 0.61 Ω, which we used in the simulation model. The measured terminal resistance is equal to 0.75 Ω. The effective electrical resistance was further increased by the voltage drop over the brushes of the motor and the inductance of the motor, which we did not account for in the model. The second additional energy loss was due to fact that we can not put a step function on the current. Therefore, both the arm with and without the spring mechanism were not controlled exactly as they were controlled in the optimal control study.

Due to the additional energy losses, both the arm with and without the spring mechanism attached used more energy than in the optimal control study. However, the absolute amount of energy saved per stroke, is comparable. In optimal control, the implementation of the spring mechanism caused a reduction in energy consumption of 0.29 J per stroke. On the prototype (with one DOF) this was 0.33 J per stroke.

The parameters of the morphology of the prototype were not optimized yet. We expect that the theoretical 22% energy reduction can be increased by optimizing the spring mechanism for a specific task. Parameters that can be varied include the spring stiffness and the radii of the pulleys of the spring mechanism.

During the system identification, we found higher frictional constants than expected. The main sources of friction were the gearboxes. Future research has to include the reduction of friction in the system. We expect a self-reinforcing effect: First, reducing the frictional losses will increase the energy savings due to parallel spring mechanisms. Next, due to the implementation of parallel spring mechanisms, the torque requirements will be reduced. Finally, reduced torque requirements will allow for lower gearbox ratios, which will lead to lower frictional losses, increasing the energy savings due to parallel spring mechanisms.

2.8 Conclusions

In this chapter we presented a robotic arm that uses a parallel spring mechanism to move more energy efficiently. We can conclude that with using a parallel spring mechanism, the natural dynamics of a system can be adjusted such that they support the required motion of the system. Doing so leads to a reduction in energy consumption without compromising the systems performance. Theoretically, the implementation

of the spring mechanism in the robotic arm leads to a reduction in energy consumption of 22%. In prototype experiments we confirmed that the system saves energy, for a one DOF (19% per stroke) as well as for a two DOF setup (20% per stroke).

Acknowledgement

The authors would like to thank G. Liqui Lung for helping with the electronical work and J. van Frankenhuyzen for his tips on the mechanical design. This work is part of the research programme STW, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO).

3

Review of locking mechanisms used in robotics

Michiel Plooij, Glenn Mathijssen, Pierre Cherelle, Dirk Lefeber and Bram Vanderborght,
IEEE Robotics and Automation Magazine
Vol. 22 (2015), issue 1, pp. 106-117.

Abstract

Locking devices are widely used in robotics, for instance to lock springs, joints or to reconfigure robots. This chapter classifies the locking devices currently described in literature and performs a comparative study. Designers can therefore better determine which locking device best matches the needs of their application. The locking devices are divided into three main categories based on different locking principles: mechanical locking, friction-based locking and singularity locking. Different locking devices in each category can be passive or active. Based on an extensive literature survey, the chapter summarizes the findings by comparing different locking devices on a set of properties of an ideal locking device.

.

3.1 Introduction

There are numerous robotic systems that utilize locking devices. Although the reasons for using such devices vary across applications, there are two main reasons: energy management and reconfiguration.

The first and most often cited reason for using locking devices is the energy management in robotic systems. Especially in the field of mobile robots, energy consumption is an important performance criterion. Examples include household robots [135], legged robots [39] and aerial robots [151, 209]. Energy consumption is also critical for wearable devices such as prostheses [11, 214] and exoskeletons [49, 177]. Over the years, the field of robotics has evolved from using stiff actuation to exploiting springs in series and in parallel with the actuator [228]. The advantage of using springs is that they provide the possibility of storing and releasing energy mechanically, which can lower the energy consumption of the actuator [229]. The disadvantage however is that they are non-controllable energy buffers. Two solutions have been proposed to control the realease of the energy stored in springs. The first is to use a continuously variable transmission (CVT) to adjust the position-torque relation of the spring [65]. However, those CVTs are still not developed well enough to be widely applied in robots. The second solution is to use locking devices to control the timing of the energy release. Such locking devices are discussed in this chapter.

The second most cited reason for using locking devices is to reconfigure a robotic system. Such systems consist of multiple modules that can be connected and discon-

nected to form different configurations that perform different tasks. Those modules are connected and disconnected using locking devices of various designs [62, 67, 68].

The fundamental principles of many locking mechanisms are quite old and have been described in multiple books, such as [22, 128, 154, 182, 191]. However, the number of applications of locking mechanisms in robotics is rapidly growing. Almost half of the citations on locking mechanisms are describing mechanisms that were implemented from 2010. Therefore, this chapter discusses the potential of them in robotic applications.

Each locking device principle has advantages and disadvantages, with no single design fulfilling all of the requirements of the ideal locking device. On the other hand, not every application requires a locking device that fulfills all the requirements. Therefore, this chapter provides an overview of all locking devices useful for robotic applications and discusses their properties, advantages and disadvantages, starting with the description of an ideal locking device.

3.2 Locking devices

3.2.1 What is an ideal locking device?

A locking device is a device that switches between allowing and preventing relative motion between two parts. The requirements of a locking device differ across applications. This section lists all the requirements one might have on a locking device. In the rest of the chapter locking devices are evaluated based on how well they meet each of these requirements, such that the reader will be able to select the most suitable locking device for their application. An ideal locking device has the following properties (in random order):

- **Adjustable locking directions.** The device can switch between locking in zero, one or two directions.
- **Unlocking while under load.** While there is a load on the locking device, it should still be able to unlock.
- **Low energy consumption.** While the device is (un)locked or while it is (un)locking, it should not consume energy.
- **Lockable in any position.** The device has an infinite amount of locking positions.
- **Compact.** The device should be small relative to its application.

- **Lightweight.** The device should be lightweight relative to its application.
- **Short switching time.** The device switches instantaneously.
- **Inexpensive** The device should be inexpensive relative to its application.
- **High locking force.** The device has unlimited locking torque.

Some locking devices can also be used as controllable brakes, meaning that the locking torque can be controlled and when the external torque is higher than the locking torque, the brake slips. Although this property is not necessary for a locking device, in some application it might be an advantage and therefore it will be considered as a side note in this chapter.

3.2.2 Categorization

Numerous locking devices are presented in literature. In this chapter, the locking devices are categorized into three main groups, based on three locking principles (see Fig. 3.1). The three distinguishable categories are:

1. **Mechanical locking:** The position of a mechanical component determines the locking or unlocking. Examples of such components are wedges and pawls. This position can be determined by an actuator or can depend on e.g. the position of a joint or the direction of the velocity. These locking devices are discussed in section 3.3.
2. **Friction-based locking:** Engaging or disengaging two friction surfaces determines if the joint is locked or unlocked. This engagement can be determined by an actuator or can depend on e.g. the position of a joint or the direction of the velocity. These locking devices are discussed in section 3.4.
3. **Singularity locking:** Singularities in mechanisms cause a transfer ratio[1] to go to infinity. In such a singular position, the locking device features an infinitely high locking force and an infinitely small unlocking force. These locking devices are discussed in section 3.5.

Each of these three groups can be subdivided into active locking devices and passive locking devices (see Fig. 3.1). Contrary to passive locking devices, active locking devices use an actuator to change the timing of the locking, the locking position or

[1] Throughout this thesis, we use the term *transfer ratio* for both a constant and varying ratio. The latter is often refered to as the *mechanical advantage*

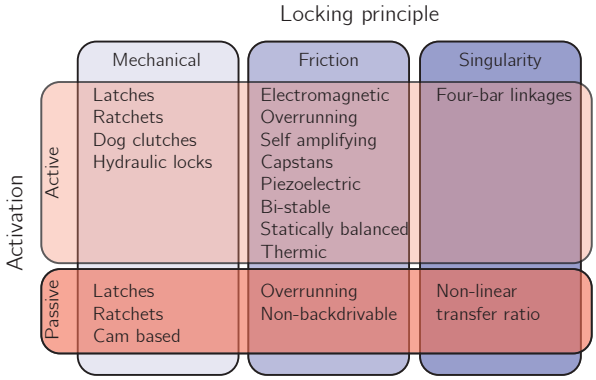


Figure 3.1: Classification of the locking devices into three main categories: mechanical locking, friction-based locking and singularity locking. All three can be divided into actuated and passive devices.

the locking torque. Therefore, where passive devices do not require any electronics or control, active devices often use some kind of state machine controller. Section 3.6 compares all the different devices based on the properties of an ideal locking device described above and section 3.7 provides a guide for the selection of a suitable locking mechanism.

3.3 Locking devices based on mechanical locking

Mechanical locking devices all use some kind of obstruction of a part by another part. For instance, in the latch in Fig. 3.2a, the hook obstructs the pawl. Sometimes, it is hard to distinguish between mechanical locking and friction based locking devices. The criterion for the categorization in this chapter is whether the device also works in a world without friction. If so, it is categorized as a mechanical locking device, if not, as a friction based device. This section describes mechanical locking devices in literature and indicates which are active and which are passive.

3.3.1 Latches

Latches consist of a pawl and a hook that can generally lock at one position (see Fig. 3.2a). Active latches use an actuator to change the position of the pawl or the hook for two reasons. Firstly, the positions can be changed to determine if the hook and pawl obstruct each other at the locking position, and thereby enabling or disabling the latch mechanism. Secondly, the positions can be changed to adjust the locking position itself. Passive latches are latches of which locking and unlocking is

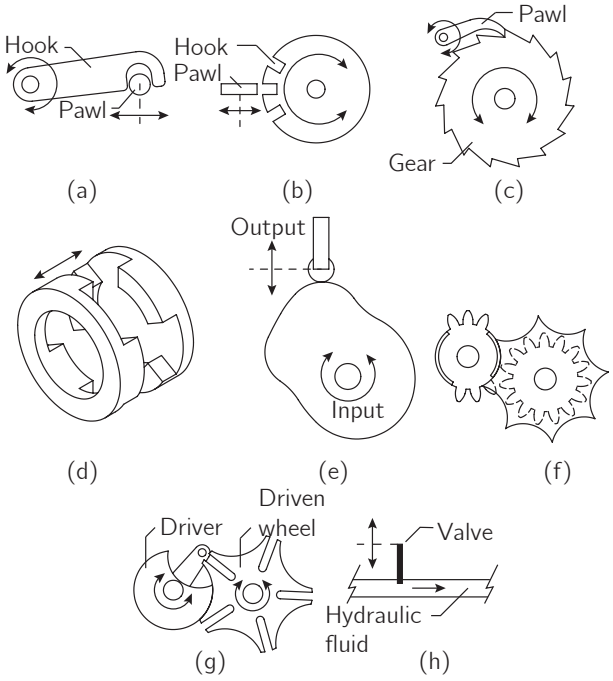


Figure 3.2: This figure shows the types of mechanical locking devices with: (a) A latch with one locking position, (b), a latch with multiple locking positions, (c) a ratchet, (d) a dog clutch, (e) a cam-based locking device: cam follower, (f) a cam-based locking device: mutilated gears, (g) a cam-based locking device: geneva mechanism and (h) an hydraulic lock.

caused by the position or velocity of components of the lock. This can be used in robots that have to lock or unlock based on the state of the robot.

Active latches have been used in several legged robots and medical devices. Firstly, Collins and Kuo [40] used a latch in their energy recycling foot prosthesis to temporarily lock a loaded spring. At the beginning of the push-off phase, the latch was released and the energy returned. Secondly, Collins and Ruina [38] and Wisse et al. [243] used latches in the knees of their bi-pedal walking robots to lock and unlock the knee joints, depending on the phase of the walking cycle. And thirdly, Karssen [99] used a latch in the knee of the bi-pedal running robot Phides to attach a parallel spring to the knee joint during the stance phase and detach it during the swing phase.

There are other examples of robotic applications that use active latches. Firstly, Tavakoli et al. [124] used a latch to lock two trunks of a flexible gripper. An SMA actuator was used to unlock the latch and disconnect the trunks again. Secondly, Wright et al. [245] used a latch in their snake robot to (un)lock a brake on the segments of the robot. This latch was actuated by an SMA actuator and held in place by a bi-stable spring. And finally, multiple modular robots use latches to join different modules [68]. A recent design of a latching mechanism in modular robots was made by Parrot et al. [160], who designed a genderless latching mechanism that can be disconnected by any of the two connected modules.

Latches can also lock at multiple positions, for instance having one pawl and multiple hooks (see Fig. 3.2b). Such latches have been used by Mitsui et al. [137] in a robotic hand to lock joints in an underactuated finger. While one DOF in the fingers is locked, the other DOFs are moved by the actuator. This allows the hand to perform different kinds of grasps with a limited amount of actuators, causing the arm to be lightweight and compact.

Unal et al. [226] used passive latches in their ankle-knee prosthesis. Based on the phase of the walking cycle, multiple latches lock and unlock in order to control the energetic coupling between the ankle and the knee during the swing phase and the stance phase.

3.3.2 Ratchets

A ratchet consists of a round gear or linear rack with teeth and a pivoting, actuated pawl that engages with the teeth and performs the locking (see Fig. 3.2c). In active ratchets, the pawl is controlled by an actuator that determines engagement or disengagement. There are two possibilities for the loading of the pawl: compression (shown in Fig. 3.2c) or tension. The pawl of the ratchet mechanism can also be powered by a spring instead of an actuator, making it passive. Such a passive ratchet

allows continuous linear or rotary motion in only one direction while preventing motion in the opposite direction.

Active ratchets are used in multiple prostheses. Firstly, Geeroms et al. [64] used a ratchet in the weight acceptance mechanism of the knee in an active knee-ankle prosthesis. During the stance phase the knee behaves like a spring, so the spring is locked parallel to the knee joint. A disadvantage of this device is that it is difficult to unlock under load, which in this application is not necessary since the ratchet is not highly loaded at the end of the stance phase. Secondly, Brackx et al. [25] used a ratchet mechanism in the ankle prosthesis AMPFoot 1 to change the internal configuration of the foot between the loading phase and the push-off phase.

A passive ratchet is used by Li et al. [117] in an energy storage device for a spherical hopping robot. Wiggin et al. [238] also used passive ratchets to design a 'smart clutch'. This smart clutch stores energy in the parallel springs to provide mechanical assistance during the stance phase and allow free rotation during the swing phase.

3.3.3 Dog clutch

A dog clutch consists of two parts that match each other's shape (see Fig. 3.2d). When the two parts are engaged, the relative rotation between the two parts is blocked; otherwise, the two parts can rotate independently. Dog clutches are discrete by nature, but have a large locking torque to weight ratio.

Although the dog clutch is a relatively uncommon locking mechanism, it has been used in several robots. However, only examples of active dog clutches are found in literature. Elliott et al. [54] used such a clutch to attach and detach a parallel spring to the joint of a knee exoskeleton. Kossett et al. [108, 109] used a dog clutch to switch between two modes of the robot: ground mode and flight mode. Palpacelli et al. [156] used dog clutches to lock one or more degrees of freedom of their spherical joints. Finally, a special type of dog clutch was designed by Kern et al. [102]. They use a rope that runs through several parts with mating surfaces. When the rope is pulled, the parts are pulled together, locking the parts. This locking device was inspired by the mammalian spine.

3.3.4 Cam-based locking devices

Cam-based locking devices consist of two separate cam surfaces that have complementary shapes and are engaged (see Figs. 3.2e, 3.2f and 3.2g). In principle, the engagement of the two surfaces can be actuated, resulting in an active cam-based locking device. The examples found in literature, however, are passive cam-based locking devices that lock due to the position of components of the lock. Although

the working principle itself is independent of friction, the relative motion between the cams induces friction in the system. The shocks introduced at the transition between locked and unlocked phase are, on the other hand, relatively small compared to for example ratchets.

As described by Bickford and Martin [22, 128], a wide variety of intermittent mechanisms is equipped with a cam-based locking device. Intermittent mechanisms consist of two members, the driver and driven member. As the name depicts, an intermittent mechanism transforms the continuous movement of the driver to an intermittent movement of the driven member. As such, the driven member has a dwell and a motion phase. During the dwell phase, a cam-based locking device ensures the driven member is locked. A first example is the Geneva drive or Maltese cross, which is widely described in literature (see Fig. 3.2f). Another example of an intermittent mechanism with cam-based locking devices is the mutilated gear mechanism (see Fig. 3.2g) which has recently been adopted by Mathijssen et al. [130, 131] in a novel compliant actuator in order to lock parallel springs.

3.3.5 Hydraulic lock

A mechanical locking principle that is very different from the other principles is hydraulic locking. In hydraulic locking devices, the fluid is blocked by closing a valve in the hydraulic circuit. The advantages of hydraulic locks are that the locking force is high with respect to the actuation force and that the system can be locked in every position. However, such a lock can only be used in hydraulic systems, which might be undesired for other reasons such as leakage or friction.

Such a locking device was used by Mauch [132] in the SWING-N-STANCE above knee prosthesis. In this prosthesis, the knee is locked or unlocked, depending on the task that is performed. Another example of a prosthesis that uses an hydraulic lock is the Otto Bock 3R80 knee prosthesis [155], where a stance phase valve is closed when the body weight is put on the knee joint. When the body weight is removed, at the end of the stance phase, the damping is reduced and the knee is able to flex.

3.4 Friction-based locking devices

Friction based locking devices depend on friction in order to prevent motion between two parts. As described in detail by Orthwein [154], the friction force can be generated using various mechanisms, such as disk brakes, drum brakes, cone brakes and band brakes. Since the amount of friction between two surfaces is limited by the normal force, friction based locking devices generally have a limited locking torque.

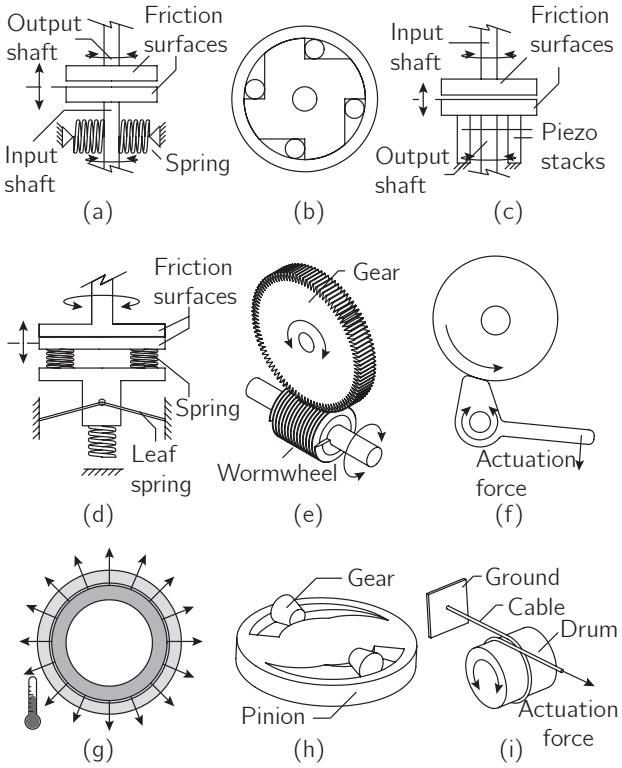


Figure 3.3: This figure shows the types of friction based locking devices with: (a) a bi-stable brake, (b) an overrunning clutch, (c) a piezo actuated brake, (d) a statically balanced brake, (e) a wormwheel, (f) a self-engaging brake, (g) a thermic lock, (h) a self-engaging pinion-gear mechanism and (i) a capstan.

On the other hand, friction based locking devices can lock at every position and can often be used as a controllable brake.

Most friction based locking devices have some kind of force amplifier, to amplify the actuation force perpendicular to the two surfaces, leading to a high friction. If no force amplification is used, the actuation force needs to be high. This section describes the friction based devices used in robotic applications and indicates which are active and which are passive. The descriptions also include the type of force amplification that is used.

3.4.1 Electromagnetic brake

A well known type of brake in robotics is the electromagnetic brake. In this type of brake, two friction surfaces are engaged by the attractive force between a permanent

magnet and an electromagnet. Such brakes are relatively simple and cheap, but often consume a relatively large amount of energy. An electromagnetic brake can be locked when powered, or locked when unpowered. Since at least switching from one state to the other requires activation of the electromagnet, an electromagnetic brake is considered active.

Multiple robots use an electromagnetic brake in parallel with a motor to either increase safety or decrease the energy consumption. Hirzinger et al. [87] used brakes on all joints of their robotic arm in order to increase the safety. These brakes make sure that the robot stands still when it is powered down. Sugahara et al. [212] used electromagnetic brakes on the joints of their bi-pedal walking robot to lower the energy consumption. When the robot stands still, the brakes hold the joints in place. This energy saving principle was also used by Rouse et al. [185] in their prosthetic knee. Another reason for using electromagnetic brakes is to adjust the joint impedance. This was done by Morita and Sugano [144] on their robotic arm and by Sarakoglou et al. [187] in their actuator with controllable mechanical damping. The latter design does not incorporate a classical electromagnet but a DC motor with a ball screw to actuate the brake.

3.4.2 Overrunning clutch

An overrunning clutch has an inner and outer raceway similar to bearings, with cylinders or balls (rollers) between the two raceways and a wedge on one side (see Fig. 3.3b). The relative rotational speed of the two raceways determines whether the overrunning clutch locks or not. The rollers of an overrunning clutch can also be pushed in the wedge using small springs or an actuator. The equivalent linear locking device uses a friction lever which is mounted around a translating stick. When this friction lever is rotated, it locks the stick.

An active overrunning clutch is the bi-directional overrunning clutch by Hild and Siedel et al. [86, 202]. In this clutch, the balls are replaced by wedges that can be placed in both directions, making locking possible in two directions.

In the asymmetric compliant antagonistic joint developed by Tsagarakis et al. [222], a two side acting passive overrunning clutch mechanism was deployed to achieve efficient regulation and maintenance of the pretension of the spring. As a result, the electric motor is unloaded when not rotating, while still a low friction and highly backdrivable linear transmission system can be used for the motor. A miniature passive overrunning clutch was designed by Controzzi et al. [42] for implementation in the fingers of their robotic hand. Li et al. [118], designed a knee brace for energy harvesting using an overrunning clutch. And in the knee orthosis by Shamei et al.

[198], an overrunning clutch is used to attach a support spring during the stance phase of the gait. A linear variant of a passive overrunning clutch was used in the ankle prosthesis of Collins and Kuo [40], to store energy in a spring, lock the spring and release the energy at a specific moment during the gait.

3.4.3 Non-backdrivable gearing

Non-backdrivable gearing are gears that can only be driven from one side. In robotics, mostly lead-screw and worm drives (see Fig. 3.3e) have been employed. The non-backdrivability is due to the shear friction, which also results in a very low efficiency. As a result, non back-drivable gears are passive locking devices.

There are two main reasons for using non-backdrivable gearing. Firstly, the non-backdrivability is a virtue to protect actuators during human robot interaction (HRI), such as in the social robot Probo [69] and the robot fingers designed by Morita and Sugano [143]. Secondly, the non-backdrivability avoids energy losses and overheating of the motors by static load cancellation. Examples are worm drives in the motors in the 1 DOF anthropomorphic arm by Gu et al. [75], the worm drive and lead screw after the variable stiffness motor in the MACCEPA actuator for the step rehabilitation robot ALTACRO [30], the powered elbow orthosis of Vanderniepen et al. [230] and the reconfigurable robot of Baca et al. [14].

3.4.4 Self amplifying brakes

Some locking devices used in robotics are based on self amplifying brakes (see Figs. 3.3f and 3.3h). While only a small force is required to engage both sides of the brake, the self amplifying effect enables to lock high forces. The self amplifying effect depends on the direction of the relative motion between the two components of the mechanism. In one direction, the mechanism will amplify the normal force since the friction between the two friction surfaces will pull them together. In the other direction, the mechanism will weaken the normal force and will therefore not lock as strongly. As such, this principle is suitable for one-direction locking only. An example of this is shown in Fig. 3.3f. When the small force to engage both surfaces is delivered by a spring, the amplifying brakes are passive. When this is delivered by a motor, the amplifying brakes are active and allow to not lock in either of both directions. The applications found in literature are active self ampifying brakes.

Kim and Choi [106] used an active self amplifying brake for an automotive clutch. In their clutch, the normal force is amplified by a wedge-like pinion gear mechanism, which transfers a relative rotational motion into a small translational motion, pushing the friction plates stronger together (see Fig. 3.3h). Peerdeman et al. [164] used an active self amplifying mechanism in their underactuated robotic hand to lock the

joints of the fingers in order to perform certain grasps. This application is similar to the robotic hand discussed in section 3.3.1.

3.4.5 Capstan

A special type of a self amplifying brake is the capstan. Capstans use the friction between a pulley and a cable to brake the cable with respect to the pulley (see Fig. 3.3i). When the cable is tightened around the pulley, the pulley pulls on the cable, which tightens the cable even more. In robotics, a capstan is mainly used as a means to actuate a cable. Werkmeister et al. [236] and Baser et al. [16] studied the capstan drive stiffness and slip error respectively. A capstan cable drive is used, for example, in the WAM^{TM} Arm from Barret Technology, in an anthropomorphic dexterous hand [208], in a low-cost compliant 7-DOF robotic manipulator [178], and in a five degree-of freedom haptic arm exoskeleton [205].

By controlling the force on the cable, a capstan can also be used as an active locking device. Instead of tightening a cable to increase the friction with respect to the pulley, a torsion spring is tensioned to reduce the outer diameter and wrap around the pulley. The elasticity of the spring facilitates the unlocking compared to a cable. However, locking requires more force due to the bending stiffness. This principle is used in the clutch on the parallel spring of the actuator proposed by Haeufle et al. [77].

3.4.6 Fluid brakes and clutches

Fluid brakes or clutches use a fluid which consists of micrometer-sized particles mixed with any kind of liquid. Of these fluids, electro-rheological (ER) and magneto-rheological (MR) fluids are studied most in literature. The fluid is placed around a rotating part like a shaft causing a small damping. By changing the electric or magnetic field around this fluid, the particles are aligned, which increases the damping. So by controlling the magnetic field, the damping of a joint can be controlled [207].

As predicted by Wang and Meng [233], fluid devices are currently being adopted increasingly in robotics applications. One of the applications is prosthetics where MR brakes are used to provide controllable resistance. This is for example interesting in a knee orthosis or prosthesis where significant portions of the gait consist of negative knee power, which can be delivered by a damper. This is done in the orthoses of Weinberg et al. [235], Chen and Liao [29], Kikuchi et al. [104] and the prosthesis of Herr and Wilkenfeld [85]. MR dampers are also incorporated in robotic arms [247] and haptic devices [181]. More recently a magnetic particle brake was installed by Shin et al. [199] in combination with Pneumatic Artificial Muscles (PAM) in a hybrid actuation concept to improve the control performance of the muscles.

3.4.7 Piezo actuated brake

Piezo actuated brakes use piezo actuators to create a normal force between two friction surfaces (see Fig. 3.3c). These actuators typically have a small stroke and therefore the alignment of the components is crucial. However, they are suited for generating a large force for a large amount of time, making them suitable for actuating brakes. The idea of a piezoelectric brake was already patented in 1989 by Yamatoh et al. [246] and was also used in the patented actuator of Hanley et al. [80]. Piezo actuated brakes are active locking devices since an electric field should be provided to initiate the locking.

Piezoelectric brakes have been used in robotic applications as well. Firstly, such a brake was used in an early version of the DLR arm as a safety brake for when the power is down [88]. Secondly, the passive haptic robot PTER used piezoelectric brakes to brake its joints [149]. And finally, Laffranchi et al. [114] used a piezo actuated brake to vary the damping coefficients of joints. In order to do so, the normal force is varied, depending on the desired damping and the joint velocity.

3.4.8 Bi-stable brakes

One way to reduce the actuation needed for maintaining the normal force in friction based locking devices is using a bi-stable mechanism (see Fig. 3.3a). Such a mechanism has two stable equilibrium positions with one unstable equilibrium position in between. It doesn't require force once the mechanism is switched, but it does require force to switch from one side of the unstable equilibrium position to the other and is therefore an active locking device. In bi-stable brakes, this spring is used to switch between the engaged and disengaged state of the brake.

This idea was already patented in 1973 by Parmerlee [159], but has not been widely used in robotics. Bi-stable locking devices have been used by Cho et al. [36] in electronics, to hold a lens in place in optical board-to-board communication. Although this example might not be very applicable to robotics, the principle can also be applied to larger brakes.

3.4.9 Statically balanced brakes

A friction based locking mechanism that completely decouples the friction force and the actuation force is the statically balanced brake by Plooij et al. [168] (see Fig. 3.3d). This brake comprises three groups of springs of which the total potential energy is constant. Therefore, all positions of the brake are equilibrium positions, while the position of the brake determines the normal force between the two friction surfaces. Since the actuator now only has to apply a force to move a small part, the

energy consumption is very low. Although this brake has not been implemented in a robot yet, it is useful in applications that require a low energy consumption, a small actuator, unlocking under load, a large amount of locking positions and an adjustable locking torque. The intended application of the brake is the locking of a spring in a novel parallel spring mechanism, which is currently being developed.

3.4.10 Thermic lock

The thermic lock uses the difference in thermal expansion coefficients of different materials to obtain a lock with force closure (see Fig. 3.3g). The mechanism consists of a shell that can freely rotate around a core. The material of the core has a higher thermal expansion coefficient than the shell. The temperature control is achieved by a resistance wire that heats up the brake and which then locks the joint. Since heating up and cooling down take time, the response time of these brakes is relatively large. There are no applications of this locking mechanism in robotics known to the authors.

3.5 Singularity locking devices

Singularity locking devices are characterized by a position dependent transfer ratio. In its singular position such locking devices have an infinitely high transfer ratio, featuring an infinitely high locking torque. This section describes the active and passive singularity locking devices found in literature.

3.5.1 Four bar linkage

One approach to realize singular positions is to use four bar mechanisms (see Figs. 3.4a and 3.4b). Such mechanisms typically have an input rotation and an output rotation or translation. In the singular position, three of the four joints of the four bar mechanism are aligned, resulting in an infinitely high transfer ratio from the input to the output. As soon as a singular position is reached, it is impossible to open the locking by applying torque on the input rotation. Only the action of an actuator pushing the linkages out of their singular position can open the system (e.g. a torque on one of the three aligned joints). Therefore, the four bar linkages are considered active locking devices. The advantages of this mechanism are that the unlocking of the mechanism can be done when bearing its maximal load and with a very low energy consumption. The disadvantage is that due to the nature of the system, the locking is only available in one angular position.

For these reasons this kind of locking device is used in robotics to lock the knee of the bipedal walking robot by Van Oort et al. [153] and in the transfemoral prosthesis

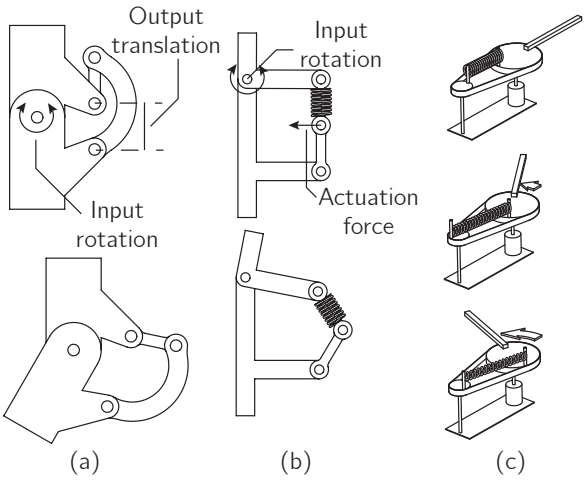


Figure 3.4: This figure shows the types of singularity locking devices with: (a) and (b) two different four bar mechanisms and (c) a non-linear spring mechanism.

AMPFoot 2.0 for realizing the 'catapult' mechanism [32]. Instead of realizing in a short time a high power for the push-off phase, a smaller and less powerful motor loads energy in a locked spring during a longer lapse of time. In the successor AMPfoot 3.0, a passive version is used which is unlocked by hitting a mechanical stop when walking [31]. A second application is the weight acceptance spring in the knee of the Cyberlegs alpha prototype [60], of which the mechanism is shown in Fig. 3.4b. During the stance phase, the spring is attached to a linkage that can be held in place with only a small actuation force. During the swing phase, the four bar mechanism is unlocked so the linkage can rotate out of the way and the knee can quickly flex to provide sufficient ground clearance for the swing phase.

3.5.2 Non-linear transfer ratio

The use of singular postures of human arms is well-known to reduce the required joint effort and avoid muscle fatigue. This virtue in nature was used by Ajoudani et al. for a robotic manipulator in [1]. The kinematic degrees of redundancy are adapted according to task-suitable dynamic costs. Arisumi et al. used the singular postures of arms to avoid actuator saturation when lifting a load [9]. The non-linear transfer ratio locking device are passive.

Non-linear transfer ratios have also been used passively. A first example of this is the non-linear spring mechanism for robotic arms designed by Plooij and Wisse [171] (see Fig. 3.4c). This mechanism consists of two connected pulleys with a spring in

between. The transfer ratio from the length of the spring to the rotation of the link becomes infinitely high at two positions and thus the spring is locked at those positions. They used this spring mechanism to reduce the energy consumption of the arm by placing the spring parallel to the motor. A second example is the variable stiffness actuator Compact-VSA of Tsagarakis et al. [223], where the stiffness regulation is achieved by a lever arm mechanism with a variable pivot axis. When the pivot moves, the non-linear amplification ratio of the lever changes from 0 to infinity. And thirdly, the knee of the humanoid Poppy by Lapeyre et al. [115] uses a spring in parallel to the knee, which locks the knee during the stance phase in a certain singular position.

3.6 Comparison

In this section, the different types of locking mechanisms are compared on the criteria given in section 3.2.1. Table 3.1 lists all types of locking devices and shows how well they score on the criteria, with a score of $++$, $+$, 0, $-$ or $--$. A $+$ always indicates that the device scores well. For instance, if the energy consumption scores $++$, this means that the device uses (almost) no energy.[2] The different locking principles (i.e. mechanical, friction, singularity) will now be discussed.

Mechanical locking devices typically have a low energy consumption. Even when they are actuated, the only thing the actuator has to do is to position the blocking part, for instance the pawl. Furthermore, mechanical locking devices typically have low weight, are small, have a low price and their locking torque is only limited by the strength of the parts. However, such locking devices also have disadvantages. Firstly, they are hard to unlock while being under load because of the friction between the two interfering parts. Secondly, their number of locking positions is limited (except for the hydraulic lock and the ratchet). And finally, the impacts that occur when a joint is blocked, will lead to shocks in the system.

Friction-based locking devices have less problems to unlock under load than mechanical locking devices. This is due to the fact that the two friction surfaces can often be disengaged, releasing the lock. Another advantage of friction based locking devices is that two friction surfaces can be engaged at any position and therefore the number of locking positions is infinite. And finally, since the locking torque depends on the friction coefficient and the normal force, the maximum locking torque can be controlled by controlling the normal force. As such, some friction based locking

[2]When a locking device scores two plusses on a criterion, it means that it performs (almost) perfectly. A single plus means that it scores above average. Similarly, a single minus indicates a below average performance and two minuses means that it performs (almost) as badly as it can get.

Table 3.1: The comparison between the different devices mentioned in this chapter. In all cases, ++ means that the property of the ideal locking mechanism is satisfied. For instance with respect to the energy consumption, ++ means that the locking device (almost) does not consume energy.

Type	*Locking principle*	*Activation*	*# of directions*	*(Un)locking while under load*	*Cont. power consumption*	*Switching power consumption*	*# of locking positions*	*Size*	*Weight*	*Switching time*	*Price*	*Locking torque*	*Locking torque adjustable*
Latches	Mechanical	Active	1	−−	++	+	−−	+	+	++	+	++	−−
Ratchets	Mechanical	Active	1	−−	++	+	+	+	+	++	+	++	−−
Dog clutches	Mechanical	Active	2	−−	++	+	−	+	+	+	+	++	−−
Hydraulic locks	Mechanical	Active	2	−	++	+	++	+	+	0	+	++	−−
Latches	Mechanical	Passive	1	−−	++	++	−−	++	++	++	++	++	−−
Ratchets	Mechanical	Passive	1	−−	++	++	+	+	+	++	++	++	−−
Cam based	Mechanical	Passive	1	++	++	++	−	0	0	−	+	++	−−
Electromagnetic	Friction	Active	2	++	−−	+	++	0	−	+	0	0	++
Overrunning	Friction	Active	1	0	++	+	++	+	+	+	0	++	−−
Self amplifying	Friction	Active	1	+	++	+	++	0	0	+	0	++	−
Capstan	Friction	Active	1	+	++	+	++	0	0	+	0	++	−
Piezoelectric	Friction	Active	2	++	+	+	++	0	0	++	−−	0	++
Bi-stable	Friction	Active	2	++	++	−	++	0	0	0	−	0	−
Statically balanced	Friction	Active	2	++	++	++	++	0	0	0	−	0	++
Thermic	Friction	Active	2	+	−−	−−	++	+	+	−−	0	0	+
Overrunning	Friction	Passive	1	++	++	++	++	+	+	+	0	++	−−
Non-backdrivable gearing	Friction	Passive	2	0	++	++	++	++	++	+	+	++	−−
Four bar linkages	Singularity	Active	2	++	++	+	−−	0	0	−	+	++	−−
Non-linear transfer ratio	Singularity	Passive	2	++	++	++	−−	−	−	−	+	++	−−

devices can be used as controllable brakes. However, these advantages come with a downside. Firstly, since the friction surfaces have to be pushed together, the energy consumption of these locks is typically high. Secondly, the locking torque is limited by the available normal force between the friction surfaces. And finally, in general friction based locking devices do not score well on size, weight and price.

Singularity locking devices are less common in robotics, although they score well on unlocking under load and on the power consumption. Also, their locking torque is only limited by the strength of the parts, similar to the mechanical locking devices. Disadvantages are that they typically have one locking position and that they are relatively large. So in comparison with the mechanical locking devices, they can unlock under load, but are larger.

3.7 Selection and development

This section provides guidelines for the selection of a locking mechanism and discusses current and future research directions.

3.7.1 Selection

Table 3.1 lists all the advantages and disadvantages of the different locking mechanisms. As a guide for designers, Fig. 3.5 shows a flow chart that can be used for the selection of a suitable locking mechanism. Following the flow chart, designers will find the locking mechanism(s) that is most specifically suited for their application. However, there might be application specific reasons to prefer another mechanism. For instance, in an application with hydraulic actuators it would be logical to prefer a hydraulic lock instead of a latch, although the application might not require infinite locking positions. Let's now look at two examples from literature to see how this flow chart would have lead the designers to their choice.

Firstly, in the knee of the bi-pedal running robot Phides [99], a parallel spring has to be attached to the joint during the stance phase and detached during the swing phase. This does not require an adjustable locking torque; (un)locking does not have to be performed under load and the number of locking positions is one. Therefore, a latch was used.

Secondly, in the transfemoral prosthesis AMPFoot 2.0 [32], the joint has to be locked while a small motor loads a spring. This does not require an adjustable locking torque; since the spring is loaded, the ankle has to be unlocked while being under load and the number of locking positions is one. Therefore, a singular locking mechanism was used.

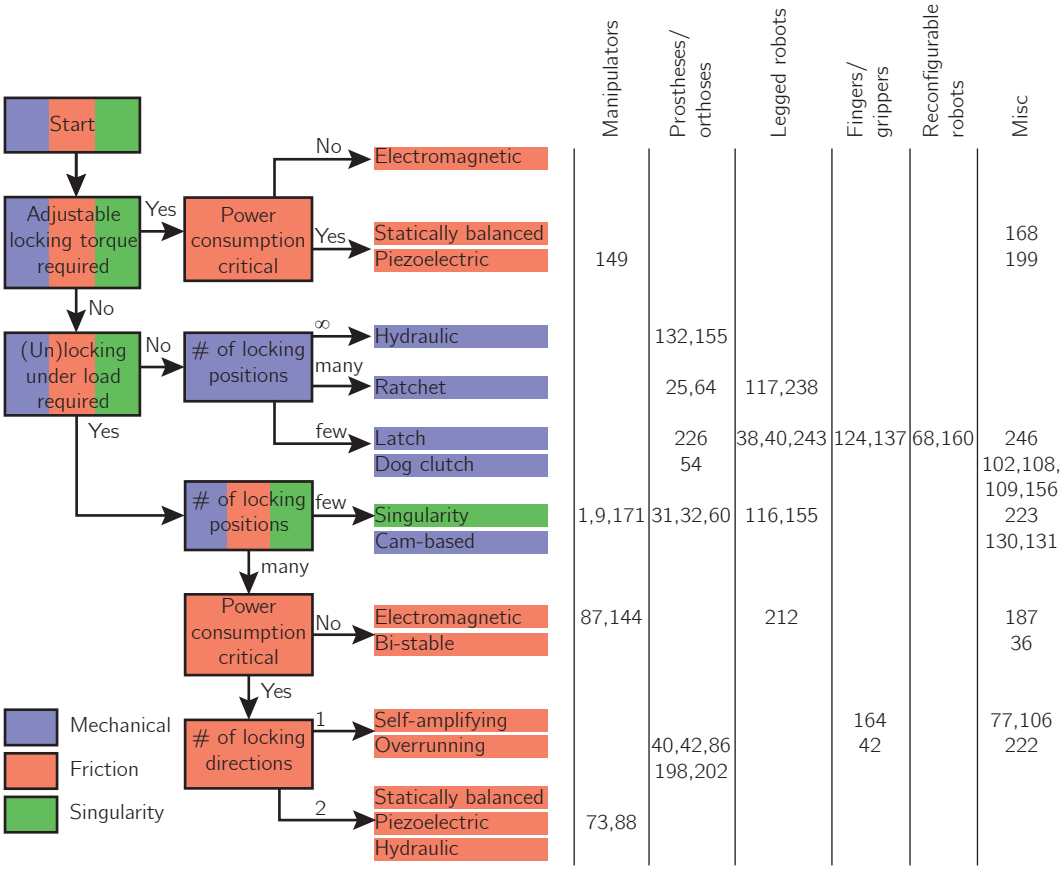


Figure 3.5: Flow chart for selecting a locking mechanism. This chart will give the most specific locking mechanism. However, there might be application specific reasons to choose another mechanism. The references in which the mechanisms are currently used are listed on the right.

3.7.2 Development

There are four trends that can be observed from recent research and will likely dominate research in the (near) future. Firstly, the amount of robots that incorporate locking mechanisms is growing rapidly. This can be seen by looking at the publication dates of the citations. Secondly, new mechanisms were developed recently that require low actuation power, but have a larger applicability than conventional mechanical or singularity locking devices. This is logical since locking mechanisms are frequently used in applications in which energy consumption is crucial. Thirdly, the use of singularities is a relatively new topic: the references date from the period 2008-2014. And fourthly, recently new actuator technology found its way into locking mechanisms. Examples include SMAs, piezo actuators and electro- and magneto-rheological fluids.

3.8 Conclusion

This chapter presented an overview of locking devices that are used in robotics. The locking devices are divided into three categories: mechanical locking, friction-based locking and singularity locking. Each category that can be split further into actuated locking devices and passive locking devices. The locking devices were then evaluated based on the properties of an ideal locking device. **Mechanical locking** devices use relatively few energy, are cheap, small and can lock high torques. **Friction based locking** devices can unlock under load, have an infinite amount of locking positions and can adjust the locking torque. **Singularity locking** devices can unlock while being under load, consume little energy and can lock high torques. A flow chart was provided that will help designers of robots to select a suitable locking mechanism and shows in which devices similar mechanisms have been used.

Acknowledgement

The authors would like to thank Josh Caputo for proofreading the manuscript.

4

Statically Balanced Brakes

Michiel Plooij, Tom van der Hoeven, Gerard Dunning and Martijn Wisse,
To appear in: *Precision Engineering*.

Abstract

Conventional brakes require a powerful actuator, leading to large, heavy and in most cases energy consuming brakes. This paper introduces a fundamentally different brake concept called statically balanced brakes (SBBs). SBBs do not require any actuation force to maintain a braking torque and only have to move a small mass to vary that torque. Therefore, their energy consumption is potentially very low. In an SBB, one of the two friction surfaces is connected through springs to a braking block. This braking block is connected through a mechanism to a second set of springs, the other side of which connects to the ground. The total energy in the two sets of springs is constant, which results in a zero-force characteristic at the braking block. The position of this statically balanced braking block determines the displacement of the first set of springs and thus the normal force between the friction surfaces. We categorize mechanisms that can be used in SBBs and show two embodiments: one with leaf springs with a range of positions with negative stiffness and one with torsion springs and a non-linear cam mechanism. Results show that the actuation force can be reduced by approximately 95-97% in comparison to regular brakes. This shows that in SBBs, the actuation force can be almost eliminated and thus showing the potential of SBBs to be small, lightweight and energy efficient.

4.1 Introduction

Conventional brakes require a powerful actuator that generates a normal force between two friction surfaces [119, 154]. The amplitude of the normal force, the friction coefficient and the geometry of the brake together determine the braking torque. There are applications in which powerful actuators are undesired due to size and weight limitations or their (potentially high) energy consumption. Therefore, researchers have worked on designing brakes that require less actuation force.

Research on the reduction of the required actuation force can be split into three categories, that are also described in the recent review paper on locking mechanisms [169]. Firstly, self-engaging brakes have been developed that use the relative motion between the friction surfaces to pull the friction surfaces together and thereby reducing the required actuation force [76, 77, 106, 163]. Disadvantages of such brakes are that they only work in one braking direction and that they can only disengage in the opposite direction of engagement. Secondly, spring brakes (also called safety brakes or parking brakes) use a spring to keep the brake engaged without actuation

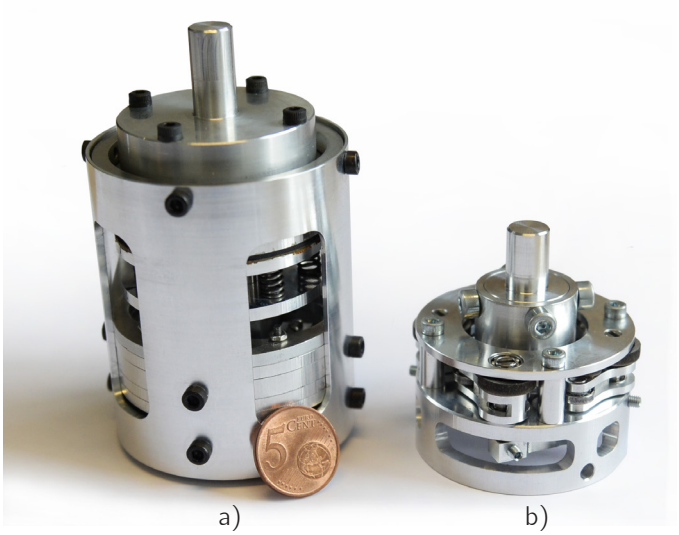


Figure 4.1: A picture of the two prototypes of statically balanced brakes. a) A prototype with leaf springs with a range of positions with negative stiffness. b) A prototype with torsion springs and a rotational cam mechanism.

force [10, 113, 119, 162]. However, these brakes still require an actuation force to keep the brake disengaged. This is solved in the third category: bi-stable brakes [36, 159]. Such brakes have a bi-stable element (e.g. a bi-stable spring), providing the brake with two stable states: the engaged state and the disengaged state. However, switching between these two states still requires a high actuation force.

Other researchers focused on implementing actuators with a high force density and a low energy consumption. The best example of this is piezo-actuated brakes [80, 88, 114, 149, 246]. Because of their high force density and low energy consumption, they are potentially very effective in solving the issues mentioned above. However, they require high voltages (that might not be available), very precise manufacturing (since they have a very small stroke) and are expensive. Furthermore, the brake construction has to be very stiff, otherwise the construction will deform, which reduces the effectiveness of piezoelectric actuators.

The problem with the state-of-the-art brakes is that the actuator has to be able to generate a force equal to the normal force between the friction plates. The goal of this paper is to introduce a brake concept in which the normal force and the actuation force are decoupled. This concept potentially reduces the actuation force by 100 %.

This new brake concept is fundamentally different from current brake concepts and

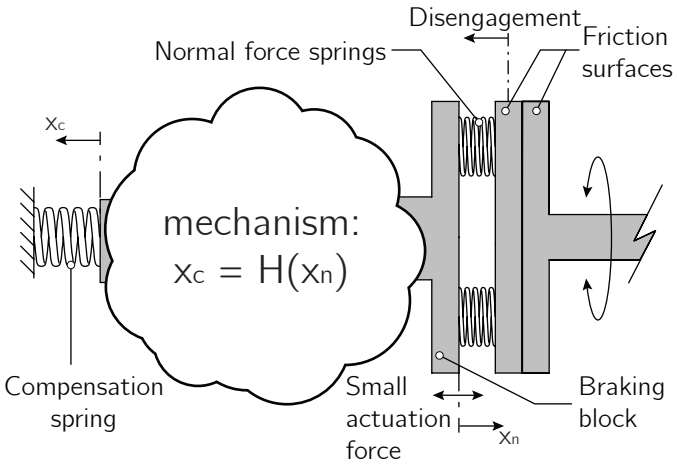


Figure 4.2: A schematic drawing of a statically balanced brake. The right friction surface is connected to a joint that has to be braked. The left friction surface is connected through the normal force springs with the braking block. A mechanism connects the braking block with the compensation springs and the other side of the compensation springs connects to the ground. This mechanism is visualized by the mechanism equation $x_c = H(x_n)$, that gives the relationship between the positions of the two sides of the mechanism. The position of the braking block determines the normal force between the friction surfaces.

is called statically balanced brakes (SBBs, see Fig. 4.1). SBBs do not require an actuation force to hold a certain braking torque and only require a small actuation force to vary that torque. Furthermore, with small adjustments, SBBs can be changed to incorporate any of the three different functionalities mentioned above (i.e. regular-, spring- and bi-stable behavior), while still only requiring a small actuation force. Statically balanced mechanisms have also been used amongst others for intrinsically safe robotic arms [225, 231], exoskeletons [84], prostheses [220] and micro and precision mechanisms [50, 219].

The rest of this paper is structured as follows. First, section 4.2 explains the concept of SBBs in more detail. Then, section 4.3 categorizes all possible embodiments of SBBs that are relatively simple and therefore small and lightweight. Sections 4.4 and 4.5 then show two prototypes of SBBs and their performance. Those results will show that the actuation forces in the prototypes are reduced by 95-97%. Finally, the paper ends with a discussion in section 4.6 and a conclusion in section 4.7.

4.2 The concept of statically balanced brakes

In this section we explain the concept of SBBs in more detail. First, we give a general formulation without assuming linear springs. Then, we work out the equations for a system with linear springs. Fig. 4.2 shows a schematic drawing of the concept of a SBB and Fig. 4.3 shows the working principle of a SBB for linear springs. The brake is engaged by pushing the two friction surfaces against each other. The friction between the surfaces is assumed to be a Coulomb type friction:

$$|F_f|_{max} = \mu F_n \tag{4.1}$$

where μ is the Coulomb friction coefficient, $|F_f|_{max}$ is the maximum absolute friction force before the surfaces start to slip and F_n is the normal force. The brake in Fig. 4.2 is statically balanced by two groups of springs. The group of normal force springs is placed between the braking block and the left friction surface. The energy in this group of springs is equal to $E_n(x_n)$, with x_n being the displacement of the springs as shown in Fig. 4.2. The force in these springs (F_n) is equal to the normal force between the friction surfaces. Multiplying F_n by the effective radius r of the brake, gives the braking torque:

$$|T|_{max} = \mu F_n r = \mu \frac{\partial E_n(x_n)}{\partial x_n} r \tag{4.2}$$

This means that the position of the braking block determines the amplitude of the braking torque. Now if this were the only group of springs, an actuator would still have to generate the force F_n to hold the braking block in a certain position. In order to decouple this normal force from the actuation force, a second spring system is used: the compensation springs. The compensation springs are placed between the ground and a mechanism that also connects to the braking block. This mechanism is depicted in Fig. 4.2 as a cloud with the mechanism equation $x_c = H(x_n)$. This equation assumes that the overall mechanism has one degree of freedom (DOF). In section 4.3, we will zoom in on this part and discuss possible mechanisms. Here we will analyse the static balance of mechanisms from an energy perspective. The energy in the group of compensation springs is $E_c(x_c)$ with x_c being the displacement of the springs as shown in Fig. 4.2. Now the system is statically balanced when $E = E_n + E_c$ is constant for all positions of the system. The transfer ratio h from the normal force springs to the compensation springs at position x_n is equal to:

$$h(x_n) = \frac{\partial H(x_n)}{\partial x_n} = \frac{\partial x_c}{\partial x_n} \tag{4.3}$$

We can now write the condition for static balance as

$$\frac{\partial E}{\partial x_n} = 0 \tag{4.4}$$

$$\frac{\partial E_n(x_n)}{\partial x_n} + \frac{\partial E_c(x_c)}{\partial x_c} h(x_n) = 0 \tag{4.5}$$

Now given the two spring characteristics, this system is statically balanced for all x_n for which it holds that

$$h(x_n) = -\frac{\partial x_c}{\partial E_c(x_c)} \frac{\partial E_n(x_n)}{\partial x_n} \tag{4.6}$$

The force that the compensation spring applies on the braking block can be expressed as:

$$F_c = \frac{\partial E_c}{\partial x_n} = \frac{\partial E_c(x_c)}{\partial x_c} h(x_n) = -\frac{\partial E_n(x_n)}{\partial x_n} = -F_n \tag{4.7}$$

It is logical that $F_c = -F_n$ because this results in force equilibrium, which is another way to consider static balancing. Now assume that both the normal force springs and the compensation springs are linear:

$$E_n = \frac{1}{2} k_n \max(x_n, 0)^2 \tag{4.8}$$

$$E_c = \frac{1}{2} k_c x_c^2 \tag{4.9}$$

where k_n and k_c are spring stiffnesses and the max operator returns the maximum value of the two inputs and models the disengagement of the friction surfaces. Eq. (4.6) now becomes:

$$h(x_n) = -\frac{k_n \max(x_n, 0)}{k_c x_c} \tag{4.10}$$

From Eq. (4.10) it follows that $h(x_n \leq 0) = 0$. This means that the mechanism is in a singular position or that the mechanism contains a clutch that decouples the two motions. From Eqs. (4.8) and (4.9) and the requirement that $E = E_n + E_c$ is constant, it follows that for linear springs, the mechanism should satisfy

$$x_c = \sqrt{\frac{2E - k_n \max(x_n, 0)^2}{k_c}} \tag{4.11}$$

Using Eq. (4.3), the transfer function becomes:

$$h(x_n) = -\frac{k_n}{k_c} \frac{x_n}{\sqrt{\frac{2E - k_n x_n^2}{k_c}}} \tag{4.12}$$

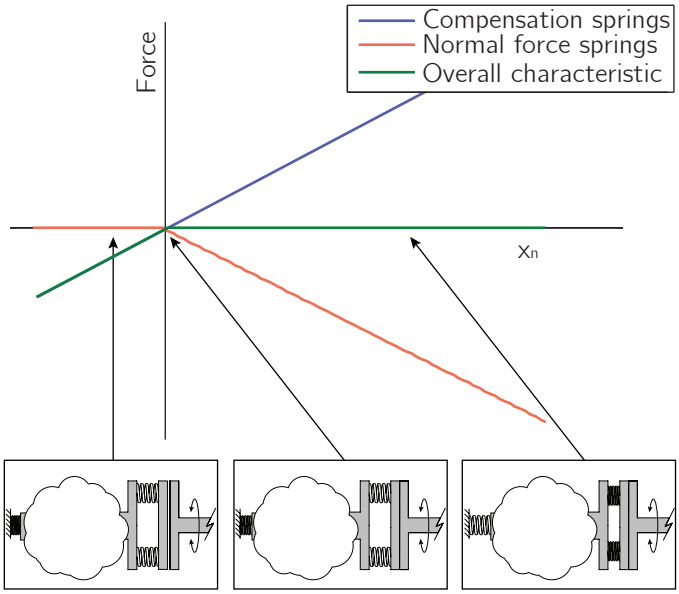


Figure 4.3: The working principle of a statically balanced brake. The compensation springs have a negative stiffness when measured at the braking block and the normal force springs have a positive stiffness. Since the stiffnesses cancel out and the equilibrium positions coincide, the overall characteristic has a range of zero force, which is the actuation stroke. A small actuator can position the brake at any position in this range, controlling the normal force and thus the braking torque.

Fig. 4.3 shows a schematic explanation of the forces in such a mechanism as function of the position of the braking block. This figure shows that the overall characteristic is equal to zero for $x_n \geq 0$, while the normal force at those positions depends linearly on the position. This means that the actuator does not have to apply any force to maintain a certain normal force between the friction surfaces. Note that in this example, $h(x_n < 0) \neq 0$, meaning that the system is not statically balanced for $x_n < 0$.

The concept of SBBs depends on a decoupling of the normal force between two friction surfaces and the force required to engage or disengage the brake. Without the static balancing, the actuator that moves the braking block would also have to deliver the force that pushes the friction surfaces together. With the static balancing, the braking block can be moved by an actuator that does not have to counteract any spring force (Eq. (4.7)). This controlled position determines the braking torque (Eq. (4.2)).

4.3 Possible embodiments

The previous section presented requirements on mechanisms for SBBs. Theoretically, when satisfying those requirements, a reduction in the actuation force of 100 % can be achieved. In order to design mechanisms that meet those requirements, in this section we categorize mechanisms that are suitable to be used in SBBs. Two of those concepts were built and the results will be shown in the next two sections.

The categorization is limited by three constraints that will lead to mechanisms that have the potential to be small and lightweight. First, we only consider one DOF mechanisms. More DOF mechanisms for SBBs are also possible, but this requires extra components and extra actuators, increasing the size and mass of the brake. Secondly, the type of spring that is used as normal force spring or compensation spring should match the type of DOF that it is attached to. For instance, we do not consider mechanisms in which a translational spring is connected to a rotating link. Such a construction does not allow for alignment of spring and the DOF and will therefore lead to an increase in size. Thirdly, we only consider mechanisms with the least amount of components. For instance, a four bar mechanism is considered, but an eight bar mechanism with one DOF is not considered.

In general, mechanisms can be divided into rigid body mechanisms and compliant mechanisms. In rigid body mechanisms, all the parts are rigid except for the springs that are either translational or rotational. For the purpose of this paper, we split rigid body mechanisms into linkage mechanisms and cam mechanisms. The overall categorization of mechanisms is shown in Fig. 4.4, that also already shows which categories are feasible. This section first analyses rigid body mechanisms and then compliant mechanisms. This analysis results in a list of feasible concepts and a description of how to construct them.

4.3.1 Rigid body: Linkages

The first class of rigid body mechanisms for SBBS is linkage mechanisms. In order to categorize one DOF linkages further, we have to realize that the mechanism should at least possess one singular position. This follows from Eq. (4.10), where the transfer function becomes zero at position $x_n = 0$. This position should be reachable to fully unload the normal force spring to obtain a zero braking torque. Therefore, we categorize both linkage and cam mechanisms further by categorizing singular mechanisms. There exists literature on singular mechanisms and how to categorize them [71, 158, 248]. However, those categorizations are based on different types of mechanical singularities and do not lead to a complete list of mechanisms. Here we introduce a new categorization that provides such a list and only incorporates simple

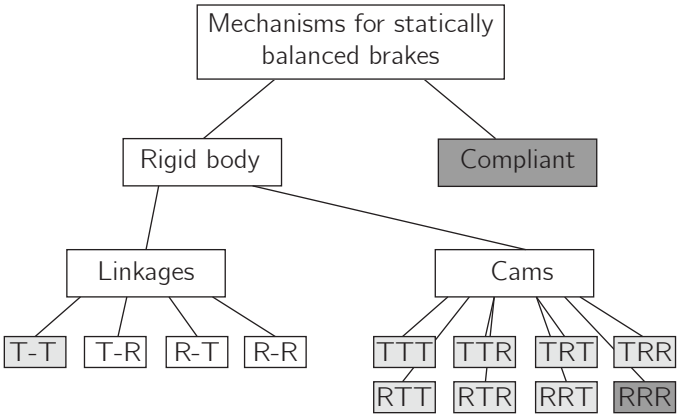


Figure 4.4: A visualization of the categorization of mechanisms for statically balanced brakes. A first division is made between rigid body mechanisms and compliant mechanisms. Rigid body mechanisms are split into linkages and cam mechanisms. Linkages are categorized based on the nature of their input and output (rotational or translational). One category of linkages leads to a feasible concept. Cam mechanisms are categorized on the nature of their input, output and cam movement (rotational or translational). In the grey categories it is possible to obtain perfect static balance. From the dark-grey categories we show a prototype in this paper.

singular mechanisms, leading to small and lightweight designs. Our categorization of singular mechanisms is based on the notion that all one DOF singular mechanisms have one input motion and one output motion. In simple singular mechanisms, these input and output motions are either translational or rotational motions. In linkages this leads to four categories:

1. translational input - translational output (see Fig. 4.5a)
2. translational input - rotational output (see Fig. 4.5b)
3. rotational input - translational output (see Fig. 4.5b)
4. rotational input - rotational output (see Fig. 4.5c)

All mechanisms use the same notation. x_n and x_c denote the displacements of the springs and can be rotational or translational. x_i and x_o denote the position of the input and output translations or rotations. l and θ refer to constant distances and angles, respectively. k_n and k_c denote the stiffnesses of the normal force springs and compensation springs. And finally d and γ refer to distances and angles that change when the position of the mechanism changes.

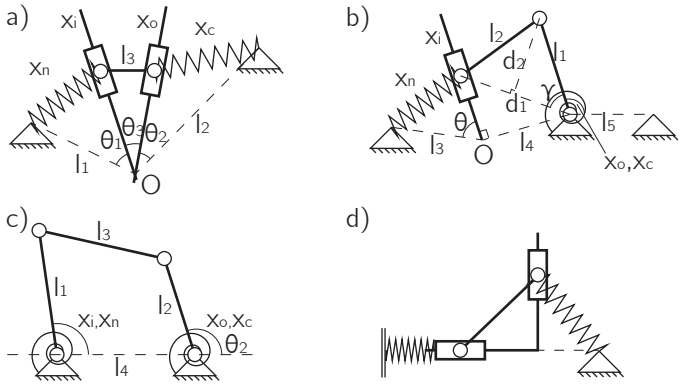


Figure 4.5: The collection of possible simple singular linkage mechanisms. a) A mechanism with translational input and output. This mechanism can be statically balanced. b) A mechanism with translational input and rotational output. c) A mechanism with rotational input and output. d) A statically balanced version of a.

The four categories of mechanisms will be discussed below. Before discussing them, it should be noted that the placement of translational normal force springs becomes impractical when they are both rotating and translating; see for example the left spring in Fig. 4.5a. Therefore, translational normal force springs can only connect to a slider that is in line with the spring. This also ensures that at a certain position, the force in the normal force springs becomes zero.

Translational input - translational output

A generalized version of this mechanism is shown in Fig. 4.5a. It consists of two sliders that intersect in O. θ_3 denotes the angle between the two sliders and x_i and x_o denote the positions of the sliders measured from O. The link between the two sliders has length l_3 and each slider connects to a spring. The other sides of the springs are connected to the ground at distances l_1 and l_2 from O under angles of θ_1 and θ_2. Since the left spring should be in line with the left slider, it is given that $\theta_1 = 0$. The energy in the system can be obtained by applying cosine rules:

$$\begin{aligned} x_n^2 &= l_1^2 + x_i^2 - 2 l_1 x_i \\ x_c^2 &= l_2^2 + x_o^2 - 2 l_2 x_o \cos(\theta_2) \\ l_3^2 &= x_i^2 + x_o^2 - 2 x_i x_o \cos(\theta_3) \\ E &= \frac{1}{2} k_n x_n^2 + \frac{1}{2} k_c x_c^2 \end{aligned} \tag{4.13}$$

$$\begin{aligned} E =& C_1 + \frac{1}{2}(k_n - 1)x_i^2 + \frac{1}{2}(k_c - 1)x_o^2 \\ &+ x_i x_o \cos(\theta_3) - k_n l_1 x_i - k_c l_2 x_o \cos(\theta_2) \end{aligned} \tag{4.14}$$

where C_1 is the constant term:[1]

$$C_1 = l_3^2 + \frac{1}{2}k_n l_1^2 + \frac{1}{2}k_c l_2^2 \tag{4.15}$$

We can derive x_o as function of x_i from Eq. (4.13) and fill it into Eq. (4.14). Now for static balance, Eq. (4.4) should hold for all x_i, which is only true when $\theta_2 = \theta_3 = 0.5\pi$, $l_1 = 0$ and $k_n = k_c$. Such a mechanism is depicted in Fig. 4.5d, where the ground can be moved freely along the dashed line. The normal force tension spring is changed into a compression spring in this example and is connected to friction plates. The fact that this category leads to a feasible solution is indicated in Fig. 4.4. The feasible mechanism in this category is not a new mechanism [83]. However, here we proved that this mechanism is in fact the only feasible mechanism in this category.

Translational input - rotational output

A generalized version of this mechanism is shown in Fig. 4.5b. It consists of one slider with a zero position $x_i = 0$ at O and a crank mechanism with links of lengths l_1 and l_2. One spring is placed between the slider and the ground and the other is placed between the crank and the ground. Since the left spring should be in line with the left slider, it is given that $\theta = 0$. The energy in this system can be derived as follows. First, we define d_2 as the distance between the joint that connects the two bars of the crank and the line d_1:

$$\begin{aligned} d_2 =& l_1 \cos(\gamma) \\ d_1 =& \sqrt{l_1^2 - d_2^2} + \sqrt{l_2^2 - d_2^2} \\ x_i =& \sqrt{d_1^2 - l_4^2} \\ x_n =& (x_i - l_3) \\ E =& \frac{1}{2}k_n x_n^2 + \frac{1}{2}k_c (x_c - x_{c,0})^2 \end{aligned} \tag{4.16}$$

where $x_{c,0}$ is the equilibrium position of the rotational spring. Again for static balance, Eq. (4.4) should hold for all x_i, which is only the case when $k_n = k_c = 0$. Since the

[1] In Eq. (4.14), we added and substracted the same term from E: $E = E + a l_3^2 - a l_3^2$, where a is taken as 1 Jm^{-2}. The remaining equation incorporates l_3 and θ_3, where l_3 is constant and θ_3 varies.

stiffnesses should be larger than zero, it is impossible to use this mechanism for a perfectly statically balanced brake, as indicated in Fig. 4.4. The use of imperfectly statically balanced mechanisms will be discussed in section 4.6.3.

Rotational input - translational output

This system is the same as the system with translational input and rotational output in Fig. 4.5b, with the difference that the input and output are switched. The energy in the system can now be calculated by:

$$\begin{aligned} x_i =& \sqrt{d_1^2 - l_4^2} \\ x_n^2 =& l_3^2 + x_i^2 - 2 l_3 x_i \cos(\theta) \\ E =& \frac{1}{2} k_n x_n^2 + \frac{1}{2} k_c (x_c - x_{c,0})^2 \end{aligned} \tag{4.17}$$

Since it is impossible to satisfy Eq. (4.4) with non-zero stiffnesses, it is impossible to perfectly statically balance this system, as indicated in Fig. 4.4.

Rotational input - rotational output

A generalized version of this mechanism is shown in Fig. 4.5c. The input and output links and the link in between form a four bar mechanism with lengths l_1, l_2, l_3 and l_4. One rotational spring is placed between link 1 and the ground. The second rotational spring is placed between link 2 and the ground. The energy can be calculated by:

$$E = \frac{1}{2} k_n (x_n - x_{n,0})^2 + \frac{1}{2} k_c (x_c - x_{c,0})^2$$

Again, such a mechanism cannot be perfectly statically balanced, as indicated in Fig. 4.4.

This leads to the conclusion that the only feasible simple linkage mechanism for SBBs is the one depicted in Fig. 4.5d.

4.3.2 Compliant mechanisms for SBBs

Instead of having rotational or translational springs and a rigid body mechanism in between, compliant mechanisms could be used. Applying compliant mechanisms varies from using springs with a range of positions with negative stiffness to making the whole mechanism out of one part. Compliant mechanisms are harder to categorize than the rigid body mechanisms earlier in this section.

In their handbook of compliant mechanisms, Howell et al. [93] categorized compliant mechanisms in two ways: based on the used components and based on their application. Here, we only indicate that there is a difference between SBBs with one

compliant mechanism that fulfills the function of both the normal force springs and the compensation springs in Fig. 4.2 and SBBs with a spring with a range of positions with negative stiffness that is used in a configuration with two spring systems. The latter option will be exploited in the embodiment in section 4.4. Two common types of springs that are known to exhibit the capacity to have range of positions with negative stiffness are leaf springs [51] and disk springs (also called Belleville springs) [7].

4.3.3 Rigid body: Cams

The second class of rigid body mechanisms are cam mechanisms. Cams can be categorized based on the same principle as we used to categorize linkages. Therefore, the normal force spring and the compensation springs can be translational or rotational. However, in cams mechanisms, the cam itself can also be rotational or translational, leading to eight categories (see Fig. 4.4).

In order to keep the discussion of those categories brief, we discuss the building blocks from which those categories can be constructed (see Figs. 4.6a-d). Due to the design freedom of cams, all eight categories lead to feasible solutions (as indicated in Fig. 4.4), although some have clear advantages or disadvantages. Fig. 4.6e shows one of the eight classes: a rotational cam, a rotational compensation spring and a rotational normal force spring. Here we discuss the components of such mechanisms, which can be split into:

1. A translational spring on a translational cam (Fig. 4.6a)
2. A translational spring on a rotational cam (Fig. 4.6b)
3. A rotational spring on a translational cam (Fig. 4.6c)
4. A rotational spring on a rotational cam (Fig. 4.6d)

In the analysis, we make three assumptions for simplicity. Firstly, we assume that the rollers on the cam have a radius of zero. How the analysis changes with a non-zero radius is described in [224]. Secondly, we assume that the grounds at which the springs are connected are in line horizontally with the center of the rotational cam (see Figs. 4.6b and 4.6d). Grounds that are not in line are also possible and would add an offset to the equations. Thirdly, we assume that the translational springs are placed horizontally (see Figs. 4.6a and 4.6b). Translational springs under an angle are also possible, but would make the analysis unnecessarily complicated.

The notation that is used is as follows. The lengths of the springs are denoted by x_n for the normal force springs and x_c for the compensation springs. $x_{n,0}$ and $x_{c,0}$

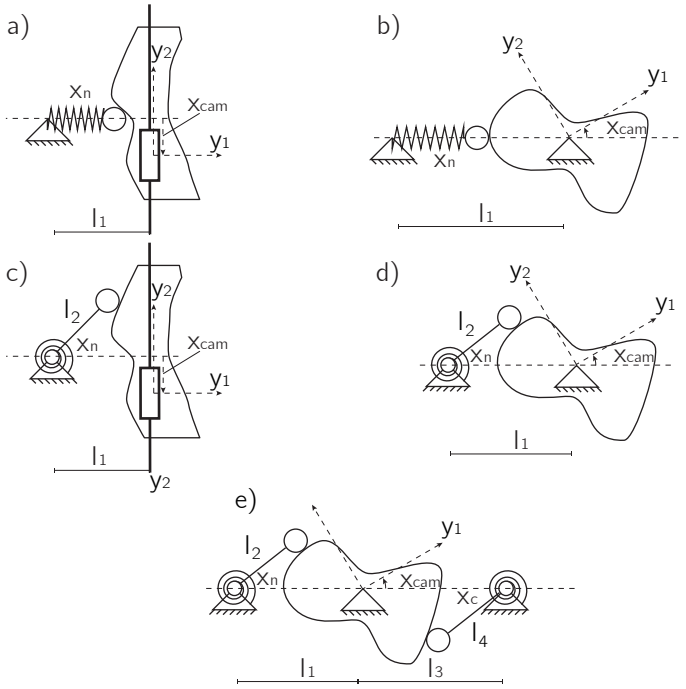


Figure 4.6: The collection of possible simple cam mechanisms which can be combined to form to SBBs. The mechanisms have rotational or translational inputs, outputs and cams. Combining two simple cam mechanisms to obtain a SBB leads to eight categories: a-a, a-c, c-a, c-c, b-b, b-d, d-b and d-d. e) shows a SBB of configuration d-d.

denote the equilibrium positions of the springs, meaning that the displacements of the springs are $x_n - x_{n_0}$ and $x_c - x_{c_0}$. x_{cam} denotes the position of the cam, which is translational or rotational. The surface of the cam is obtained in the body fixed workspace coordinates y_1 and y_2. l_1 and l_3 denote the distances between the two grounds and the center of the cam. And finally, l_2 and l_4 denote the length of the links connected to the rotational springs (when present).

In statically balanced cam mechanisms, the cam determines the relationship $x_c = H(x_n)$. Now, x_n can be chosen freely as function of the position of the cam. Then, by filling in Eq. (4.11), we obtain:

$$x_c(x_{cam}) = \sqrt{\frac{2E - k_n(x_n(x_{cam}) - x_{n,0})^2}{k_c}} + x_{c,0} \tag{4.18}$$

Using Eq. (4.18), a trajectory for $x_c(x_{cam})$ can be found as function of the trajectory $x_n(x_{cam})$. Designing a cam now splits into three steps. First, a displacement function

$x_n(x_{cam})$ should be chosen. Secondly, using this function the displacement function $x_c(x_{cam})$ can be calculated. And finally, when $x_n(x_{cam})$ and $x_c(x_{cam})$ are known, the cam surface can be obtained in the body fixed workspace coordinates y_1 and y_2. We will now present the equations to obtaining a cam surface with the desired follower behaviour for the four building blocks in Fig. 4.6. Afterwards, we will give an example that takes all three steps.

Translational spring - translational cam

An example of such a submechanism is shown in Fig. 4.6a. The shape of the cam surface for the normal force springs in the workspace coordinates y_1 and y_2 can now be obtained with:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_n(x_{cam}) - l_1 \\ x_{cam} \end{bmatrix} \tag{4.19}$$

This defines the cam surface for the side with the normal force springs. The side with the compensation springs works the same.

Translational spring - rotational cam

An example of such a submechanism is shown in Fig. 4.6b. The cam shape can be obtained with:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} (x_n(x_{cam}) - l_1)\cos(x_{cam}) \\ (x_n(x_{cam}) - l_1)\sin(x_{cam}) \end{bmatrix} \tag{4.20}$$

Rotational spring - translational cam

This submechanism is shown in Fig. 4.6c. The cam shape in body fixed workspace coordinates can be obtained with:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} l_2\cos(x_n(x_{cam})) - l_1 \\ l_2\sin(x_n(x_{cam})) + x_{cam} \end{bmatrix} \tag{4.21}$$

Rotational spring - rotational cam

A schematic drawing of this submechanism is shown in Fig. 4.6d. The cam shape can be obtained using the equations:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = R(x_{cam}) \cdot \begin{bmatrix} l_2\cos(x_n(x_{cam})) - l_1 \\ l_2\sin(x_n(x_{cam})) - l_1 \end{bmatrix} \tag{4.22}$$

where $R(x_{cam})$ denotes the rotation matrix for a rotation of x_{cam}.

Cam mechanism example (RRR)

As an example, Fig. 4.6e shows the mechanism with two rotational springs and a rotational cam. A preliminary study showed that this concept was most promising of all rigid body mechanisms in terms of compactness. Therefore, we also built a prototype of this concept (see section 4.5). For the analysis we must first choose a displacement function $x_n(x_{cam})$, the stiffnesses and the equilibrium positions:

$$x_n(x_{cam}) = \sin(x_{cam}) \tag{4.23}$$

$$k_n = k_c = k \tag{4.24}$$

$$x_{n,0} = l_1 = x_{c,0}, l_3 \tag{4.25}$$

From Eq. (4.18) it follows that the displacement function $x_c(x_{cam})$ is equal to:

$$x_c(x_{cam}) \quad = \quad \cos(x_{cam}) \tag{4.26}$$

Here we chose a sine as a spring displacement function of the normal force springs, which leads to a cosine in the displacement function of the compensation springs. For every choice for a spring displacement function, a displacement function of the compensation springs exists that leads to a perfectly statically balanced mechanism. The choice for a sine will prove to result in a cam that can rotate 360 degrees. The cam trajectory for both springs is obtained by filling in Eq. (4.22):

$$\begin{bmatrix} y_{1,n} \\ y_{2,n} \end{bmatrix} = R(x_{cam}) \cdot \begin{bmatrix} l_2 \cos(\sin(x_{cam}))) - l_1 \\ l_2 \sin(\sin(x_{cam}))) - l_1 \end{bmatrix} \tag{4.27}$$

$$\begin{bmatrix} y_{1,c} \\ y_{2,c} \end{bmatrix} = R(x_{cam}) \cdot \begin{bmatrix} l_4 \cos(\cos(x_{cam}))) - l_3 \\ l_4 \sin(\cos(x_{cam}))) - l_3 \end{bmatrix} \tag{4.28}$$

There are two options to create this cam surface. Firstly the cam can be split in two halfs where one half connects to the normal force spring and the other half connects to the compensation spring. Secondly the position of the compensation spring can be altered and placed vertically in Fig. 4.6e. By doing so a 0.5π phase shift is obtained in x_{cam} for the compensation spring trajectory. Since $\sin(x_{cam}) = \cos(x_{cam} - 0.5\pi)$, the same cam surface can be used as for both the normal force spring and the compensation spring. Therefore, the cam is statically balanced for the full 360 degrees. This concept is implemented in the embodiment that will be explained in section 4.5.

4.4 Example 1: Compliant mechanism: bi-stable leaf springs

This section presents the first of two prototypes that were build to verify the performance of SBBs. The prototype in this section has compression springs as normal force springs and leaf springs with a range of positions with negative stiffness as compensation springs (see Fig. 4.7). This concept has a third group of springs, called the counter springs, to compensate for the non-linear behavior due to disengagement of the friction surfaces in Eq. (4.8). We chose to build a prototype of this category to illustrate the possibilities and issues of this type of SBBs. The results in this section will show that the actuation force is reduced by 95 % in comparison to a regular brake.[2]

4.4.1 Static balancing

The energy in the normal force springs is given by Eq. (4.8). The leaf springs with a range of positions with negative stiffness are the compensation springs in this embodiment. In the range with negative stiffness, their energy can be approximated by

$$E_c = \frac{1}{2} k_c x_c^2 + E_0 \tag{4.29}$$

where k_c is negative and E_0 is a constant. The third group of springs in this embodiment (i.e. the counter springs) engage when the normal force springs disengage. The energy in this spring system is equal to

$$E_{ctr} = \frac{1}{2} k_{ctr} \min(x_{ctr}, 0)^2 \tag{4.30}$$

where the min operator returns the lowest value of the two inputs. Now if $k_n = k_{ctr} = -k_c$ and $x_n = x_c = x_{ctr}$, the total energy in the system is constant:

$$E = E_n + E_c + E_{ctr} = E_0 \tag{4.31}$$

4.4.2 Detailed design

Fig. 4.8 shows a section view CAD drawing of this embodiment. The amount of positive stiffness of the two groups of compression springs was tuned manually, as will be explained in the next section.

[2] The aim of both prototypes was to reach a braking torque of approximately 0.8-1 Nm, because the intended application was the BIC-PEA (see Chapter 5). However, the main result of the prototypes is not whether or not they reach this braking torque, but how much the required actuation force is reduced.

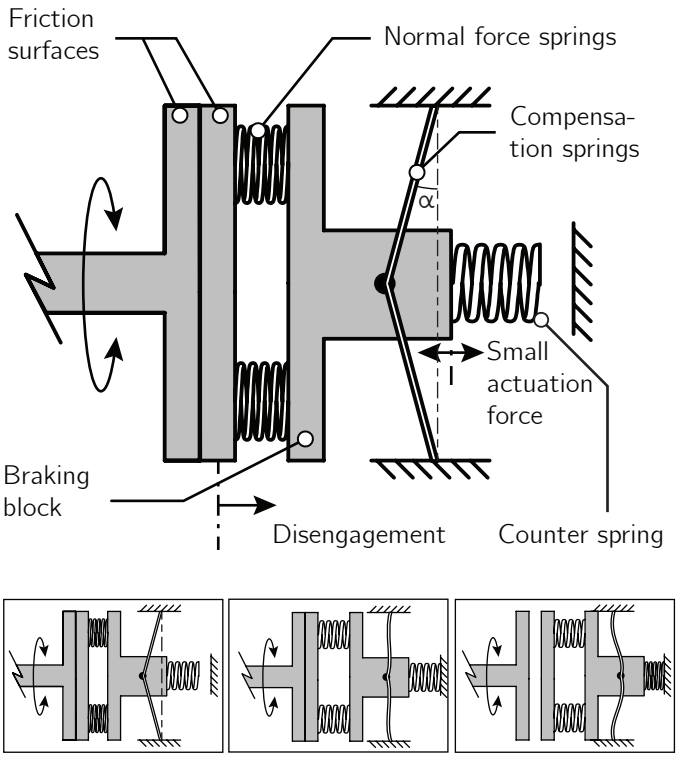


Figure 4.7: A schematic drawing of the statically balanced brake with leaf springs. The left friction surface is connected to a joint that has to be braked. The right friction surface is connected through the normal force springs with the braking block. The braking block is connected to the ground through leaf springs with a range of positions with negative stiffness and through the counter springs that engage when the friction surfaces disengage. The three configurations at the bottom show how the springs are deformed when the braking block changes position. They show an engaged brake (left), a disengaged brake (right) and a brake that is on the verge of engagement (middle).

The used friction materials are the specialized friction material Vulka SF-001 and rubber. This leads to a friction coefficient of almost 0.8 and does not lead to sticking behavior. The use of rubber has the disadvantage that rubber tends to wear fast when there is relative movement of the friction surfaces at the moment a normal force is applied. However, in clutches, there should be no relative movement when the device is engaged. If a large amount of relative motion during engagement is to be expected, SF-001 should be used for both friction surfaces, which leads to a friction coefficient of 0.5.

Note that the design of this prototype does not include an actuator, because the main purpose of the prototype is to verify the performance of SBBs and not to be directly

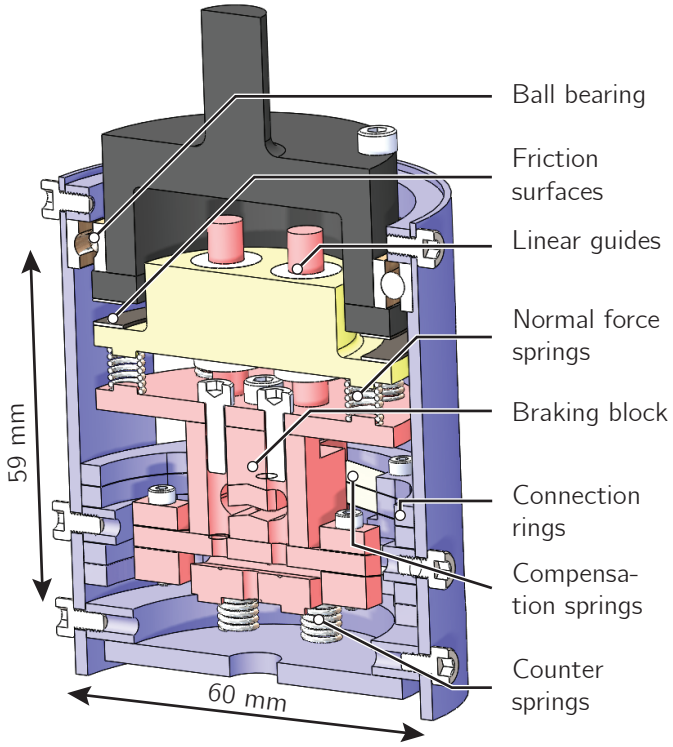


Figure 4.8: A section view of the embodiment with leaf springs. Parts with the same color (other than grey) are connected and move as one part. The compensation springs (leaf springs) are arranged in eight pairs of two. One side of the leaf springs is connected to the braking block (the red parts), the other side is connected to the ground (the blue parts) through a set of connection rings. The braking block is connected to the bottom friction (the yellow parts) surface by seven normal force springs (compression springs) with a total stiffness of 66.8 N/mm. Four axes that are connected to the braking block are connected to the bottom friction surface through linear ball bearings. These linear guides prevent a torsional load on the normal force springs. At the bottom of the braking block, there is a group of counter springs (compression springs) with a total stiffness of 54.0 N/mm. The top friction surface is connected to the joint (the black part at the top) that has to be braked, which is connected to the ground by a ball bearing. The colors refer to the online version of this article.

implemented in an application. For testing purposes, we connected this prototype to a standard test setup that includes a motor and a loadcell. The same counts for the prototype in section 4.5.

Table 4.1: The parameters of the seven cases

Case	l (mm)	t (mm)	α (deg)	u (mm)
Case 1	10.0	0.10	10.0	0.0
Case 2	10.2	0.10	10.0	0.0
Case 3	10.0	0.11	10.0	0.0
Case 4	10.0	0.10	10.5	0.0
Case 5	10.0	0.10	10.0	0.2
Case 6	10.2	0.09	9.5	-0.05
Case 7	9.8	0.11	10.5	0.2

4.4.3 Stiffness tuning

The stiffnesses of the three groups of springs have to be balanced. Therefore, we used a leaf spring model in the software package ANSYS™ of which the correctness was verified in [51]. The used leaf springs are made of stainless steel with an E-modulus of 200 GPa and a Poisson ratio of 0.3. The springs have a width of 7.5 mm, a thickness of 0.1 mm and a length of 10 mm. The angle between the leaf springs and the vertical (see Fig. 4.7) is $\alpha = 10$ deg. The length and width were chosen to fit in the housing of the brake. The thickness was chosen such that the maximum stress is slightly smaller than the yield strength.

In order to test the sensitivity of the characteristic to manufacturing inaccuracies, we derived the characteristics for one pair of springs with seven slightly different parameter sets (see Table 4.1). Case 1 uses the intended parameters. In Case 2-5, the length, thickness, angle and pretension were varied. Pre-tension means that the distance between the braking block and the ground is reduced. Then, in Case 6 and 7, the worst and best cases in terms of maximum force were tested.

Fig. 4.9 shows the seven characteristics. It shows that the characteristic is very sensitive to certain manufacturing inaccuracies. Especially, a pre-tension changes the characteristic drastically. The different parameter variations influence the maximum force, the stroke of the springs and the stiffness.

Since the final characteristic is very sensitive to inaccuracies, we decided to tune the stiffnesses in the final design manually. This tuning consisted of three steps. First we measured the characteristic of the group of leaf springs to determine the amount of positive stiffness k_n and k_{ctr} that should be added, which turned out to be k_n = 66.8 N/m and k_{ctr} = 54.0 N/m. The two groups of springs have a different total stiffness in order to better match the sinusoidal-like characteristic of the leaf springs. Second, we added the positive stiffnesses and measured the characteristic again to

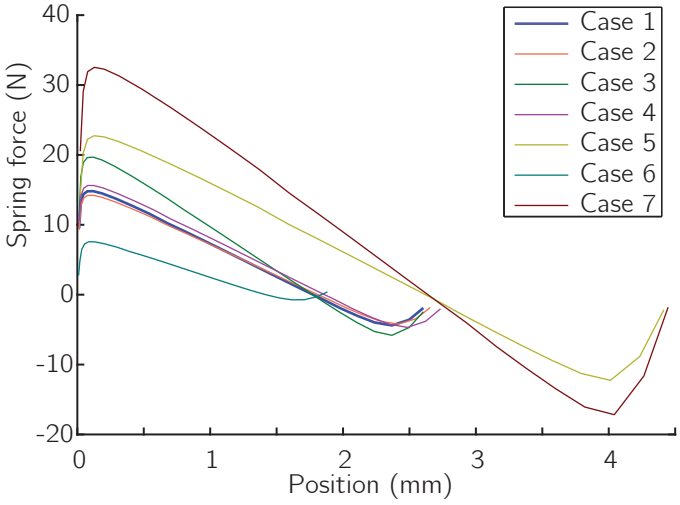


Figure 4.9: The characteristics of seven sets of springs with slightly varied parameters.

determine the off-set in x_n and x_{ctr}. Finally, we adjusted the off-sets in x_n and x_{ctr} accordingly and verified the balancing by measuring the overall characteristic. The result of this tuning is presented in the next section.

4.4.4 Reduction in actuation force

Fig. 4.10a shows the characteristic of the leaf springs and the tuned characteristic of the three spring systems combined. The maximum actuation force is 5.9N and the maximum normal force is 109.6N. Therefore, the actuation force is reduced with 95% in comparison to a regular brake. The average actuation force within the actuation stroke is 1.83 N, which is an improvement of 97% in comparison to a regular brake.

4.4.5 Braking characteristic

We measured the braking torque manually at 12 positions by applying torque until the brake starts to slip and thus measuring the static friction. Fig. 4.10b shows the braking torque as function of position and the fit through the data. The results show that the braking torque is a piecewise linear function of the position of the braking block. The maximum braking torque is 1.08 Nm.

4.5 Example 2: Cam mechanisms: RRR

This section shows our prototype of the concept based on a rigid body approach. Rigid body mechanisms have the advantage that they are easier to model than com-

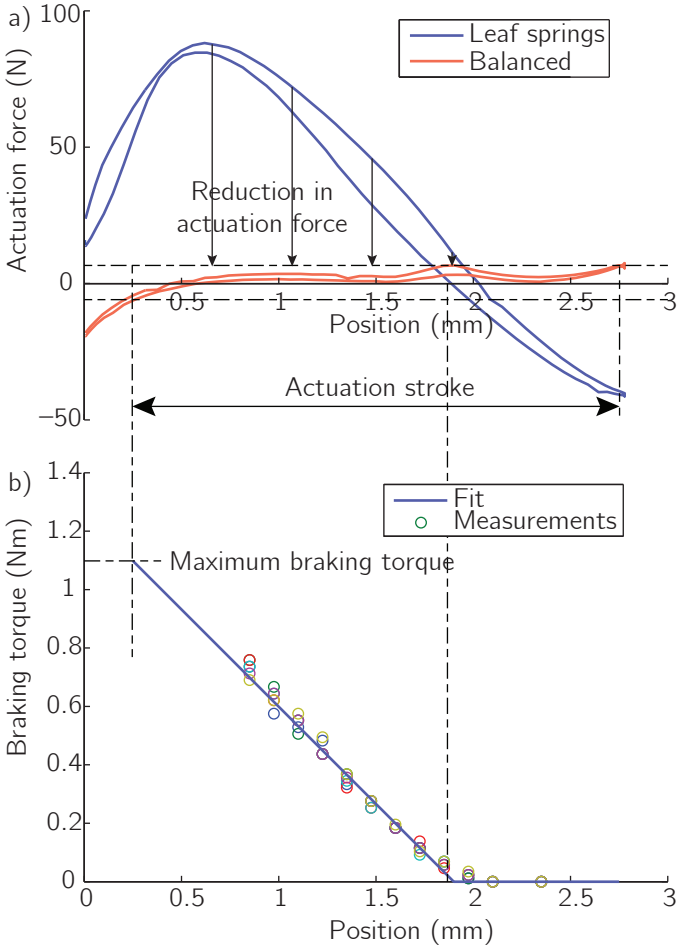


Figure 4.10: Measurements on prototype 1. a) The characteristics of the leaf springs and the complete statically balanced mechanism. In both cases, the force is measured while moving in both directions. The difference between the two measurements is due to hysteresis. The maximum actuation force is only 5 % of the force in the normal force springs. This means that the actuation force is reduced with 95 % b) The braking torque as function of the position.

pliant mechanisms. We chose to build a prototype with a rotational cam surface with torsion springs connected to the rotational input and output (see Fig. 4.6e). The mechanism in Fig. 4.6e is implemented twice, leading to a total of four torsion springs. An initial case study showed that this concept was most promising in terms of torque density and the results in this section will show that the actuation torque is reduced by 97 % in comparison to a regular brake.

4.5.1 Detailed design

Fig. 4.11a shows a CAD drawing of this prototype. The cam shaft is the braking block to which an actuator can be connected. When the cam shaft is rotated, the four followers track the motion induced by the cam surface and the torsion springs are deflected accordingly. The axles running through the followers connect to two components: two connect to the braking arms, the other two connect to the ground. This connection is clarified in Fig. 4.11b. The axles are connected to the followers through torsion springs with an individual stiffness of 0.64 Nm/rad. The springs are laser cut out of 3mm thick RVS301 spring steel, which has a E-modulus of 189 GPa. The springs have a wall thickness of 1.4 mm and makes 1.75 revolutions.

Fig. 4.11b depicts the braking side of the mechanism with the brake arms and the friction surface disk of the robot axis. The radius of the friction surface disk is 16mm. Clearly, the robot axis can pass through the entire braking device without obstructions. This has the advantage that this type of brake can also be used in the middle of an axis. The used friction materials are the specialized friction material Vulka SF-001 and rubber (the same as embodiment 1), which have a friction coefficient of 0.8.

4.5.2 Reduction in actuation force

Fig. 4.12 shows both the actuation torque of the cam shaft and the estimated torque in two torsion springs connected to the brake arms. Every 90 degrees, the system has a singular configuration in which the brake is either fully braking or disengaged. In those singular configurations, the actuation torque is 0 Nm which is a 100% reduction of the torque in the springs. In the non-singular configurations the maximum actuation torque on the cam shaft is 0.04 Nm which is a 97% reduction of the combined actuation torque in the torsion springs connected to the brake arms.

4.5.3 Braking characteristic

Fig. 4.13 shows the braking torque as function of the position of the cam. The maximum braking torque in clockwise direction is 0.83 Nm and the maximum braking torque in counter clockwise direction is 0.75 Nm. The cause for this difference is unclear. Part of the difference can be explained by a self-engaging effect similar to that in [163]. For the self-engaging effect to explain the complete difference, the friction force vector should pass the center of the axis of the brake arms at a distance of 0.8 mm. 0.2 mm can be explained by the friction surface that was slightly thicker than anticipated and manufacturing inaccuracies. Another 0.1 mm can be explained by the bending of the brake arms when they apply force on the braking disk. For the remainder of the difference, we do not have an explanation.

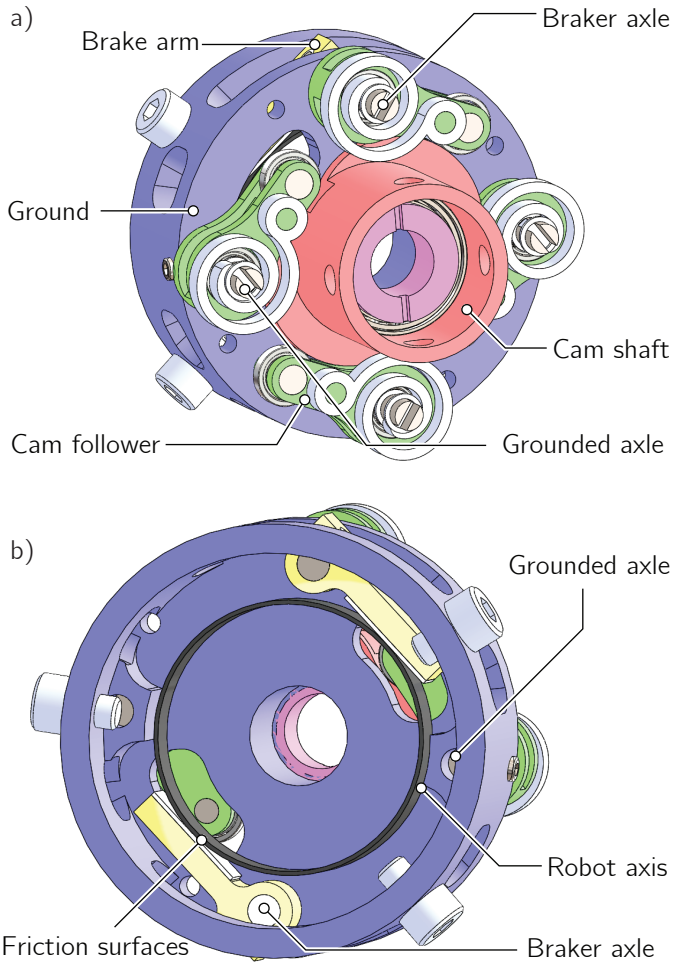


Figure 4.11: Two CAD drawings of the second embodiment. a) An inside view of the embodiment with torsion springs. b) The back side of the embodiment with torsion springs showing the connection with the braking arms. Four followers (shown in green) roll over the surface of the cam (shown in red). Two connect to the ground (shown in blue) and two connect to the brake arms (shown in yellow). The brake arms then push against a ring on the robot axis (shown in black).

Fig. 4.12 shows the actuation torque for this embodiment when only the two normal force springs are connected. The resulting actuation torque can be used to estimate the actual stiffness of the torsion springs. The calculated maximum input torque was 0.29 Nm and the actual maximum input torque was 0.24 Nm. This leads to the conclusion that the springs have a 20% lower stiffness than expected from the model

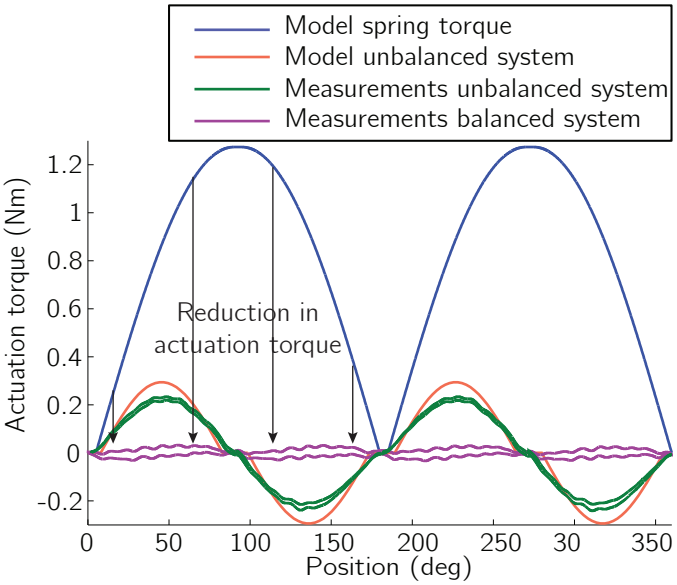


Figure 4.12: Actuation torque of embodiment 2 for a full rotation of the input cam. The blue solid graph is the estimated torque in the braking springs, the purple line is the measured input torque. The maximum input torque is only 3 % of the maximum spring torque. This means that the actuation torque is reduced with 97 %.

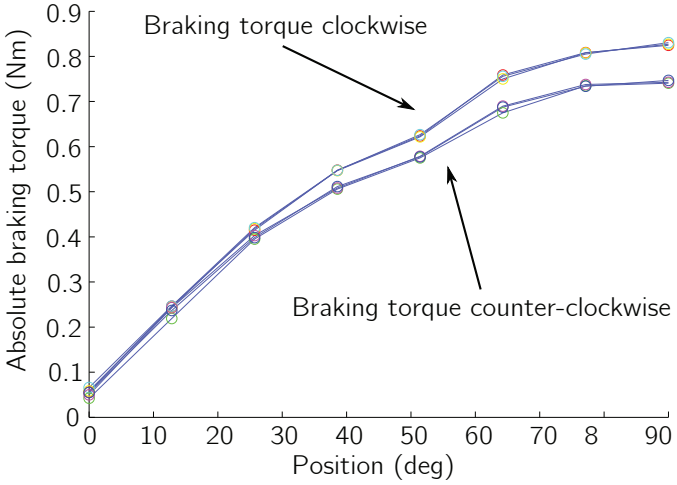


Figure 4.13: Braking torque of embodiment 2 over a range between 0 degrees (disengaged) to 90 degrees (fully braking) of the cam.

Table 4.2: Performance of the embodiments

Criterion	Leaf springs	Rotational cam
Braking torque	1.08 Nm	0.83 Nm
Actuation	5.9 N (= -95%)	0.035 Nm (= -97 %)
Bi-directionality	++	+
Size	∅60x59 mm	∅55x23 mm
Mass	170 g	92 g

and the maximum torque the springs produce is 0.64 Nm per spring instead of the calculated 0.80 Nm. Note however, that although the springs do not have the same stiffness as modeled, the brake is still statically balanced as long as the springs are linear.

4.6 Discussion

This paper introduced the concept of SBBs, categorized all relatively simple mechanisms that can be used in SBBs and showed two embodiments. The results show that the actuation force/torque can be reduced by 95-97% in comparison to a regular brake. We will now discuss the performance, mechanism selection, the use of imperfectly balanced mechanisms, the energy consumption of SBBs and the applications.

4.6.1 Performance

Table 4.2 summarizes the performance of the two embodiments in terms of the braking torque, actuation force/torque, bi-directionality, size, mass. We will now discuss those performance criteria separately. A full comparison between SBBs and other locking mechanisms is provided in [169].

The braking torques of the embodiments are 1.08 Nm, and 0.83 Nm. There are two options to increase the braking torque. The first option is to increase the radii of the friction surfaces and the second option is to increase the stiffnesses of the springs.

The maximum actuation force and torque are 5.9 N and 0.035 Nm for the two embodiments. Since one embodiment has an actuation force and the other has an actuation torque, the two numbers are hard to compare. However, in embodiment 1, the actuation force is only 5 % of the force in the normal force springs and in embodiment 2, the actuation torque is only 3 % of the torque in the normal force springs. This means that the brakes reach reductions of 95 % and 97 % in comparison to regular brakes. In both cases, the fact that the actuation force is not zero is due

to hysteresis and imperfect balancing. Both account for approximately 50% of the maximum actuation force/torque.

For the bi-directionality, we use the qualitative scoring from [169], where $++$ denotes a perfect bi-directionality and $--$ denotes locking only in one direction. The prototype with leaf springs has the same braking torque in both direction and thus scores a $++$. The prototype with a rotational cam has a slightly larger braking torque in one direction and thus it scores a $+$.

The size and mass of embodiment 1 are significantly larger than that of embodiment 2. The challenge in both designs is to miniaturize the mechanism for the compensation springs. In embodiment 1, the leaf springs and their mounting contribute most to mass and size. One possibility to lower the mass and size would be to replace the leaf springs by Belleville springs (also known as disk springs) [7]. More space and mass can be saved by optimizing the design. For example, the braking block, friction surfaces and housing are not optimized for mass and size. In embodiment 2, the cam mechanism and springs are all in one plane, making the design more compact.

4.6.2 Mechanism selection

In section 4.3, we categorized all relatively simple mechanisms that can be used between the braking block and the compensation springs. There are no hard rules that, when followed, will automatically lead to the best design. However, here we provide three considerations for selection of a suitable mechanism.

Firstly, rigid body mechanisms in combination with regular springs are easier to model than compliant mechanisms. As shown in section 4.4, compliant mechanisms can be very sensitive to manufacturing inaccuracies. On the other hand, compliant mechanisms are potentially very compact because the whole mechanism can be made out of one part.

Secondly, rigid body designs are in general smaller when the spring, the actuator and the input and output match. For instance, when a linear actuator is used, it is inconvenient to connect it to a rotational DOF.

Thirdly, the actuator should not be placed on a joint that can reach a singular position. For instance in Fig. 4.5b, placing a linear actuator on the vertical slider is not a good idea, because it will not be able to leave the position where the bar is vertical. Instead, a rotational actuator could be placed between the bar and one of the sliders.

4.6.3 Imperfectly balanced mechanisms

Section 4.3.1 discusses three categories of linkage mechanisms that cannot be perfectly statically balanced. Therefore, we did not consider them to be applicable. However, these mechanisms can be used when an approximately statically balanced mechanism suffices. Moreover, with imperfectly balanced mechanisms, the different functionalities mentioned in the introduction can be obtained: the regular brake, the safety brake and the bi-stable brake. These functionalities can also be obtained in cam mechanisms and compliant mechanisms. Here we explain how these adjustments can be obtained in embodiment 1.

The characteristic in Fig. 4.10 shows an almost statically balanced mechanism. At positions smaller than 0.5mm, the spring force is negative and at all other positions, the spring force is slightly larger than zero. This means that without any actuation force, the system will move to the 0.5mm position, at which the brake is engaged. This behavior is equal to that of a safety brake: when not actuated, the brake engages.

Changing the zero positions of the normal force or counter springs does not change the stiffness and thus it only shifts the total characteristic up and down. The amplitude of the shift influences the maximum actuation force. When the position at which the counter springs engage with the ground is changed, the behavior can be changed to that of a bi-stable brake. Such a brake has two stable positions: one in which the brake is engaged and one in which the brake is disengaged. When also changing the position at which the friction surfaces engage (and thus effectively changing the zero position of the normal springs), the behavior can be changed to that of a regular brake, which is only braking when actuated.

4.6.4 Energy consumption

Throughout this paper we only considered the actuation force and suggested that this relates to the energy consumption. Only considering the actuation force has the advantage that it is independent of the specific actuator that is used. To get an idea of the actual energy consumption, we will now briefly discuss the power consumption of a DC motor as a brake actuator. The power of a DC motor that is standing still is equal to:

$$P = \frac{F_n^2 \cdot R}{n^2 \cdot k_t^2} \tag{4.32}$$

where R is the motor resistance, n is the transfer ratio from the position of the motor to the position of the brake and k_t is the motor constant. This power consumption goes to zero when the transfer ratio n goes to infinity. However, since this also

increases friction, size and mass, this transfer ratio cannot be chosen too large. Now given a certain n, k_t and R, we see that the power consumption scales quadratically with the actuation force. In most other actuators, the energy consumption will scale with the actuation force and therefore, a higher actuation force will result in a higher energy consumption.

There are actuators in which the energy consumption (theoretically) is independent of the actuation force. Examples are electro-static and piezo-electric actuators. However, those actuators have other disadvantages that make them less suitable for application in brakes, as explained in the introduction.

4.6.5 Applications

The intended application of the brakes we introduced in this paper is robotics. The use of locking devices in robotics is increasing [169]. Such locking devices are mainly used to reconfigure robots, decrease actuator load when standing still, and control the energy release of springs. Especially in mobile robots, such as household robots or walking robots, components that do not consume energy are advantageous.

The main reason for using a SBB in comparison to other brakes is when only a small actuation force is available and the brake should be able to brake in two directions. Other possible applications for statically balanced brakes include torque limiters, cars, trains, buses, trucks and bikes. Especially the safety/parking brake version of the brake that we showed in this paper is applicable in vehicles such as buses and trucks that often use such brakes to stand still.

As stated in [169], brakes are often used as locking mechanisms or clutches. Using a brake as a clutch (instead of quickly switching clutches such as ratchets) has two advantages. Firstly, a brake can disengage while under load and secondly, the braking torque is independent of the position of the joint.

4.7 Conclusion

In this paper we introduced a new type of brakes: statically balanced brakes. The goal of SBBs is to eliminate the actuation force required in regular brakes. With small adjustments, SBBs can also be used as safety brakes or bi-stable brakes, with a reduced actuation force. We conclude that the concept of SBBs is promising and that the required actuation force can be reduced by 95-97% in comparison to regular brakes. Furthermore, cam mechanisms seem to be the most promising approach for balancing of the two spring systems because of their design freedom and the fact that they are relatively easy to model.

Acknowledgement

This research was sponsored by Technology Foundation STW (project numbers 11282 and 11832)

5

The Bi-directional Clutched Parallel Elastic Actuator

Michiel Plooij, Martijn Wisse and Heike Vallery,
Submitted to: Transactions on Robotics.

Abstract

Parallel elastic actuators (PEAs) have shown the ability to reduce the energy consumption of robots. However, regular PEAs do not allow to freely choose at which instant or configuration to store or release energy. This chapter introduces the concept and the design of the Bi-directional Clutched Parallel Elastic Actuator (BIC-PEA), which reduces the energy consumption of robots by loading and unloading the parallel spring in a controlled manner. The concept of the BIC-PEA consists of a spring that is mounted between the two outgoing axes of a differential mechanism. Those axes can also be locked to the ground by two locking mechanisms. At any position, the BIC-PEA can store the kinetic energy of a joint in the spring such that the joint is decelerated to zero velocity. The spring energy can then be released, accelerating the joint in any desired direction. Such functionality is suitable for robots that perform rest-to-rest motions, such as pick-and-place robots. Simulations show that the energy consumption of our one DOF setup can be reduced by 73 %. In hardware experiments, we reached reductions up to 65 %, which is larger than other concepts with the same functionality.

5.1 Introduction

Energy consumption of robots is an increasingly important performance criterion. Especially mobile systems that rely on batteries require a low energy consumption for a long operating time and low weight. Examples include domestic robots [135], walking robots [39], exoskeletons [49, 177] and prostheses [11, 165, 214]. One of the most promising approaches to reduce the energy consumption is recapturing negative work. Two main approaches are commonly applied in robots: generators and potential energy storage devices such as springs.

Electric motors can function as generators and transfer negative mechanical work into electrical energy to charge the battery. This has the advantage that the energy exchange between the energy storage and the joints is fully controllable. However, electrical motors suffer from copper losses, which decreases the efficiency of the energy recapture. Moreover, such motors often require gearboxes with large gearbox ratios that also increase the joint friction and thus increase the energy consumption of the robot. The efficiency of the energy recapture is further reduced by batteries that do not have a 100 % efficiency. A state-of-the-art robot that uses motors as generators is the Cheetah robot [192]. Seok et al. report an efficiency of energy

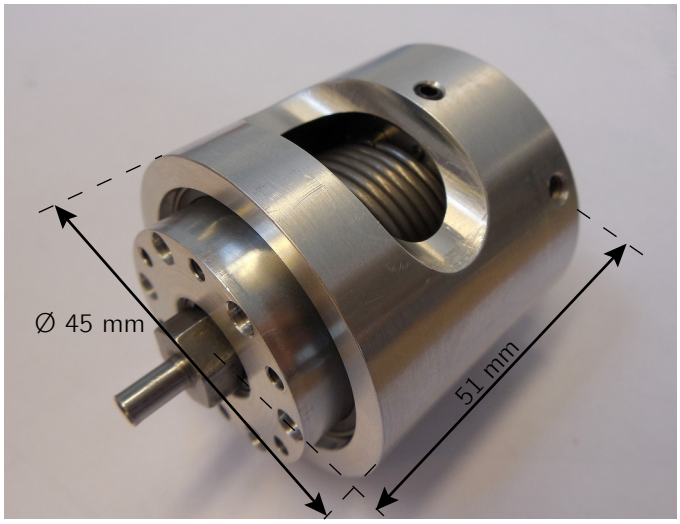


Figure 5.1: A photograph of the BIC-PEA. It weighs 202 g and fits in a cylinder with a length of 51 mm and a diameter of 45 mm.

recapture from joint to battery in the Cheetah robot of 63 %. Assuming that the efficiency from the battery to the joint is the same, the total efficiency is only 40 %.

Storing energy in the form of potential energy is potentially more energy efficient. Springs typically exhibit a hysteresis of 1-5 % and thus high efficiencies seem achievable. Therefore, springs have been placed in parallel with actuators in various applications (see e.g. [13, 77, 100, 103, 131, 153, 171, 180, 231]). With a regular parallel spring, the position of the joint directly determines the energy in the spring. This spring characteristic can be adjusted to match the task (see e.g. [103, 165, 171, 180]). The parallel spring then provides part of the torque required to perform the task, lowering the required actuator torques, and thereby reducing the energy consumption. Reduction of actuator torques should also allow for smaller gearbox ratios, leading to less gearbox friction, which again reduces the energy consumption.

Regular springs however, do not allow adjusting their characteristic during operation. This means that at any position, the amount of energy stored in the spring only depends on the design and not on the control. This greatly reduces the versatility of the robot in three ways. First, when energy is stored in the spring, the spring exerts a torque on the joint, which is undesirable when the joint has to stand still. Secondly, the timing of energy capture and release cannot be controlled independently of the joint position. And thirdly, the energy that is stored in the spring while decelerating

in one direction can only be used to accelerate in the opposite direction.

The solution to these problems is to make the energy inflow and outflow of the spring controllable. This could theoretically be realized by placing an infinitely variable transmission (IVT) between the joint and the spring, as proposed in [210]. However, there are no studies showing the working principle or effectiveness of such an IVT. Therefore, other researchers proposed to use locking mechanisms to control the energy inflow and outflow of the spring [169]. Locking the spring when standing still prevents the spring from exerting torque on the joint, solving the first problem mentioned above [13, 171]. The second problem mentioned above is addressed by Haeufle et al. [77], who proposed a clutched parallel elastic actuator (CPEA) in which the parallel spring can be connected to a joint by a controllable clutch. They report a reduction of 80 % in the energy consumption of their emulated knee joint. However, the problem that still has not been addressed is that the energy that is stored in the spring while decelerating in one direction can only be used to accelerate in the opposite direction.

In order to solve this problem, we propose to use the Bi-directional Clutched Parallel Elastic Actuator (BIC-PEA, see Fig. 5.1), which consists of one differential mechanism, two locking mechanisms and one spring that is placed inside the differential mechanism. The parallel spring can be connected to the joint at an arbitrary position or speed when the kinetic energy has to be stored in the spring. This energy can be released at an arbitrary later point in time, accelerating the joint in any desired direction. This principle is especially suited for rest-to-rest motions, which we will study in this chapter. We recently presented the working principle and a preliminary study on the performance of the BIC-PEA at a conference [170].

In this chapter, we analyse how much the energy consumption can be reduced by implementing the BIC-PEA in systems with rest-to-rest motions. Although the BIC-PEA provides more functionality than the CPEA, the goal is to reach a reduction similar to that reported for the CPEA [77].

The rest of the chapter is structured as follows. Section 5.2 provides a more detailed explanation of the working principle of the BIC-PEA. Next, section 5.3 discusses the design of our prototype. Then, section 5.4 explains our methods for analysing the energy consumption in simulation and in hardware experiments. The results in sections 5.5 and 5.6 show how implementation of the BIC-PEA can lead to a decrease in the energy consumption up to 65 %, depending on the specific task. The chapter ends with a discussion in section 5.7 and a conclusion in section 5.8.

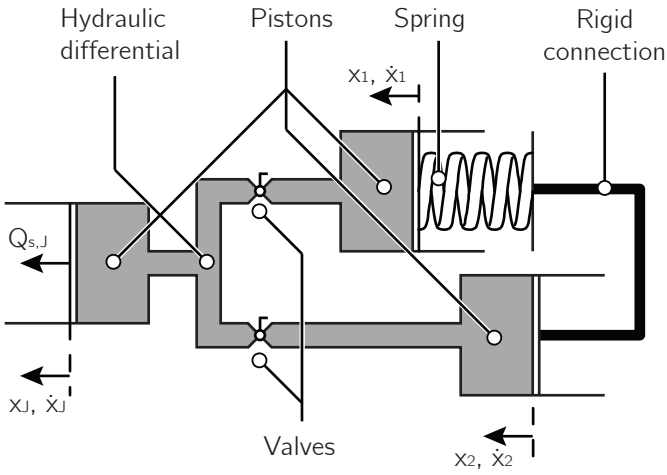


Figure 5.2: A schematic drawing of the concept of the BIC-PEA, visualized as a hydraulic system. It consists of a differential, a spring and two valves (locking mechanisms). The differential mechanism causes the joint piston to move with the sum of the velocities of the other two pistons. The valves can lock the two pistons.

5.2 Working principle

In this section, we explain the working principle of the BIC-PEA in more detail. First, we will explain the components of the BIC-PEA (one differential mechanism, two locking mechanisms and a spring) and how they are connected. Second, we will explain the operating principle of the BIC-PEA. A schematic drawing of the BIC-PEA is shown in Fig. 5.2. This figure shows a hydraulic mechanism for explanation purposes. The final design of the BIC-PEA is a geared version and will be explained in section 5.3.2.

5.2.1 Differential mechanism

A differential mechanism is a mechanism with three coupled axes. The coupling is described by one constraint that can be written in the form:

$$x_j = n_1 x_1 + n_2 x_2 \tag{5.1}$$

where n_1 and n_2 are constant positive transfer ratios, x_j is the position of the input and x_1 and x_2 are the positions of the output of the differential. These three positions are the three axes of the differential mechanism. Examples of such mechanisms are a planetary gear, a planetary differential, an automotive differential and a 'movable pulley and cables' differential. We call a differential mechanism ideal if $n_1 = n_2$,

which is generally the case for the planetary differential, automotive differential and 'movable pulley and cables' differential. In the BIC-PEA, the input position x_j is connected to the joint of a robot, hence the subscript j.

5.2.2 Locking mechanisms

A locking mechanism is a component that can switch between allowing and preventing relative motion between two other components (see [169]). In the BIC-PEA, the two locking mechanisms are placed between the ground and the two output positions x_1 and x_2. The discrete states of the two locking mechanisms are denoted by L_1 and L_2, which have value 0 if the output position is locked to the ground and have value 1 if the output position is not locked. Using this notation, eq. (5.1) can be re-written in terms of velocities:

$$\dot{x}_j = n_1\dot{x}_1 + n_2\dot{x}_2 \tag{5.2}$$

with $\dot{x}_i = 0$ if $L_i = 0$, $i \in 1, 2$.

5.2.3 Spring

A spring is a compliant component with (in our case) two connection points. The potential energy that is stored in the spring is a function of the relative position of the two connection points. In the BIC-PEA, a spring is placed between the two output positions of the differential mechanism. Therefore, the displacement of the spring is equal to

$$\Delta x = x_1 - x_2 \tag{5.3}$$

where the positions are defined such that $\Delta x = 0$ is an equilibrium position of the spring. The potential energy in the spring is a function of this displacement:

$$E_s = g(\Delta x) \tag{5.4}$$

The generalized force $Q_{s,i}$ exerted by the spring on the i-th axis is given by

$$Q_{s,i} = -\frac{\partial \Delta x}{\partial x_i}\frac{\partial g(\Delta x)}{\partial \Delta x} \tag{5.5}$$

where force is defined in the same direction as position. Examples of commonly used springs are compression springs, extension springs, torsion springs and spiral springs. Most springs have a constant and positive stiffness, which means that $\frac{\partial g(\Delta x)}{\partial \Delta x}$ is a monotonically increasing and linear function of Δx:

$$\frac{\partial g(\Delta x)}{\partial \Delta x} = k\Delta x \tag{5.6}$$

where k is the spring stiffness. The apparent spring stiffness at the joint is k_j, which is equal to

$$k_j = \left(\frac{\partial \Delta x}{\partial x_j}\right)^2 k \tag{5.7}$$

and the maximum displacement of the spring is called Δx_{max}.

5.2.4 Operating principle

Using the two locking mechanisms, there are four modes of operation:

1. $\boldsymbol{L_1 = 1}$ **and** $\boldsymbol{L_2 = 1}$**:** The two output positions are free to move and therefore the spring deflection Δx is independent of the joint position x_j. Although the spring might deflect due to inertia and friction, these deflections are negligible as long as the spring stiffness is sufficiently large.

2. $\boldsymbol{L_1 = 0}$ **and** $\boldsymbol{L_2 = 1}$**:** Output position 1 is locked, meaning that $\dot{x}_1 = 0$. Since x_1 is now constant, the spring deflection linearly depends on the joint position. Combined with eq. (5.1), this results in:

$$\Delta x = c_1 - x_2 \tag{5.8}$$

$$= (1 + \frac{n_1}{n_2})c_1 - \frac{1}{n_2}x_j \tag{5.9}$$

   where c_1 is equal to x_1 at the moment that x_1 was locked. Substituting in eq. (5.5) leads to:

$$Q_{s,j} = \frac{1}{n_2} \cdot \frac{\partial g(\Delta x)}{\partial \Delta x} \tag{5.10}$$

3. $\boldsymbol{L_1 = 1}$ **and** $\boldsymbol{L_2 = 0}$**:** Output position 2 is locked, meaning that $\dot{x}_2 = 0$. Since x_2 is now constant, the spring deflection again linearly depends on the joint position:

$$\Delta x = x_1 - c_2 \tag{5.11}$$

$$= \frac{1}{n_1}x_j - (1 + \frac{n_1}{n_2})c_2 \tag{5.12}$$

   where c_2 is equal to x_2 at the moment that x_2 was locked. This leads to the generalized force:

$$Q_{s,j} = -\frac{1}{n_1}\frac{\partial g(\Delta x)}{\partial \Delta x} \tag{5.13}$$

   In comparison to mode 2, this force is reversed.

4. $\boldsymbol{L_1 = 0}$ **and** $\boldsymbol{L_2 = 0}$**:** The two output positions are locked and thus the joint position is locked as well. If the spring is deflected, it will remain deflected while being in this mode of operation.

From eqs. (5.10) and (5.13), it follows that for the same deflection of the spring, the joint force can be negative or positive. Now suppose that $\dot{x}_j$ is positive. If we want to decelerate the joint, we switch to mode 3, where the force on the joint is negative. Once the joint is decelerated to zero velocity and the kinetic energy is transferred to potential energy in the spring, we switch to mode 4. From mode 4, we can release the potential energy by accelerating in positive or negative direction (respectively mode 3 and 2). Similarly, if $\dot{x}_j$ had been negative, we would have switched to mode 2 to decelerate the joint.

From eqs. (5.10) and (5.13), it is also clear why an ideal differential is advantageous: if $n_1 = n_2$, the stiffnesses while accelerating in positive and negative direction are identical.

5.3 Prototype Design

We built a prototype of the BIC-PEA in order to test how much it can reduce the energy consumption of a standard robotic joint actuated by a DC motor. In this section we first discuss selection of the spring. Then, we describe the hardware design of the BIC-PEA. And finally, we present a model and show how the model corresponds to previous measurements in [170].

5.3.1 Spring selection

The spring is one of the most important component of the BIC-PEA. Selection of the spring will influence the amount of energy that is saved in certain tasks. The tasks are defined by a certain allocated time to move t_f and distance to move x_f. We hypothesize that the savings will be highest in tasks where the 'natural time to move' t_n corresponds with the required time to move t_f. The natural time to move is the time it takes to move from the start to the goal position in a frictionless system, without actuation. We will verify the hypothesis in simulation in section 5.5.3. Following this hypothesis, we will now calculate the time to move for a frictionless system with BIC-PEA and without actuation and derive the spring stiffness that leads to equality of t_f and t_t.

Therefore, assume a frictionless system in which the only force that acts is the spring force and where the spring is exploited up to a maximum excursion Δx_{max}. The motion of this system can be split into three phases: an acceleration phase, a phase

with constant velocity and a deceleration phase. Since the motion is symmetrical, the first and the last phase take an equal amount of time. We start at state $x_j = 0$ and $\dot{x}_j = 0$. In the first phase, the motion is described by

$$x_j(t) = -\Delta x_{max} \cos\left(\sqrt{\frac{k_j}{J}}t\right) + \Delta x_{max} \tag{5.14}$$

where J is the inertia of the robot. This phase ends at time

$$t_1 = \frac{1}{2}\pi\sqrt{\frac{J}{k_j}} \tag{5.15}$$

The velocity at the end of this phase is equal to

$$\dot{x}_j = \Delta x_{max}\sqrt{\frac{k_j}{J}} \tag{5.16}$$

and since the distance that has to be traveled in the second phase is $x_f - 2\Delta x_{max}$, this phase has a duration of

$$t_2 = \frac{x_f - 2\Delta x_{max}}{\Delta x_{max}\sqrt{\frac{k_j}{J}}} \tag{5.17}$$

The total time it takes for this system to move to x_f is

$$t_n = 2t_1 + t_2 = \sqrt{\frac{J}{k_j}}\left(\pi - 2 + \frac{x_f}{\Delta x_{max}}\right) \tag{5.18}$$

This t_n is the natural time to move. Now assuming that this time corresponds with the time to move (i.e. $t_n = t_f$), we can write the stiffness as function of the system and task parameters t_f and x_f:

$$k_j = J\left(\frac{\frac{x_f}{\Delta x_{max}} + \pi - 2}{t_f}\right)^2 \tag{5.19}$$

When choosing the BIC-PEA parameters Δx_{max} and k_j, two things should be taken into account. First, the minimum distance $x_{f,min}$ that the BIC-PEA should move is twice the maximum displacement of the spring Δx_{max}. Therefore, this maximum spring displacement should be chosen to match the smallest distance that the system is expected to move:

$$\Delta x_{max} = \frac{1}{2}x_{f,min} \tag{5.20}$$

Second, since the energy savings are expected to drop for tasks that are slower or faster than the natural time to move, k_j should be optimized to fit the different task

parameters. A good approximation is to take the average optimal k_j. This means, finding the mean k_j for all expected tasks combinations of $x_{f,i}$ and $t_{f,i}$ using

$$k_j = \frac{1}{N} \sum_{i=1}^{N} J \left(\frac{\frac{2x_{f,i}}{x_{f,min}} + \pi - 2}{t_{f,i}} \right)^2 \tag{5.21}$$

where N is the number of expected task combinations.

5.3.2 Mechanical design

In this section, we explain the design of our hardware prototype (see Fig. 5.1). A schematic drawing of the design is shown in Fig. 5.3. We respectively discuss the differential mechanism, the spring, the locking mechanisms and the mechanical properties.

As a differential mechanism, we use a planetary differential (see Fig. 5.3). The positions and velocities in this mechanism are rotational. Therefore, the generalized forces are actually torques. A planetary differential consists of two internal gears, one or multiple pairs of planet gears and a joint axis. Within one pair of planet gears, each gear engages with a different internal gear and the gears engage with each other. Our prototype has two pairs of planet gears, as can be seen in Fig. 5.4b. The angular displacements x_1 and x_2 of the separate sides of this planetary differential are described by:

$$x_1 = x_j + r_p x_{p1} \tag{5.22}$$

$$x_2 = x_j + r_p x_{p2} \tag{5.23}$$

where r_p is the effective radius of the planet gears and x_{p1} and x_{p2} are the positions of the planet gears (see Fig. 5.3). Since the two planets mesh with each other such that $x_{p1} = -x_{p2}$, the overall motion is described by:

$$x_j = \frac{1}{2} x_1 + \frac{1}{2} x_2 \tag{5.24}$$

This differential mechanism is an ideal differential (i.e. $n_1 = n_2 = 0.5$).

We chose to use a torsion spring with a spring stiffness of 0.17 Nm/rad and a maximum displacement of 0.5π rad. We attached two pegs to each side of the spring to connect to the internal gear (see Fig. 5.4a). The parameters of the BIC-PEA were chosen for the task with a distance to move of $x_f = 2\pi$ rad and a time to move of $t_f = 1$ s.

In order to test the spring mechanism, we used two large brakes as locking mechanisms to lock the internal gears. These brakes consist of rubber plates that are each pushed against a Vulka SF-001 braking disk by a solenoid (see Fig. 5.5).

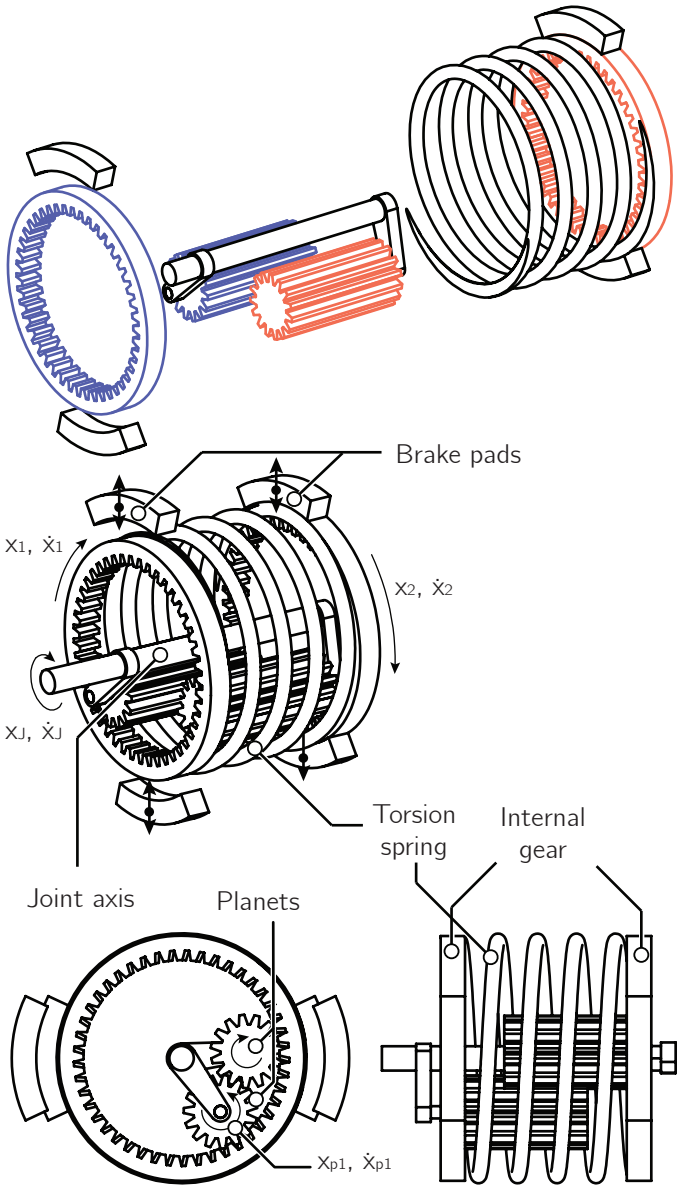


Figure 5.3: A schematic drawing of the final design of the BIC-PEA. It consists of a planetary differential, a torsion spring and two brakes. The planetary differential consists of two internal gears, two planet gears and one planet carrier (the joint axis). The planets have a lateral offset, such that each planet meshes with the different internal gear of the corresponding color and they mesh with each other in between the internal gears. The planet axes are connected to the joint axis by cranks so that the distances between the joint axis and the planet axes are fixed. The planets can rotate around their axes independent from the rotation of the joint axis. The spring connects the two internal gears. The differential mechanism causes the joint axis to rotate with the average velocity of the two internal gears. The brakes can lock the internal gears with respect to the ground.

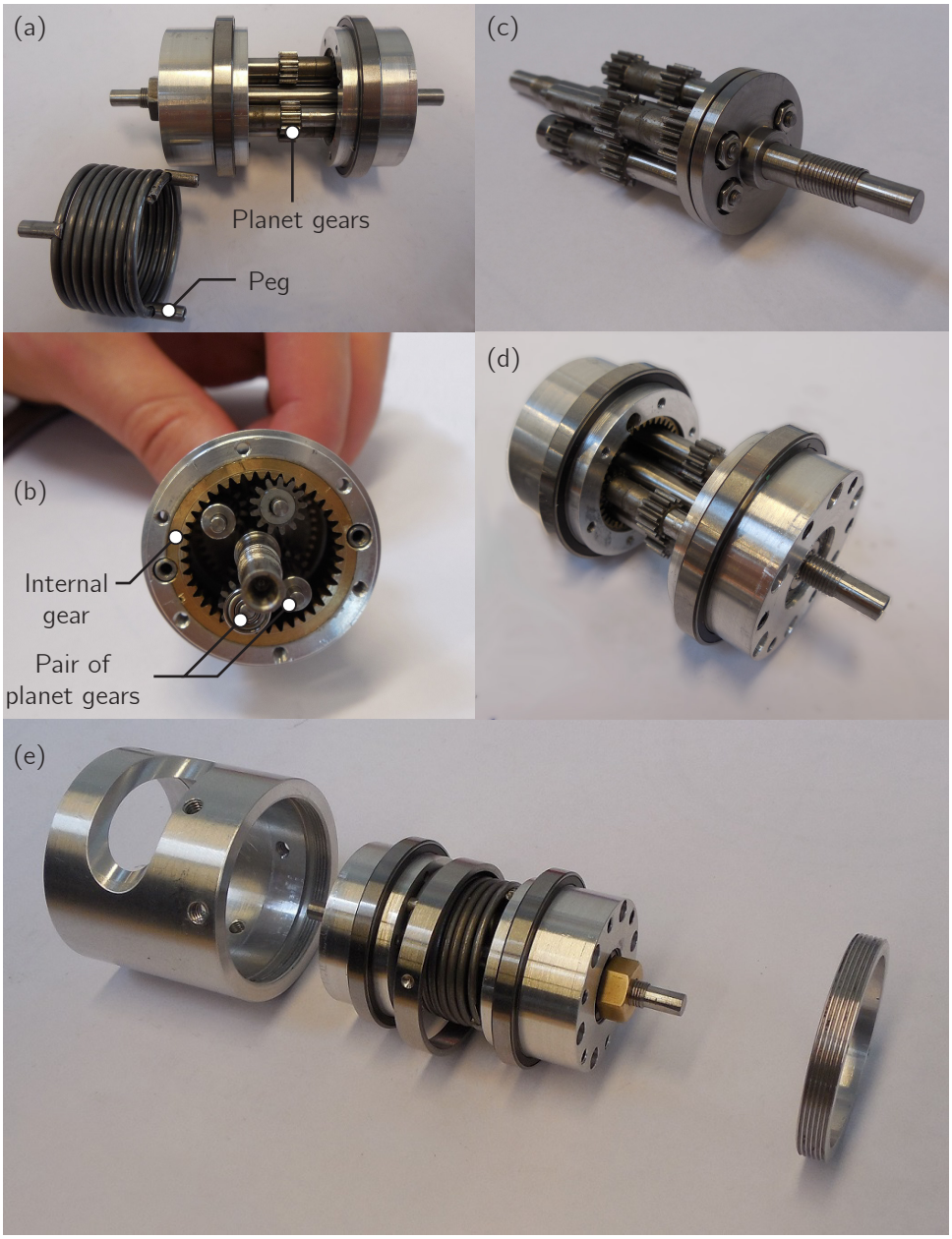


Figure 5.4: Photographs of the inside of the prototype.

The BIC-PEA without brakes as shown in Fig. 5.1 weighs 202 g and fits in a cylinder with a length of 51 mm and a diameter of 45 mm. The transfer ratio of the joint axis to the spring (when one brake is locked) is 1:2. This means that the apparent spring stiffness at the robot joint is 0.68 Nm/rad.

5.3.3 Modeling the BIC-PEA

In [170], the characteristic of the BIC-PEA is measured. In order to generate energy-optimal trajectories, a model is required. We model the characteristic of the BIC-PEA in mode 2 and 3 as

$$Q_{s,j} = k_j(x_0 - x_j) - T_c \text{sign}(\dot{x}_j) \tag{5.25}$$

where x_0 is the joint's position when the spring is in equilibrium and T_c is the Coulomb friction torque. Coulomb friction is mainly generated by the gears and is the main cause for hysteresis in the mechanism. We used the Nelder-Mead simplex direct search algorithm of `fminsearch` in MATLAB to identify the values for k_j and T_c that lead to the least squared error between measurements [170] and model. The obtained values are $k_j = 0.71$ Nm/rad and $T_c = 0.12$ Nm. The fit has a root mean squared error of 0.08 Nm and a peak error of 0.17 Nm. In mode 1, $Q_{s,j}$ is equal to zero and in mode 4, the acceleration of the joint is zero.

5.4 Experimental setup

In this section we explain the experimental setup we used to measure the reduction in energy consumption. First, we describe the measurement setup. Then, we explain how we find energy-optimal trajectories in simulation. And finally, we describe the hardware experiments.

5.4.1 Measurement setup

Fig. 5.5 shows the test setup we used to analyse the energy consumption. It consists of a Maxon RE-30 brushed DC motor that connects to a body with inertia through a timing belt. The inertia is represented by an aluminum tube of length 650 mm that connects to the joint in its center. At both end points of the tube, a brass block is placed inside the tube to increase the inertia. This setup is not influenced by gravity. We chose such a setup because it was already shown in [231] that gravity can be compensated by parallel springs. The motor was chosen based on availability. The identified parameters of the test setup are listed in Table 5.1. We used a gearbox ratio of 1:18 because initial simulations showed that this ratio leads to the lowest energy consumption when the BIC-PEA is not attached. Gearbox selection is discussed further in section 5.7.4.

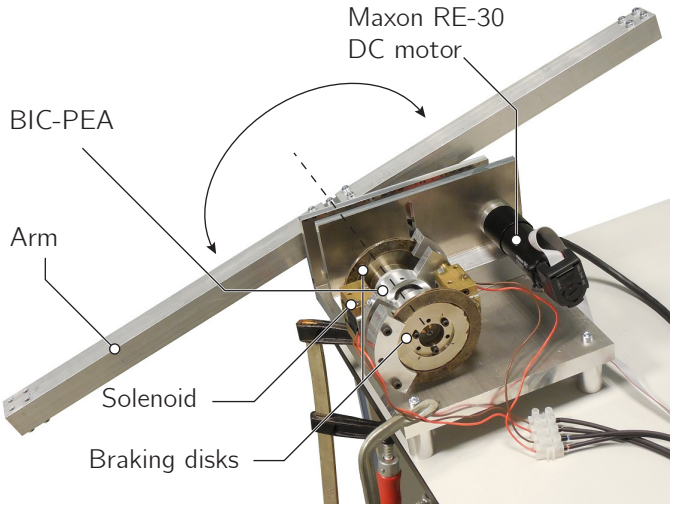


Figure 5.5: The setup to test the influence of the BIC-PEA on the energy consumption. It consists of a symmetric arm that rotates in a vertical plane. When the BIC-PEA is attached, it directly connects to the axle of the arm. A Maxon RE30 DC motor with a gearbox of 1:18 drives the system through a timing belt.

Table 5.1: The model parameters of the one degree of freedom test setup. The values are obtained through a system identification of the setup.

Parameter arm	Symbol	Value
Inertia	J	0.029 kgm^2
Coulomb friction	T_{cf}	0.06 Nm
Torque-dependent friction	μ_{tf}	21 %
Torque constant	k_t	25.9 mNm/A
Motor resistance	R	1.5 Ω
Gearbox ratio	n	1:18
Spring stiffness	k_j	0.71 Nm/rad

5.4.2 Simulation experiments

Since we are focusing on rest-to-rest motions, the task that has to be performed is to move from one position to another, both with zero velocity. First, we analysed how the performance depends on the two task parameters t_f and x_f. Therefore, we varied t_f between 0.7 s and 1.7 s and x_f between π rad and 3π rad.

Secondly, we varied the stiffness, to verify the stiffness selection hypothesis in section 5.3.1. Meanwhile, we keep Δx_{max} constant. While varying the stiffness, we obtain the tasks for which the energy savings are maximal. If the hypothesis is true, these tasks satisfy eq. (5.18).

In order to fairly compare the system with and without the BIC-PEA, we have to know how it should move in order to consume as little energy as possible in each case. Therefore, we used optimal control to find energy-optimal trajectories off-line. The model of the arm that we used is:

$$\ddot{x}_j = \frac{T_m + Q_{s,j} + T_f}{J} \tag{5.26}$$

where T_m is the torque applied by the motor, T_f is the friction torque and J is the total inertia about the joint. The motor torque is equal to

$$T_m(t) = k_t n I(t) \tag{5.27}$$

where I is the current through the motor, k_t is the torque constant of the motor and n is the transfer ratio of the gearbox. T_f is equal to

$$T_f = -(T_{cf} + \mu_{tf}|T_m|)\text{sign}(\dot{x}_j) \tag{5.28}$$

where T_{cf} and μ_{tf} are the Coulomb friction torque and the torque-dependent friction coefficient respectively.

We used the optimal control software package GPOPS-II [161] to find energy-optimal trajectories between the two positions. As solver we used the SNOPT version that comes with GPOPS II. The cost function that we used is the integral over the electrical motor power P, which is equal to

$$P = T_m \dot{x}_j + I^2 R \tag{5.29}$$

where R is the terminal resistance of the motor windings. Note that the power can become negative when the motor torque opposes the velocity. This would lead to a

decrease in the cost function and models recapturing energy electrically. Therefore, using this cost function, we compare the energy consumed by the system with the BIC-PEA to a system that can use the motor as a generator. The optimization has a free end time t^* with a maximum of t_f. The overall optimal control problem is described by:

$$\begin{aligned} \underset{T_m(t)}{\text{minimize}} \quad & \int_0^{t^*} P \mathrm{d}t \\ \text{subject to} \quad & t^* \leq t_f \\ & x(0) = 0 \\ & x(t^*) = x_f \\ & \dot{x}(0) = \dot{x}(t^*) = 0 \end{aligned} \tag{5.30}$$

We assume that it is optimal to switch the BIC-PEA depending on the initial and the final positions. Therefore, for $x_j < \Delta x_{max}$ the generalized force $Q_{s,j}$ is equal to

$$Q_{s,j} = k_j(\Delta x - x_j) - T_c \text{sign}(\dot{x}_j) \tag{5.31}$$

for $x_j > x_f - \Delta x_{max}$, it is equal to

$$Q_{s,j} = k_j(x_f - \Delta x - x_j) - T_c \text{sign}(\dot{x}_j) \tag{5.32}$$

and $Q_{s,j} = 0$ otherwise.

5.4.3 Hardware experiments

In the hardware experiments, the energy-optimal trajectories that were calculated off-line, are followed by a manually tuned PD controller in combination with the feedforward controller that follows from the optimization. For the system without the BIC-PEA attached, the tuned controller gains are $[K_P, K_D] = [1,\ 0.2]$[Nm/rad, Nms/rad]. In the system with the BIC-PEA, a smaller proportional gain appeared to be optimal: $[K_P, K_D] = [0.5,\ 0.2]$[Nm/rad, Nms/rad].

The solenoid brakes have a switching time that is not negligible. Therefore, they are switched at a fixed duration before reaching the switching positions. When disengaging, the brake is switched 0.1 s before reaching the switching position. When engaging, this is 0.02 s.

The trajectories that will be tested on the hardware setup have a time to move that varies between 0.7 s and 1.7 s and a fixed distance to move of 2π rad. In practice, a task description will consist of several times to move and moving distances. To show the versatility of the BIC-PEA, we composed a task where the arm has to move

between four positions. First, it moves from 0 rad to 3π rad in 1 s. Then, it moves to 4.5π rad in 1.5 s. Next, it moves to 1.5π rad in 1.5 s. And finally, it moves back to 0 rad in 1 s.

The energy consumption of those motions is calculated by numerically integrating the electrical power:

$$E = \int_0^{t_f} UI \mathrm{d}t \tag{5.33}$$

where U is the voltage that is applied on the motor. Note that this energy consumption does not include the energy consumed by the brakes. There are two types of motions, which we will call *type 1* and *type 2*. In motions of type 1, the next motion continues in the same direction as the previous motion. So after deceleration, the arm will accelerate while moving in the same direction. In motions of type 2, the next motion is in the opposite direction of the previous motion. This type of motion leads to back-and-forth motions. We repeated every task 18 times: 9 times for type 1 and 9 times for type 2. The results are then averaged.

5.5 Simulation results

In this section, we first show the results of the optimal control study for a nominal task. Then, we show the results for multiple task parameters. And finally, we verify the stiffness selection hypothesis from section 5.3.1.

5.5.1 Energy savings for a nominal task

Fig. 5.6 shows the optimized motions for moving one revolution in one second, with and without the BIC-PEA attached. As can be seen in Fig. 5.6d, when the BIC-PEA is attached, the robot consumes 72 % less energy.

5.5.2 Influence of task parameters

Fig. 5.7 shows the energy savings as function of the task parameters. This figure shows that the maximum percentage of energy that can be saved is approximately 73 % and that there is a manifold of tasks in which this can be achieved. The figure shows that the savings for other tasks are lower, but for a considerable amount of tasks, the system with BIC-PEA still consumes less energy than the system without BIC-PEA.

5.5.3 Influence of the spring stiffness

Fig. 5.8 shows two sets of data. First, it shows the manifolds for x_f and t_f that follow from eq. (5.19) for nine different stiffnesses. These manifolds represent the natural

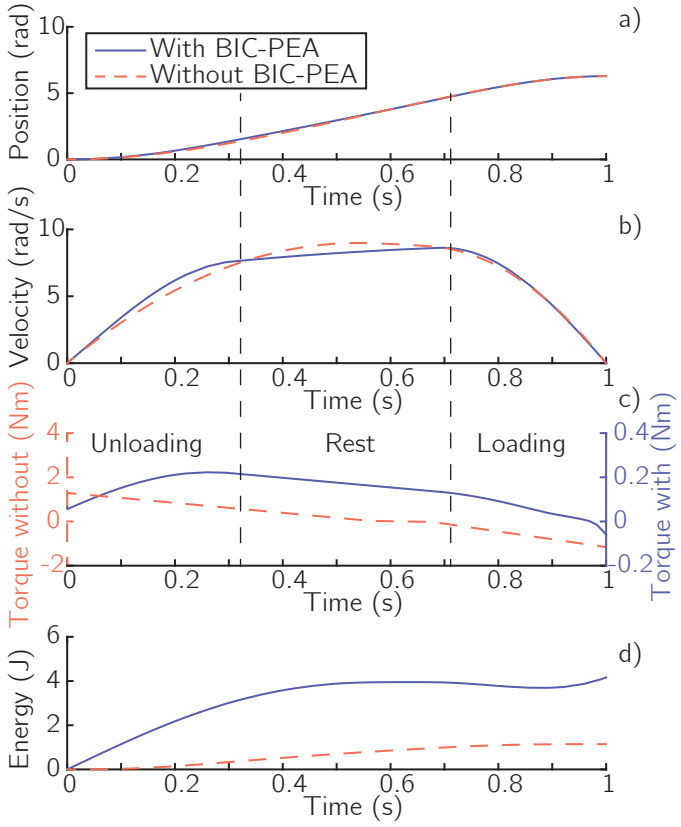


Figure 5.6: The optimized trajectories to move one rotation in one second with and without the BIC-PEA attached. Note that in c), the torques for the motion with BIC-PEA are multiplied by a factor 10. The vertical striped lines indicate the switching times of the locking mechanisms. Therefore, they divide the motion into three phases: a phase in which the spring unloads, a phase in which the spring is in rest and a phase in which the spring is loading.

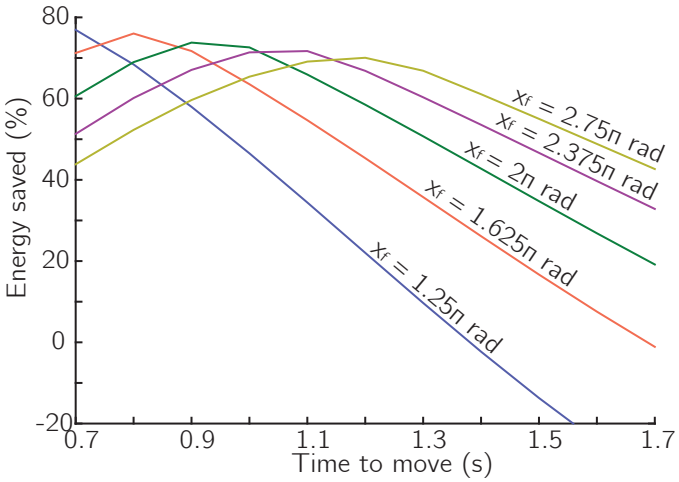


Figure 5.7: The energy savings as function of the task parameters. For every distance to move, there is a time to move that leads to approximately 73 % energy savings. This maximum percentage of energy saved decreases when the distance to move increases.

motion of the system. Second, it shows the manifolds with the maximum energy savings in simulation for the same nine stiffnesses. It shows that the two manifolds are similar, especially at high stiffnesses. The similarity between the analytical manifolds and the optimal manifolds in simulation shows that the tasks in the optimal manifolds in simulation indeed coincide with the natural motion of the system. The differences between the two sets of manifolds is due to the effects of friction, which is larger at low stiffnesses.

5.6 Hardware results

Fig. 5.9 shows the energy that is consumed by the hardware setup with and without the BIC-PEA attached. The figure shows that the results in simulation and hardware experiments are comparable. For the system without the BIC-PEA attached, the energy consumption decreases when the time to move is increased. For the system with BIC-PEA, the same holds until a time to move of approximately 1 s, after which the consumed energy is constant. The latter is caused by the fact that for larger times to move, the energy-optimal trajectory is to move in approximately 1 s and then stand still at the goal position.

Fig. 5.10 shows the percentage of energy that is saved by implementing the BIC-PEA as function of the time to move. The figure shows the same trend in simulation and

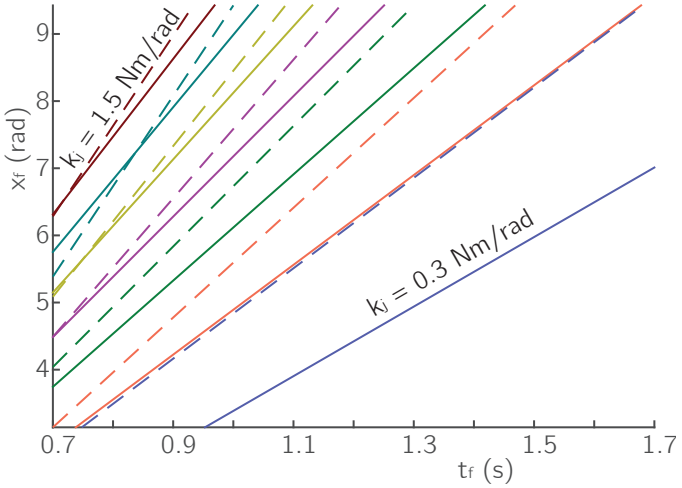


Figure 5.8: The optimal task manifold for nine different stiffnesses. The figure shows the manifold both from an idealized model (solid lines) and from simulations with a realistic model (dashed lines). It shows that the two models are comparable, especially at high stiffnesses.

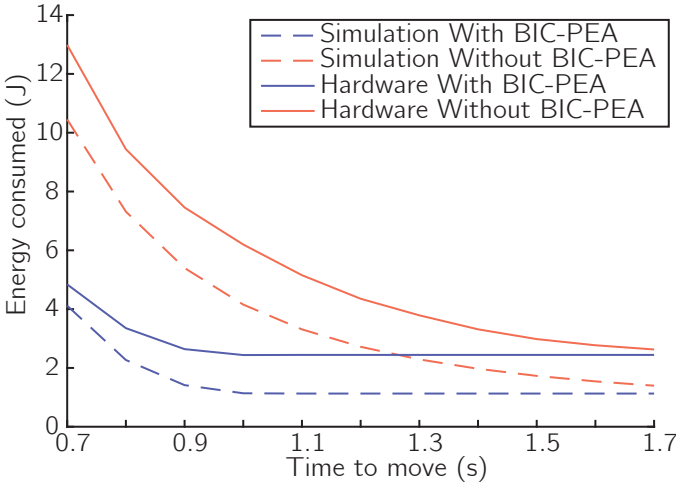


Figure 5.9: The energy consumption of the arm as function of the time to move for a distance to move of 2π rad. The lines show the energy consumed in simulation and in hardware experiments for the system with and without the BIC-PEA attached.

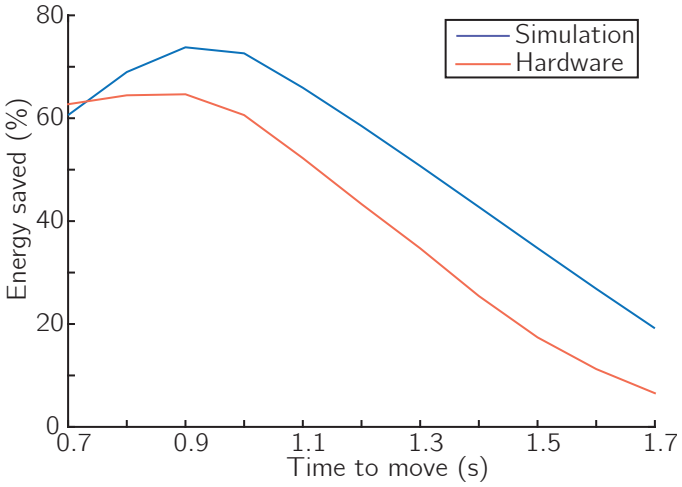


Figure 5.10: The energy savings of the arm as function of the time to move for a distance to move of 2π rad. The lines show the energy consumed in simulation and in hardware experiments.

hardware experiments: there is a peak at a time to move of approximately 0.9 s and the savings decrease when the time to move is increased. The peak in hardware experiments in lower than in simulation: 65 % in hardware experiments and 73 % in simulation.

Fig. 5.11 shows the hardware results for performing the realistic task. Figs. 5.11a and 5.11b show the positions as function of time for the system with and without the BIC-PEA for 10 runs. Fig. 5.11c shows the energy consumed over time. The system without BIC-PEA consumed 25.0 ($\pm$ 0.1) J to complete the task. The system with BIC-PEA consumed 11.8 ($\pm$ 1.4) J. This means that implementation of the BIC-PEA resulted in a 53 % lower energy consumption for this task.

5.7 Discussion

In this chapter, we presented the concept of the BIC-PEA and showed its potential in reducing the energy consumption of robots. In this section we discuss the performance, the applicability, the use of locking mechanisms, selection of the gearbox and other embodiments of the BIC-PEA.

5.7.1 Performance

The simulation results showed that theoretically 73 % energy can be saved by implementing the BIC-PEA. Hardware results showed the same trends as simulation

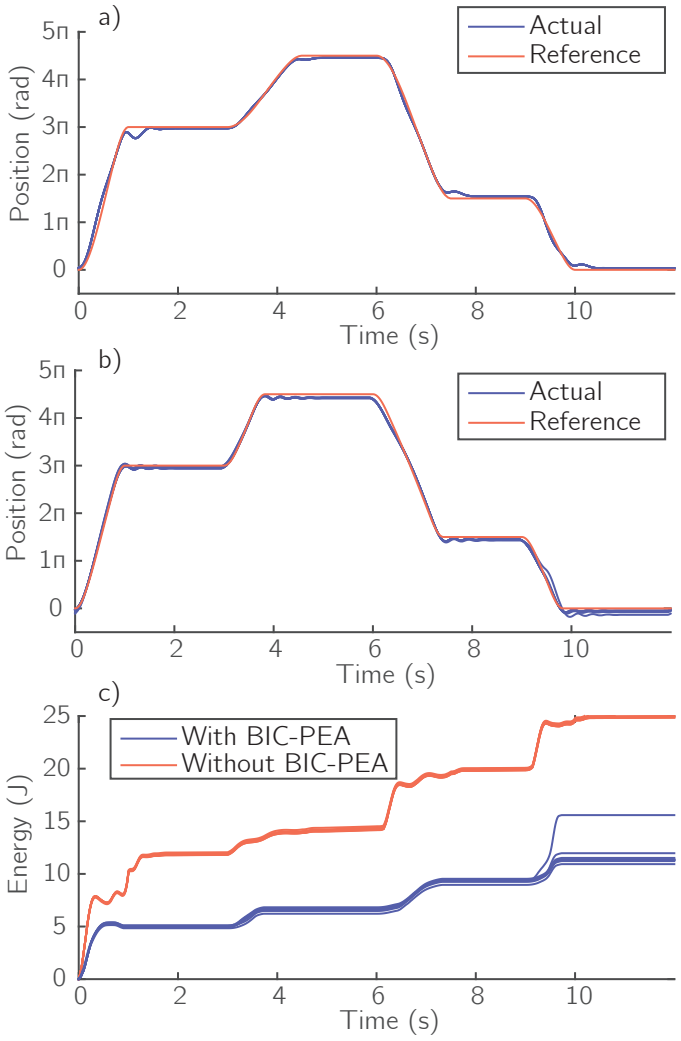


Figure 5.11: The hardware results for performing a realistic task. The arm moves from 0 rad to 3π rad to 4.5π rad to 1.5π rad and then back to 0 rad. The times to move are 1 s, 1.5 s, 1.5 s and 1 s. a) The position of the arm without the BIC-PEA. b) The position of the arm with the BIC-PEA. c) The energy consumed over time for the system with and without the BIC-PEA.

results, but the maximum energy savings were 65 %. Furthermore, in hardware experiments, the robot consumed more energy than in simulation. In the system without the BIC-PEA, this is caused by an imperfect model, introducing the need for feedback. In the system with BIC-PEA, another cause for the larger energy consumption is the imperfectness of the locking mechanisms. Since the brakes have a non-zero switching time and sometimes slip slightly, the efficiency of the BIC-PEA decreases.

In hardware experiments, the energy consumption was measured for rest-to-rest motions of *type 1* and *type 2*. In the results, the energy consumptions of these two types of motions were averaged. It should be noted that the results were quantitatively different. For instance, for the motion with a time to move of 0.9 s, the energy consumptions for the system with BIC-PEA attached were 2.94 ($\pm$0.20) J and 2.34 ($\pm$0.46) J for type 1 and type 2 motions respectively. For the system without BIC-PEA attached this was 8.20 ($\pm$0.57) J and 6.71 ($\pm$0.13) J. Although the exact energy consumptions were different, the energy savings for both types of motions are similar: 64 % and 65 % respectively.

The fact that the energy consumption of the two types of motions is different is caused by steady-state errors at the goal positions. As is also visible in Fig. 5.11, the position has a steady-state error by stopping just before reaching the goal position. When the next motion is in the opposite direction, the arm starts with a lead with respect to the reference. When the next motion is in the same direction, the arm starts with a lag. The feedback control effort it takes to compensate for this lag causes the higher energy consumption in continuing motions.

The two main losses in robotic arms are friction and copper losses. The energy consumption of the arm without BIC-PEA consists for 71 % of copper losses and 29 % of friction losses, for the motion displayed in Fig. 5.6. When attaching the BIC-PEA, the friction losses remain approximately equal, while the copper losses are decreased. The copper losses then account for 11 % of the energy consumption and the friction accounts for 89 %. Therefore, future research should focus on reducing the friction losses in robots in order to further reduce the energy consumption.

As mentioned in the introduction, there are two concepts that compete with the BIC-PEA in terms of energy efficiency. First, mechanical energy could be recaptured electrically, such as in the Cheetah robot [192]. The advantage of this is that the energy flow remains fully controllable. However, as mentioned in the introduction, the energy that can be saved using this approach is only 40 %. Furthermore, the 65 % savings achieved with the BIC-PEA are in comparison with a system that also recaptures energy electrically (note the descent of the energy consumed at the end

of the motion in Fig. 5.6d). To achieve the 40 % reduction, specialized motors are required. Secondly, an existing clutched parallel elastic actuator (CPEA) could be used [77]. This CPEA has shown to be able to decrease the energy consumption with 80 %. However, such a CPEA cannot use the potential energy in the spring to accelerate in any desired direction and therefore decreases the versatility of the arm. For instance, the task in Fig. 5.11 cannot be performed using this existing CPEA.

5.7.2 Applicability

The BIC-PEA is suitable for all robots that perform rest-to-rest motions. The most common example of such a task in industry is a pick-and-place task with a robotic arm. However, the applications of BIC-PEAs far exceed robotic arms. In fact, all machines that decelerate and accelerate again in a repetitive manner would benefit from an efficient way of energy recapture. For instance, one could imagine that a wheeled robot that constantly has to come to a standstill and then accelerate again could benefit from implementation of a BIC-PEA. Also, a manufacturing process where belts transport a batch of products in intermittent fashion (to allow discrete operations to be performed on the products) could benefit.

Future research on CPEAs should focus on other configurations with more locking mechanisms that increase the applicability of the BIC-PEA. For instance, an extra clutch between the axis of the robot and the input shaft of the BIC-PEA would allow for the robot to move while the spring remains loaded.

5.7.3 Locking mechanisms

In the current prototype, we used solenoid actuated brakes to lock the internal axes of the differential mechanism. In the results, we excluded the energy consumption of the solenoids, because this assumes an ideal locking mechanism. Various types of locking mechanism have been used in robotics [169]. A locking mechanism that was specifically designed to have a low energy consumption is the statically balanced brake [168]. An interesting direction of future research would be to consider different locking mechanisms. For instance in some applications, active latches can be used in which the position of locking and unlocking can be set by an actuator.

5.7.4 Gearbox selection

In this chapter, we used the same gearbox ratio for the system with and without the BIC-PEA. In the introduction we stated that reduction of actuator torques should also allow for smaller gearbox ratios, leading to less gearbox friction, which again reduces the energy consumption. Initial simulation studies showed that a gearbox ratio of 1:6.2 would be optimal for the task of moving 2π rad in 1 s. However, with

such a small gearbox ratio, the feedback control action leads to high copper losses. Therefore, in our experience, the gearbox ratio cannot be taken too low, so we used a gearbox ratio of 1:18.

5.7.5 Other configurations

There are numerous configurations of BIC-PEAs that all have the same functionality. However, there are two configurations that we would like to mention because of their distinct advantages: the hydraulic and the planetary gear versions.

A schematic drawing of a hydraulic version is shown in Fig. 5.2. The advantage of this configuration is that both the differential and the locking mechanisms are relatively simple: the differential is a three-way tube and the locking mechanisms are valves. Of course, such a system would only work if the leakage and friction are minimal.

The planetary gear version is shown in Fig. 5.12. The advantage of this configuration is that the differential is relatively compact. The disadvantage however is that a planetary gear is not an ideal differential as described in section 5.2.

5.8 Conclusion

In this chapter, we presented the concept of the Bi-directional Clutched Parallel Elastic Actuator (BIC-PEA). The BIC-PEA is suitable for implementation in robots that perform rest-to-rest motions such as pick-and-place robots. We conclude that using the BIC-PEA, the energy consumption of such robots can be reduced significantly without compromising their capabilities. Results show that implementing the BIC-PEA leads to a decrease of the energy consumption up to 73 % in simulation, in comparison to the same system that also allows for recapturing energy electrically. In hardware experiments a reduction of 65 % was reached. This is larger than all other concepts with the same functionality.

Acknowledgement

This work is funded by research programme STW, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO), and by a Marie-Curie career integration grant PCIG13-GA-2013-618899.

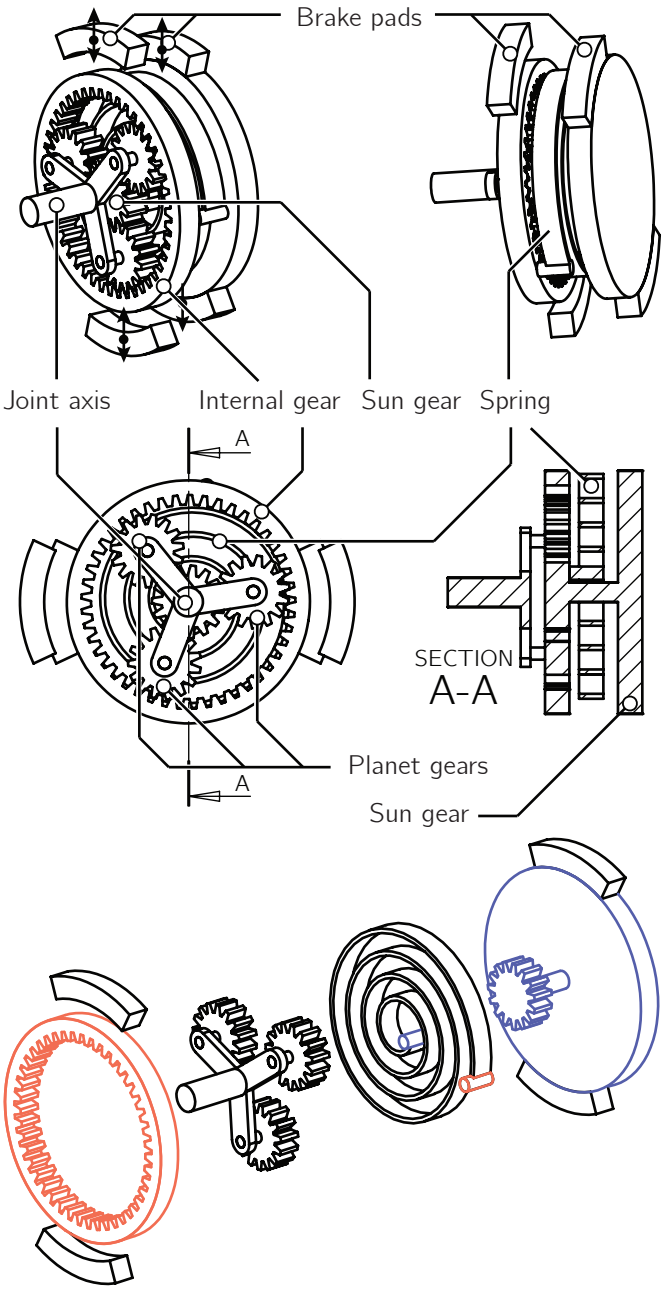


Figure 5.12: A planetary gear version of the BIC-PEA, which is compact in axial direction. The spring is placed between the sun and the internal gear of the planetary gear, as indicated by the corresponding colors.

6

Clutched Elastic Actuators

Michiel Plooij, Wouter Wolfslag and Martijn Wisse,
Submitted to: Transactions on Mechatronics.

Abstract

This chapter identifies the class of actuators called clutched elastic actuators (CEAs). CEAs use clutches to control the energy flow into springs. CEAs in exoskeletons, prostheses, legged robots and robotic arms have shown the ability to reduce the energy consumption and motor requirements such as peak torque and peak power. Because of those abilities, they are increasingly used in robotics. In this chapter, we categorize existing CEA designs, identify trends in those designs and provide a taxonomy to analyze their functionality. Based on a literature survey, current CEA designs are placed in nine categories, depending on their morphology. The main trend is that CEA designs are becoming more complex, meaning that the number of clutches and springs increases. We show with the introduced taxonomy that the functionality can be analyzed with a constraint matrix, a stiffness matrix and multiplication of a clutch dependent diagonal matrix with a oriented incidence matrix. This taxonomy eases the analysis of the functionality of CEAs. Furthermore, it can lead to new CEA designs in which the number of resulting stiffnesses grows exponentially with the number of springs and clutches.

6.1 Introduction

In autonomous robots, energy consumption is an important performance criterion, because this directly influences their uptime. Clear examples include walking robots [39], household robots [135], prostheses [11, 214] and orthoses [49, 177]. One of the most effective techniques to obtain a low energy consumption is the efficient recapture of negative work. There are multiple options to store the energy recaptured from the robot (e.g. electrical, chemical, potential, etc.), of which potential energy is the most promising in terms of efficiency. This efficiency can even approach 100 %. Compared to other potential energy storages, springs are relatively compact and therefore preferred. Actuators that use springs to temporarily store energy are called elastic actuators (EAs). The two most well known EAs are series elastic actuators (SEAs) [175] and parallel elastic actuators [13, 83] (PEAs).

The problem of EAs is that the timing of energy storage and release is not independently controllable from the position of the joints and/or motor. This lack of control of the energy storage limits the versatility of robots with EAs, especially when using PEAs. For example, Shirata et al. [200] designed a walking robot with a leg that includes a spring mechanism to statically balance the robot when standing on that leg. During the stance phase, this spring mechanism reduces the torques that the

motors have to deliver. However, during the swing phase, the motors have to counteract these springs. If the stiffness of those springs would be fully controllable, the stiffness could be eliminated during the swing phase.

In recent years, the use of clutches in EAs has become popular as a tool to control the energy flow in springs. We call such EAs clutched elastic actuators (CEAs). Such CEAs have mostly been applied in legged robots [55, 56, 58, 213], underactuated systems [136] and robotic arms [13]. An alternative for using clutches to control the energy flow in the springs is the use of continuously variable transmissions (CVTs). CVTs have mainly been used to control the stiffness of springs in series with the actuator [79, 228]. In principle, such CVTs could also be used to control the stiffnesses of parallel springs. With a CVT, the energy flow of the springs would become fully controllable [210]. However, current CVT designs are not applicable. They are typically based on a wheel rolling on a surface, meaning that in order to be able to transfer high forces, the wheel should be pushed strongly against the surface to prevent slip perpendicular to the rolling direction. This increases the friction in the overall system, leading to a limited life time due to wear and a low energy efficiency. Furthermore, it makes it difficult to vary the transfer ratio. Clutches are inherently simpler components, because they switch between completely locking and completely unlocking. Furthermore, they are widely available in various forms [169].

Different applications require different functionalities from their CEAs and thus various types of CEAs have been designed. However, the full capabilities and limitations of CEAs have never been studied. Having a method to analyze the functionality of CEAs will become increasingly important, due to their increasing complexity. We envision future CEAs consisting of many springs and many clutches. Such CEAs show resemblance with human muscles that consist of many elastic elements that can be locked. In order to prepare for those future CEAs, we provide a method to categorize and analyze both existing and future CEA designs.

Therefore, the goal of this chapter is threefold. The first goal is to categorize existing CEA designs. The second goal is to identify trends in those designs. The third goal is to introduce a taxonomy to analyze functionalities of all possible CEA designs. These goals are reached using the following structure. Section 6.2 gives a definition of CEAs and their components. Section 6.3 provides an overview of existing CEA designs and identifies trends. Section 6.4 evaluates how we can define what functionalities a certain CEA design incorporates. Section 6.5 then provides a taxonomy for analyzing the functionalities in CEAs. Synthesis and different configurations of CEAs are discussed in section 6.6. Finally, the chapter ends with a discussion in section 6.7 and a conclusion in section 6.8.

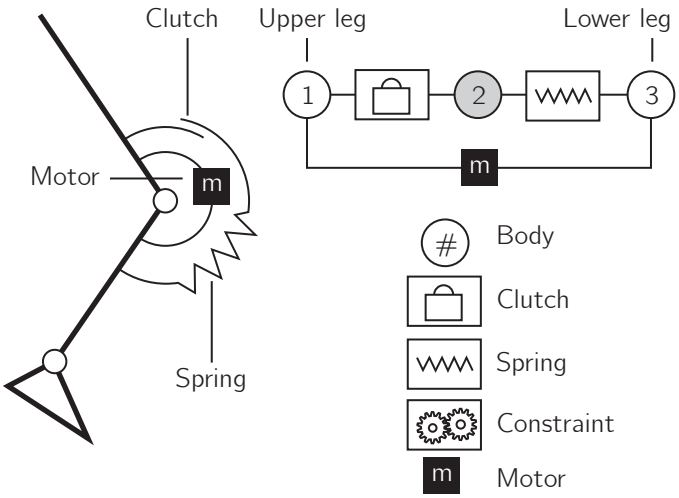


Figure 6.1: An example of a CEA, showing the visualization that is used in this chapter. CEAs consist of bodies that are connected with clutches and springs. In the leg on the left, one spring is placed in series with a clutch between the lower and the upper leg. A motor is also placed between the lower and the upper leg. The schematic on the right shows a visualization of this mechanism. Body 1 connects to the upper leg and body 3 connects to the lower leg. Body 2 is an internal body in between the clutch and the spring and is therefore visualized in grey. Note that this example does not include a constraint.

6.2 Components of CEAs

In this section, we briefly define both the terminology and the visualization we use in this chapter. We define CEAs as all mechanisms that consist of at least one spring and one clutch and possibly incorporating gears or differentials. The connections of those components connect to bodies that have a continuous state (position and velocity).

A *spring* is an elastic element and in this chapter we assume that it has two connections. The potential energy in the spring is a function of the positions of its connections. A *clutch* is a locking device that switches between allowing and preventing relative motion between two bodies [169]. This means that when the clutch is engaged, the relative velocity between the two bodies is zero. In this chapter, we consider all clutches to be ideal locking mechanisms as defined in [169]. This means that we disregard the differences between locking mechanisms such as brakes, ratchets, latches, non-backdrivable gearing and singular locking mechanisms. Section 6.7.3 will discuss how different locking mechanisms might be included in figure work. A *gear* determines a constant transfer ratio between two bodies. A generalized

version of a gear is a *differential*, which imposes a linear velocity constraint on two or more bodies.

Fig. 6.1 shows an example CEA in a robotic knee on the left and the schematic visualization of this CEA on the right. The CEA consists of one springs and one clutch. The spring is placed in series with the clutch. The clutch-spring combination and the motor are placed between the upper and the lower leg. This mechanism has three bodies: body 1 represents the upper leg, body 3 represents the lower leg and body 2 is the position of the connection between the spring and the clutch.

Note that in the proposed representation, all body positions are absolute positions. This means that the position of a joint to which the CEA connects, equals the relative position between two bodies. In Fig. 6.1, the knee angle is equal to the difference between the angle of body 1 and the angle of body 3. This also means that even when the knee is locked, the bodies of the CEA can still move with the same velocity in the same direction.

6.3 Current CEA designs

The goal of this section is to provide a categorization of existing CEA designs and to identify trends in the use of CEAs. We performed a literature survey by searching for papers that include either 'clutch', 'latch', 'locking mechanism' or 'lock' and 'spring', 'elasticity' or 'compliance'. We also searched the references of the papers we found.

We categorize the CEAs based on their morphology and briefly discuss the results that were achieved by implementing them. Existing CEAs can be subdivided into nine types that are visualized in Fig. 6.2. The order of the types was chosen based on the number of clutches that are used. In this section we discuss the types in Figs. 6.2a, b, d, f and i, because they are used most often and are significantly different from each other.

The names that we use for the types of CEAs are defined by their morphology, excluding other components such as motors. Per body, we describe how that body connects to a body with a higher number. For instance, body 1 in the CEA in Fig. 6.1 connects to body 2 through a clutch. Therefore, its name starts with 2C, where the numbers indicate the bodies it connects to and the C indicates a clutch. The connections of different bodies are separated by a dash. Since body 2 connects to body 3 through a spring, the name for the CEA in Fig. 6.1 is 2C-3S. Differentials are denoted by the numbers of the connecting bodies between brackets, followed by a D (see Fig. 6.2d). An example of the notation is a type 4S-4C-4C. This name

means that body 1 connects to body 4 through a spring, body 2 connects to body 4 through a clutch and body 3 also connects to body 4 through a clutch.

6.3.1 Type 2C-3S

The first type of CEA we discuss, consists of a spring placed in series with a clutch and has three bodies (see Fig. 6.2a). In most type 2C-3S CEA designs, body 1 connects to one side of the joint and body 3 connects to the other side. When the CEA also includes a motor, that motor is usually also placed between bodies 1 and 3. When the clutch is locked, bodies 1 and 2 are connected and the elongation of the spring depends on the relative motion of body 1 and 3. Therefore, the CEA functions as a PEA when the clutch is locked. When the clutch is unlocked, the position of body 2 does not depend directly on the position of the output bodies and the spring goes to its equilibrium position. Therefore, only the motor applies forces to the output joints when the clutch is unlocked.

Type 2C-3S have been applied both in parallel to a motor [12, 57, 77, 99] and without a motor [34, 41, 53, 54, 60, 197, 238]. There are four studies with a completely different implementation of this type of CEA. Two designs placed this CEA in series with a motor instead of in parallel [129, 222]. One design placed the motor between bodies 1 and 2, such that the rotation of the motor can be locked [186]. And another design used bodies 1 and 2 as output bodies [33] . Then, by placing a motor between body 1 and 3, they can load the spring while not applying torque on the output joint.

This type of CEA has been used most widely and the results show that even the implementation of a simple CEA can lead to significant reductions of the energy consumption, peak torque and peak power, both in robots and in humans. In walking robots and exoskeletons, implementation of such a CEA has been shown the ability to reduce the electric energy consumption of walking robots and exoskeletons up to 80 % [77, 129]. In humans, a reduction of the metabolic cost of transport by 7 % was achieved [41].

6.3.2 Type 2CS

The second type of CEA in literature consists of a spring in parallel with a clutch (see Fig. 6.2b). Typically, the CEA also includes a motor in between the two bodies. When the clutch is locked, bodies 1 and 2 are connected and thus the output is locked. When the clutch is unlocked, the output bodies are not coupled directly and both the motor and the spring apply a force on the output bodies. This type of CEA can be used when the joint has to stand still while the spring is loaded. The 2CS type has been used to lower the energy consumption of robotic arms [13], to insert the

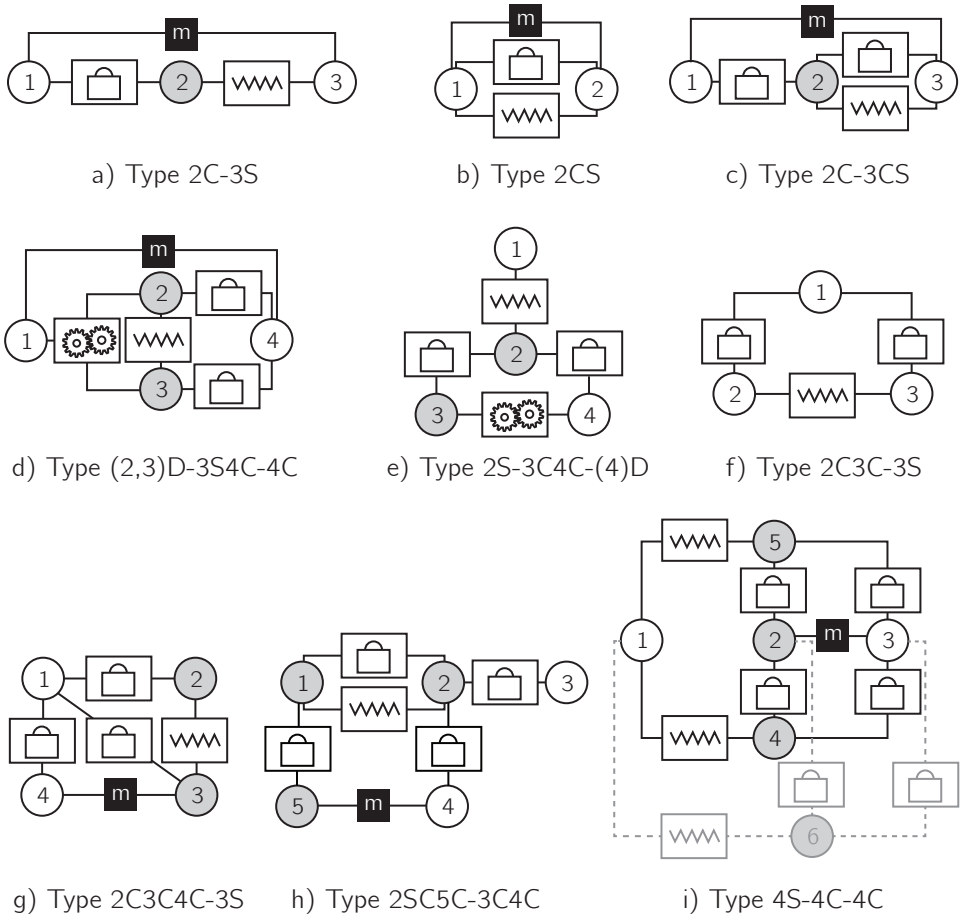


Figure 6.2: The nine different types of CEAs used in literature. The output bodies and the placement of the motor are visualized in the most used form in literature. The white bodies are the output bodies. In i), three modules of a series parallel elastic actuator are visualized. Two modules are shown in black and one module is shown in grey.

same amount of energy in every step of a walking robot [38] and to switch between a rigid and soft connection between a human and a backpack [184].

6.3.3 Type (2,3)D-3S4C-4C

The third type of CEA we discuss, has four bodies, one spring, two clutches and a differential. Body 1 and 4 are the output bodies (see Fig. 6.2d). As shown in Fig. 6.2d, bodies 1, 2 and 3 are connected with a differential mechanism. This differential causes the first body to move with the mean velocity of bodies 2 and 3. When only the top clutch is locked, body 2 connects to body 4 and the spring connects bodies 1 and 4 through body 3. When only the bottom clutch is locked, the spring connects the output bodies through body 2. This means that depending on the state of the clutches, the force on the output joints is positive or negative and therefore, energy in the spring can be released in clockwise or counterclockwise direction. Plooij et al. [170] used such a CEA to reduce the energy consumption of robotic arms with pick-and-place tasks by 65 %.

6.3.4 Type 2C3C-3S

The fourth type we discuss, consists of two clutches and one spring, and has three bodies (see Fig. 6.2f). The purpose of the mechanism is the same as that of type 2S-3C4C-(4)D and this type of CEA has also only been used as a passive CEA. Collins and Kuo [40] use this type of CEA in their energy recycling foot prosthesis. The lower leg connects to body 1, the heel connects to body 2 and the front of the foot connects to body 3. Here, the foot actually consists of two independently moving parts. This design makes it possible to store energy in the spring during heel strike and release it during push-off. This led to a reduction of the metabolic energy consumption by 8 % in comparison to a conventional prosthesis.

6.3.5 Type 4S-4C-4C

The last type of CEA is called the series parallel elastic actuator (SPEA) [131]. It consists of three main bodies (1, 2 and 3) and modules of which multiple can be connected between those bodies (see Fig. 6.2i). Those modules consist of a spring and two clutches. Note that for the name of this type, we only considered one module. Fig. 6.2i shows a system with two modules in black and one in grey. The joint is placed between bodies 1 and 3 and the motor is placed between bodies 2 and 3. By engaging and disengaging the clutches, the springs can be connected either in series or in parallel with the motor. When in series, the elongation of the springs can be controlled; when in parallel, the torque in the springs does not pass through the motor. Preliminary results show that this concept can reduce the energy consumption of robots by tens of percents in comparison to a stiff actuator.

6.3.6 Other designs

There are four other CEA designs that are depicted in Fig. 6.2. The CEA type in Fig. 6.2c is used to reduce the energy consumption in robots, specifically robotic arms [45, 171]. The CEA type in Fig. 6.2e is used to mimic normal human ankle torques using only one spring [226]. The CEA type in Fig. 6.2g is used to obtain an actuator that mimics some aspects of biological muscles [116]. And finally, the CEA type in Fig. 6.2f is used to constantly input small amounts of energy into the spring that are later released to the joint [66].

6.3.7 Trends

Two trends can be observed from literature. First, the use of CEAs is increasing. More than 60 % of the papers cited above, date from 2012 or later. Before 2012, the maximum amount of papers per year that mentioned the use of a CEA was 2 (in 2010). In the first half of 2015 alone, 5 papers were found in which CEAs were used. Secondly, the complexity of CEAs is increasing. Five out of seven types of CEAs with more than one clutch were introduced in 2010 or later, while CEAs with only one clutch were used more often before 2010. While less complex CEAs are easy to analyze by hand, the functionalities of more complex CEAs are less obvious. Therefore, there is a need for a method to analyze the functionalities of CEAs.

6.4 Functionalities

Each CEA has different functionalities and properties, as shown in the previous section. This makes them suitable for different applications. In this section we list all functionalities that can be obtained in CEAs. In the next section, we will describe CEAs mathematically to analyze which CEAs incorporate which functionalities. Functionalities can be described in terms of the resulting generalized force Q_y on the output joint. This force depends on its turn on the position of the joint y, the resulting stiffness k_y and the equilibrium position y_0:

$$Q_y = -k_y(y - y_0) \tag{6.1}$$

where Q_y and y are defined in the same direction. Note that here we assume one output joint, while CEAs could also be used to transfer energy from one joint to another. The analysis would not change for such a CEA, because the output joints can be analyzed one by one. Now there are three main modes of operation that could be obtained with CEAs:

1. **Moving freely:** In this mode, the joint can move while no springs are loaded or unloaded and thus $k_y = 0$. Depending on the specific design and state of the clutches, springs that were loaded before entering this mode remain loaded or will move to their equilibrium position.

2. **Spring connected:** In this mode, there is a fixed relationship between the position of the joint and the energy in the spring(s) and thus $k_y > 0$. This relationship can be characterized by the resulting spring stiffness and the equilibrium position. In some CEAs, the stiffness and equilibrium position can be set before entering this mode of operation.

3. **Output locked:** In this mode, the joint and the springs are locked, meaning that no motion is possible. This can also be seen as an infinite stiffness.

We will now discuss those modes of operation in more detail, including how they are obtained in current CEA designs.

6.4.1 Moving freely

The moving-freely mode is available if for a certain state of the clutches, k_y is zero. This mode is available in all CEA designs in literature except the type 2CS CEA. In principle, the moving freely mode is available in the CEA in Fig. 6.2i. However, Mathijssen et al. [131] integrated the functionality of the two clutches into one locking mechanism: mutilated gears. Therefore, the states of the clutches in their CEA are not independent and the moving-freely mode is not available. In principle, mode 1 can be achieved in every CEA design by the addition of one extra clutch. See for instance how the addition of one spring to a type 2CS CEA results in a type 2C-3CS CEA.

Mode 1 is used extensively in the knees of walking robots and exoskeletons [34, 53, 54, 60, 77, 99, 186, 197, 221]. The reason is that the torques in a human knee resemble a stiff spring during the stance phase and a weak spring during the swing phase. Therefore, during the swing phase, the stiff spring should be decoupled from the output joint. Another application of the moving freely mode is robotic arms with pick-and-place tasks [170]. In such tasks, there is a varying distance between the pick and the place position. Therefore, the spring characteristic should be variable. This can be done by adjusting the amount of distance the arm travels while being in the moving-freely mode [170].

Current CEA designs show two different aspects of the moving freely mode. First, the moving freely mode implies the spring does not apply a force on the output

bodies. However, depending on the design, the motor can or cannot apply a force on the output bodies while being in the moving-freely mode. In most CEAs, the motor is placed in parallel with the CEA [12, 57, 77, 99, 170, 171], meaning that the motor can still actuate the joint while the spring is disconnected. In other designs, the motor is placed in series with the CEA and thus when the CEA is in the moving freely mode, the motor also does not apply forces on the output bodies [129, 221, 222].

The second aspect of the moving freely mode is whether the spring remains loaded or not. For instance, in a type 2C-3S CEA, the energy in the spring will be lost when the clutch is unlocked while the spring is loaded. This can reduce the energy efficiency of the CEA significantly. In order to prevent this, multiple clutches are necessary. Current CEA designs in which the spring can be locked while the output can move freely are the CEA in Figs. 6.2c, g and h.

6.4.2 Spring connected

The spring-connected mode is always available, since CEAs are designed to be able to (un)load the springs. The different transfer ratios from the joint to the spring(s) determine k_y and y_0. CEAs differ in how many choices for k_y and y_0 there are and if the CEA can switch between those without entering mode 1 or 3. For instance, the CEA in Fig. 6.2d has two different modes 2. Whether the CEA can switch between two modes 2 without entering mode 1 or 3 depends mainly on the type of clutches that is used and the CEA design. Theoretically, all switches are possible. However, in the CEA in Fig. 6.2d, switching between two spring-connected modes is not possible because the clutches do not switch infinitely fast and perfectly timed. Therefore, the mechanism will either enter the output locked mode or the moving-freely mode. An example in which switching is possible is the CEA in Fig. 6.2i, because the output bodies are never locked with respect to each other since the connection between the two always passes through springs. Another issue with switching between different modes 2 is that springs that are loaded might become free to move and thus their energy is lost. This of course reduces the efficiency of the CEA and therefore, this should be avoided.

The trend towards CEAs with more clutches and more springs introduces the possibility of multiple modes 2. The simplest CEA designs in Fig. 6.2a-c have only one mode 2. In other CEAs, there are two different ways in which multiple modes 2 two are obtained. The CEA in Fig. 6.2d uses a differential to change the resulting equilibrium position y_0. This principle is used to reverse the resulting force Q_y on the output joint. This CEA was designed for robotic arms with pick-and-place tasks [170]. With this CEA, the energy in the spring can be released while accelerating in

any direction that is required by the task. A similar functionality is obtained by the CEA in Fig. 6.2e, in which the force is reversed using a gear ratio of -1.

The CEA design in Fig. 6.2i has a unique functionality because it has multiple springs that can be placed in series or in parallel by the clutches. In such a CEA, the force that the springs apply on the output joint is fully controllable.

6.4.3 Output locked

The output locked mode is available if for a certain state of the clutches, the output bodies cannot move with respect to each other. This mode is available in all current CEA designs except the CEAs in Figs. 6.2a and i. In principle, this functionality can be added to any CEA with one additional clutch. Note that switching to mode 3 can induce shocks in the system when the system has a non-zero velocity.

There are three situations in which a locked output joint is useful. First, when the output joint has to stand still, it might be beneficial to lock the joint in order to prevent the motor from having to counteract spring forces, gravitational forces or external forces. For this reason, researchers locked the output joint in robotic arms with repetitive tasks [13, 170]. Secondly, the joint can be locked while the spring is being loaded by the motor. Cherelle et al. [33] used a 2C-3S CEA in their ankle-foot prosthesis. They connected body 1 to the lower leg and body 2 to a lever arm that connects to the foot. Then, by placing a motor between body 1 and 3, they can load the spring while not applying torque on the ankle joint. Thirdly, locking the joint can ease the timing and precision of the release of the energy in the springs. Collins and Ruina [38] used a 2CS CEA in the ankle of their 3D walking robot. In the ankle, a spring connects the calf and the back of the foot. This spring is stretched by a small motor that lifts the front of the foot. During push-off, the energy in the spring is released at once. This set-up has the advantage that at every step, the same amount of energy is released.

There are also CEAs in which only the motor is locked. Rouse et al. [186] used a 2C-3S CEA for implementation in their knee prosthesis. Body 1 connects to the upper leg and body 3 to the lower leg. The motor is then placed between bodies 1 and 2, such that the rotation of the motor can be locked. Using this CEA, the motor does not have to deliver torque when the knee torque acts like a spring, but when needed, the system acts like a regular series elastic actuator.

6.5 Taxonomy description

For the selection of CEAs, it is important to be able to analyze the functionalities of different CEA designs. The goal of this section is to introduce a method for

finding all values for k_y and y_0 in Eq. (6.1) that are possible within a certain CEA design. Therefore, we now introduce a taxonomy that makes it easy to describe all possible CEAs and analyze their functionality. A CEA always consists of multiple bodies, spring(s), one or more clutches and possibly a number of constraints on the bodies. We will now discuss these components separately and give a mathematical description. Throughout the description, two examples will be explained to clarify the method: the type 2CS CEA in Fig. 6.2b and the type (2,3)D-3S4C-4C CEA in Fig. 6.2d.

6.5.1 Bodies

Bodies are movable components between which springs, clutches and differentials can be placed. Their position is denoted as x_i, where i is the number of the body. Most CEAs in literature have two external bodies and one or multiple internal bodies.

The external bodies are connected to the robot. For instance, they can be connected to the two sides of a joint, placing the CEA on that joint. An example of a bi-articular CEA can be found in [226]. CEAs with more than two external body are also possible and could be used to transfer energy between multiple joints [40]. The internal bodies are not directly connected to the robot. We call the vector with all body positions x:

$$x^T = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix} \tag{6.2}$$

where n is the number of bodies.

Example 2CS

This CEA has two bodies and thus $n = 2$.

Example (2,3)D-3S4C-4C

This CEA has four bodies and thus $n = 4$.

6.5.2 Differentials

Gears and differentials can both be described as linear holonomic constraints between bodies. These constraints all have the form:

$$c_j^T x = \begin{bmatrix} c_{j,1} & c_{j,2} & \dots & c_{j,n} \end{bmatrix} x = 0 \tag{6.3}$$

where the values for $c_{j,i}$ determine the constraint and j is used to enumerate the constraints. A set of m constraints can be written as:

$$C_D x = \begin{bmatrix} c_{1,1} & c_{1,2} & \dots & c_{1,n} \\ c_{2,1} & c_{2,2} & \dots & c_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m,1} & c_{m,2} & \dots & c_{m,n} \end{bmatrix} x = 0 \tag{6.4}$$

where 0 is a null vector in this case. The time derivative of this constraint also holds:

$$C_D \dot{x} = 0 \tag{6.5}$$

Example 2CS

This CEA does not include a gearbox or a differential and thus C_D is an empty matrix.

Example (2,3)D-3S4C-4C

The differential in this CEA is a standard differential and thus the constraint matrix C_D is equal to

$$C_D = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} & 0 \end{bmatrix}$$

6.5.3 Springs

The potential energy in a spring is a function of the positions of its two connected bodies, say x_i and x_j. Assuming a linear spring, the potential energy E in the spring is equal to

$$E = \frac{1}{2} k (x_i - x_j - x_0)^2 \tag{6.6}$$

where k is the spring stiffness and x_0 is the equilibrium position. If E is known as function of x, the stiffness matrix K and the pretension vector Q_0 can be calculated using:

$$K = \frac{\partial^2 E}{\partial x^2} \tag{6.7}$$

$$Q_0 = \frac{\partial E}{\partial x} - x^T K \tag{6.8}$$

So the energy in the springs can be re-written as function of x:

$$E = \frac{1}{2} x^T K x + Q_0^T x \tag{6.9}$$

Example 2CS

The spring is placed between bodies 1 and 2. Therefore, K and Q_0 are equal to

$$K = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix}$$
$$Q_0 = \begin{bmatrix} -kx_0 & kx_0 \end{bmatrix}$$

Example (2,3)D-3S4C-4C

The spring is placed between bodies 2 and 3. Therefore, K and Q_0 are equal to

$$K = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & k & -k & 0 \\ 0 & -k & k & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$
$$Q_0 = \begin{bmatrix} 0 & -kx_0 & kx_0 & 0 \end{bmatrix}$$

6.5.4 Clutches

A clutch switches between allowing and preventing relative motion between two bodies. We define the state of the i-th clutch as l_i, which has a value of 1 when the clutch prevents relative motion and a value of 0 otherwise. The number of clutches in the mechanism is p. Now define the matrix L as the diagonal matrix of $[l_1, l_2 \dots l_p]$ and L_I as a $p \times n$ oriented incidence matrix. Then the constraint imposed on the system by the clutches is

$$L_I L \dot{x} = C_L \dot{x} = 0 \tag{6.10}$$

Note that this is a constraint on the velocities and not on the positions, because the relative positions are set at the moment a clutch is locked. Therefore, we could also write this as a constraint on the positions with an offset. As an alternative, this offset can also be set in the pretension vector of the springs Q_0. Since all options lead to the same answers, we will consider the constraint on the positions and assume that Q_0 depends on the clutching positions.

Example 2CS

For this CEA, $n = 2$ and $p = 1$. The clutch is placed between bodies 1 and 2, meaning that L_I and L are equal to

$$L_I = \begin{bmatrix} 1 & -1 \end{bmatrix}$$
$$L = \begin{bmatrix} l_1 \end{bmatrix}$$

where l_1 and l_2 are equal to 0 or 1, depending on the state of the clutches.

Example (2,3)D-3S4C-4C

In the example in Fig. 6.2d, $n = 4$ and $p = 2$. The clutches are placed between bodies 2 and 4 and between bodies 3 and 4. This means that the matrix L_I and L are equal to

$$L_I = \begin{bmatrix} 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

$$L = \begin{bmatrix} l_1 & 0 \\ 0 & l_2 \end{bmatrix}$$

6.5.5 Determining the motion space

Based on Eqs. (6.5) and (6.10), we can find a basis B for the motion space of the mechanism:

$$Z = \begin{bmatrix} C_D \\ C_L \end{bmatrix} \tag{6.11}$$

$$B = \mathrm{null}\,(Z) \tag{6.12}$$

where the null operator returns a matrix with vectors that form a basis for the null space of a matrix. Since B is a basis for all possible motions of x, it can be used to define a minimal set of generalized coordinates q for these motions. From this vector, the positions of the individual bodies can be calculated:

$$x = Bq \tag{6.13}$$

Example 2CS

The clutch in the CEA in Fig. 6.2b has two states. When the clutch is locked, the mechanism can only move as a whole, meaning that q is a scalar and B is equal to:

$$B = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

When the clutch is not locked, q is a vector of length two and B is equal to:

$$B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Here, the two bodies can move independently and x is equal to q. Note that the number of columns of B is equal to the number of degrees of freedom of the mechanism plus one, because even when the mechanism is locked, the mechanism can still move as a whole. The rows of B give information on the possible motion of the bodies. If row i contains only zeros, then the i-th body is locked.

Example (2,3)D-3S4C-4C

For the possible states of the clutches in this CEA, the matrices B_{l_1,l_2}, containing vectors that form a basis for the null space, now become

$$B_{0,0} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \qquad B_{1,0} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 2 & -1 \\ 0 & 1 \end{bmatrix}$$

$$B_{0,1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 2 & -1 \\ 2 & -1 \end{bmatrix}; \qquad B_{1,1} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Those four B-matrices show the motion space for the four different states of the clutches. When all clutches are locked, the only motion that is allowed is when all bodies rotate in the same direction with the same velocity. When only one clutch is locked, the motions of bodies 2 and 3 are a function of the motions of bodies 1 and 4. And when none of the bodies is locked, the mechanism can move freely.

6.5.6 Determining the k_y and y_0

Now, we are interested in the force-displacement relationship between two of the bodies. Typically, we are interested in the bodies that are connected to the two sides of a joint or in the bodies between which the motor acts. We call the bodies we are interested in the i-th and the j-th bodies. Then the output displacement y is the difference between the two bodies:

$$y = x_i - x_j = d^T x \tag{6.14}$$

where d is a vector containing two unity entries, of which one is negative. In order to determine the relationship in Eq. (6.1), we have to find x as a function of y. Unfortunately, Eq. (6.14) cannot be solved for x, because d is not invertible. Simplifying the problem to a static analysis, we can find the motion of the whole mechanism by minimizing the energy in the springs, while still satisfying the constraints in Eqs. (6.5) and (6.10). This can be solved using Lagrange multipliers, leading to the minimization problem:

$$\min_{x,\lambda_1,\lambda_2} \quad \mathcal{L}(x, \lambda_1, \lambda_2) \tag{6.15}$$

where λ_1 and λ_2 are Lagrange multipliers and the Lagrangian $\mathcal{L}$ is equal to

$$\mathcal{L} = \frac{1}{2} x^T K x + Q_0^T x + \lambda_1^T Z x + \lambda_2^T (d^T x - y) \tag{6.16}$$

This can be solved by setting the partial derivatives of $\mathcal{L}$ to x, λ_1 and λ_2 to zero:

$$\frac{\partial \mathcal{L}}{\partial x} = Kx + Q_0 + Z^T\lambda_1 + d\lambda_2 = 0 \tag{6.17}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_1} = Zx = 0 \tag{6.18}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_2} = d^T x - y = 0 \tag{6.19}$$

In matrix form this becomes

$$\begin{bmatrix} K & Z^T & d \\ Z & 0 & 0 \\ d^T & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} -Q_0 \\ 0 \\ y \end{bmatrix} \tag{6.20}$$

Here λ_2 is equal to Q_y since it is the force that ensures that Eq. (6.14) holds. Now, often the matrix in Eq. (6.20) is singular, because there are multiple solutions for x that satisfy the constraints and for which the energy in the springs is minimal. An example is a mechanism that can rotate as a whole in absolute sense, while the relative motion is constrained. This singularity means that the matrix in Eq. (6.20) is not invertible and that the solution is the sum of the pseudo inverse times a vector plus any vector that is in the null-space of the matrix. In general, the null-space of the matrix in Eq. (6.20) contains all motions that do not influence the length of any spring. Therefore, this space is not interesting for the calculation of k_y and y_0 and we can take the pseudo inverse:

$$\begin{bmatrix} x \\ \lambda_1 \\ Q_y \end{bmatrix} = \begin{bmatrix} K & Z^T & d^T \\ Z & 0 & 0 \\ d & 0 & 0 \end{bmatrix}^\dagger \begin{bmatrix} -Q_0 \\ 0 \\ y \end{bmatrix} \tag{6.21}$$

$$= \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} -Q_0 \\ 0 \\ y \end{bmatrix} \tag{6.22}$$

From this equation, it follows that the parameters in Eq. (6.1) are equal to

$$k_y = -A_{33} \tag{6.23}$$

$$y_0 = \frac{A_{31}Q_0}{A_{33}} \tag{6.24}$$

Note that the pseudo inverse also exists when the i-th and j-th bodies are locked with respect to each other. Therefore, it should be checked if Eq. (6.20) holds for the found x, λ_1 and λ_2. If it does not hold, then the bodies are locked.

Example 2CS

The output bodies of this CEA are bodies 1 and 2, meaning that d is equal to

$$d^T = \begin{bmatrix} 1 & -1 \end{bmatrix}$$

For the two possible states of the clutches, this gives the following outcomes:

- $\begin{bmatrix} l_1 \end{bmatrix} = [0]$: the spring is (un)loaded when moving the joint, meaning that this is a mode 2. $k_y = k$, $y_0 = Q_0 k$.
- $\begin{bmatrix} l_1 \end{bmatrix} = [1]$: all bodies are locked, meaning that it is a mode 3.

Example (2,3)D-3S4C-4C

The output bodies of this CEA are bodies 1 and 4, meaning that d is equal to

$$d^T = \begin{bmatrix} 1 & 0 & 0 & -1 \end{bmatrix}$$

For the four possible states of the clutches, this gives the following outcomes:

- $\begin{bmatrix} l_1 & l_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix}$: the output can move freely, meaning that this is a mode 1. $k_y = 0$, $y_0 = 0$.
- $\begin{bmatrix} l_1 & l_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \end{bmatrix}$: the spring is (un)loaded when moving the joint, meaning that this is a mode 2. $k_y = 4k$, $y_0 = -2Q_0 k$.
- $\begin{bmatrix} l_1 & l_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \end{bmatrix}$: the spring is (un)loaded when moving the joint, meaning that this is a mode 2. $k_y = 4k$, $y_0 = 2Q_0 k$.
- $\begin{bmatrix} l_1 & l_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \end{bmatrix}$: all bodies are locked, meaning that it is a mode 3.

Note that when one of the clutches is engaged, the stiffness is always equal to $4k$. On the other hand, the force between bodies 1 and 4 can differ depending on the clutch that is locked, because the equilibrium position y_0 is different for the two modes 2.

6.5.7 Switching between modes

As mentioned in section 6.4.2, switching between modes can lead to energy loss, because loaded spring might become disconnected from the output bodies without being locked by clutches. This energy loss is undesirable because it decreases the energy efficiency of the CEA. Therefore, before performing a switch, the energy in the springs before and after the switch can be calculated. Using Eq. (6.21), the position vector x can be calculated before and after the switch. Then, the energy in the system can be calculated using Eq. (6.9).

6.6 The future of CEA design

We envision future CEAs consisting of many springs and many clutches. Section 6.3 showed that there already is a trend towards such CEAs and Section 6.5 provided a taxonomy that eases the analysis of current and future CEA designs. This section focuses on the design of future CEAs. First, we discuss the synthesis of new CEAs. Secondly, we discuss the CEAs for stiffness control and force control. Thirdly, we propose a new CEA design in which the number of reachable combinations of k_y and y_0 grows exponentially with the number of clutches and springs.

6.6.1 Choosing a suitable CEA

The question now is how to select a suitable CEA for a certain application. We suggest three approaches that span the space of possible approaches from manual to automated design: optimization, heuristics and using building blocks.

In an optimization approach, the configuration of the clutches, springs and differentials are included in an optimization algorithm. This algorithm can either have the goal to approach a certain functionality or can be part of a co-optimization that optimizes both design and control. It should be noted that the amount of possible mechanisms grows fast with the amount of used components. Therefore, the search space might have to be limited or an optimization algorithm should be used that can handle large and odd-shaped spaces.

In a heuristic approach, the designer determines the functionality that he or she thinks is desired and tries to find a CEA that matches that functionality. The limitations of this approach are the skills, the knowledge and the time of the designer.

There is a third approach that helps the designer to find a suitable CEA: using building blocks. Here we mention four building blocks that are particularly interesting. First, a type 2C-3CS CEA is a building block that can be placed between two bodies and can be used to arbitrarily connect those bodies, decouple them, or place a stiffness between the two. Secondly, a type (2,3)D-3S4C-4C CEA is a building block that can be used to reverse the torque between two joints. Thirdly, multiple type 2CS CEAs in series are a building block that can switch between being disconnected, being locked or having multiple different stiffnesses. And finally, a module of a type 4S-4C-4C CEA is a building block to switch a spring between being in series or in parallel with the motor.

6.6.2 Force control and stiffness control

The two main reasons for using CEAs is to control the force and to control the stiffness. These two reasons lead to different CEA designs. For controlling the

force Q_y, there are two techniques that can be used. First, when considering only one spring, the force on the output joint changes when changing the transfer ratio between the spring and the output joint. This influences both k_y and y_0. This technique is exploited in two existing CEA designs: the ones in Figs. 6.2d and 6.2e. Secondly, when using multiple springs, the springs can be placed in parallel, such that they connect and disconnect from the joint. An example would be to place multiple type 2C-3CS CEAs in parallel. Another example of such a CEA is the type 4S-4C-4C CEA in Fig. 6.2i. In such a CEA, the force is even fully controllable, because the motor can change the equilibrium positions of the springs.

For controlling the stiffness, the same techniques can be used as for controlling the force, since those techniques also change k_y. However, there is a third technique to change the stiffness that does not involve changing the force: placing multiple type 2CS CEAs in series (see Fig. 6.3a). The challenge in such a design is to keep the energy efficiency high, as discussed in section 6.5.7. The general rule for keeping the efficiency high is to only unlock springs in which the force is similar to the force in the springs that are already unlocked. A new CEA design based on this principle is introduced in the next section.

6.6.3 A newly proposed CEA design

Based on the building clock method, we propose a new CEA design in which the number of combinations of k_y and y_0 grows exponentially with the number of clutches and springs. It is based on the CEA in Fig. 6.3a, in which the number of resulting stiffnesses is equal to $2^p - 1$, where p is the number of clutches. This CEA also has a locked mode. However, it has no moving-freely mode and the Q_y cannot be reversed arbitrarily.

To increase the functionality of the CEA in Fig. 6.3a further, we propose to use this CEA as a building block again and combine it with the type (2,3)D-3S4C-4C CEA in Fig. 6.2d. This results in the CEA in Fig. 6.3b, where bodies 1, 2 and 3 connect through a differential. This CEA includes a moving-freely mode and a locked mode. Furthermore, given the states of the clutches $3 \ldots p$, Q_y can be positive or negative depending on the states of clutches 1 and 2. The number of resulting stiffnesses is $2^{p-2} - 1$. Since y_0 can be set by changing the states of clutches 1 and 2, the number of possible combinations of k_y and y_0 is $2^{p-1} - 1$. Note that in order to reach this exponential growth in resulting stiffnesses, the stiffnesses of the individual springs should all be different.

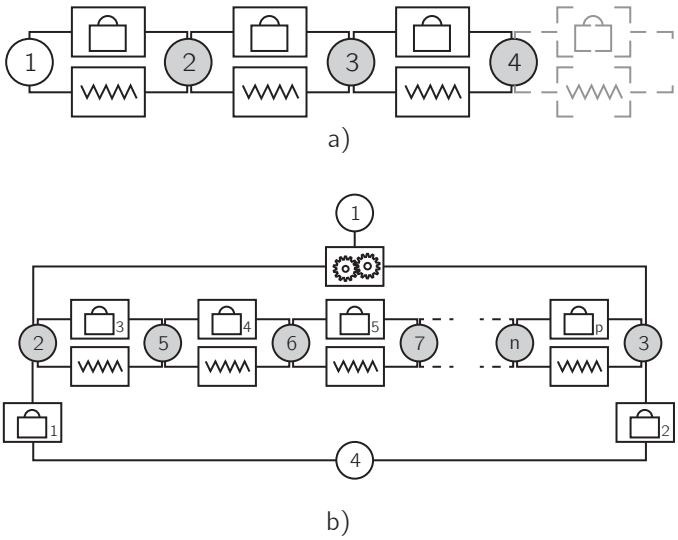


Figure 6.3: A visualization of two CEAs consisting of building blocks. a) A CEA where a type 2CS CEA is placed in series multiple times. b) A CEA in which the CEA as visualized in a) is placed inside a differential mechanism, allowing the force to be reversed.

6.7 Discussion

In this chapter we analyzed the class of mechanisms called clutched elastic actuators (CEAs). Although CEAs existed for decades, most studies that use CEAs date from after 2010. We provided a unified mathematical description that can be used to easily analyze the functionality of different CEAs. In this section, we discuss the current use of CEAs, series and parallel elasticity, the components that are used and the use and limitations of the introduced taxonomy.

6.7.1 Current use of CEAs

The nine types of existing CEAs show that very different CEA designs have been implemented in robots. However, their functionalities are still limited. This follows from the fact that only one type of CEA has more than one spring and only a few have four or more clutches. As the reason for using multiple springs and many clutches, Mathijssen et al. mention that they 'believe that the way transmissions and springs are used, needs drastic innovation' [131]. In this chapter we showed that with more complex CEAs, the possible combinations of k_y and y_0 can grow exponentially with the number of springs and clutches, meaning that the energy inflow and outflow of the springs can be controlled better.

Most CEAs are implemented to reduce the energy consumption, peak torque or peak power. The underlying principle to reduce the energy consumption is that springs are more efficient at storing and releasing energy than motors. By recapturing negative energy and releasing it at a later instance in time, the motor consumes less energy. Typically, reductions in the energy consumption between 20 % and 80 % are reported. Since the energy efficiency of springs themselves is typically 90 %-95 %, we expect that reported reductions in the energy consumption will be even higher in the future. The challenge will be to reduce the energy consumption while keeping the versatility of the robot at an acceptable level. With the same line of reasoning, it can be shown that CEAs lead to a reduced peak torque and peak power.

6.7.2 Series elastic and parallel elastic

The taxonomy of CEAs includes both clutched parallel elastic actuators and clutched series elastic actuators. In some designs, the spring(s) can even be switched between a series and a parallel configuration. However, in most CEAs, the springs are placed in parallel to a motor. This matches with the reasons for designers to use CEAs: energy efficiency, peak torque and peak power. The peak torque is not reduced by SEAs, because the torque on the joint also passes through the motor. For the same reason, the reduction in energy consumption is limited using SEAs, because the copper losses scale quadratically with the torque. The peak power can be reduced in some applications using SEAs, because the spring can be used as an energy buffer, separating high torques and high velocities in time. Therefore, we expect the best results from CEAs that are in parallel with the motor or from hybrid systems, such as the series parallel elastic actuator, which is a type 4S-4C-4C CEA.

6.7.3 Components

The components of CEAs we considered in this study are springs, clutches and differentials. Springs are well studied components with high energy efficiencies and are commercially available. One way in which springs might be improved, is the use of materials that can store much energy per volume or per mass. The extent to which differentials are studied varies per amount of bodies that are involved. A two-body differential is the same as a gear, which is a well known component. Three-body differentials are also studied widely, since they are used in cars and planetary gearboxes. However, when they are to be widely applied in robots, they will need more development, especially in terms of compactness. Furthermore, four-and-more-body differentials are hardly studied at all. Clutches are still subject to many recent studies. The main challenge is to make clutches with a high torque density that consume little

energy and can lock at every position. Recent clutch designs include piezoelectric clutches [114], statically balanced brakes [168] and electro static clutches [48].

Two components that are related to CEAs were omitted from this study: continuously variable transmissions (CVTs) and dampers. A CVT can be seen as a gear ratio that can be varied by a controller. Such a component could be interesting in combination with clutches and springs because it would allow for a more continuous variation of the stiffness instead of a discrete variation. A similar idea was proposed by Wang and Zhu [234] who called their device a geared infinitely variable transmission. Current limitations to CVTs include their compactness, efficiency and reliability. Dampers were omitted from this study because they are dissipative elements that are not ideal from an energy perspective. However, they might be useful to improve the control bandwidth of robots [70, 157]. Therefore, they might be implemented in CEAs in the future. Such actuators might then be called clutched impedance actuators (CIA).

6.7.4 Limitations of the introduced taxonomy

The taxonomy that was introduced in this chapter can be used to analyze the different values for k_y and y_0 that can be reached given a certain morphology. In the taxonomy, we used ideal clutches to categorize different CEA designs. In practice, the choice for a certain type of clutch has an impact on the working of the CEA. Section 6.4.2 already discussed that with clutches that do not switch infinitely fast and perfectly timed, the functionality decreases.

In general, switches between modes can only be performed when it only requires one clutch to be switched. However, this general rule might not hold when passive clutches are used. An example of a passive clutch is a one way clutch that locks the motion in one direction and allows motion in the other direction. Such clutches react on a load and therefore react on the state of the other clutches. Therefore, future work should address the question how the choice for a certain type of clutch influences the functionality.

6.8 Conclusion

This chapter presented an overview of current CEA designs and provided a taxonomy for analyzing functionalities of current and future CEAs. We conclude that functionality can be analyzed using a constraint matrix, a stiffness matrix and multiplication of a clutch dependent diagonal matrix with a oriented incidence matrix. With these matrices, it is possible to calculate the resulting stiffnesses and equilibrium positions and thus the resulting force on the output bodies. Current CEA designs are split into

nine types, based on the placement of the spring(s), clutch(es) and differential(s). Although CEA designs are becoming more complex, their full potential will only be reached with multiple clutches, springs and differentials. Based on the introduced taxonomy, we proposed a new CEA design, showing that in CEAs, the number of resulting stiffnesses and equilibrium positions can grow exponentially with the number of clutches and springs.

Acknowledgement

This work is part of the research programme STW, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO). The authors thank Steve Collins, Heike Vallery and Bram Vanderborght for their useful comments on the manuscript of this paper.

Part II

Feedforward control in robots

7

Robust feedforward control of robotic arms with friction model uncertainty

Michiel Plooij, Wouter Wolfslag and Martijn Wisse,
Robotics and Autonomous Systems
Vol. 70 (August 2015), pp. 83-91.

Abstract

To design feedforward controllers for robots, a model that includes friction is important. However, friction is hard to identify, which causes uncertainty in the model. In this chapter we consider rest-to-rest motions of robotic arms that use only feedforward control. We show that it is possible to design feedforward controllers such that the final position of the motion is robust to uncertainty in the friction model. We studied a one DOF robotic arm in the horizontal plane, of which we show analytical, simulation and hardware results and we also show simulation results of a planar two DOF arm. Our friction model includes three types of friction: viscous, Coulomb and torque dependent friction. The results show that it is possible to eliminate the sensitivity of the final state to uncertainty in the three types of friction.

7.1 Introduction

We are fascinated by the question what is still possible without feedback on robotic arms. Control without feedback is called feedforward control or open loop control. Although intuition tells us that accuracy in the presence of model uncertainty and disturbances using only feedforward control is hopeless, humans use feedforward control for fast motions and are still able to perform their tasks accurately [47]. In this chapter, we present the surprising result that the final state of rest-to-rest motions of robotic arms can be made insensitive to uncertainty in the friction model. Possible fields of application include environments with heavy radiation, such as nuclear disaster areas or space, where feedback might be difficult and applications in which a large amount of agents are controlled with one input signal [19].

Generally, feedforward control laws use models of the system to compute control signals. These models often include friction, which is hard to model accurately despite the amount of literature on friction identification [21, 23, 46, 52, 152]. Such inaccurate modeling introduces model uncertainty. Usually, the controller relies on feedback to compensate for uncertainty in the model. We are interested if it is also possible to incorporate robustness of the final position of motions to model uncertainty in the feedforward controller of robotic arms. As case studies, we use robotic arms with one and two DOF in the horizontal plane that are controlled with only a feedforward controller.

Multiple researchers share our fascination for only feedforward control and have shown positive results. Firstly, Schaal and Atkeson showed that it is possible to

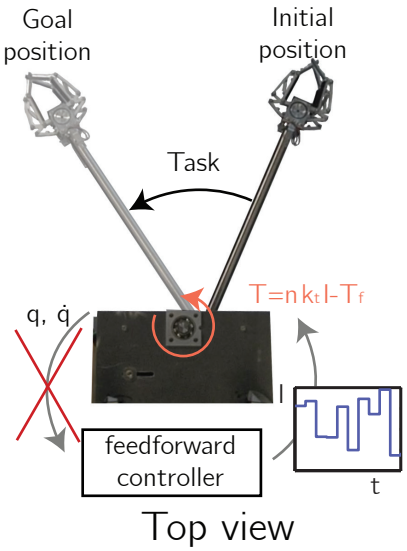


Figure 7.1: A schematic representation of the subject of this chapter. The one DOF robotic arm in this picture has to perform a rest-to-rest motion: move from the initial to the goal position. The controller is a feedforward controller, which means that the state (i.e. position q and velocity $\dot{q}$) is not used to determine the control signal. The control signal is a current I, which is only a function of time. The resultant torque on the arm is the motor torque minus the friction torque. In this chapter, we investigate the sensitivity of feedforward motions to uncertainty in the friction parameters. These model uncertainties usually cause the arm to end up in a different state than the goal state. We aim to eliminate this error in the final state. (figure from [167])

perform robot juggling with an open loop controller [189]. Seyfarth et al. showed a similar example in which feedforward control schemes for the swing leg retraction improved the stability of running in a humanoid robot [196]. And finally, Mombaur et al. showed that even stable walking and running are possible by creating open loop stable periodic motions [138, 139]. Although interesting, these researches did not account for model uncertainty.

Also robustness of feedforward controllers to model uncertainty has been studied before. Firstly, Singhose, Seering and Singer showed that input shaping to reduce vibrations can be performed in open loop while being robust to uncertainty in the natural frequency and damping of the system [203, 204]. Secondly, Akella and Mason showed that the result of several pushing actions in planar manipulation can be made robust to uncertainty in the initial position of the object [2]. Thirdly, Becker and Bretl showed that using open loop control on differential drive robots, the final

position of a motion can be made robust to uncertainty in the wheel diameter [18]. They also showed that this works for balls with an uncertain diameter that roll over a moving plate [17]. All previous examples of feedforward control with robustness to model uncertainty concern systems with first order dynamics. Recently we showed the potential of using feedforward control on robotic arms with an inaccurate model, which is a second order system [167]. There is one technique all these examples use: a certain *task redundancy*[1] is exploited in order to make the controller robust. We are continuing on the last example, by exploiting task redundancy in rest-to-rest motions of robotic arms.

Exploiting the task redundancy to compensate for model uncertainty has also been observed in humans. The human nervous system introduces large time delays of typically 150 ms [44, 216] and therefore humans must partially rely on feedforward control [47]. Since the internal model of humans is inaccurate [101] they exploit task redundancy to minimize the error due to model uncertainties. Error-minimizing feedforward signals have been reported in eye movements [82], but also in the games of darts and skittles [37, 146, 147, 206].

The task we consider in this chapter is a pick-and-place task, which also possesses redundancy (see Fig. 7.1). The task consists of rest-to-rest motions where only the initial and final position matter and the path in between can be chosen freely. We demonstrate that by choosing the right feedforward controller, it is possible to essentially eliminate the sensitivity of the final state to uncertainty in the friction model.

We extend our research in [167] in three ways. Firstly, we start with an analytical approach to gain more understanding of the principles behind feedforward control of robotic arms with friction model uncertainty. Secondly, we show improved results on a one DOF simulation model with multiple uncertainties at the same time. And thirdly, we show simulation results on a two DOF simulation model. For completeness, we also show the hardware results from [167].

The rest of this chapter is structured as follows. Section 7.2 provides the problem formulation of the problem that is considered in this chapter. Sections 7.3, 7.4 and 7.5 show analytical, numerical and hardware studies respectively, on one and two DOF systems. The chapter ends with a discussion in section 7.6 and a conclusion in section 7.7.

[1] Task redundancy means that there are multiple ways to fulfill the task perfectly.

7.2 Problem formulation

This section describes the problem of feedforward control with friction model uncertainty. First, the problem is formulated in general terms, then two case studies are introduced that will be studied in this chapter and finally, the task description of the arms will be discussed.

7.2.1 General problem

We consider mechanical systems of the form

$$\dot{x}(t) = f(x(t), u(t), p) \tag{7.1}$$

where x is the state containing the positions q and the velocities $\dot{q}$, u is the input and p are the friction parameters. The motions we consider have an initial state x_0 and a final state x_f, leading to the following initial condition and motion constraint:

$$x(0) = x_0 = \begin{bmatrix} q_0 \\ \dot{q}_0 \end{bmatrix} \tag{7.2}$$

$$x(t_f) = x_f = \begin{bmatrix} q_f \\ \dot{q}_f \end{bmatrix} \tag{7.3}$$

where t_f is the time to move. We now define y as the state at t_f, which is a function of x_0, u and p:

$$y(x_0, u(t), p) = x(t_f) = x(0) + \int_0^{t_f} f(x(\tau), u(\tau), p) d\tau \tag{7.4}$$

The goal is to make the final state of the arm insensitive to the friction parameters. We define sensitivity as:

$$S_{ij} = \left| \frac{\partial y_j(x_0, u(t), p)}{\partial p_i} \right| \tag{7.5}$$

$$S(u(t)) = \sum_{i=1}^{i=k} \sum_{j=1}^{j=n} c_{ij} S_{ij} \tag{7.6}$$

where S_{ij} is the sensitivity of final state j to parameter i, S is the total sensitivity and c_{ij} is the weighing factor corresponding to parameter p_i and the jth state.[2] In this chapter we weigh all sensitivities equally, meaning that c_{ij}=1 for all i and j. The sensitivity S_{ij} is equal to zero when the final state j is insensitive to the friction parameter p_i. Note that this definition differs from the classical definition of sensitivity in control [61][3]. This goal can be transformed into an optimization as follows:

$$\underset{u(t)}{\text{minimize}} \quad S(u(t)) \tag{7.7}$$

$$\begin{aligned} \text{subject to} \quad & |u(t)| \leq u_{max} \qquad \forall t \\ & q(t_f) = q_f \\ & \dot{q}(t_f) = \dot{q}_f \end{aligned} \tag{7.8}$$

where k is the number of friction parameters, n is the number of states, and u_{max} is the maximum input.

7.2.2 Case studies

There are two case studies we consider in this chapter: a one DOF arm and a two DOF arm, both in the horizontal plane (see Fig. 7.2a and 7.2b). We also show results from hardware experiments on a one DOF arm to confirm the results from simulation (see Fig. 7.2c).

The equations of motion of the models will be given in the sections discussing the specific models (sections 7.3.1, 7.3.2, 7.4.1 and 7.4.2). In all models, the actuation torque applied by the DC motors on the joints is equal to:

$$T = nk_t I \tag{7.9}$$

Where T is the actuation torque k_t is the motor constant, n is the gearbox ratio and I is the current through the motor. Since we use motor controllers with current control, the current is used as the input u.

Our frictional model consists of three commonly used types of friction: viscous friction, Coulomb friction and torque dependent friction:

[2] The original publication defines k and n in the subsequent paragraph: k is the number of friction parameters and n is the number of states.

[3] We decided to still use the term *sensitivity* here, because it describes best the meaning of this measure

Figure 7.2: The three configurations we studied: (a) Simulation models of a one DOF arm in the horizontal plane. (b) A simulation model of a two DOF system in the horizontal plane. (c) Hardware experiments on a one DOF robotic arm to show that the principles found in the simulation studies work on a hardware setup.

$$T_p = \left\{ \begin{array}{ll} p_v \dot{q} + \mathrm{sign}(\dot{q})(p_c + p_t|T|) & \text{for } \dot{q} \neq 0 \\ \min(p_c + p_t|T|; |T|)\mathrm{sign}(T) & \text{for } \dot{q} = 0 \end{array} \right\} \tag{7.10}$$

where T_p is the frictional torque, p_v is the viscous friction coefficient, p_c is the Coulomb friction constant and p_t is the torque dependent friction coefficient. Torque dependent friction is less commonly used than the other two. The way we use it is similar to the force dependent friction term in [52]. We included this type of friction because model identification showed the presence of torque dependent friction.

7.2.3 Task description

We consider rest-to-rest motions, which are very common motions for robotic arms. For instance, in industry pick-and-place tasks depend on accurate rest-to-rest motions. In practice, a robotic arm performing rest-to-rest motions has to move between a number of positions. In the analytical studies, we will keep the task description generic. In the numerical results, we will show specific results where the one DOF arm moves from 0 rad to 1 rad and the two DOF arm moves from $[-0.5, 0]$ rad to $[0.5, 0]$ rad, all with a time to move of $t_f = 1$s. In the one DOF case, we will analyse how the results change when changing the task. This will show that other initial and goal positions lead to identical results as long as the distance between the two does not exceed a certain threshold.

7.3 Analytical Studies

In this section, we study the problem analytically, in order to understand the principles behind feedforward control of robotic arms with friction model uncertainty. For this purpose, we study two one-DOF models: a model with only viscous friction and a model with only Coulomb friction. The first model is a negative example, showing that for this model uncertainty in the viscous friction cannot be compensated for. The second model is a positive example, showing that uncertainty in the Coulomb friction can be compensated for.

7.3.1 With only viscous friction

In the one DOF model with only viscous friction, the equations of motion are:

$$\frac{d}{dt}\begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ \frac{1}{J_{joint}}(k_t n I - p_v \dot{q}) \end{bmatrix} \tag{7.11}$$

where J_{joint} is the inertia about the joint. If we now integrate the second row of this equation with respect to time we get

$$J_{joint}\ddot{q} = k_t n I(t) - p_v \dot{q} \tag{7.12}$$

$$J_{joint}\Delta\dot{q} = \int k_t n I(t) dt - p_v \Delta q = 0 \tag{7.13}$$

where Δq is the change in angular position and $\Delta\dot{q}$ is the change in angular velocity. $\Delta\dot{q}$ is equal to zero since we start and end with zero velocity. Here, $I(t)$ can be chosen freely, as long as the motion satisfies the constraints in (7.8). Suppose that $I^*(t)$ is a feedforward control signal that satisfies the constraints and add $\delta(t)$ to the signal, which represents all possible changes in the feedforward control signal:

$$\Delta q = \frac{k_t n}{p_v} \int (I^*(t) + \delta(t)) dt \tag{7.14}$$

$$= \frac{k_t n}{p_v} \left(\int I^*(t) dt + \int \delta(t) dt \right) \tag{7.15}$$

Since the final position of the motion should not change in order to satisfy (7.8), all possible changes $\delta(t)$ of the feedforward control signal are constrained by the following equation:

$$\int \delta(t)dt = 0 \tag{7.16}$$

Combining (7.15) and (7.16), leads to

$$\Delta q = \frac{k_t n}{p_v} \int I^*(t)dt \tag{7.17}$$

From (7.17), we see that $\frac{\partial \Delta q}{\partial p_v}$ is independent of changes of the feedforward control signal that reach the same end state for the nominal value of p_v. This shows that all the changes in the feedforward control signal that satisfy the constraints, do not influence the sensitivity of the final position of the arm to the viscous friction.

Apparently, on the system we are considering it is impossible to achieve the goal of this chapter: sensitivity of the feedforward motion to uncertainty in the friction model is independent of the chosen feedforward controller.

7.3.2 With only Coulomb friction

So now consider a one DOF system with only Coulomb friction. In this system, the equations of motion are:

$$\frac{d}{dt} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ \frac{1}{J_{joint}}(I(t)k_t - p_c \mathrm{sign}(\dot{q})) \end{bmatrix} \tag{7.18}$$

The changes in velocity and position are equal to:

$$\Delta \dot{q} = \int \frac{I(t)k_t n}{J_{joint}} \, \mathrm{d}t - p_c \int \frac{\mathrm{sign}(\dot{q})}{J_{joint}} \, \mathrm{d}t \tag{7.19}$$

$$\Delta q = \iint \frac{I(t)k_t n}{J_{joint}} \, \mathrm{d}t^2 - p_c \iint \frac{\mathrm{sign}(\dot{q})}{J_{joint}} \, \mathrm{d}t^2 \tag{7.20}$$

In (7.19) and (7.20), we see that there are three components that influence the final position of the arm: the current through the motor, the sign of the velocity and the amplitude of the Coulomb friction p_c. Note that the Coulomb friction is a bang-bang torque. A change in the Coulomb friction constant has two effects:

1. The amplitude of the bang-bang Coulomb friction torque scales

Table 7.1: The model parameters of the arm with only Coulomb friction. All inertial terms are combined in the inertia about the joint (J_{joint}).

Parameter	Symbol	Value
Viscous friction	p_v	0 Nms/rad
Coulomb friction	p_c	0.19 Nm
Torque dependent friction	p_t	0 %
Inertia	J_{joint}	0.17 kgm^2
Motor constant	k_t	26.7 mNm/A
Gearbox ratio	n	1:54

2. The switch times of the $\mathrm{sign}(\dot{q})$ function change.

If we assume the second effect is negligible, then the total effect of changing the Coulomb friction is a scaling of the frictional term in (7.19) and (7.20). If we now choose our feedforward control signal such that:

$$\int \mathrm{sign}(\dot{q})\,\mathrm{d}t = 0 \tag{7.21}$$

and

$$\iint \mathrm{sign}(\dot{q})\,\mathrm{d}t^2 = 0 \tag{7.22}$$

then a change in the amplitude of the Coulomb friction torque has no effect on the final position and velocity.

The assumption above is not valid, as we will now show with an example, but does lead to small values of the partial derivatives in (7.5). To show this, we used a simulation of a robotic arm of which the parameters are given in Table 7.1. Fig. 7.3 shows a motion that satisfies Eqs. (7.21) and (7.22). The motion has three phases: first, it starts with a negative velocity, second, it continues with a positive velocity and finally, it ends with a negative velocity again. Figs. 7.3e and 7.3f show that the derivatives of the final position and velocity to the Coulomb friction are approximately zero (0.009 (Nms)$^{-1}$ and 0.012 (Nm)$^{-1}$). The derivatives are not exactly equal to zero because of the change in switch times.

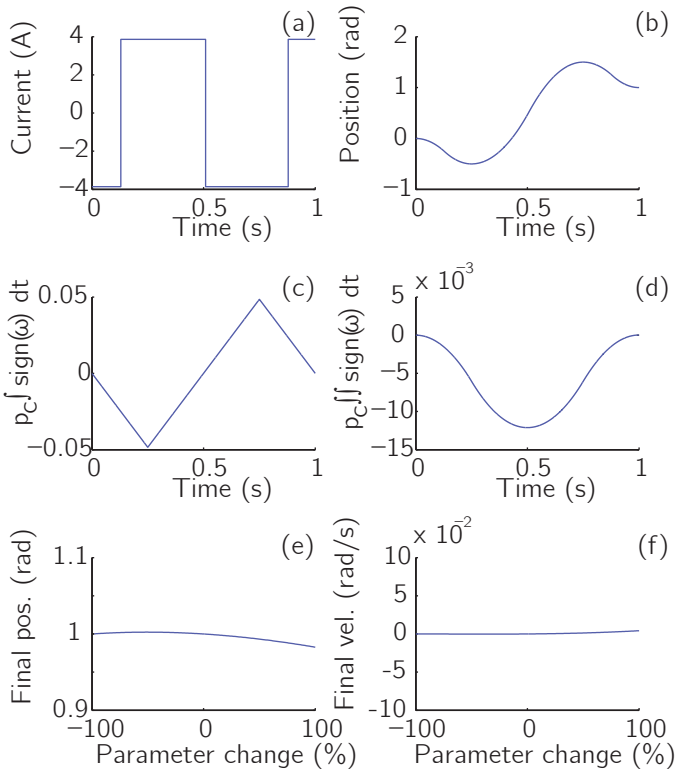


Figure 7.3: A simple example of a motion that satisfies the requirements to eliminate the effect of an uncertain Coulomb friction on the final position and velocity. Both the single and the double integral of the Coulomb friction over the time are approximately equal to zero. They are not exactly equal to zero, because the amplitude of the Coulomb friction also influences the zero crossing of the angular velocity. (a) The current through the motor as function of time. (b) The position of the arm as function of time using the nominal friction values. (c) The integral of the Coulomb friction over time. (d) The double integral of the Coulomb friction over time. (e) the final position of the arm as function of the Coulomb friction parameter change. (f) the final velocity of the arm as function of the Coulomb friction parameter change.

7.4 Numerical Studies

In the previous section, we showed analytical results for a one DOF system with only viscous friction or with only Coulomb friction. However, when the complete friction model is considered (i.e. viscous, Coulomb and torque dependent friction) or when the dynamics are non-linear, obtaining analytical results becomes infeasible. Therefore, in this section, we perform numerical studies with a complete friction model (viscous, Coulomb and torque dependent friction). Although we showed in the

Table 7.2: The model parameters of the arm with the complete friction model, based on a system identification of our one DOF arm. All inertial terms are combined in the inertia about the joint (J_{joint}).

Parameter	Symbol	Value
Viscous friction	p_v	-0.05 Nms/rad
Coulomb friction	p_c	0.19 Nm
Torque dependent friction	p_t	22 %
Inertia	J_{joint}	0.17 kgm^2
Motor constant	k_t	26.7 mNm/A
Gearbox ratio	n	1:54
Maximum current	I_{max}	10 A

previous section that in a system with only viscous friction, uncertainty in the viscous friction cannot be compensated for by the choice of the feedforward controller, in this section we will show that compensation is possible when the two other friction terms are non-zero. The two systems we consider are a one DOF robotic arm and a two DOF robotic arm. We will show that on both systems, the sensitivity to all three friction model uncertainties can be eliminated.

7.4.1 One DOF robotic arm

Method

The equations of motion for the one DOF model with complete friction model are given by

$$\frac{d}{dt}\begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ \left(\frac{k_t n I - T_p}{J_{joint}}\right) \end{bmatrix} \tag{7.23}$$

where T_p is the frictional torque from Eq. (7.10). For the one DOF numerical study, we used the parameters values in Table 7.2, which are based on a system identification of the robotic arm in Fig. 7.2c.

The sensitivities in Eq. (7.5) are calculated using a finite difference approximation, with a 0.1% difference in the model parameters. In the numerical optimizations, we parameterized the controller as a piecewise constant current signal with N controller steps of equal length. An example of an input signal with $N = 3$ is shown in Fig. 7.4. These N controller steps are used as decision variables in the optimization in Eq. (7.7), constrained by Eq. (7.8). Since numerical optimization does not exactly reach zero, we set a threshold below which we call the outcome of an optimization

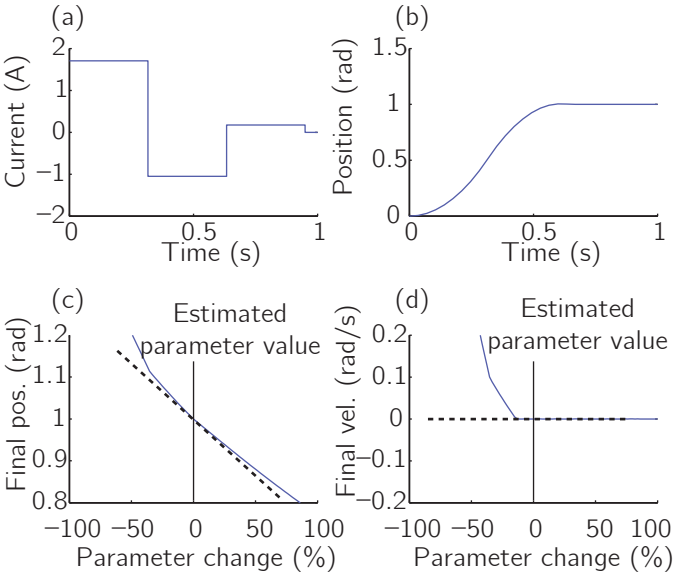


Figure 7.4: An example of a feedforward control signal and the motion that results from the controller. The signal represents the current through the motor as function of the time. There are 4 set points in total ($t_0 \ldots t_3$) and the current is constant in between the set points. The duration of the signal is 1 second. (b) Shows the position of the arm as function of time using the nominal friction values. (c) and (d) show the final position and final velocity as function of the parameter value of the Coulomb friction. The dotted lines show the partial derivatives of the final position and velocity with respect to the parameter value. The goal of the optimization is to minimize a weighed sum of these derivatives

equal to zero. This threshold is equal to the function tolerance of the optimization algorithm, which was set to 10^{-6}.

We used two techniques to speed up the optimizations. Firstly, instead of propagating the equations of motion using an ODE solver, we used the fact that the equations of motion of this one DOF system are piecewise linear, with switching times at changes in control signal and when the velocity crosses zero. Therefore we can calculate the exact end state with a number of calculation steps smaller than $2N$. Secondly, we made use of the fact that the goal velocity of rest-to-rest motions is zero ($\dot{q}_f = 0$). Due to the Coulomb friction in the system, an infinitely small velocity will be reduced to zero by coasting ($u = 0$) for a small amount of time. Therefore, in order to satisfy $\frac{\partial q_f}{\partial p} = 0$, we let the system coast from $t = t_f - 0.05s$ to $t = t_f$. Now the optimization has to satisfy two equality constraints (see Eq. (7.8)) and has to minimize one partial derivative per uncertain parameter ($\frac{\partial q_f}{\partial p_i}$). Therefore, we used $N = 2 + k$ where k is the number of uncertain parameters.

The optimizations were performed with a multistart of the MATLAB function `fmincon` and were run on an Intel DuoCore i7-2620M CPU.

Results

Firstly, we obtained results with only one uncertain parameter at a time and thus $N = 3$. The 50 starts of the optimization had an average duration of 26 ms. In all the three optimizations, the partial derivatives (Eq. (7.5)) were all smaller than 10^{-9} which is smaller than the function tolerance of the optimization algorithm and are therefore considered zero. Lowering the function tolerance to 10^{-12} did lower the partial derivatives to 10^{-13}, while the motions did not change significantly. We chose to use the slightly larger function tolerance of 10^{-6} to keep the computational cost low. The results with an uncertain Coulomb friction are shown in Fig. 7.5. It shows that the final position of the arm is approximately 1 rad, even if p_c changes with 100% of the estimated value. Furthermore, it shows that the arm first moves away from the goal position, before moving towards it. We observed this behavior in all the three cases: with an uncertain p_v, p_c and p_t. Although optimization results can be hard to interpret, we saw the same behavior in the previous section in Fig. 7.3. Therefore, in all the three cases, probably some kind of condition as in Eq. (7.20) is satisfied.

Fig. 7.6 shows the partial derivatives as function of the goal position. It shows that every partial derivative is zero up until a certain threshold of the goal position. This threshold is smallest for the viscous friction (approximately 1.8 rad), is slightly larger for the torque dependent friction (approximately 2.3 rad) and is largest for the Coulomb friction (approximately 6 rad). These values depend on the amount of friction, the inertia and the maximum torque. The latter is a combination of the motor constant, the gearbox ratio and the maximum current.

Secondly, we obtained results with three uncertain parameters at the same time and thus $N = 5$. The 100 starts of the optimization had an average duration of 53 ms. The results of the optimization are shown in Fig. 7.7. We again see that the arm first moves away from the goal position. This was to be expected since the three separate optimizations showed this behavior. Furthermore, Fig. 7.7 shows that the final position is most sensitive to an uncertain torque dependent friction. The partial derivatives (Eq. (7.5)) were all zero.

7.4.2 Two DOF robotic arm

Methods

We performed the same optimizations as on the one DOF model on a two DOF SCARA type arm model (see Fig. 8.3), to see how the results extrapolate to a

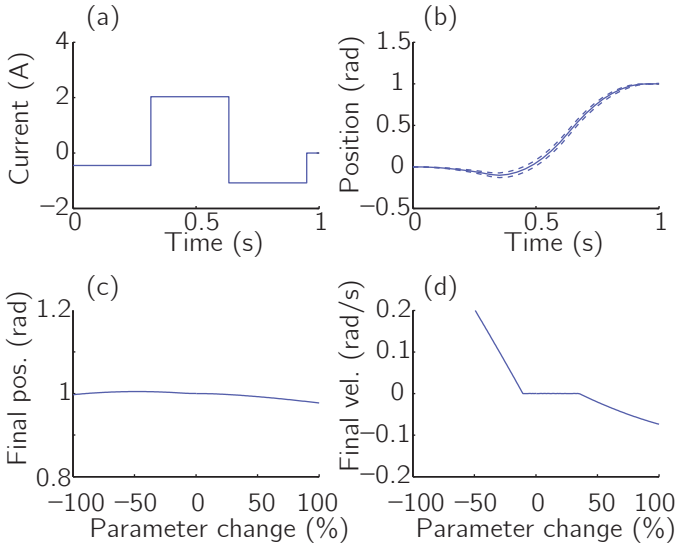


Figure 7.5: The results of the optimization with an uncertain Coulomb friction. (a) The current through the motor as function of time. (b) The position of the arm as function of time using the nominal friction values. The dashed lines show the position over time with -40% and +40% parameter change. (c) The final position as function of the parameter value of the Coulomb friction. (d) The final velocity as function of the parameter value of the Coulomb friction. The two bottom graphs show that the partial derivatives of the final state with respect to the parameter are zero.

Table 7.3: The model parameters of the two DOF arm.

Parameter	Symbols	Values	Unit
Viscous friction	p_{v1}, p_{v2}	-0.05, -0.14	Nms/rad
Coulomb friction	p_{c1}, p_{c2}	0.19, 0.60	Nm
Torque dependent friction	p_{t1}, p_{t2}	22, 32	%
Inertia about COM	J_{C1}, J_{C2}	0.023, 0.178	kgm^2
Mass upper arm	m_1, m_2	0.809, 1.502	kg
Length of link upper arm	l_1, l_2	0.41, 0.43	m
Position COM	l_{C1}, l_{C2}	0.07, 0.33	m
Motor constant	k_{t1}, k_{t2}	26.7, 28.1	mNm/A
Gearbox ratio	n_1, n_2	1:54, 1:198	
Maximum current	I_{max1}, I_{max2}	10, 10	A

robotic arm with non-linear dynamics. We used the TMT method [121] to obtain the equations of motion of the two DOF simulation model, which are too long to include in this chapter. In the two DOF model, a motor actuates the absolute angle

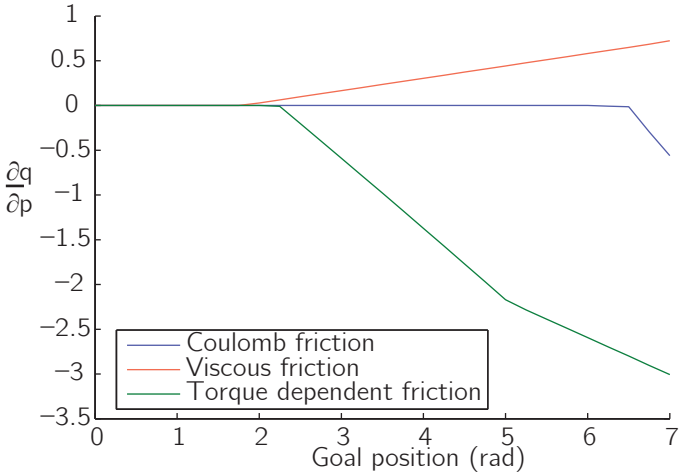


Figure 7.6: This figure shows the partial derivative of the goal position to the parameter changes as function of the goal position. The partial derivatives for all the three parameter variations are zero until a certain threshold on the goal position.

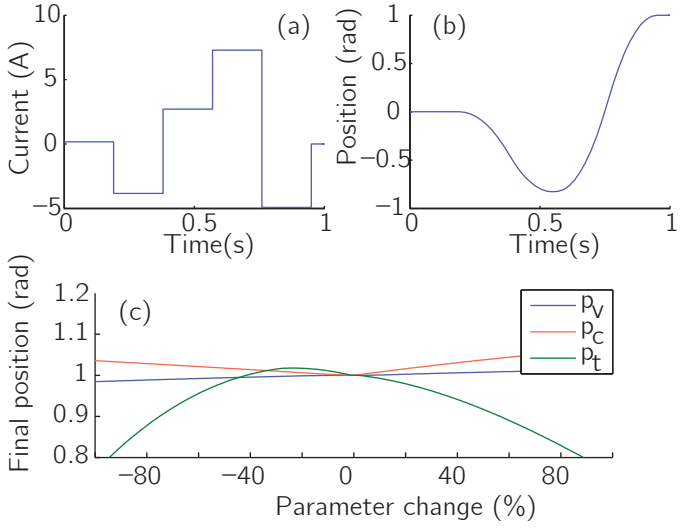


Figure 7.7: The results of the optimization on a one DOF arm with three uncertain friction parameters at the same time. (a) The current through the motor as function of time. (b) the position of the arm as function of time using the nominal friction values. (c) The final position of the arm as function of the three parameter changes. The bottom graph shows that the partial derivatives of the final position with respect to the three parameters values are zero.

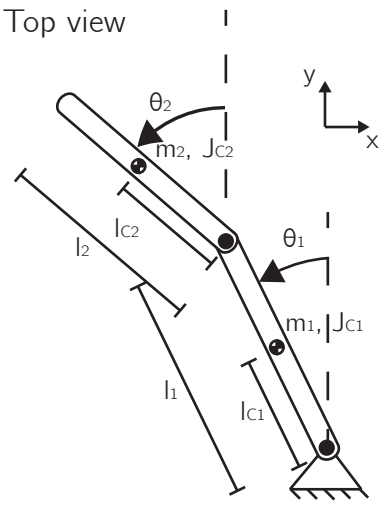


Figure 7.8: Top view of the two DOF SCARA type arm model. The second joint is actuated through a parallel mechanism (not shown in this figure), such that the angle of the second arm is an absolute angle. The friction acts on the absolute angles of the joints.

of the second joint. Since approximately all friction is caused by the motor, the friction in the second joint is a function of the absolute angular velocity of that joint. The friction between the first link and the second link is assumed to be zero.

The parameter values of the two DOF model are listed in Table 7.3. In this model, there are multiple parameters that influence the inertia of the upper and lower arm: the inertias about the centers of mass (COMs), the lengths of the arms, the masses of the arms and the positions of the COMs. The 3 friction parameters per joint lead to 6 uncertain parameters and thus 12 partial derivatives that should be equal to zero. Therefore, on both joints we took $N = 8$.

Results

Figs. 7.9c and 7.9d show the final positions of the two arms as functions of the parameter changes. The values of the partial derivatives are smaller than the function tolerance of the optimization and are therefore considered zero.

Figs. 7.9b shows the positions of the two joints as functions of time. The first joint shows the same behavior as the one DOF system: it first moves away from the goal position before moving towards it.

One optimization of the two DOF system takes about 10 hours, with the current state of technology, which is too long to be applicable. We will discuss this issue in

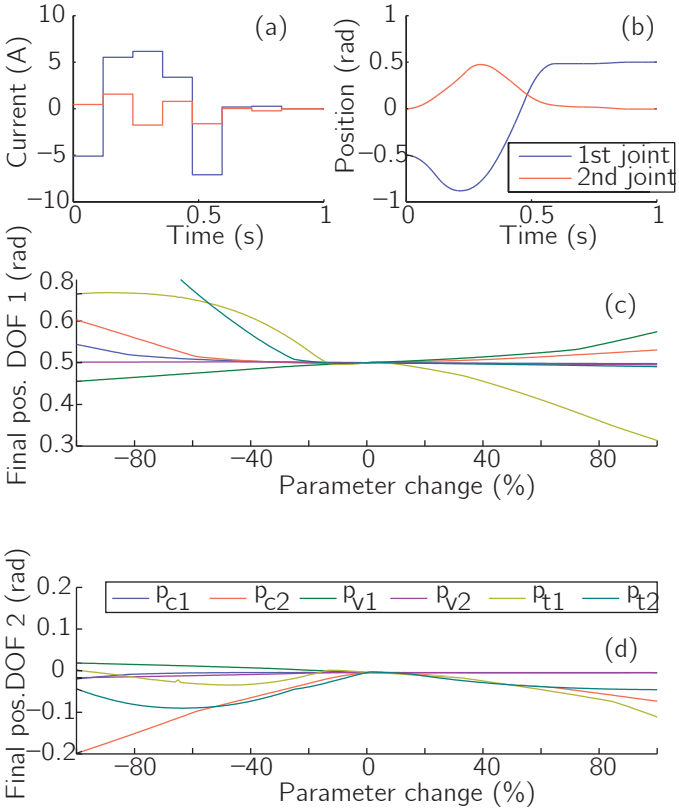


Figure 7.9: The results of the optimization on a two DOF arm with three uncertain friction parameters at the same time. (a) The currents through the motors as function of time. (b) the positions of the joints as function of time using the nominal friction values. (c) The final position of the first joint as function of the six parameter changes. (d) The final position of the second joint as function of the six parameter changes. The two bottom graphs show that the partial derivatives of the final position with respect to the parameters values are approximately zero.

section 7.6.

7.5 Hardware Study

In the previous sections we saw that in theory, feedforward controlled motions of our robotic arm can be made insensitive to friction model uncertainty. In this section, we show that this is also possible on the robotic arm itself. We verify the results with an uncertain Coulomb friction on our one DOF robotic arm and show that the sensitivity to this friction uncertainty can be eliminated to negligible levels. Firstly,

we will explain the test set up, secondly, we will explain our test protocol and finally, we will show the results.

7.5.1 The robotic arm

Fig. 7.2c shows a picture of the one DOF robotic arm, which is the same arm as in [171], but without the spring mechanism. The DOF consists of an 18x1.5mm stainless steel tube connected with a joint to the ground. A weight of 1 kg is connected to the end of the tube, which represents the weight of a gripper plus a product. The motor is placed on a housing and AT3-gen III 16mm timing belts are used to transfer torques within the housing. The joint is actuated by a Maxon 60W RE30 motor with a gearbox ratio of 18:1. The timing belts provide an additional transfer ratio of 3:1. The model parameters as shown in Table 7.2 are based on a system identification of this robotic arm.

To change the Coulomb friction, we designed a mechanism that adds Coulomb friction by clamping a nylon sleeve bearing on the motor axis. The Coulomb friction can be increased by tightening the screw of the clamping mechanism. Every time after we changed the Coulomb friction, we ran a system identification to determine the amount of friction that was added.

7.5.2 Test protocol

We performed an online optimization to find the feedforward controller that is most robust to uncertainty in the Coulomb friction. This optimization consisted of testing a grid of 32 different feedforward controllers and selecting the best one. Each feedforward controller had three current set points of which the first one was determined by a grid of 32 set points. The other two set points were determined by the constraints on the final position and final velocity in simulation.

Instead of using the derivative as a performance measure, we used the root mean squared (RMS) of the position errors using three different values for the Coulomb friction: 0.19 Nm, 0.22 Nm and 0.25 Nm.

7.5.3 Results

Fig. 7.10 shows the current and the position as function of time for a typical experimental run. The two controllers shown are the two resulting in the minimum and maximum errors in the hardware experiments. It also shows that the current profiles with minimal and maximal error in hardware experiments correspond to the current profiles with minimized and maximized sensitivity in simulation. For systems with more DOFs it is less feasible to perform a grid search and thus the current profiles from simulation should be used.

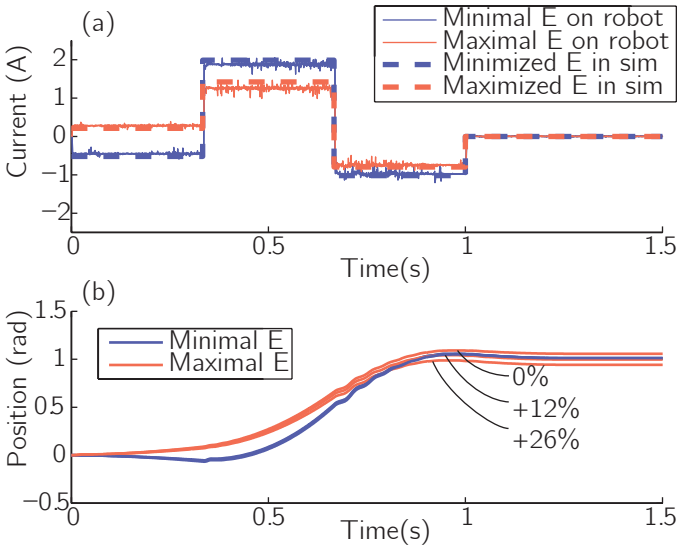


Figure 7.10: A typical example of the data obtained from hardware experiments. This figure shows the minimized and maximized motions for a changing Coulomb friction. a) The feedforward current as function of the time. The solid lines correspond to the motions with minimized and maximized error in hardware experiments. The dashed lines are optimized current profiles in simulation. This graph shows that the current profiles optimized in simulation are the same as the current profiles with minimal and maximal error in hardware experiments. b) The position of the arm as function of the time. We clearly see that the spread in the final position of the minimized motion is smaller than that of the maximized motion. For the minimized motion, it is hard to distinguish the three lines, for the maximized motion the spread is clearly visible. (Figure from [167])

The minimal and maximal RMS were 0.002 rad and 0.092 rad respectively. These hardware experiments confirm the main conclusions of the simulation study: the errors due to uncertainty in the friction model can be reduced to approximately zero (at least for the Coulomb friction). Furthermore, again the optimal motion first moves in negative direction before moving towards the goal position.

7.6 Discussion

In this study we researched motions of a one and two DOF robotic arm, controlled by a feedforward controller. The task consisted of fixed initial and goal positions and a fixed time per stroke. We showed that the choice of the motions in between the initial and goal positions is important for the sensitivity of the final state to friction model uncertainty. For all systems, this sensitivity can be eliminated.

7.6.1 Implications

The results of this study are important to consider when implementing feedforward control. The correct use of feedforward control improves the performance of the system and this study shows that the performance can even be improved in such a way that the friction parameters do not have to be known accurately. The interesting result is that all optimized motions do not move from the initial to the goal position directly, but first move away from the goal position. We showed why this strategy is advantageous in one DOF with an uncertain Coulomb friction and we expect that similar explanations hold for other friction uncertainties and systems.

This study also has implications on the field of human motion control. Recent studies in the field of human motion control focused on the uncertainty (i.e. noise) in the control signals [82]. It would be interesting to research the accuracy of the internal models of humans and the influence of this accuracy on the motions humans choose. Another interesting topic for future research would be the influence of noise on the performance of feedforward control in robotic systems.

The one DOF system is a system with linear dynamics and non-linear friction and the two DOF system is a system with non-linear dynamics. We expect that the results of this research can be extrapolated to a variety of other systems. We did not consider gravity in this study. Adding DOFs that are influenced by the gravity adds non-linearity to the system. We expect that sensitivity minimizing feedforward controllers can still be found on such systems.

7.6.2 Linearization

In this study, we used a linearization of the influence of a parameter on the final state as a measure of the performance of a feedforward controller. This linearization only accounts for infinitely small parameter changes and thus it does not tell anything about finite parameter changes. However, many of our results showed that the errors in the final position were even small with large parameter changes (see e.g. Fig 7.5 and Fig. 7.7). A first alternative to linearizing is to sample a couple of parameter values around the estimated value (e.g. estimated value -10% and estimated value +10%) and minimize a sum or RMS of the errors at those parameter values, as we did in [167]. However, such an optimization takes longer since more decision variables are needed. A second alternative would be to also take into account higher order partial derivatives, as in [18]. We do not expect that this will lead to a better performance since the results already show a low second order derivative (see Figs. 7.5, 7.7 and 7.9).

7.6.3 Adding feedback

In most applications, some kind of feedback is available. Combining a separately designed feedback controller and a robust feedfoward controller will in general not lead to good results, since in general the superposition principle does not hold. An approach could be to first design a feedback controller and then optimize the combination of that controller with a feedforward controller to be robust to model uncertainty. Although showing the feasibility of this approach is part of future work, an interesting approach was recently proposed by Ansari and Murphey [8].

7.6.4 Optimization duration

For the one DOF system, the duration of one optimization was 5.3 seconds. This is still too slow for performing optimizations while the arm is performing its task. However, there are obvious ways to solve this problem, like creating a database of motions and only use the optimization to adjust the closest motion from the database to fit the required motion. Such an optimization should not require a multistart and should therefore take about 50 ms for the considered system. In general, the optimization would be fast enough if the optimization time is shorter than the time to move, which in our example was 1s.

A problem arises when the systems become more complex: more DOFs or more uncertain parameters. An example of this is the two DOF system in section 7.4.2, for which an optimization had to run for 10 hours to find a reasonable solution. This problem is caused by the large number of local minima in the optimization function. These local minima are caused by a combination of highly non-linear dynamics and a relatively large number of decision variables. Although this problem is suited for parallelization, it would require more than 36000 cores to speed up the optimization to 1 second per motion. Therefore, the best approach would be to search for methods that do not have this large amount of local minima.

Because of this large optimization time, we did not perform an elaborate two DOF study on the influence that the initial and goal positions have on the results. However, trying a couple of random tasks showed that similar results can be obtained for other tasks. We expect that what we showed for the one DOF system, also holds for the two DOF system: the sensitivity can be reduced to zero if the distance between initial and goal positions does not exceed a certain threshold.

7.6.5 Other optimization goals

In this chapter, we optimized feedforward controllers such that the sensitivity to friction model uncertainty was minimized. However, both in industry and in human

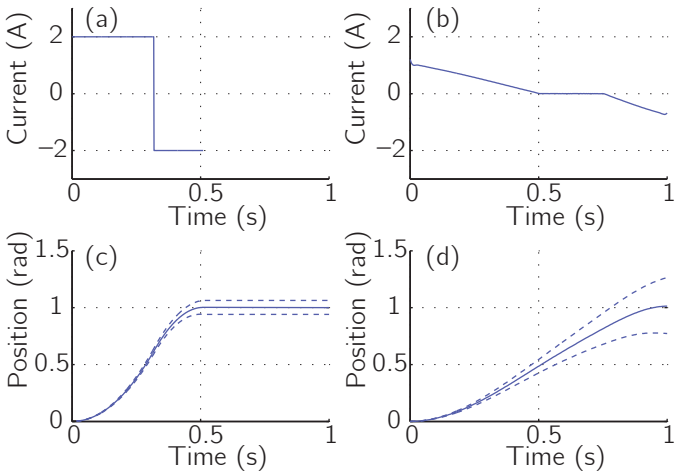


Figure 7.11: The time and energy optimal motions as function of time. (a) The current through the motor of the time optimal motion. (b) The current through the motor of the energy optimal motion. (c) The position of the arm of the time optimal motion. (d) The position of the arm of the energy optimal motion. The dashed lines show the position over time with -40% and +40% parameter change.

motion control, other cost functions are used as well. Two common cost functions are the time per stroke and the energy consumption per stroke. The question that remains is how these commonly used cost functions perform with an uncertain friction model in comparison to the results of the optimization in this chapter.

Fig. 7.11 shows the energy and time optimal motions of the one DOF system with a complete friction model. The time optimal torque profile was obtained by minimizing the time the system reaches the goal state. The energy optimal torque profile was obtained by minimizing the integral of the electrical motor power over time (see [171]). The values of the partial derivatives are shown in Table 7.4. This shows that both the energy and time optimal motions are very sensitive to friction model inaccuracies, and so a trade-off has to be made between energy, time and robustness.

7.7 Conclusion

In this chapter, we optimized feedforward controllers for robustness to uncertainty in the friction model. On both the one and two DOF system, we eliminated the sensitivity of the final position of rest-to-rest motions to parametric uncertainty in the friction model. Interestingly, all motions that are robust to friction model uncertainty, first move away from the goal position before moving towards it. Such

Table 7.4: The partial derivatives of energy and time optimal motions

	p_c	p_v	p_t
Energy optimal			
$\frac{\partial q}{\partial p_i}$	-0.626	0.163	-0.499
$\frac{\partial \dot{q}}{\partial p_i}$	-1.318	0.357	-0.839
Time optimal			
$\frac{\partial q}{\partial p_i}$	-0.154	0.074	-0.499
$\frac{\partial \dot{q}}{\partial p_i}$	-0.344	0.180	-1.109

motions eliminate the sensitivity by reducing the integral of the effect of the friction torque to zero.

Acknowledgement

This work is part of the research programme STW, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO).

8

Open Loop Stable Control in Repetitive Manipulation Tasks

Michiel Plooij*, Wouter Wolfslag* and Martijn Wisse
* These authors contributed equally to this chapter,
IEEE International Conference on Robotics and Automation 2014
pp. 949-956.

Abstract

Most conventional robotic arms depend on sensory feedback to perform their tasks. When feedback is inaccurate, slow or otherwise unreliable, robots should behave more like humans: rely on feedforward instead. This chapter presents an approach to perform repetitive tasks with robotic arms, without the need for feedback (i.e. the control is open loop). The cyclic motions of the repetitive tasks are analyzed using an approach similar to limit cycle theory. We optimize open loop control signals that result in open loop stable motions. This approach to manipulator control was implemented on a two DOF arm in the horizontal plane with a spring on the first DOF, of which we show simulation and hardware results. The results show that both in simulation and in hardware experiments, it is possible to create open loop stable cycles. However, the two resulting cycles are different due to model inaccuracies. We also show simulation and hardware results for an inverted pendulum, of which we have a more accurate model. These results show stable cycles that are the same in simulation and hardware experiments.

8.1 Introduction

The vast majority of robotic manipulators require sensory feedback in order to perform their tasks. Humans, on the other hand, use both feedback and feedforward (or open loop) control when controlling their body [47]. Using feedforward allows humans to control their body despite having large time delays (typically 150 ms [43, 215]). Although robots have faster feedback loops, feedforward still has advantages. First, feedforward can anticipate on future states of the system, second, is can offer cheaper control when the cost is critical, and third, it is suitable for systems with slow and imprecise feedback, such as camera based feedback.

In a previous study, we showed the remarkable result that sensitivity to some modeling inaccuracies can be eliminated by choosing the right feedforward controller [174]. This result was similar to observations in humans, who minimize the influence of uncertainty on the final position of feedforward controlled movements [81]. Where the previous study only considered short motions (one second), this chapter takes those ideas a step further by considering long term stability of open loop controlled robotic arms (see Fig. 8.1), inspired by (human) stable walking motions.

One of the commonly cited disadvantages of open loop controllers is that they cannot directly compensate for perturbations, since those perturbations are not fed back into

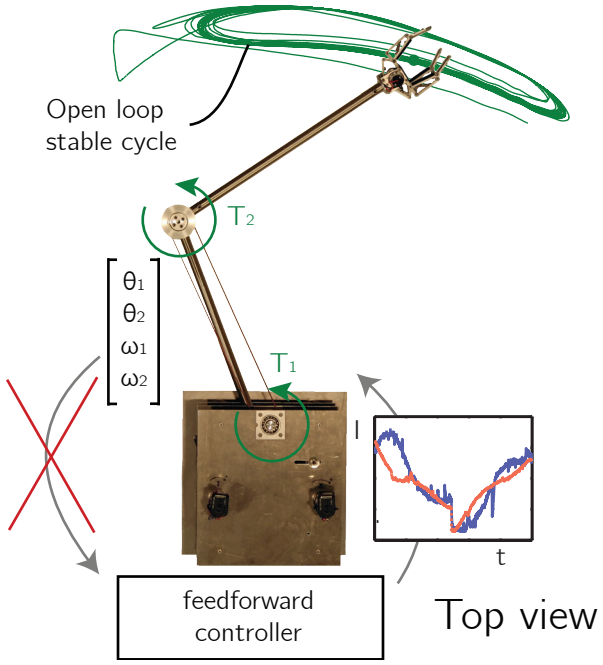


Figure 8.1: This figure shows the top view of the concept of open loop stable manipulation for our two DOF robot setup. Since there is no feedback available, the torque signal is a function of time only. Using numerical optimizations, we find torque signals that result in open loop stable cycles and allow the robot to perform repetitive tasks. Both the path displayed and the current signals are obtained from hardware experiments.

the controller. Since most tasks of robotic arms are repetitive, we propose to view them as cycles and consider the inherent stability of those cycles. If the trajectories we perform are stable, perturbations will simply decay over time, without the need for sensory feedback.

Other researchers have already used the inherent stability of specific motions in order to create functional robots. Those robots can be split into two groups: robots with only limited state feedback and robots without any state feedback.

A well known example of robots with limited state feedback is given by Schaal and Atkeson [188], who studied open loop stable juggling with a robotic arm. In their case, open loop means that the state of the ball is not used as an input for the controller, but the arm itself is position controlled. Other examples include a timed position controlled swing leg retraction to stabilize running [195] and rope turning without measuring the state of the rope [105].

The group that is more related to this study is the group of robots without any state feedback. The most striking result in this group was obtained by McGeer, who introduced the concept of passive dynamic walking [133].Those walkers do not have motors and thus they do not use any feedback control, while their walking motion is stable. The stable walking motions do not rely on the motion being stable at each point in time, rather they work due to the existence of stable cyclic motions, called limit cycles. Such cycles were later on used in so called limit cycle walkers in combination with feedback control [73, 89, 95]. The work most strongly related to this paper is that by Mombaur et al. [140, 142]. They found stable open loop controllers for walking and running robots by optimizing the open loop controllers for both stability of the motion and energy consumption. These results on a variety of systems indicate that open loop stable control can be an effective approach for robotic arms as well.

The goal of this chapter is to introduce a new approach in robot manipulator control: *open loop stable manipulation* and to show the first results using this approach. We show results both in simulation studies and in hardware experiments.

The remainder of this chapter is structured as follows. Section 8.2 explains the configurations we studied and the optimization method we used. Section 8.3 shows the results for the two DOF robotic arm, including a description of the simulation model, the simulation results and the hardware results. Section 8.4 shows the results for the inverted pendulum, including a description of the simulation model, the simulation results and the hardware results. Finally, the chapter ends with a discussion in Section 8.5 and a conclusion in Section 8.6, where we will conclude that it is possible to perform manipulation tasks with an open loop controller by performing open loop stable cycles.

8.2 Methods

We studied open loop stable task execution of robotic arms by optimizing the open loop controller such that the task is performed by a stable cyclic motion. In this section, we discuss the configurations we studied and the optimization method.

8.2.1 Configurations

We studied four configurations, which are also shown in Fig. 8.2:

(a) **A simulation model of a two DOF robotic arm**. On this model we optimized the open loop controller such that the arm makes stable cyclic motions while performing the specified task.

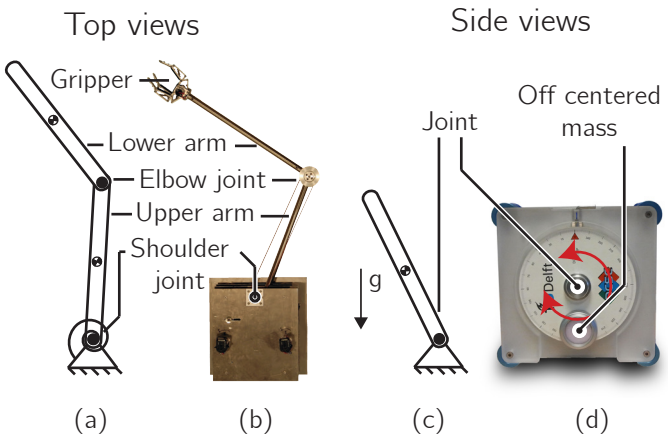


Figure 8.2: The four configurations we studied: (a) a simulation model of a two DOF robotic arm, (b) a two DOF robotic arm, (c) a simulation model of an inverted pendulum and (d) an inverted pendulum.

(b) **A prototype of a two DOF robotic arm**. We implemented the controller obtained in (a) on a two DOF robotic arm to test how the controller performs on a real system. The results will show that the open loop controller generates stable cycles on the robot, but does not converge to the same trajectory as in (a). In Section 8.5.2, we will argue that this is caused by a bending of the second DOF due to gravity.

(c) **A simulation model of an inverted pendulum**. We used the same techniques as used in (a) to obtain open loop stable motions of an inverted pendulum. We show that on this system, the results have an intuitive explanation.

(d) **A prototype of an inverted pendulum**. We implemented the controller obtained in (c) on an inverted pendulum to test how the controller performs on a real system for which an accurate model is known.

8.2.2 Optimization method

The stability of limit cycles can be assessed with a number of different measures, such as Lyapunov stability [211] or contraction analysis [122, 127]. We used the classic notion of Poincaré maps [211] to find the stability of the open loop controlled motions, which we will now explain.

Consider a non-linear system described by the following differential equation:

$$\dot{x} = f(x, u(t)) \tag{8.1}$$

Since we use open loop control, we can consider the time as an extra state. Using transverse coordinates [15, 126, 201] with the time as phase variable in this appended state space is the same as using error dynamics in the original state space. This means that we can set a Poincaré section at $t = t_f$. We calculated the error dynamics along the trajectory $x^*(t)$ that results from the input $u^*(t)$. Since the controller is open loop, the error dynamics are simply given as the difference between the current state and the trajectory state, both at the same time:

$$\dot{x^*} = f(x^*, u^*(t)) \tag{8.2}$$

$$\delta x = x - x^* \tag{8.3}$$

$$\dot{\delta x} = f(x, u^*(t)) - f(x^*, u^*(t)) \tag{8.4}$$

Linearizing the dynamics along the trajectory results in:

$$\dot{\delta x} = \frac{\partial f(x^*, u^*(t))}{\partial x} \delta x \tag{8.5}$$

$$= A^*(t)\delta x \tag{8.6}$$

Where A^* is the linearized system matrix, and δx the error-state. We constrained the motions to be cyclic with period t_f (see below), which results in a cyclic $A^*(t)$ with the same period. Since system (8.6) is linear, we can take the state transition matrix Ψ from $t = t_0 = 0$ to $t = t_f$. Because we know $A^*(t)$ analytically, we can find Ψ by numerically computing the solution to the following initial value problem:

$$\dot{\Psi} = A^*(t)\Psi, \ \Psi(0) = I \tag{8.7}$$

Similar to the monodromy matrix in Poincaré map analysis of limit cycles, the motion is stable if the eigenvalues of Ψ have an absolute value smaller than one:

$$|\lambda(\Psi(t_f))|_{max} < 1 \tag{8.8}$$

To find a trajectory, we now use an optimal control approach similar to [142]. We used the above condition as a constraint rather than to minimize the left hand side of it, because of two reasons. First, the above condition specifies the convergence rate of the limit cycle, and does not specify other stability related factors, such as basin of attraction [90] and robustness against model uncertainties. Second, in practice other performance issues are also of concern, such as energy consumption and speed. We chose to use the integral of the squared input as objective, resulting in the following optimization:

$$\underset{u(t)}{\text{minimize}} \quad \int_{t_0}^{t_f} u(t)^2 dt \tag{8.9}$$

$$\begin{aligned} \text{subject to} |\lambda(\Psi(t_f))|_{max} &< 0.9 \\ |u(t)| &< u_{max} \\ x(0) &= x_{pick} \\ x(t_1) &= x_{place} \\ x(t_f) &= x_{pick} = x(0) \end{aligned} \tag{8.10}$$

Where $u(t)$ is the input, u_{max} is the maximum input and x_{pick} and x_{place} are the pick and place states (see Section 8.3.2). The constraint that the final state is equal to the initial state causes the resultant motion to be cyclic.

In case of a setup with multiple motors, we used the integral of the sum of squared inputs as cost function. We used this cost function because it is often used in other control applications. Furthermore, the resulting controllers are relatively smooth, whereas a time-optimal controller would be bang-bang. Such a bang-bang controller is undesirable in robot experiments, because it is more likely to be affected by unmodeled effects such as backlash. The optimization is performed using the optimal control package GPOPS [179] in Matlab.

8.3 Two DOF manipulator

We implemented open loop stable manipulation on a SCARA type arm with two DOFs: two revolute joints moving two links in the horizontal plane (see Fig. 8.2b and 8.3). Since there is no spring present on the second joint of the arm, an open loop controlled motion must depend on dynamic effects for stabilization. Specifically, the zero input does not lead to a stable result. Since the dynamic effects are highly non-linear, it is not obvious whether stable motions exist. This section starts with a system description, including a description of the robotic arm and the simulation model. Next, we explain the task the arm has to perform, followed by simulation results and hardware results.

8.3.1 System description

Fig. 8.1 shows a picture of the two DOF robotic arm [171]. The DOFs are created by two 18x1.5mm stainless steel tubes, connected with two revolute joints, with a spring on the first joint. A mass of 1 kg is connected to the end of the second tube, which represents the weight of a gripper with product. The motors are placed on a housing and AT3-gen III 16mm timing belts transfer the torques within the housing. The joints are actuated by Maxon 60W RE30 motors with gearbox ratios of respectively 18:1 and 1:66. The timing belts provide an additional transfer ratio of 3:1 on the first joint and 5:3 on the second joint.

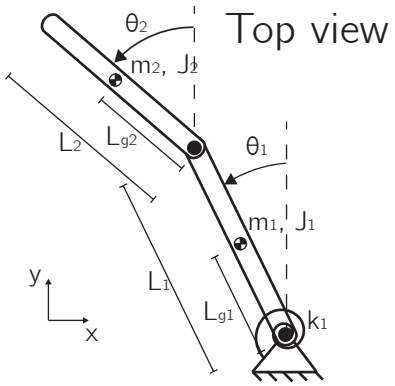


Figure 8.3: Top view of the two DOF system with a linear spring on the first joint. The second joint is actuated through a parallel mechanism (not shown in this figure), such that the angle of the second arm is an absolute angle. The friction acts on the absolute angles of the joints.

Table 8.1: The model parameters of the two DOF arm.

Parameter	Symbol	Value	Unit
Coulomb friction	μ_{c1}, μ_{c2}	0.0481, 0.0218	Nm
Viscous friction	μ_{v1}, μ_{v2}	0.03, 0.03	Nms/rad
Torque dependent friction	μ_{tf1}, μ_{tf2}	21.87, 31.91	%
Inertia	J_1, J_2	0.0233, 0.0871	kgm^2
Mass	m_1, m_2	0.809, 1.599	kg
Length	l_1, l_2	0.410, 0.450	m
Position of COM	l_{g1}, l_{g2}	0.070, 0.325	m
Motor constant	k_{t1}, k_{t2}	26.7, 28.1	mNm/A
Gearbox ratio	n_1, n_2	1:54, 1:110	rad/rad
Spring stiffness	k_1	1.6	Nm/rad

We used the TMT method [120] to obtain the equations of motion of the simulation model of this arm, which are too long to include in this chapter. The model includes 19 parameters, which are listed in Table 8.1. We included three types of frictional losses: Coulomb friction, viscous friction and torque dependent gearbox friction. Torque dependent gearbox friction is less commonly used than the other two, however, from the parameter values obtained through a system identification of the motor it is clear that this type of friction is not negligible (see Table 8.1). The way we implemented it is similar to the force dependent friction term in [52]. The

friction in a joint is equal to:

$$T_f = -\mu_v \cdot \omega - \text{sign}(\omega) \cdot (\mu_c + \mu_t \cdot |T|) \tag{8.11}$$

for $\omega \neq 0$

$$T_f = -\min(\mu_c + \mu_t \cdot |T|; |T|) \cdot \text{sign}(T) \tag{8.12}$$

for $\omega = 0$.

ω is the velocity of the joint, T is the torque exerted by the motor on the joint, and μ_v, μ_c and μ_t are the coefficients of viscous, Coulomb and torque dependent friction respectively.

The simulation model includes a DC motor. The torque applied by the DC motor on the joint is equal to:

$$T = n \cdot k_t \cdot I \tag{8.13}$$

Where k_t is the motor constant, n is the gearbox ratio and I is the current through the motor. The current through the motor is constrained to 1 A (see eq. 8.10).

Coulomb friction and torque dependent friction introduce discontinuities in the equations of motion, which are difficult for GPOPS to handle. Therefore, we optimized the open loop controller on a model with only viscous friction, and added a torque afterwards to compensate for the Coulomb friction and torque dependent friction. Such a compensation can only be done when it does not effect the stability of the cycle, so when A^* is independent of both the compensated friction and the input. Both the Coulomb friction and the torque dependent friction depend on the state through a sign function, which is a piecewise constant function. Since A^* is the result of linearizing along the state, A^* does not depend on those friction terms (neglecting the always stabilizing effect of the discontinuity in the sign function). To make A^* independent of the input, we consider momenta instead of velocities. The equations of motion then become:

$$\dot{x} = f(x) + Bu(t) \tag{8.14}$$

With x, the state consisting of positions and momenta, and B a constant matrix. Following the definition in eq. (8.6), we see that A^* is therefore independent of u. Such a transformation is possible for many mechanical systems. Although such a transformation is not necessary, our specific optimization was faster with the transformation. For easier interpretation, we will show the velocities in the results.

8.3.2 Task

The robotic arm has to perform a pick-and-place task. The important task parameters are the pick state, the place state and the time per stroke. We show the results

of the optimization for a motion which starts at $t = 0$ at the pick state x_{pick}, goes to the place state x_{place} at t_1 and then returns to the pick state at t_f with

$$x_{pick} = \begin{bmatrix} -0.7 \text{ rad} \\ -0.85 \text{ rad} \\ 0 \text{ rad/s} \\ 0 \text{ rad/s} \end{bmatrix}; x_{place} = \begin{bmatrix} 0.7 \text{ rad} \\ -0.3 \text{ rad} \\ 0 \text{ rad/s} \\ 0 \text{ rad/s} \end{bmatrix} \tag{8.15}$$

Where x is the vector consisting of the positions of the first and second arm, and their respective angular velocities. t_1 and t_f are free parameters in the optimization, but bounded as follows:

$$0.1 \text{ s} \leq t_1 \leq 1.2 \text{ s} \tag{8.16}$$

$$0.1 \text{ s} \leq t_f - t_1 \leq 1.2 \text{ s} \tag{8.17}$$

The goal is to find a path that statisfies the task constraints (that also include a stability-enforcing constraint) according to eq. (8.10), and then minimize the integral of the square of the current (see eq. (8.9)). Although we only show the result of one specific task, similar results were obtained using other task constraints. In none of these cases multi-starts or tuned initial conditions were needed, even though the optimization is non-convex.

8.3.3 Simulation results

Fig. 8.4 shows the position of the gripper in the workspace. The gripper does not start at the stable cycle, but converges to it. The sharp corners in the motion are the pick and place positions. Similar results were obtained for different pick and place positions.

Fig. 8.5 shows an example motion converging to the stable cycle in state space. The motion starts at a distance from the pick state $x_{start} = x_{pick} + [0.07; -0.15; 0; 0]$ and converges to the open loop stable cycle.

Fig. 8.6 shows the motion and current profile that result from the optimization. In Fig. 8.6a and 8.6b, we see that the motion starts and ends at the pick state, while at t $\approx$ 1.2 s, the arm is at the place state.

8.3.4 Hardware results

Fig. 8.7 shows the position of the gripper in the workspace. It clearly shows that the gripper does not start at the stable cycle, but converges to the cycle. Comparing the hardware results (Fig. 8.7) with the simulation results (Fig. 8.5) leads to the conclusion that although both the simulation and hardware results show convergence

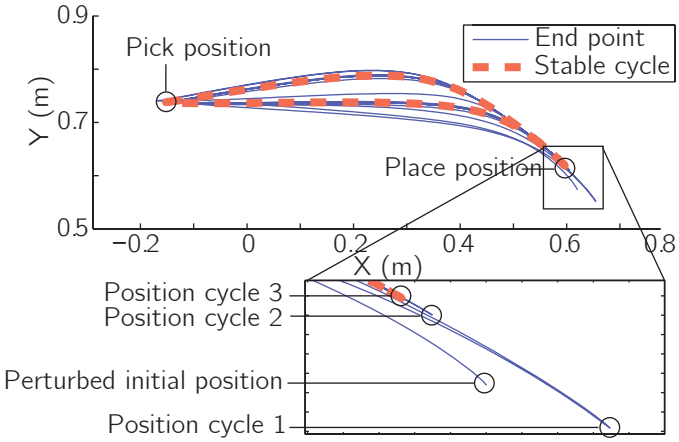


Figure 8.4: This figure shows the plot of the end point of the arm in simulation. The plot shows the stable cycle from simulation (thick dotted red line) and the motion of the robotic arm that starts from a perturbed position and converges to the stable cycle (thin blue line).

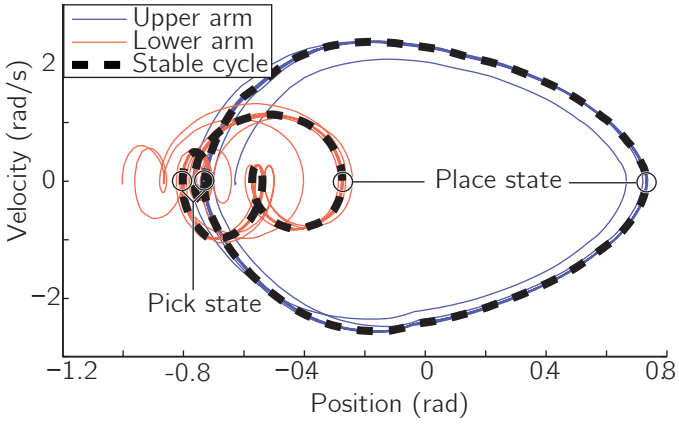


Figure 8.5: This figure shows the state space plot of the simulation data for the two DOF robotic arm. The plot shows the stable cycle (thick dotted red lines) and a motion that starts from a perturbed position and converges to the open loop stable cycle (thin lines).

to a stable cycle, the stable cycles themselves are different. In Section 8.5.2 we will argue that the difference is probably caused by a bending of the second DOF due to gravity. In Section 8.4, we will show that for a simpler system (i.e. an inverted pendulum), our model is accurate enough to predict the exact cycle.

Fig. 8.8 shows the motion of the arm in state space. We see that the motion converges to a stable cycle in state space. When comparing Fig. 8.8 with Fig. 8.5,

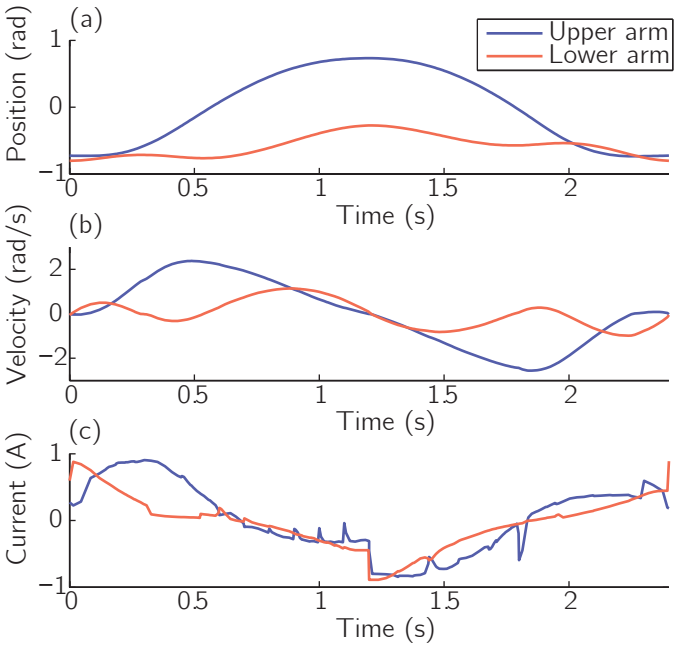


Figure 8.6: This figure shows the simulation data for the two DOF robotic arm as function of time. The figure shows the positions of the joints (a), the velocities of the joints (b) and the torques about the joints (c).

we see that range of positions of the upper arm is smaller in hardware experiments, and that the range of positions of the lower arm is larger in hardware experiments.

Fig. 8.9 shows the time series of a typical cycle of the robotic arm in hardware experiments. Again we see that the hardware results differ from the simulation results. In Fig. 8.9b, we notice that the velocity signals show a vibration at approximately 10 Hz. This vibration is caused by the elasticity of the timing belts between the motors (with encoders) and the joints.

The accompanying video shows a demonstration of the disturbance recovery of the two DOF arm. This video also indicates that the basin of attraction is large. In Section 8.5.2, we will show this basin of attraction in more detail.

8.4 Inverted pendulum

In the previous section, we showed the results of a two DOF SCARA type arm. These results show that pick and place motions can be performed in an open loop stable

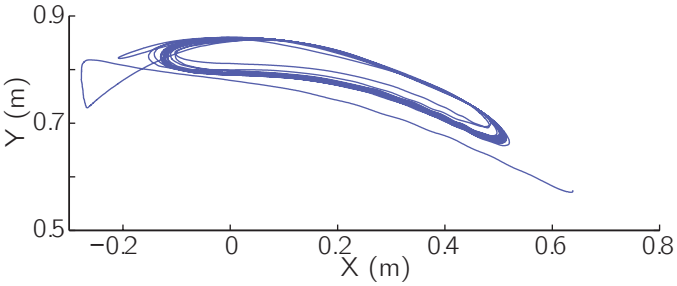


Figure 8.7: This figure shows the plot of the gripper in hardware experiments. The plot shows the motion of the robotic arm that starts from a perturbed position and converges to the stable cycle.

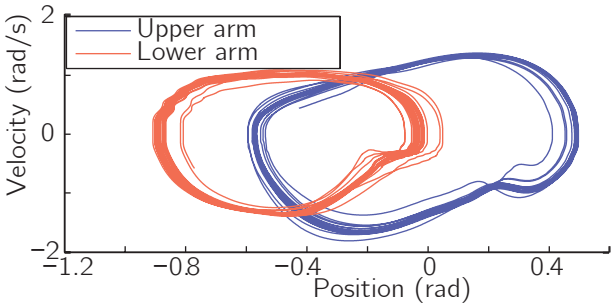


Figure 8.8: This figure shows the state space plot of the hardware experiments on the two DOF robotic arm. The plot shows the motion of the robotic arm that starts from a perturbed state and converges to the open loop stable cycle. In order to obtain a smooth graph, the velocity data is filtered with a fifth order Butterworth filter with the cutoff frequency at 10 Hz.

manner. However, there was a difference between simulation and hardware results due to model inaccuracies. In this Section we move to a simpler system, allowing us to both give an intuitive explanation for how the stabilization works, and indicate that with a more accurate model hardware experiments and simulation can be made to match. For this purpose an inverted pendulum is used (see Fig. 8.10).

8.4.1 System description

Fig. 8.2d shows a picture of the inverted pendulum setup we used. The pendulum consists of a disk with an off-centered mass, which is connected to a DC motor without a gearbox in between. This direct drive actuation results in low friction, which makes the system easier to model. The DC motor is voltage controlled with a maximum voltage of 5 V (see eq. 8.10).

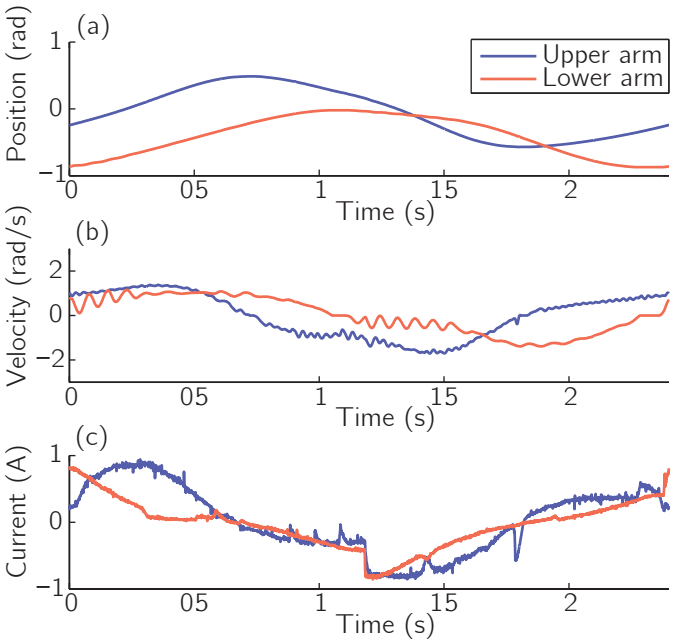


Figure 8.9: This figure shows the data of a typical motion of the hardware experiments on the two DOF robotic arm as function of time. The figure shows the positions of the joints (a), the velocities of the joints (b) and the current through the motors (c).

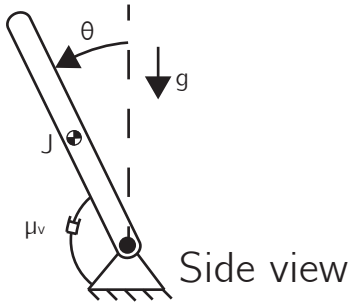


Figure 8.10: Side view of the one DOF system with linear viscous friction and gravity.

The differential equation for this system is

$$\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \omega \\ c_1 \sin(\theta) + c_2 U + c_3 \omega \end{bmatrix} \tag{8.18}$$

Where, U is the input voltage, and c_1, c_2 and c_3 are the model parameters, which we identified through a system identification and are listed in Table 8.2. In this model,

Table 8.2: The model parameters of the two DOF arm. The values are obtained through a system identification of the inverted pendulum.

Parameter	Symbol	Value	Unit
Gravitational term	c_1	112.9	s^{-2}
Motor parameters term	c_2	28.0	$V^{-1}s^{-2}$
Damping term	c_3	-1.8	s^{-1}

c_1 can be seen as the gravitational term, c_2 as the motor parameters term and c_3 as the damping term, which includes the back-emf term of the motor. All these terms have the inertia of the pendulum included.

We again used GPOPS to optimize an open loop controller such that we obtain an open loop stable motion for the task described below.

8.4.2 Task

The task we will look at is a motion with the initial and final position of the arm both at $^1/_5\pi$ (nearly upright position). Simply not moving will not result in stability, since by itself this is an unstable position. So at first, it seems impossible to find an open loop stable solution. However, this is possible when we allow the pendulum to swing to the region between $^1/_2\pi$ and $^3/_2\pi$ (around the lower equilibrium) during the motion. The goal is to find a path that is stable according to (8.8), and then minimize the integral of the square of the voltage (see eq. (8.9)). We limit the duration of the motion to 1.2 seconds.

8.4.3 Simulation results

Fig. 8.11 shows the result of the optimization, which can be explained intuitively. Since eq. (8.18) is linear in the velocity and the input, the A^*-matrix of the pendulum only depends on the position. That is, being in a state with θ between $-\pi/2$ and $\pi/2$ has a destabilizing effect, and being in a state with θ outside that range has a stabilizing effect. Loosely speaking, a stable motion requires that the stabilizing effects compensate the destabilizing ones. This means that the system should spend enough time in sufficiently stabilizing positions to counter the time it spends in the destabilizing positions. In Fig. 8.11, we see that the pendulum moves directly from the destabilizing initial position to stabilizing positions, where it spends most of the time before moving back at the end of the motion. We use the condition in eq. (8.8) to evaluate if the stabilizing effects are compensating the destabilizing effects.

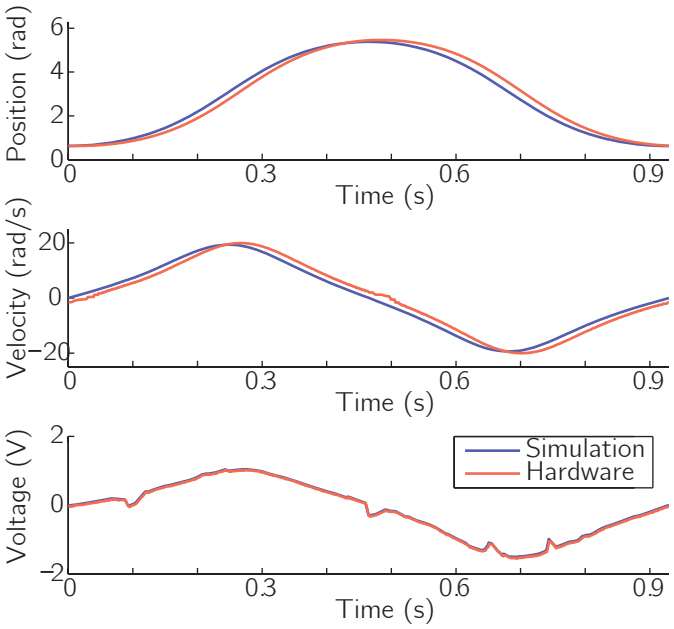


Figure 8.11: This figure shows the data for the single pendulum in the vertical plane as function of time. The figure shows the position of the joint (a), the velocity of the joint (b), and the torque about the joint (c) for both the simulation as the hardware experiments.

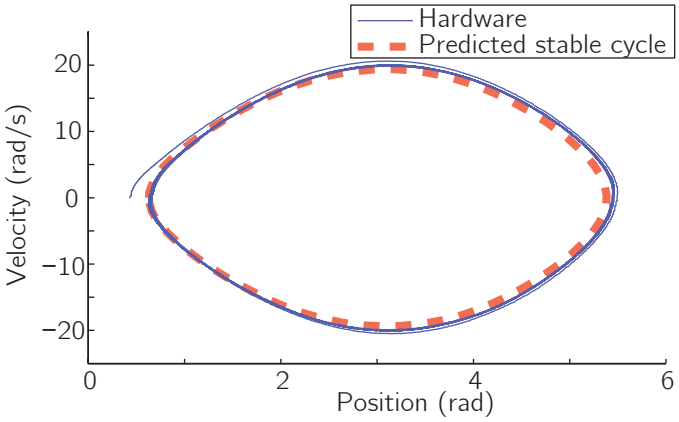


Figure 8.12: This figure shows the state space trajectory for the single pendulum in the vertical plane. The plot shows the stable cycle as determined in simulation (thick dotted line) and a motion on the hardware setup that starts from a perturbed position and converges to the stable cycle (thin line).

8.4.4 Hardware results

Fig. 8.12 shows the results of applying the input signal found in simulation on the hardware setup. It shows a motion being initialized in a perturbed position, after which it converges towards a cycle close to the one predicted by simulation. Fig. 8.11 shows that the motion over time after convergence is the same in simulation and hardware experiments. This shows that when an accurate model is available, the method we used finds a controller that performs the desired task on the real system. Extending the method to allow task performance even when accurate models are not readily available is an important next step in the research on open loop stable manipulation.

8.5 Discussion

8.5.1 Model mismatch

For the two DOF robotic arm, the results from simulation (see Fig. 8.4, 8.5 and 8.6) are clearly different from those of the hardware experiments (see Fig. 8.7, 8.8 and 8.9). However, both simulation and hardware experiments show convergence to a stable cycle. These results show that the specific cycle the system converges to is sensitive to unmodeled behavior. For the inverted pendulum, we showed the results from simulation and hardware experiments are the same. This is due to the fact that the inverted pendulum is easier to model accurately.

There are three possible ways to reduce errors due to unmodeled behavior and thereby improve the control performance. First, the model of the system can be extended to include more of the unmodeled dynamics. Since modeling inaccuracies will always exist, a second approach would be to include the sensitivity to modeling inaccuracies in the optimization as in [174]. However, since sensitivity to modeling inaccuracies cannot always be reduced [174], we expect the best results from the third approach: tuning or learning open loop stable cycles online.

8.5.2 Basin of Attraction

In this chapter we focused on the calculation of open loop stability, which is a minimal requirement for making open loop manipulation work, but it gives little information about the rejection of realistic (i.e. finite) disturbances. Therefore, we simulated the arm using a grid of initial positions in order to see if they converge to the intended stable cycle. Fig. 8.13 shows this data, which is a slice of the 4D basin of attraction. This shows that the majority of initial positions (within the mechanical limits of

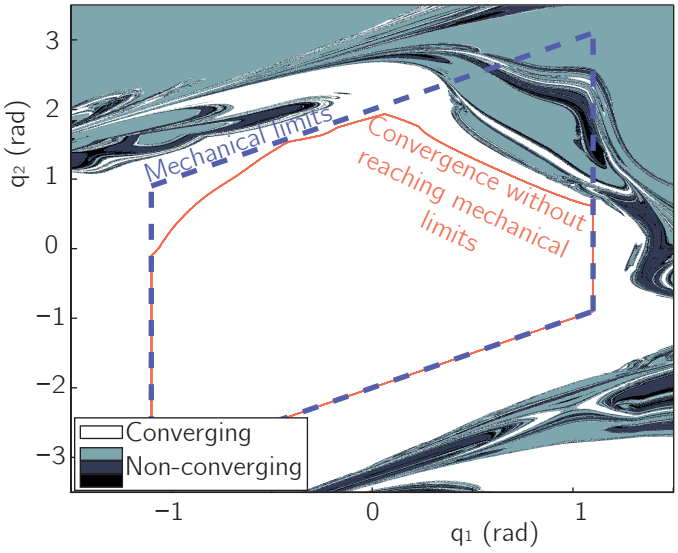


Figure 8.13: This figure shows the basin of attraction of the open loop stable cycle for different initial positions. The white region depicts all initial positions that result in convergence to the cycle. The blue dotted line depicts the mechanical limits of the robotic arm. All initial positions within the red line result in convergence while staying within the mechanical limits. The non-converging initial positions are divided into three groups. First, light blue depicts convergence to the intended cycle with an error in θ_2 of 2π rad; second, dark blue depicts an error in θ_2 of 4π rad and third, black depicts an error in θ_2 of 6π rad or more. So all initial positions converge to the same cycle, although in some cycles, the second joint has rotated for exactly one or multiple revolutions.

the arm) result in convergence to the cycle. This result means that precise initial positioning of the arm is not required for converging to the intended cycle.

Interestingly, the basin of attraction analysis shows that all initial positions converge to the same cycle, although in some cycles, the second joint has rotated for exactly one or multiple revolutions. While implementing several feedforward controllers on the robotic arm, we found that most feedforward controllers (including the one shown in this chapter) result in two stable cycles: one with negative θ_2 and one with positive θ_2. We expect that this is caused by a bending of the arm due to gravity, which is larger around $\theta_2 = \pm\frac{\pi}{2}$ rad than around $\theta_2 = 0$ rad. We suspect that this difference between the model and the arm is the main cause for the mismatch of the cycles in simulation and in hardware experiments. In order to improve the prediction of the cycle by the model, a stiffer arm can be used or this bending can be modeled. A second cause of the model mismatch could be the friction, which is hard to model in general.

8.5.3 Implications

The idea of using open loop stability of periodic motions to analyze repetitive manipulation tasks is a new approach in robot manipulator control. It allows for robotic arms to be controlled when feedback is too slow (e.g. using camera feedback or control over great distance), too imprecise (e.g. due to cheap, noisy sensors [112]) or even impossible (e.g. in micro-scale applications or due to radiation), or when the input is limited and planning is required.

There is no fundamental reason why stable cycles would not exist in robotic arms with more DOFs, or for more complex tasks (e.g. obstacles or interactions with the environment). However, we expect that it will be more difficult to find such cycles and maybe impossible to find cycles that include the pick- and place positions. Finding trajectories can be made easier by tuning the dynamics of the system, such that all trajectories are stable, i.e. the system is contractive. Such tuning could be done by adding springs on all joints or changing the mass distribution.

8.5.4 Applicability

The concept of open loop stable manipulation as presented in this chapter is not fully applicable yet, since it consists of finding one stable cycle. In practice, tasks consist of moving between multiple positions in a certain (not necessarily predefined) order. Since the computations are too complex to perform online, we see two approaches to make open loop stable manipulation fully applicable in the future.

The first approach is to move in a stable periodic motion which covers most of the task space. Whenever necessary, the controller can make open loop controlled deviations from this cycle to the specified positions after which the manipulator returns to the main cycle.

The second approach is to divide the space into a grid and build a library of feedforward controllers for moving between the grid positions. The controller can then create various stable cycles by combining multiple trajectories from this library into a stable cycle that tracks the specified positions.

8.6 Conclusions

In this chapter we introduced an approach to control robotic arms, without the need for state feedback. We conclude that this approach is promising for robotic arms that perform repetitive position tasks without feedback. Small disturbances on the state of the system will decay over time and the robotic arm will asymptotically return to its original trajectory. Both simulation and hardware experiments show convergence

to an open loop stable cycle, but the cycles they converge to are not the same, probably caused by a bending of the second DOF due to gravity. We expect that this problem can be solved by online learning of the stable cycles.

Acknowledgement

This work is part of the research programme STW, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO).

9

Learning robustly stable open-loop motions for robotic manipulation

Wouter Wolfslag*, Michiel Plooij*, Robert Babuška and Martijn Wisse
* These authors contributed equally to this chapter,
Robotics and Autonomous Systems
Vol. 66 (April 2015), pp. 27-34.

Abstract

Robotic arms have been shown to be able to perform cyclic tasks with an open-loop stable controller. However, model errors make it hard to predict in simulation what cycle the real arm will perform. This makes it difficult to accurately perform pick and place tasks using an open-loop stable controller. This chapter presents an approach to make open-loop controllers follow the desired cycles more accurately. First, we check if the desired cycle is robustly open-loop stable, meaning that it is stable even when the model is not accurate. A novel robustness test using linear matrix inequalities is introduced for this purpose. Second, using repetitive control we learn the open loop controller that tracks the desired cycle. Hardware experiments show that using this method, the accuracy of the task execution is improved to a precision of 2.5cm, which suffices for many pick and place tasks.

9.1 Introduction

This research aims at future applications where sensing and feedback are undesirable due to costs or weight; or difficult due to small scale, radiation in the environment or frequent sensor faults. A recent example of such an application is the control of a swarm of nano-scale medical robots [19]. These applications inspire us to investigate an extreme case of feedback limitations: solely open-loop control on robotic arms. Control without any feedback can only be effective if two key problems are addressed: disturbances (e.g. noise and perturbations) and model inaccuracies.

The first problem, handling disturbances on an open-loop controlled robot, has mainly been addressed by creating open-loop stable cycles. The best known examples of this are passive dynamic walkers, as introduced by McGeer in 1990 [134]. Since those walkers do not have any actuators, there is no computer feedback control. The walking cycle of those walkers is stable, which means that small perturbations will decay over time. Such stable cyclic motions are called limit cycles. Limit cycle theory was later used to perform stable walking motions with active walkers, of which the closest related work is that by Mombaur et al. [139, 141]. They optimized open-loop controllers for both stability and energy consumption and performed stable walking and running motions with those robots. Open-loop stable motions have also been used before to perform tasks with robotic arms. In 1993, Schaal and Atkeson showed open loop stable juggling with a robotic arm [189]. Even though their controller had no information about the position of the ball, they showed that any perturbation in this position decays over time, as long as a specific path of the robotic arm itself

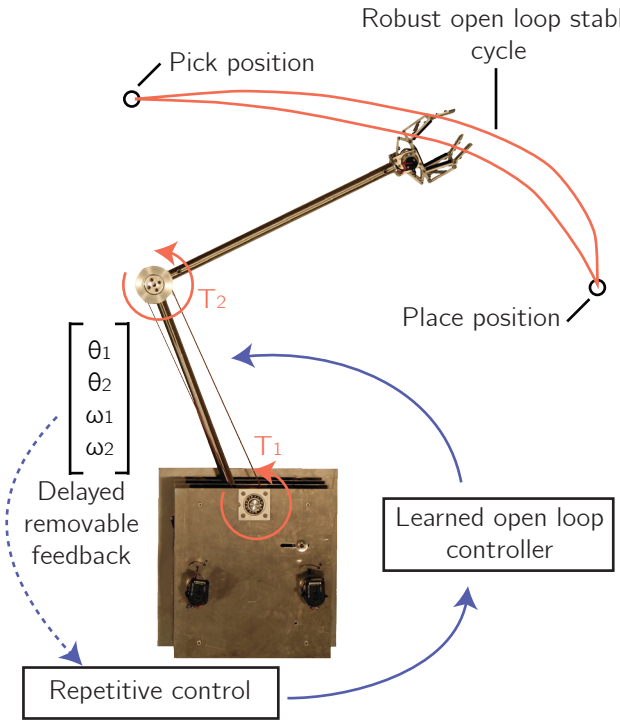


Figure 9.1: This figure shows the top view of the concept of robust open-loop stable manipulation. We first optimize a cycle that stands still at the pick and place positions for open-loop stability. Next, we check the cycle for robustness to model uncertainty. Then, using repetitive control on the robotic arm, we learn an open-loop controller that tracks the cycle. After the learning, the open-loop controller performs the task without any feedback.

can be tracked. In a recent study, we showed that it is possible to perform repetitive tasks on a robotic arm with solely an open-loop current controller [172].

The second key problem with feedforward control (i.e. model inaccuracies) prevents the approach in [172] to be fully applicable: it causes a difference between the motion as planned in simulation and as performed in hardware experiments. Handling model inaccuracies on open-loop controlled robots has recently become the subject of research. Singhose, Seering and Singer [203, 204] researched vibration reducing input shaping of open-loop controllers while being robust to uncertainty in the natural frequency and damping of the system. Becker and Bretl [18] researched the effect of an inaccurate wheel diameter of unicycles on the performance of their open-loop velocity controller. In their case, open-loop control means that the position of the unicycle is not used as an input for the controller, but the velocity of the wheels is.

In a previous paper, we showed that on a robotic arm, different open-loop current controllers have different sensitivities to model inaccuracies [174]. We found open-loop current controllers of which the end position of the motion is independent of the friction parameters. However, the motions that handle the model inaccuracy problem of feedforward control, the stability problem still exists, i.e., disturbances acting on these motions will grow over time.

Since these two problems of disturbances and model inaccuracies in open-loop control have only been addressed separately, no applicable purely open-loop control scheme has been devised. This chapter shows that repetitive tasks can be performed stably by robotic arms with an open-loop voltage controller, even when an accurate model is not available.

In order to achieve this goal, the problem is split into two phases (see Fig. 9.1). In the first phase the robustness of the system is analyzed with a novel method based on linear matrix inequalities (LMI)[190]. In the second phase repetitive control (RC)[123] is used to learn the exact control input, such that the desired positions are reached accurately. During this learning phase, very slow feedback is allowed, this feedback can be removed after the learning has been completed.

The rest of this chapter is structured as follows. Section 9.2 shows why the problem can be split into two phases, and explains the robustness analysis method and the repetitive control scheme. Next, Section 9.3 shows the experimental setup we used to test our approach. Then, Section 9.4 shows the results of both the numerical and the hardware experiments. Finally, the chapter ends with a discussion in Section 9.5 and a conclusion in Section 9.6.

9.2 Methods

In this section we explain our methods. First, in Section 9.2.1 we discuss the basic concept of the stability analysis. Second, Section 9.2.2 explains our approach to perform robustly stable open-loop cycles. Then we will describe the two steps of this approach separately: a robust stability analysis (Section 9.2.3) and learning of an open-loop controller (Section 9.2.4).

9.2.1 Open-loop stable manipulation

A system described by the differential equation $\dot{x} = f(x, u)$ can be linearized along a trajectory x^* caused by input $u^*(t)$:

$$\frac{d\bar{x}}{dt} = A^*(t)\bar{x} \tag{9.1}$$

with

$$A^*(t) = \left.\frac{\partial f}{\partial x}\right|_{x^*(t),u^*(t)} \tag{9.2}$$

where $\bar{x}(t) = x(t) - x(t)^*$ is the state error. For the ease of notation, the time dependency of variables is occasionally dropped if it is unambiguous to do so. For example $A^*(t)$ will be written as A^*.

If both trajectory and input are cyclic with period t_f, stability can be assessed by by discretizing the system using a time step t_f. To be able to draw upon the research in stability of limit cycles, note that such discretization is the same as a Poincaré map of the system with the time appended to the state vector. The Poincaré section is then taken as $t = t_f$, and the time is reset to 0 after crossing this section. Previously (notably in [141] and [172]), verifying stability was done using the eigenvalues of the linearized discrete system. But that approach does not allow incorporating model uncertainty in the stability analysis.

To obtain a method that does allow uncertain models, we use a quadratic Lyapunov function, $J = \bar{x}^T M(t)\bar{x}$, with positive definite $M(t)$. The idea is that for a stable system, an $M(t)$ can be found such that the norm J is always decreasing over time. For cyclic systems this means the following two constraints should be satisfied (cf. [217]):

$$M(t)A(t)^* + \dot{M}(t) + A(t)^{*^T}M(t) \prec 0, \quad \forall t \in [t_0, t_f] \tag{C1}$$

$$M(t_f) - M(t_0) \succ 0 \tag{C2}$$

where $\prec$ and $\succ$ are used to indicate negative/positive definiteness respectively, $\dot{M}$ is the time derivative of M and the subscripts 0 and f denote initial/final time. The first of these constraints ensures that the Lyapunov function is decreasing at each time instant. The second constraint makes sure that it becomes stricter after each cycle, i.e., that having the same error ($\bar{x}$) as a cycle before means that the Lyapunov function has increased. Note that only one of the two inequalities needs to be strict in order for stability to hold.

When there are model inaccuracies, two changes occur that make the above conditions invalid. Firstly, $A^*(x^*(t))$ is no longer accurate when in state $x^*(t)$. Secondly, when using a fixed open-loop controller on an uncertain system, the trajectory is not fully predictable, so in general $x(t) \neq x^*(t)$, when using the input $u^*(t)$. In the next section we will outline our approach to solve these two issues.

9.2.2 Robust open-loop approach

To find motions that are open-loop stable even when the model is not accurately known, we will focus on input affine systems with constant input matrix, i.e., systems that are described by the following differential equation:

$$\dot{x}(t) = f(x(t)) + Bu(t) \tag{9.3}$$

where B is a constant matrix. The equations of motion of serial chain robots can be written in this form, by considering phase-space rather than state space (i.e. using momenta rather than velocities). Systems where the control input enters the system via a constant matrix have the advantage that the linearized error dynamics do not depend on u (cf. Eq. (9.2)):

$$A^*(t) = \left.\frac{\partial f}{\partial x}\right|_{x^*(t)} \tag{9.4}$$

Because the local stability of the motion only depends on the linearized error dynamics, the stability for motions of such a system only depends on the states of the motion, and not on the inputs used. This is the key insight that allows robust open-loop control, by splitting the problem into two stages:

1. Finding a trajectory through the phase-space that is stable, even if the (linearized) system dynamics differ from their nominal value by some uncertain amount.

2. Learning the inputs for that trajectory. When done online, this learning can take into account the uncertain dynamics.

With this two stage approach, an open-loop controller is found that accurately controls the robot in a way that is both accurate and stable. These two properties hold even when facing modeling errors, which is important for hardware implementation.

There are two important remarks to be made about this approach. Firstly, the robust trajectory found should be a valid trajectory for the physical system, and thus for all realizations of the uncertainty in the model. That means there should exist an input for which the trajectory given is a solution to the differential equation. In our case this means the inertia matrix has to be known accurately in order to translate momenta into the velocities that correspond to the time derivatives of the positions in the planned trajectories. Secondly, for the learning step some feedback is required, which means the robot is no longer purely open-loop controlled. However, such

feedback can be delayed and can be removed once the feedforward signal is known. This allows opportunities for external sensing, such as cameras based feedback during the learning phase.

In Section 9.2.3 the methods used to find a robust trajectory will be explained. Section 9.2.4 explains how Repetitive Control is used to learn the open-loop controller.

9.2.3 Finding robustly stable trajectories

The first step in our approach is to find trajectories that are stable. To do so, we use an optimization approach, further explained in Section 9.3. Then we test if the resulting trajectory is robustly stable. In this section we derive a novel robustness test, which is summarized in Algorithm 1.

For this section, we assume that we have a known trajectory for which we want to determine how robust it is. We model the uncertainty of the system through an integer number of uncertain parameters δ_j, that enter the linearization affinely using known state and time dependent matrices $\Delta_j(t)$, i.e.

$$A(t) = \left.\frac{\partial f}{\partial x}\right|_{x=x^*(t)} + \sum_j \Delta_j(t)\delta_j \tag{9.5}$$

where we will take $A^*(t) = \left.\frac{\partial f}{\partial x}\right|_{x^*(t)}$ as the certain part of the dynamics, and $\Delta(t) = \sum_j \Delta_j(t)\delta_j$ as the uncertain part. Here the Δ_j are known matrices describing how the uncertain parameters δ_j enter the system dynamics. As a result, we have a time varying uncertain system $\dot{\bar{x}} = A(t,\Delta)\bar{x} = (A^*(t) + \Delta(t))\bar{x}$.

From theory of Linear Matrix Inequalities [190, Prop. 5.3], we know that if each δ_j is constrained to some interval $\delta_j \in [\underline{\delta}_j, \bar{\delta}_j]$, then the constraint C1 holds for all $\Delta(t)$, if it holds for the $\Delta(t)$ created by the vertices of the hypercube of allowed δ_j. The $\Delta(t)$ values connected to these vertices will be called $\Delta_0(t)$, which is a finite set. The choice for $\Delta_0(t)$ depends on the expected model inaccuracies. The robust stability constraint now becomes:

$$MA^* + \dot{M} + A^{*T}M + \Delta_0^T M + M\Delta_0 \prec 0 \tag{C3}$$

Note that this constraint is a sufficient, but not necessary condition for robust stability. Making M a function of Δ would reduce the conservativeness [190]. However, in implementation this would also result in much greater complexity and computation time, so we have elected not to do so.

Furthermore, constraint C3 determines whether the trajectory is robustly stable or not, but it does not give a continuous measure of robustness. To add this measure, we try to find the maximal constant ϵ_x, such that the following constraint holds:

$$MA^* + \dot{M} + A^{*T}M + \epsilon_x \Delta_0^T M + \epsilon_x M \Delta_0 \prec 0 \qquad \text{C4}$$

The value of ϵ_x is the robustness measure.

What is left is a way to find an $M(t)$ that satisfies the constraints. Before describing our approach, we will briefly discuss two approaches that involve optimizing a paramet rized $M(t)$, and are unsuitable in this case, but at first sight might seem applicable.

The first of these approaches is the most straightforward conceptually and has been used in literature [35, 125, 217]. The idea in that research is to parameterize $M(t)$, and then optimize these parameters. The literature referred to establishes that this parameter optimization can be cast as a convex problem by using Sum Of Squares programming. The disadvantage of this approach for our problem is that the optimization of $M(t)$ requires many (> 1000) decision variables and is therefore computationally expensive and sensitive to numerical errors.

The second approach is to parameterize $M(0)$ only, and rework equation C1 or C3 into a differential equation, which can be integrated to find $M(t)$. The basic example would be

$$\dot{M} = -\epsilon_s I - A^{*T}M - MA^* \qquad (9.6)$$

with ϵ_s a small positive constant. This allows to optimize $M(0)$, such that $M(t_f)$ found using integration satisfies equation C2. This approach of integrating $\dot{M}$ could potentially be extended to incorporate the uncertain dynamics into the differential equations, for instance by viewing M as an ellipsoid, and using a minimum volume ellipsoid covering algorithm. However, even then the method of searching for $M(0)$ and using forward integration is troublesome. The number of parameters is greatly reduced compared to approaches where $M(t)$ as a whole is approximated and optimized, but at the cost of losing convexity. This greatly increases the computation time and introduces the risk of finding local minima.

Since these two approaches that involve searching for $M(t)$ do not work, we propose a method which does not involve such optimization, but rather immediately computes a (suboptimal) $M(t)$. This method is based on the fact that all time varying systems with periodic coefficients are reducible [63, Sec. XIV.3], i.e. can be transformed into a time invariant system. The idea is to find a Lyapunov function for the certain part of this time invariant system and then transform this Lyapunov function back to the time variant system for the robustness check.

The transformation to the time invariant system is based on the state transition matrix of the nominal system $\bar{x}(t) = \Phi(t)\bar{x}(0)$, which can be found by integrating the following initial value problem:

$$\dot{\Phi}(t) = A^*(t)\Phi(t), \quad \Phi(0) = I \tag{9.7}$$

Now if we define $L(t) = \Phi(t)\Phi(t_f)^{-\frac{t}{t_f}}$ and the transformation $\bar{x} = L(t)y$, we get for the state transition matrix of y:

$$\dot{y} = \frac{1}{t_f}\ln(\Phi(t_f))y \tag{9.8}$$

which is a time invariant, but possibly complex-valued system [63, Sec. XIV.1]. For this system we can then find a Lyapunov function $y^T M_y y$ by solving for the positive definite matrix M_y using standard LMI techniques. In principle it would be possible to incorporate a worst-case transformed $\Delta(t)$ in that equation, but for simplicity we optimize to find a Lyapunov function with the smallest time derivative, i.e. maximize ϵ_y in:

$$\frac{1}{t_f}\ln(\Phi(t_f))^T M_y + M_y\frac{1}{t_f}\ln(\Phi(t_f)) + \epsilon_y I \prec 0 \tag{9.9}$$

The resulting M_y is then transformed back to $\bar{x}$ coordinates:

$$M(t) = L(t)^{-T} M_y L(t)^{-1} \tag{9.10}$$

Finally, the maximum ϵ_x that satisfies constraint C4 at all times is found using a numerical LMI solver. The time derivative of M that is needed for this step is readily determined analytically. We omit these expressions because they are too lengthy.

As Eq. (9.7) can only be solved numerically, a natural time sampling approach is used. That is, the constraint C4 is only tested at the sampling times that are used by the solver that integrates Eq (9.7). In this case, we used the MATLAB ode45 solver. This time sampling is inspired by [217], which discusses the correctness of such a sampling procedure for a similar robustness test.

In many cases one or more of the singular values of the state transition matrix $\Phi(t_f)$ are nearly zero [183]. This means that some errors are reduced to nearly zero after one cycle. This can happen for instance with a high damping constant, which could arise from using voltage control.

Such a nearly singular state transition matrix makes the numerics of the above scheme ill-conditioned. Furthermore, the resulting LMI-test is very sensitive to small changes

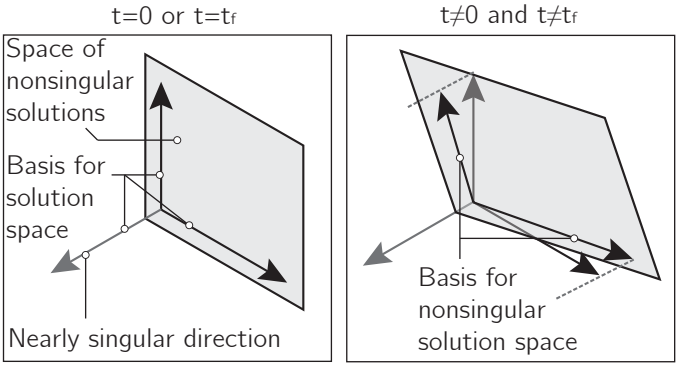


Figure 9.2: Explanation of the transformations required for model reduction. At time $t = 0$ or $t = t_f$, the singular value decomposition can be used to find a basis for the solutions space. In the left figure it can be seen that this basis has the advantage that the singular directions are clearly separated. The right figure shows that by using the same basis at other times, the non-singular dimensions are described using more basis vectors than are required to span the space. The solution, also indicated in the right figure is to use a projection, in our case by computing the column echelon form. This particular projection ensures the basis is periodic, which allows the robustness analysis to be performed.

in the nearly singular directions. This numerical sensitivity is in contrast with the fact that in many cases, such near singularity is some intrinsic property of the system or trajectory, and is not influenced by some small changes in model parameters. In those cases the above scheme is sensitive to errors that are irrelevant. Therefore we disregard these near-singular directions by assuming the error in those directions is always 0.

Since the system is time varying, the relevant directions are also time varying. So what is needed is to find a time varying basis for the non-singular space of the error dynamics, see Fig. 9.2. The robustness test described previously requires a periodically time varying system, therefore the basis for the non-singular dimension should also be periodically time varying.

First, to disregard the near singular directions, use the singular value decomposition of the state transition matrix at time t_f: $\Phi(t_f) = U\Sigma V^T$. Define U_χ as the columns of U that correspond to the singular values that are not nearly 0. These columns are an orthonormal basis for the not-nearly-singular space of $\Phi(t_f)$. In particular, define $\tilde{x}(t)$ coordinates by the transformation $\bar{x} = U\tilde{x}$. Then

$$\tilde{\Phi}(t) = U^{-1}\Phi(t)[U_\chi, 0] \tag{9.11}$$

is the state transition matrix for $\tilde{x}$, where the errors in the singular dimensions are

immediately set to zero (by the multiplication with $[U_\chi, 0]$). The columns of $U\tilde{\Phi}(t)$ now form a time varying basis for the non-singular space of the $\bar{x}$-dynamics. However, this basis is not periodic yet.

This periodicity can be enforced by using the following transformation, which defines $\hat{x}(t)$:

$$\bar{x}(t) = U\text{rcef}\big(\tilde{\Phi}(t)\big)\,\hat{x}(t) \tag{9.12}$$

where rcef signifies the reduced column echelon form. To see that it works, it is necessary to realize that after exactly one cycle, the columns of $\tilde{\Phi}$ span the same space. Furthermore, $\tilde{\Phi}(0)$ is simply the identity matrix with the last diagonal entries turned into 0. Therefore $\tilde{\Phi}(t_f)$ also spans this same space, and the reduced column echelon form then finds the correct basis: the identity matrix with the last entries turned into 0. As a result $\text{rcef}(\tilde{\Phi}(0)) = \text{rcef}(\tilde{\Phi}(t_f))$. This means that the transformation in Eq. 9.12 is periodic, and can therefore be used in the robustness test. The procedure of the robustness test, including this transformation is denoted in Algorithm 1, in which each step is described and the relevant equations for that step are then referred to.

This leaves only two remarks on the implementation of this transformation. First, the reduced column echelon form of a smoothly varying matrix is smoothly varying, meaning the time derivative of the transformation exists. To speed up computation we use finite differences to compute this derivative. Second, the transformation as defined above does not actually reduce the number of states in the computation, but rather sets the states to zero. By substituting U_χ for $[U_\chi, 0]$ in Eq. 9.11, the size of the state vector is reduced, which again speeds up computation.

9.2.4 Repetitive control

The stability of the trajectories computed by the optimization is independent of the input and finite modeling errors. The next step is to learn the input that tracks the trajectory on the robotic arm. When the trajectory is learned, the feedback is disconnected and the task is performed stably with the learned open-loop controller.

There are two (similar) learning algorithms which are commonly used for learning of repetitive motions: Iterative Learning Control (ILC) and Repetitive Control (RC) [123]. Both algorithms use the input and error from the previous iteration (cycle) to compute the input in the current iteration. The only difference between the two is that in ILC every iteration starts at the same state, where in RC every iteration starts at the final state of the previous iteration. We use RC, because it allows us to immediately start a new iteration at the end of every movement cycle, whereas ILC would require a reinitialization after every cycle to start the new iteration, which

Algorithm 1 Robustness test

Compute state transition matrix $\Phi(t_f)$, Eq. (9.7)
if nearlySingular($\Phi(t_f)$) **then**
    Reduce system dimension, Eqs. (9.11)-(9.12)
end if
Make system time invariant by transformation, Eq. (9.8)
Find M_y, the Lyapunov function for that system, Eq. (9.9)
Initialize robustness measure, $\epsilon_x \leftarrow 1000$
for all t in time discretization **do**
    Find $M(t)$, Lyapunov function of first system, Eq. (9.10)
    Numerically find $\dot{M}(t)$ as required in constraint C4.
    $\bar{\epsilon} \leftarrow$ maximum ϵ_x that satisfies constraint C4 at time t
    $\epsilon_x \leftarrow \min(\bar{\epsilon},\epsilon_x)$
end for

would take time and require fast and precise state feedback. The repetitive control runs on a real time target with a sampling frequency of 500 Hz. Therefore, we will use discrete time notation with time step k.

The repetitive control scheme we used is

$$u(k) = u(k-p) + \alpha(k) \cdot (\Delta u_P(k) + \Delta u_D(k)) \tag{9.13}$$

with

$$\Delta u_P(k) = \sum_{i=0}^{p-1} \phi(i) \cdot P \cdot e(k-p+i) \tag{9.14}$$

$$\Delta u_D(k) = \sum_{i=0}^{p-1} \sigma(i) \cdot D \cdot (e(k-p+i) - \bar{x}(k-2p+i)) \tag{9.15}$$

$$\bar{x}(k) = x^*(k) - x(k) \tag{9.16}$$

where $u(k)$ is the control input, p is the number of time steps in one iteration, $\bar{x}(k)$ is the error, x^* is the desired state and x is the actual state. $\alpha(k)$, $\phi(i)$, $\sigma(i)$, P and D are explained below.

The learning rate $\alpha(k)$ is a function of time and can vary between 0 and 1. A varying $\alpha(k)$ reduces the chance of converging to a sub-optimal solution.

The filter gains $\phi(i)$ and $\sigma(i)$, for $i = 1, \ldots, p$, determine how much the different errors in the previous iterations contribute to the change in the input signal. The

filtering accounts for measurement noise and prevents oscillations in the RC. The sum of the elements of $\phi(i)$ and the sum of the elements of $\sigma(i)$ are both equal to 1, to obtain a (weighted) moving average filter.

The learning gains P and D are $N_u \times N_s$ matrices (with N_u the number of inputs and N_s the number of states). The P and D we used have the following structures:

$$P = \begin{bmatrix} P_{11} & 0 & P_{13} & 0 \\ 0 & P_{22} & 0 & P_{24} \end{bmatrix} \tag{9.17}$$

$$D = \begin{bmatrix} D_{11} & 0 & D_{13} & 0 \\ 0 & D_{22} & 0 & D_{24} \end{bmatrix} \tag{9.18}$$

Using this structure, errors in the state of one joint have no direct influence on the control signal in another joint (i.e. there are no cross terms). The Δu_P term can be seen as the proportional term of the RC, since it leads to a change in u proportional to the errors in the previous iteration. The Δu_D term can be seen as the damping term of the RC between two iterations, since it leads to a change in u proportional to the change in errors between two iterations.

9.3 Experimental setup

We tested our approach on a two DOF SCARA type arm. This type of arm can perform industrially relevant tasks with a simple mechanical design. In the experiment, the arm has to perform a rest to rest motion, typical for a pick and place task. In this section the hardware setup and task description will be addressed, along with parameters in our approach that were set specifically for the experiments. Note however that the general approach does not depend on the exact values of the parameters, which were manually tuned.

9.3.1 Hardware setup

Fig. 9.1 shows a picture of the experimental two DOF robotic arm [171]. The arm consists of two 18x1.5mm stainless steel tubes, connected with two revolute joints, with a spring on the first joint. A gripper is connected to the end of the second tube. The motors are placed on a housing and AT3-gen III 16mm timing belts are used to transfer torques within the housing. The joints are actuated by Maxon 60W RE30 motors with gearbox ratios of respectively 66:1 and 18:1. The timing belts provide additional transfer ratios of 5:4 on both joints. The parameters of this robotic arm are listed in Table 9.1.

Table 9.1: The model parameters of the two DOF arm.

Parameter	Symbol	Value	Unit
Damping	μ_{v1}, μ_{v2}	7.48, 0.56	Nms/rad
Inertia	J_1, J_2	0.0233, 0.0871	kgm^2
Mass	m_1, m_2	0.809, 1.599	kg
Length	l_1, l_2	0.410, 0.450	m
Position of COM	l_{g1}, l_{g2}	0.070, 0.325	m
Motor constant	k_{t1}, k_{t2}	25.9, 25.9	mNm/A
Gearbox ratio	g_1, g_2	82.5:1, 22.5:1	rad/rad
Spring stiffness	k_1	1.6	Nm/rad

The large damping terms are caused by the back-emf of the motors (since we use voltage control and not current control). The viscous and Coulomb friction are neglected in this chapter since they are small compared to the back-emf induced damping. The equations of motion and the transformation to momenta can be derived using standard methods [121, Chap. 3], and are omitted from this chapter because they are too long. Because the second joint is connected to its motor via a parallelogram mechanism (see [171]), the angle of the second arm is taken as the absolute angle, i.e., relative to the world frame.

9.3.2 Task description

We let the manipulator perform a cyclic pick and place motion, with pick and place positions at [0.4, 0.1] rad and [-0.2, -1.2] rad respectively. At the pick and place position, the arm has to stand still for 0.2s, which would be required to pick or place an object. The time to move between the pick and the place position is 1.4s. Hence, the total time of one cycle is 3.2s.

9.3.3 Optimization

To find a robustly stable trajectory an optimization is used. First, the trajectory is optimized for fast convergence, by minimizing the maximal eigenvalue modulus of the linearized Poincaré map, see the stability measure in [172]. The input was taken as a piecewise linear input, consisting of 20 pieces, with an absolute maximum value of 5V. Then, the robustness of the resulting trajectory is verified using the approach outlined in Section 9.2. The uncertain dynamics $\Delta(t)$ consist of two uncertainties: uncertainty on the linearized stiffness on the first and second joint. The bounds on these uncertainties are constant and taken as ± 0.3 and ± 1.0 Nm/rad respectively.

Table 9.2: The parameters used in optimization and RC

Symbol	Value	Symbol	Value
$t_{RCfinal}$	300 s		
P_{11}	1.5 V/rad	D_{11}	0 V/rad
P_{22}	0.6 V/rad	D_{22}	0.4 V/rad
P_{13}	0.3 Vs/rad	D_{13}	0 Vs/rad
P_{24}	0.4 Vs/rad	D_{24}	0 Vs/rad

9.3.4 Repetitive control parameters

We let the robotic arm learn the cycle from simulation while $t < t_{RCfinal}$. During this learning period, the learning gain decreased linearly from $\alpha = 1$ at $t = 0$ s to $\alpha = 0$ at $t = t_{RCfinal}$ s.

The filter gains we used are equal to

$$\phi(i) = \sigma(i) = \frac{-(i-1)(i-50)}{|-(i-1)(i-50)|} \qquad \forall i \leq 50 \tag{9.19}$$

$$\phi(i) = \sigma(i) = 0 \qquad \forall i > 50 \tag{9.20}$$

The result of using these filter gains is that the change in input signal depends on errors in steps $k-p$ to $k-p+50$, see Eq. 9.13. Therefore the RC scheme is non-causal. This filter has two purposes. First, taking a weighed average over multiple measurements reduces the influence of sensor noise. Second, the non-causality takes into account that the input relates to the acceleration, which means it takes some time to have a measurable effect on the state, which consists of the positions and the velocities.

The remaining parameters are shown in Table 9.2. All PD and filter gains we used in the repetitive control algorithm are based on experience rather than on extensive calculations. Also, note that in this chapter we do not not prove that the repetitive controller is stable. Experience on the robot shows that the repetitive controller is stable as long as the gains P_{ij} and D_{ij} are not too high.

9.4 Results

This section presents the results from both the optimization in simulation and hardware experiments. First, using the LMI based robustness analysis, we optimize the

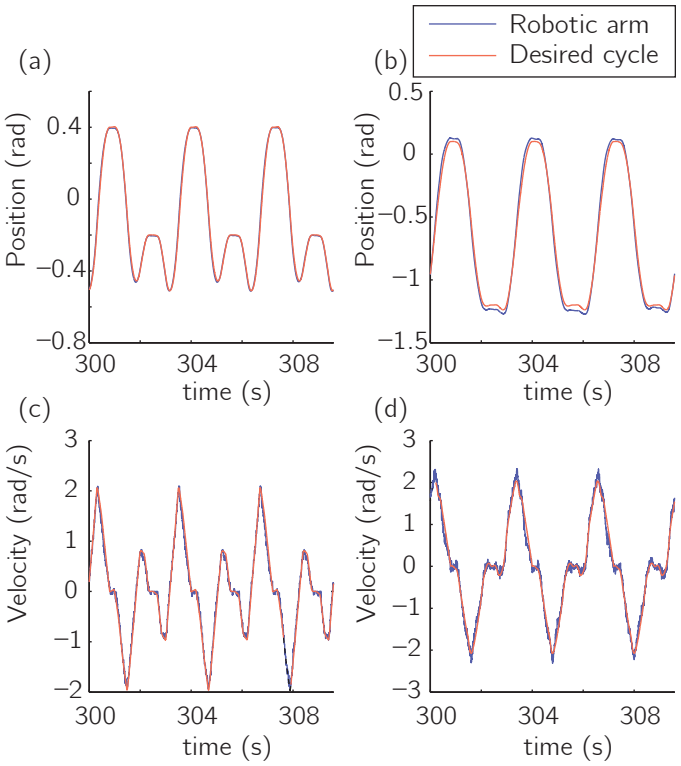


Figure 9.3: The cycle of the robotic arm after learning. The red line shows the desired cycle that was obtained from simulation. The blue solid line shows the cycle of the robotic arm after 300s of learning. (a) The position of the first joint, (b) the position of the second joint, (c) the velocity of the first joint and (d) the velocity of the second joint.

trajectory for both stability and robustness. Second, we learn the open-loop controller that tracks this trajectory. When the controller is learned, the feedback is disconnected and the robotic arms can perform its task with solely open-loop control. The video accompanying this chapter shows the hardware experiments.

9.4.1 Simulation results

The red lines in Figs. 9.3a-d show the cycle obtained from optimization. The cycle has a maximal eigenvalue modulus of 0.70 and a robustness value of ϵ_x= 0.038. These results are interpreted in Section 9.5.2.

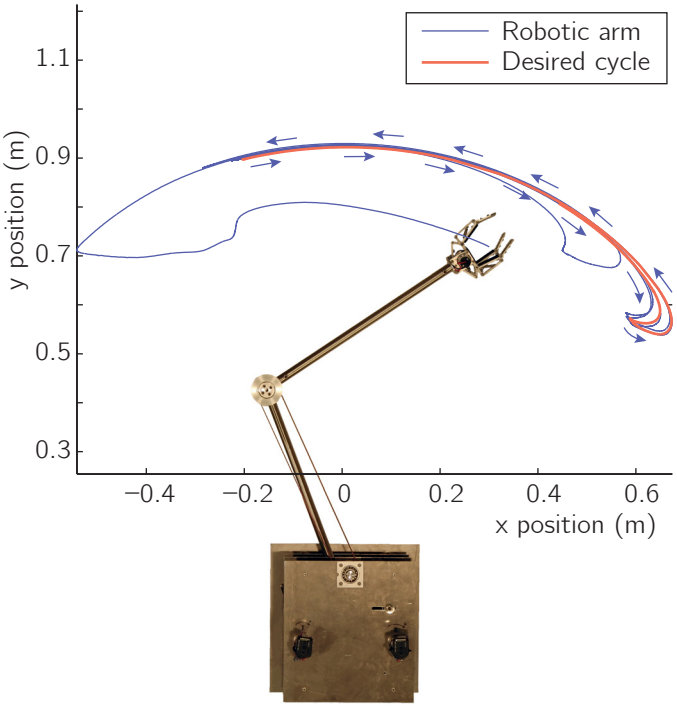


Figure 9.4: This figure shows the motion of the arm in workspace. The red line shows the desired cycle that was obtained from optimization. The blue solid line shows the motion of the robotic arm that does not start on the cycle but converges to the cycle. After approximately two cycles, the arm returns to tracking the desired cycle with its final accuracy of 2.5 cm.

9.4.2 Hardware results

Fig. 9.5 shows the learning curve of the repetitive control. It shows two curves that correspond to the absolute position error of the end effector at the pick positions (blue line) and the place position (red line). These errors were calculated by taking the errors at the time in the cycle when picking and placing are to take place. The learning phase takes about 40 cycles, which corresponds to 128s. After the cycle is learned, the error at the pick position is approximately 1cm and the error at the place position is approximately 2.5cm. These errors are small enough to perform many basic pick and place tasks, as found for instance in the food and packaging industry. Furthermore, these errors are also within the grasping-ranges of modern robotic grippers [26][110].

The blue solid lines in Figs. 9.3a-d show the cycle after it was learned on the robotic arm. The graphs show that the cycle on the robotic arm is the same as the one

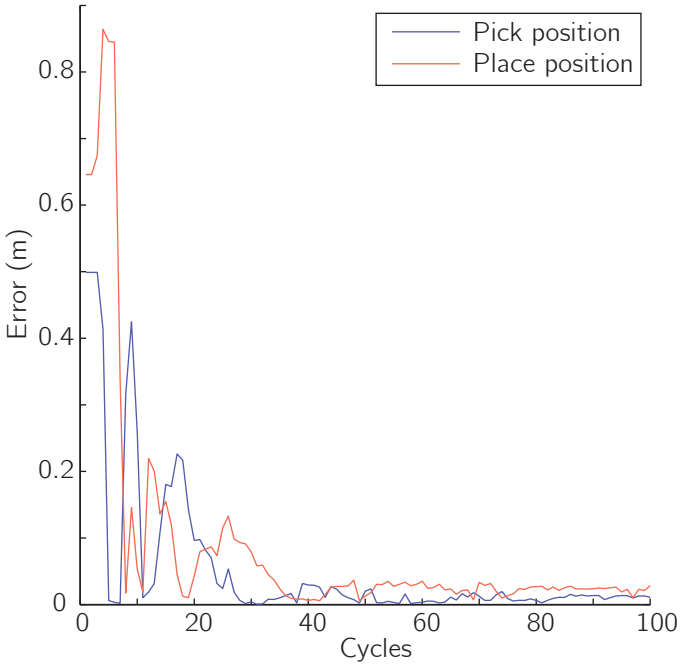


Figure 9.5: The absolute error in the pick and place positions while learning. This graph shows that the absolute error decreases to approximately 1 cm for the pick position and 2.5 cm for the place position.

obtained from simulation.

Fig. 9.4 shows the motion of the gripper in workspace. The red line shows the cycle that was obtained from simulation and the blue solid line shows the robotic arm that starts at a perturbed state and converges to the desired cycle. Within two cycles, the arm follows the desired cycle.

9.5 Discussion

9.5.1 Applicability

The techniques presented in this chapter make open-loop stable manipulation applicable. Previous work showed that model uncertainties lead to a cycle that differs too much from the intended cycle to be applicable in e.g. pick and place tasks [172]. With the learning of the open-loop controller, the errors in the position of the gripper decreased to 1-2.5cm. Furthermore, when the arm is perturbed, it converges back to the desired cycle within approximately two cycles.

Some applications, however, might require a more accurate controller (i.e. with errors smaller than 2.5 cm). There are no fundamental problems that limit the accuracy of our approach. There are however, two problems specific to our robotic arm and implementation that do limit the accuracy. Firstly, velocity estimation is not accurate due to quantization noise due to a coarse encoder. This makes it hard to learn the desired cycle. Secondly, the state is filtered in our implementation of repetitive control, leading to a smooth control signal. However, a non-smooth control signal might be required to track the desired cycle more accurately. In conclusion, depending on the application, the setup and implementation can be changed to increase the accuracy to the desired level.

The learned feedforward approach is applicable to many tasks. A new cyclic task can be handled by the same procedure of learning a feedforward trajectory, testing the robustness with Algorithm 1, and learning the input signal with repetitive control. At the moment the learning takes the most time, 300s. The optimization (1s) and robustness test (5s) take much less time. Reducing the time needed to learn a new task will further increase the applicability and is therefore an important part of our future research.

The main drawback of the current approach is that some feedback is required during the learning phase. However, this feedback can be delayed (almost a whole cycle) and can be removed after the learning has finished. The (delayed) feedback could therefore be provided by e.g. cameras. One interesting direction for future work would be to research repetitive control schemes without full state feedback. This could mean omitting certain states or reducing the frequency of the feedback.

This feedback requirement also comes up in situations where large disturbances are expected to occur after learning. In such situations the robot should know whether it has converged back to the desired cycle. This requirement can be satisfied with cameras, or even more basically, by using a switch at a certain position that checks the timing of passing that position.

9.5.2 Interpreting robustness measure

The robustness test depends on the choice of Δ, which is to some degree arbitrary. In previous experiments on this robotic arm, bending of the second joint was hypothesized to be one of the main causes of model-reality mismatch [172]. Therefore Δ was chosen to emphasize this error, which depends only on the position of the second joint.

For the chosen Δ, the resulting robustness measure ϵ_x of 0.038 indicates that the spring stiffness of ± 0.038 Nm/rad could be added to the second joint. This seems

insignificant to the effective spring stiffness of around 1.6 Nm/rad that is in place on the first joint. However, note that adding any negative spring stiffness to the second joint would make that joint unstable by itself. Therefore low values of ϵ_x should be expected.

9.5.3 Hybrid systems and Coulomb Friction

Despite the fact that robustness against modeling errors is checked, there are still two modeling-related issues that are not accounted for in the current approach. Firstly, when performing an actual pick and place motion, the model will consist of two phases with different end-point masses with an impact phase in between to model the grasping. Secondly, a more accurate friction model would include Coulomb friction. Both these effects can be added by using a hybrid system model. Such hybrid models are quite generally used in analysis of systems with limit cycles, mostly to cope with impact in walking, e.g. [237]. Under some basic transversality conditions, as discussed in [28], the current approach can be extended to such hybrid models. We see that inclusion as the logical next step in the research in open-loop control of robotic arms.

9.5.4 Scalability

The experiments were done on a two DOF robotic arm. Such an arm has similar dynamics to SCARA type arms, which are often used in industry. The current approach therefore already can be used for realistic industrial situations. However, there are also many robotic arms which have a larger number of joints and rotate in 3D. For such a setup the approach remains untested.

There are three possible concerns for this approach for higher DOF robots. Firstly, there is the possibility that no stable trajectories exist. This is unlikely, and it can be ensured not to happen if springs are used on all joints to stabilize the system. Secondly, the required computation will become more extensive. However, because only a feedforward signal is required, no full state exploration is necessary. Furthermore, the number of free parameters in the robustness approach is independent of the number of DOFs on the robot. Thirdly, although earlier results on a two DOF arm suggest that the basin of attraction of our approach is quite large [172], for multiple degrees of freedom it becomes more likely that the robotic arm will converge to a different cycle after a large perturbation. Again, this could be prevented by using springs on all joints to stabilize the system. Combining all of the above, we expect that our approach scales well to higher DOF robotic arms.

9.6 Conclusion

In this chapter we showed that repetitive tasks can be performed stably by robotic arms with an open-loop controller, even when an accurate model is not available. We used an LMI-based robustness analysis to check the robustness to model inaccuracies. We then used a repetitive control scheme to track this trajectory with the robotic arm. Hardware experiments show that using this approach, position errors can be reduced to 2.5 cm, making open-loop control applicable in tasks such as picking and placing objects.

Acknowledgement

This work is part of the research programme STW, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO).

10

The effect of the choice of feedforward controllers on the accuracy of low gain controlled robots

Michiel Plooij*, Wouter Wolfslag* and Martijn Wisse
* These authors contributed equally to this chapter,
IEEE/RSJ International Conference on Intelligent Robots and Systems 2015.

Abstract

High feedback gains cannot be used on all robots due to sensor noise, time delays or interaction with humans. The problem with low feedback gain controlled robots is that the accuracy of the task execution is potentially low. In this chapter we investigate if trajectory optimization of feedback-feedforward controlled robots improves their accuracy. For rest-to-rest motions, we find the optimal trajectory indirectly by numerically optimizing the corresponding feedforward controller for accuracy. A new performance measure called the Manipulation Sensitivity Norm (MSN) is introduced that determines the accuracy under most disturbances and modeling errors. We tested this method on a two DOF robotic arm in the horizontal plane. The results show that for all feedback gains we tested, the choice for the trajectory has a significant influence on the accuracy of the arm (viz. position errors being reduced from 2.5 cm to 0.3 cm). Moreover, to study which features of feedforward controllers cause high or low accuracy, four more feedforward controllers were tested. Results from those experiments indicate that a trajectory that is smooth or quickly approaches the goal position will be accurate.

10.1 Introduction

High precision in robotics is usually achieved with high feedback gains. However, there are applications in which such high gains are undesirable or infeasible. For instance, in the presence of sensor noise or time delays, high feedback gains will make the robot unstable. A second example of robots in which high gains are undesirable are robots that interact with humans. In such robots, high feedback gains increase the risk of injury. These examples show that it is important to develop alternative techniques to obtain high precisions that work even on robots with limited feedback.

To that end, multiple researchers have taken the idea of feedback limitations to an extremum and have focused on executing tasks with robots without any feedback. A first example is the concept of passive dynamic walking, as introduced by McGeer [134]. Those walkers do not have motors and therefore no feedback control, and still walk with a stable gait. These gaits do not rely on the motion being stable at each point in time, rather they work due to the existence of stable cyclic motions, called limit cycles. Such cycles were later on combined with feedback control in so called limit cycle walkers [72, 89, 94]. Mombaur et al. [139, 141] found stable open loop

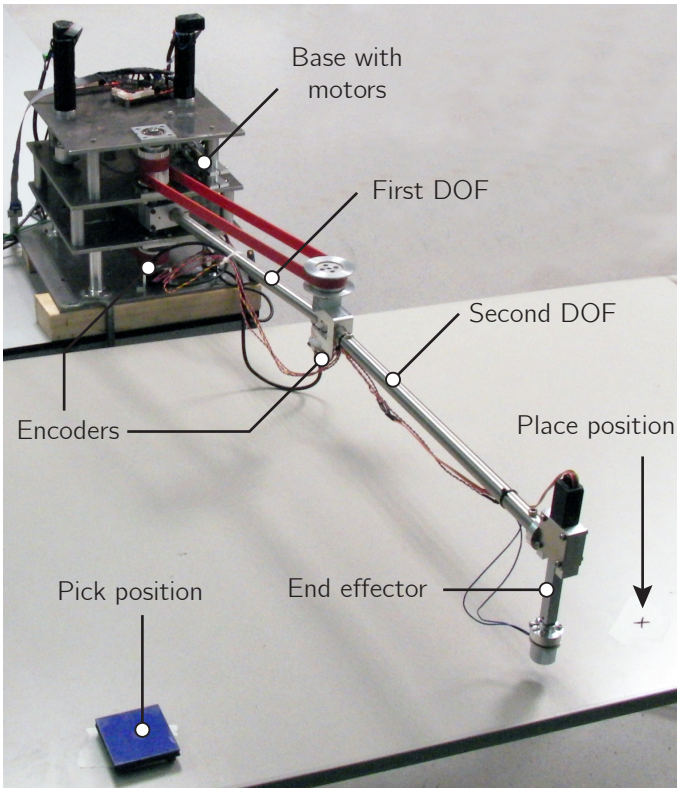


Figure 10.1: A photograph of the robotic arm we use to test our method: a two DOF SCARA type arm, which has to move between the pick and the place positions.

controllers for walking and running robots by optimizing the open loop controllers for both stability of the motion and energy consumption.

Control without feedback has also been applied on robotic arms. A well-known example is the work of Schaal and Atkeson [189], who studied open loop stable juggling with a robotic arm. In their case, open loop means that the state of the ball is not used as an input for the controller, but the arm itself is position controlled. In previous work, we showed that it is possible to perform open loop motions with robotic arms that are insensitive to model inaccuracies [167, 174] and to perform open loop stable cycles in which state errors vanish without any feedback [172, 244]. In [244], we optimized trajectories for open loop stability, and used an initial on-line learning approach to improve the precision of the purely feedforward controlled robot. In these studies, the trajectory itself was effected by the choice of feedforward controller, which was optimized. In the rest of this chapter, we consider the feedforward

controller and the trajectory to contain the same information, since they can be translated into each other using the model of the robot.

Previous examples show that the most common technique to stabilize robots without feedback is to optimize trajectories for their open loop stability [139, 141, 172, 174]. In practice, a certain amount of feedback will always be available, and therefore the advantages of both control paradigms should be exploited to achieve higher precision [112]. However, it is unclear if trajectory optimization is useful for robots with at least a little feedback.

Therefore, the question we will answer in this chapter is: *does the choice of the feedforward controller influence the accuracy of systems with (limited) feedback?* We will answer this question by studying a two degree of freedom (DOF) SCARA type robotic arm in the following way. First, in section 10.2 we explain the methods that we used, including the setup and task we study. Then, in section 10.3 we introduce the Manipulation Sensitivity Norm (MSN), which we use to estimate the lower and upper bounds of the accuracy of the arm given a certain feedback controller. Next, in section 10.4 we show the results of four alternative controllers that indicate that smooth and goal directed motions result in high accuracy. These results are discussed in section 10.5. And finally in section 10.6, we will conclude the chapter.

10.2 Methods

In this section we explain the methods we used. First, we explain the systems under consideration, including the controller. Second, we describe the robotic arm that is used as a test case. Third, we discuss the specific task that is studied. And finally, we discuss the feedforward term in the controller.

10.2.1 System description

The type of system considered in this chapter is a serial chain robotic arm moving in the horizontal plane. The equation of motion of such a system is described by a second order differential equation:

$$\ddot{q} = f(q, \dot{q}) + M^{-1}(q)\tau \tag{10.1}$$

with q the absolute angles of the links of the robot, $\dot{q}$ and $\ddot{q}$ the angular velocities and accelerations, and τ the motor torques, which are used as control inputs. Note that this system is non-linear due to the Coriolis and centrifugal terms $f(q, \dot{q})$, and the configuration dependent mass matrix M. To control the robot, both a feedback and a feedforward term are used, hence $\tau = \tau_{fb} + \tau_{ff}$.

Because the goal of this chapter is to investigate the effect of feedback gain limitations, we structure the feedback controller in such a way that it depends on only one parameter, namely ω, which is the desired natural frequency of the controlled system. For the purpose of constructing the feedback controller from this ω, the system is simplified by neglecting $f(q, \dot{q})$, and decoupling the resulting system by considering only the diagonal entries of the mass matrix at position $q = 0$. In other words, only considering the simplified system

$$\text{diag}(M(0))\ddot{q} = \tau \tag{10.2}$$

Note that this simplified system is only used to construct the feedback controllers and that the system we study is the non-linear system in Eq. (10.1). By using diagonal gain matrices K and C, we obtain the following, second order linear differential equations:

$$\text{diag}(M(0))\ddot{q}_i = -Kq - C\dot{q} \tag{10.3}$$

Because all matrices are diagonal, the differential equations are decoupled, which means that they can be solved separately. Finally, we choose the gains such that the natural frequency of all decoupled parts are set to a desired value (ω), and the damping ratio is set to 1, i.e. critically damped. Therefore, the controller gains are set by solving:

$$\sqrt{\frac{k}{m}} = \omega \tag{10.4}$$

$$\frac{c}{2\sqrt{km}} = 1 \tag{10.5}$$

for each part. The natural frequency ω is then used as the parameter to vary the gains. These feedback gains are then used to stabilize the robotic arm around a desired trajectory:

$$\tau_{fb} = -K(q - q_{des}) - C(\dot{q} - \dot{q}_{des}) \tag{10.6}$$

The feedforward term is simply a torque as function of time: $\tau_{ff}(t)$. This term leads to the desired trajectory: $\ddot{q}_{des} = f(q_{des}, \dot{q}_{des}) + M^{-1}(q_{des})\tau_{ff}(t)$. Because of this relation between feedforward controller and trajectory, we use the two terms interchangeably in this chapter.

The goal of this chapter is to study the effect of the feedforward controller on the accuracy of a rest-to-rest motion under disturbance. Our approach to studying the effect is to optimize the feedforward controller to minimize or maximize this disturbed accuracy. The optimization is done using single shooting, a basic optimal control approach.

Table 10.1: The model parameters of the two DOF arm.

Parameter	Symbol	Value	Unit
Damping	μ_{v1}, μ_{v2}	0.2, 0.2	Nms/rad
Inertia	J_1, J_2	0.0233, 0.0312	kgm^2
Mass	m_1, m_2	0.809, 0.784	kg
Length	l_1, l_2	0.410, 0.450	m
Position of COM	l_{g1}, l_{g2}	0.070, 0.195	m
Motor constant	k_{t1}, k_{t2}	25.9, 25.9	mNm/A
Gearbox ratio	g_1, g_2	82.5:1, 22.5:1	rad/rad

10.2.2 Hardware setup

To test our approach, we use a two DOF SCARA type arm [171] (see Fig. 10.1). This type of arm was chosen as it is the simplest that can perform industrially relevant tasks. The arm consists of two 18x1.5mm stainless steel tubes, connected with two revolute joints. An end effector is connected to the end of the second tube. The motors are placed on a housing and AT3-gen III 16mm timing belts are used to transfer torques within the housing. The joints are actuated by Maxon 60W RE30 motors with gearbox ratios of respectively 66:1 and 18:1. The timing belts provide an additional transfer ratios of 5:4 on both joints. Because the second joint is connected to its motor via a parallelogram mechanism (see [171]), the angle of the second arm is taken as the absolute angle, i.e., relative to the world frame. The end effector acts in the vertical plane and thus its motions do not influence the dynamics of the first and second DOF. The mass of the end effector is incorporated in the inertial terms of the second DOF. The arm is controlled through xPC-target in MATLAB at a frequency of 1 kHz. The parameters of this robotic arm are listed in Table 10.1.

10.2.3 Task description

We let the manipulator perform a cyclic pick and place motion, with pick and place positions at [-0.2, -0.3] rad and [0.2, 0.4] rad respectively. The time to move between the pick and the place position is 1.05 s. Hence, the total time of one cycle is 2.1 s.

10.2.4 The feedforward term in the controller

We test the accuracy resulting from different feedforward controllers: minimization and maximization of the novel Manipulation Sensitivity Norm (see section 10.3) and both smooth and time optimal trajectories (see section 10.4). The feedforward controller has to be parameterized in order to be able to optimize it.

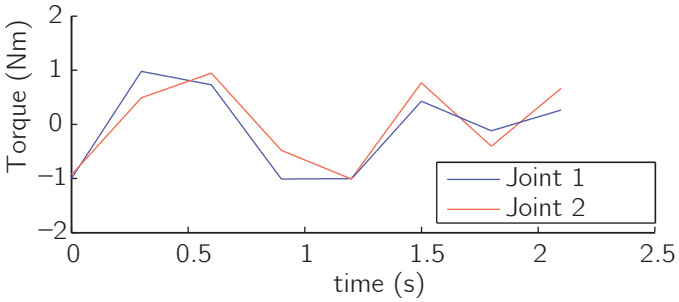


Figure 10.2: An example of a feedforward controller. The controller is parameterized as a piecewise linear function with the length of every part being 0.3 s. These torques over time are used as decision variables in the optimizations.

The optimization schemes parameterize the torque signals for both joints as a piecewise linear signal, with the length of every piecewise part being 0.3 s. The space of such signals is called $\mathcal{U}$. The system states are constrained to be on the pick and place motions at the pick and place times. This also ensures that the motion is cyclic. Finally, the absolute value of both torque signals is bounded by $\tau_{max} = 1$ Nm in order to prevent reaching the actuator limits when when feedback is needed. An example of such a feedforward control signal is shown in Fig. 10.2.

In the remainder of the chapter, the piecewise linear torque signals are optimized for various goals. These goals are expressed as a function $C(\tau_f f)$, which is either maximized or minimized. Combined with the task description, this leads to the following optimization problem:

$$\begin{aligned} \underset{\tau_{ff}(t)\in\mathcal{U}}{\text{minimize}} \quad & C(\tau_{ff}) \\ \text{subject to} \quad & |\tau(t)| \leq \tau_{max} \qquad \forall t \\ & q(0) = q_{pick} \\ & q(1.05) = q_{place} \\ & q(2.1) = q_{pick} \\ & \dot{q}(0) = \dot{q}(1.05) = \dot{q}(2.1) = [0, 0] \end{aligned} \tag{10.7}$$

10.3 Optimality study

In this section we estimate the lower bound and upper bound of the accuracy of the arm, given a certain feedback gain ω. First, we introduce a new measure for

disturbance and modeling error rejection, called the Manipulation Sensitivity Norm (MSN). Then, we use this measure to optimize feedforward controllers in simulation, both minimizing and maximizing the MSN. Finally, we apply these controllers on the hardware setup, to test their accuracy.

10.3.1 The Manipulation Sensitivity Norm

To judge the quality of a feedforward signal, a measure is needed that quantifies the feasibility of performing a manipulation task when there are disturbances or modeling errors. This section explains the novel Manipulation Sensitivity Norm (MSN), which is inspired by the gait sensitivity norm used for bipedal walking robots [91]. This inspiration comes from the insight that a pick and place task can be seen as a repetitive motion and can therefore be analyzed using limit cycles [172]. The effect of disturbances on limit cycles on bipedal robots can be captured by the gait sensitivity norm, which analyses the system based on an estimation of an input-output relation that is defined once per step. This means that the effect of a realistic disturbance profile during one step is taken as input, and a performance measure as it occurs during that step is the output. In walking robots, a possible output is the step time. A slight modification of the gait sensitivity norm can be used to analyze the performance of manipulation tasks. This modification is the MSN, and requires four steps to compute.

1. Defining output indicators
2. Defining a set of realistic disturbances as input signals
3. Obtaining the input to output relation
4. Computing the appropriate system norm of the input output relation.

The first step is to define output indicators. For pick and place tasks, output indicators are a measure of the distance of the arm to the desired path. To make the analysis as clear as possible, we use the error in the absolute angles of the links at the pick position, which is the initial position of the cycle. The MSN will compute the gain from a set of realistic disturbances to this output measure and is therefore a measure of accuracy when moving under real world disturbances.

The next step is to define the disturbances, which are used as inputs. For our analysis, we use three disturbances: a torque on the first link during the first 0.15 s of the cycle, a varying end-effector mass that represents a product that has a different weight than expected and a varying viscous friction coefficient. These inputs have

a nominal value of 0, and their value is allowed to change every cycle. Note that the last two inputs, mass and friction, are typically seen as parameter variations. For this linearized analysis, there is no mathematical distinction between such a parameter variation and a more traditional disturbance such as the torque. This justifies treating parameter variations and disturbances in the same way. Do note however, that the parameter uncertainty is lasting, which should be reflected when computing the input-output gain.

In the third step, the input-output relation in Eqs. 10.8-10.9 are obtained using a finite difference scheme. At the beginning of every cycle, very small (10^{-5}) initial condition disturbances are used to obtain a Jacobian matrix A, by comparing the initial state to the state after exactly one cycle. Then, small values (10^{-5}) of the inputs are used to obtain the input to state Jacobian B. Finally, the relation to the output is linearized, again both for initial error and inputs, obtaining C and D respectively. The procedure is described in more detail in [91]. We now obtain a state space system S of the form:

$$x(n+1) = Ax(n) + Bu(n) \tag{10.8}$$

$$y(n) = Cx(n) + Du(n) \tag{10.9}$$

where x is a vector containing the errors in the state after each cycle and y is a vector containing the errors in the positions at the pick position. In this linearized discrete system, the matrix A depends on the system dynamics and the chosen trajectory and B depends on how the inputs influence the state error. Note that for our choice, the output indicators are already linear in the state and $D = 0$.

The last step is to compute an appropriate norm, which is the main difference between the gait sensitivity norm and the MSN. For walking, it is important that the walking motion recovers to normal after the disturbance stops. For manipulation, the disturbances tend to last, meaning we are interested in the maximum error in the situation where the disturbance continues to exists, The appropriate norm is thus the induced $\mathcal{L}_\infty$ norm:

$$\|S\|_{\mathcal{L}_\infty} = \sup_{u \neq 0} \frac{\|y(u)\|_\infty}{\|u\|_\infty} \tag{10.10}$$

What remains is to compute $\|S\|_{\mathcal{L}_\infty}$. The $\mathcal{L}_\infty$-induced norm is the same as the $\mathcal{L}_1$ norm of the impulse response [24, 166]. Rather than computing the complete $\mathcal{L}_1$ norm, we approximate it by taking the sum of the first N steps, with $N = 100$, chosen sufficiently large. Take $g_{ij}(n)$ as the impulse response from input j to output

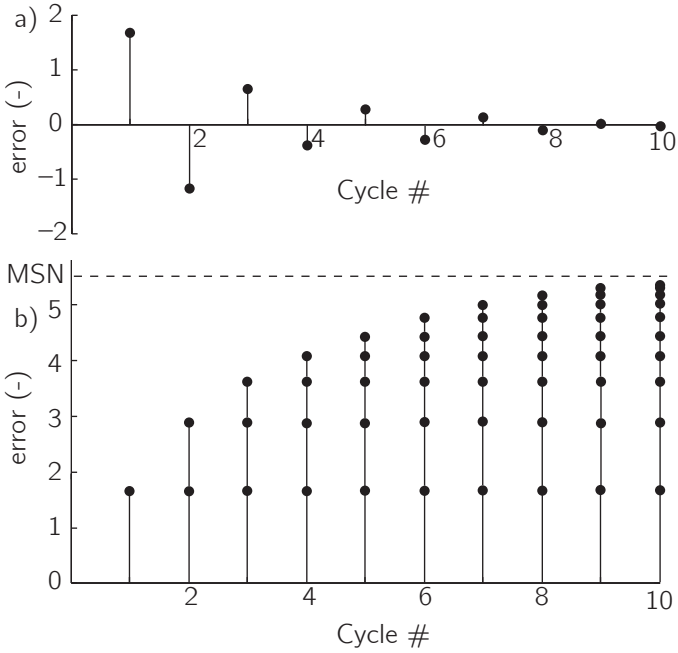


Figure 10.3: A visualization of the calculation of the MSN. a) The impulse response of the system. b) The sum of the absolute values of the impulse response. The MSN can be interpreted as the amount of error given a unit input, which would have radians as unit. Since the input for which the error is largest differs per cycle, we chose to not use a unit for the MSN.

i. Now the Manipulation Sensitivity Norm can be written as:

$$MSN = \|S\|_{msn} = \max_i \sum_{n=0}^{N} \sum_j |g_{ij}(n)| \tag{10.11}$$

The MSN is the amount of error given a unit input and therefore could have radians as unit. However, the specific input for which the error is largest differs per cycle. Therefore, we chose to not use a unit for the MSN. The calculation of the MSN is visualized in Fig. 10.3 and the overall procedure is summarized in Algorithm 2.

When computing the MSN, we should scale the size of the inputs in order to take into account the difference in the effect they have and the realistic sizes of those inputs. Since the expected disturbances depend heavily on the system under consideration, we choose a different approach, which highlights the capability of the MSN to take into account multiple disturbance sources at the same time. The input sizes are scaled such that the MSN of each of the three inputs considered separately is 1, when feedback gains specified by $\omega = 1$ s^{-1} are used for the MSN minimization.

Algorithm 2 Calculating the MSN

```
procedure MSN(τ(t), ω, N)
    Determine [q(t), q̇(t)]                          ▷ Eq. (10.1)
    Determine K                                      ▷ Eq. (10.4)
    Determine C                                      ▷ Eq. (10.5)
    Determine A                                      ▷ Finite difference on q_0
    Determine B                                      ▷ Finite difference on u
    for j = 1..J do
        for n = 1..N do
            Determine g_ij(n)                        ▷ Eqs. 10.8-10.9
        end for
    end for
    Determine MSN                                    ▷ Eq. 10.11
    return MSN
end procedure
```

The overall optimization for minimizing/maximizing the MSN is described by eq. (10.7), with as cost function

$$C(\tau_{ff} = \pm MSN(\tau_{ff}) \tag{10.12}$$

We used `fmincon` with 20 initial conditions determined by `multistart` in MATLAB® to solve the optimizations.

10.3.2 Simulation results

Fig. 10.4 shows the minimum and maximum MSN that were obtained as functions of the natural frequency parameter ω. This figure shows what was to be expected: increasing the gains results in a decrease of the MSN. The figure shows both the maximum and the minimum MSN that were obtained by optimization. At low gains, the maximization does not result in stable controllers, meaning that the MSN is infinitely large. This instability shows that at these gains the unstabilizing dynamical effects are larger than the stabilizing effects of the feedback controller. For $\omega > 5$ s^{-1}, the difference between the cycles with minimized and maximized MSN is negligible.

The red lines in Fig. 10.6a and 10.6b correspond to two optimized cycles. The cycle in Fig. 10.6a was obtained by maximizing the MSN and the cycle in Fig. 10.6b was obtained by minimizing the MSN. These cycles were obtained for $\omega = 2.7$ s^{-1}. The corresponding values for the MSN are 12.2 and 1.6 respectively. Finally, Fig. 10.7

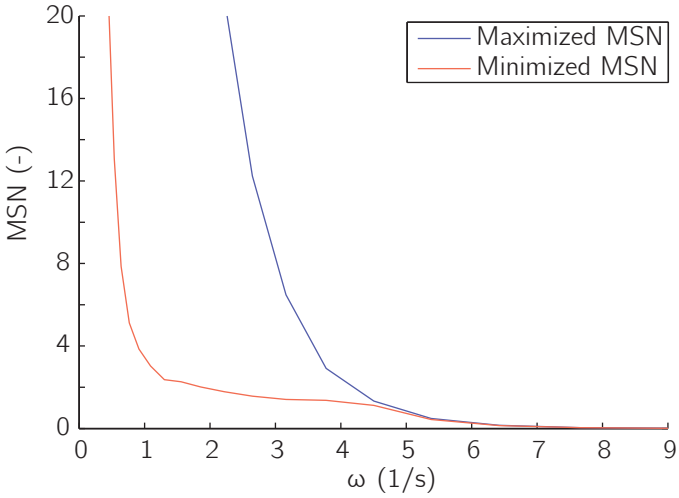


Figure 10.4: The minimized and maximized MSN as function ω, as found in simulation.

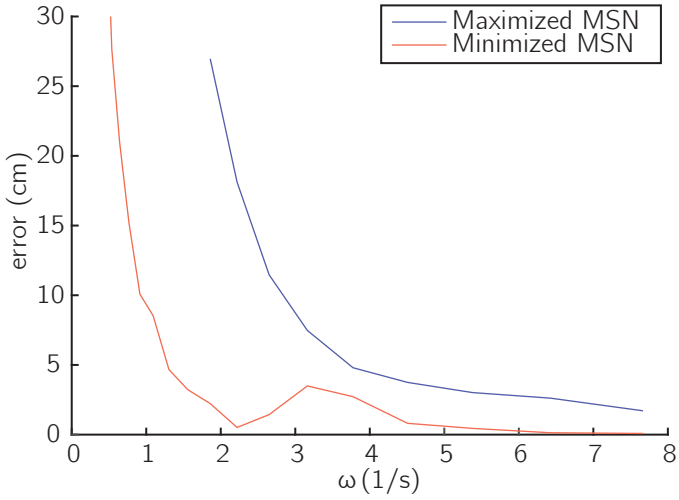


Figure 10.5: The errors at the pick position as function of ω, as found in hardware experiments. The errors are shown for trajectories with a minimized MSN and a maximized MSN.

shows the time evolution of the torque signals, the feedforward term of which was obtained in simulation.

10.3.3 Hardware results

Fig. 10.5 shows the position error of the end effector at the pick position during hardware experiments as function of the natural frequency parameter ω. These errors are the average error over 10 cycles after letting the robotic arm converge for 2 cycles initially. The standard deviation over these 10 cycles is negligible. Logically, the errors decrease when the feedback gains are increased.

At the pick position, the error of the MSN minimizing trajectory is 0.3 cm at ω = 2.1 s^{-1}, in between ω = 2.1 s^{-1} and ω = 4.5 s^{-1}, the error of that trajectory is approximately 2.5 cm and for ω >4.5 s^{-1}, the error drops to approximately 0.3 cm again. The error of the maximized-MSN-trajectory is larger than 2.5 cm, for almost the whole range of gains. Only, for ω = 7.7 s^{-1}, the error becomes 1.8 cm. This significant difference between the errors of these two trajectories means that the choice for the feedforward controller is important for the accuracy of the task execution.

Fig. 10.6 shows two typical sets of hardware results. These results were obtained for a controller with $\omega = 2.7$ s^{-1}. The plots show the state space trajectories for a maximized MSN and a minimized MSN. They show that the three trajectories differ significantly: the trajectory with maximized MSN covers a larger part of state space than with minimized MSN. The measurements on the prototype show the 10 cycles used in determining the errors used in Fig. 10.5. It can be seen that the arm has converged, and only very little variation between cycles occurs.

Fig. 10.7 shows the torques for the two different feedforward controllers with ω = 2.7 s^{-1}. The plots show both the feedforward torque and the actual torque. The difference between the two is due to the feedback controller. The plots clearly show that the feedback control effort is larger when following the trajectory with maximized MSN (Fig. 10.7a), than when following the trajectory with minimized MSN (Fig. 10.7b).

10.4 Alternative motion profiles

In the previous section, the feedforward controllers under study were determined by minimizing or maximizing the MSN. To further study which feedforward controllers lead to accurate motions, four more methods to generate a feedforward controller will now be compared. The simulation and hardware results for these controllers are found in Fig. 10.8 and 10.9 respectively.

The first of the controllers is used to compare the minimized and maximized MSN trajectories to a trajectory that is standard in industry: *a trapezoidal velocity profile*.

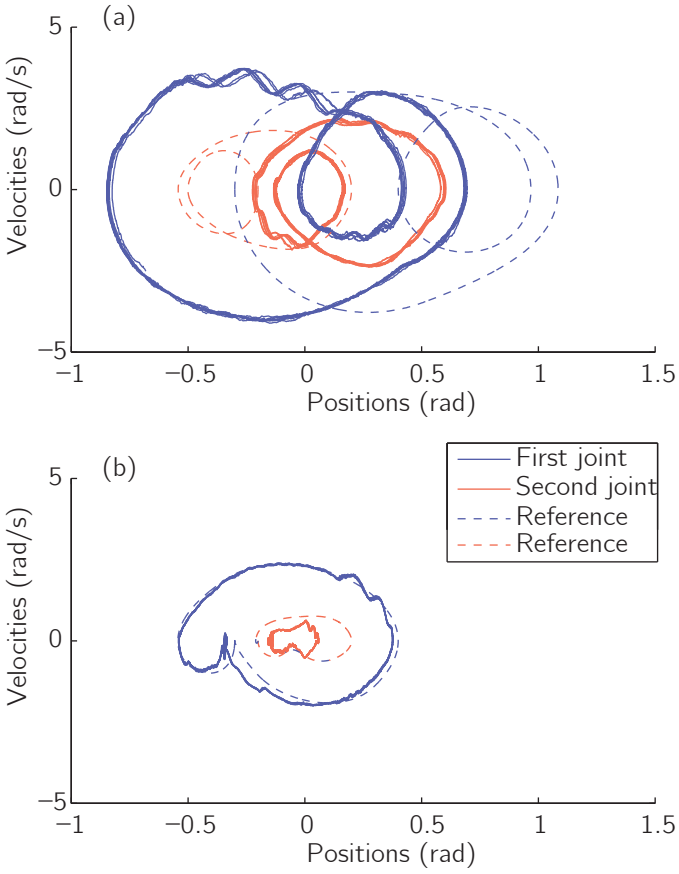


Figure 10.6: State space plots of the optimization and hardware results for ω =2.7 s^{-1}. a) The cycle with a maximized MSN. b) The cycle with a minimized MSN.

The trapezoidal velocity profile is created by dividing the time to move between pick and place position in three equal parts: one part each for acceleration, constant velocity and deceleration. The same procedure is used to move from place to pick position. Both the MSN in simulation, and the position error on the hardware show that this trapezoidal trajectory has accuracy closer to the minimized MSN trajectory than to the maximized MSN trajectory. In hardware results, the error of the trajectories with minimized MSN and trapezoidal velocity profile are not even significantly different.

So, why does this standard controller perform as accurate as the optimally accurate one? There are two potentially beneficial aspects to this trapezoidal trajectory.

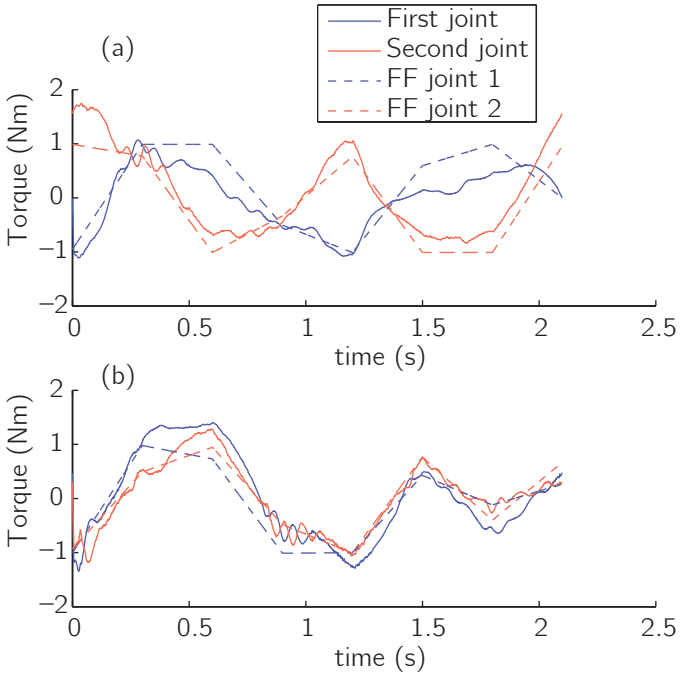


Figure 10.7: The torques as functions of time for the results with ω =2.7 s^{-1}. a) Torques for the cycle with a maximized MSN. b) Torques for the cycle with a minimized MSN.

First, it is relatively smooth, without large accelerations back and forth. Second, it approaches the goal directly, and is already close to the goal position for the last part of the motion. To test if these two effects are indeed beneficial, we compare feedforward controllers with these two specific aspects.

If smooth controllers lead to accurate motions, it should be impossible to make an inaccurate motion with a smooth controller. To test this, we performed *the MSN-maximization with a smoothness constraint*. Here we took the smoothness as a maximal torque time derivative of $t_f/4$ Nm/s. This rate allows the torque to go from maximum to minimum and back in one cycle. As can be seen in Fig.10.9, this maximization with constraint has similar accuracy as the minimization in hardware results. This indicates that smoothness is indeed beneficial for accuracy.

To test whether a quick motion towards the goal leads to low errors, we optimized a cost function that squares the error with the goal position. This new optimization

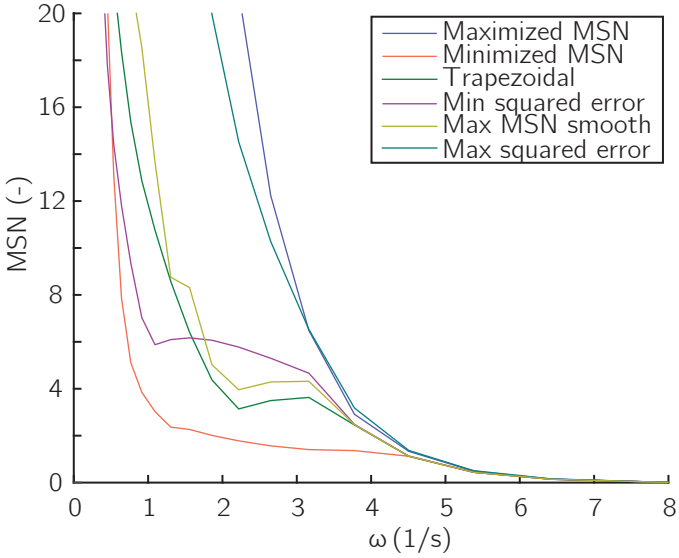


Figure 10.8: The MSN as found in simulation experiments. The errors are shown for six types of trajectories.

is otherwise the same, but minimizes the following cost function:

$$C(\tau_{ff}) = \int_0^{t_f} (q(\tau_{ff}) - q_g(t))^T (q(\tau_{ff}) - q_g(t)) dt \tag{10.13}$$

With $q_g(t)$ being the way-point position when $t < t_f/2$, and the initial position otherwise. Again, the results show that *this squared error minimization* gives accuracy close to the minimized MSN trajectory, as expected.

In order to confirm these results, we also tested the motion with a *maximize the squared error* function? Because this would lead to a motion that moves away from the target, and only reaches the target at the very last moment, this motion is expected to result in relatively large errors. Furthermore, moving away, and then rapidly towards the target is not very smooth, which is also likely to affect the accuracy adversely. Figs. 10.8 and 10.9 show that this prediction is indeed true. The trajectory performs worse than the other trajectories, although still not as poorly as the maximized MSN trajectory.

10.5 Discussion

In this section, we discuss the results as presented in this chapter. As can especially be seen in Fig. 10.5, the choice for a certain trajectory is important for the accuracy

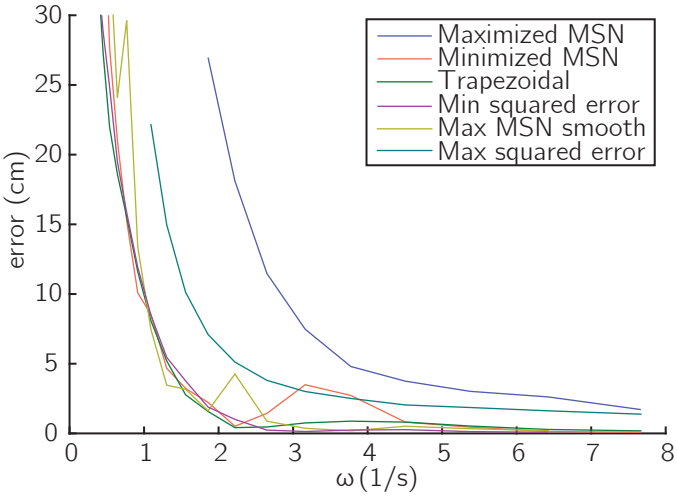


Figure 10.9: The errors at the pick position as function of ω, as found in hardware experiments. The errors are shown for six types of trajectories.

that is achieved. For high gains ($\omega > 3.2$ s^{-1}), this choice makes the difference between a negligible position error and an error of multiple centimeters. For medium gains (1.9 s$^{-1} < \omega < 3.2$ s^{-1}), this choice makes the difference between negligible position errors and errors between 5 and 27 cm. And for low gains ($\omega < 1.9$ s^{-1}) this choice makes the difference between stability and instability.

The simulation and hardware results match well. Particularly, the shape of Figs. 10.4 and 10.5 are similar. There are however two differences. First, the difference in error between the trajectories does not converge to 0 when the gains increase in hardware experiments, whereas the difference in MSN does convergence in simulation. The second difference is the hump in error and MSN that occurs around $\omega = 3$ rad/s. These differences are small and are caused by unmodeled dynamics that were not taken into account in our choice for disturbances in the MSN-computation. The most likely effects are elasticity in the timing belts and backlash. Because the simulation and hardware results are so similar, the MSN is a good approximation for accuracy, and can be used to find feedforward controllers in cases when feedback gains are low, yet accuracy is important.

The shape of Figs. 10.8 and 10.9 are similar, but there are clear differences. The first and most important difference is that only two control strategies lead to errors that are significantly larger than the minimum error. Those two control strategies are the maximized MSN and the maximized squared error. These results suggest that there

are two principles that lead to small errors: smoothness and being close to the goal position before the end of the motion. Therefore, we expect that other common profiles such as minimal energy, minimal torque, minimal acceleration and minimal jerk will also perform well. The second difference is that in the hardware results, the accuracy of the MSN-optimized feedforward controllers only give an estimate of the range of possible accuracies. This can be seen in Fig. 10.9, in which the minimized MSN controller is not the most accurate one for $2.1\ s^{-1} < \omega < 4.5\ s^{-1}$.

In simulation, there are two points where one of the comparison trajectories has an MSN that is outside the range given by the MSN minimization and maximization. Specifically, this occurs for the In both these instances, the difference is small, and caused by the choice of step size in simulation. In optimization, a sampling time of 0.01 s is used to save computation time. For Fig. 10.8, a sampling time of 0.001 s was used, because this aligns with the robot hardware.

As ω goes to zero, both the MSN and the error in hardware results get very large. This is logical since the system without feedback is not stable. Fig. 10.5 shows that for feedback with $\omega < 1.8\ s^{-1}$, the error is larger than 2.5 cm and therefore picking up objects of reasonable size will be difficult. If the feedback gains cannot be increased, more mechanical feedback has to be implemented. The most straightforward approach is to place springs at the joints. In our previous work [172, 244], we showed that with a spring at the first joint, tasks can be performed stably even when $\omega = 0$.

The results from this chapter can be improved by incorporating the feedback in a Repetitive Control (RC) scheme [123]. In an RC scheme, the feedforward controller is adjusted based on the state error in the previous cycle. In the most simple form, the feedforward controller in the current cycle is equal to that in the previous cycle plus the feedback that was applied. Such an RC scheme was used before on robotic arms to learn open loop stable trajectories [244].

There is an interesting parallel between the controller we use in this chapter and human movement control. Similar to our controller, humans also exploit the advantages of both feedforward and feedback in order to optimize their performance [47]. For fast motions, humans cannot rely on feedback at all, due to the large time delays (typically 150 ms for humans [44, 216]). Therefore, they have to rely on feedforward, in which control signals are generated based on the prediction of an internal model [101]. In slower motions, more feedback is used to correct for inaccuracies in the internal model and external disturbances.

Another interesting parallel with human motion control is the fact that smooth motions perform well. In human motion control, there is an ongoing debate about the

cost function humans use optimize their motions. Suggested cost functions are the maximum jerk [59, 92], change of torque [227] and sensitivity to motor noise [81]. Other researchers suggest that humans perform some kind of stochastic optimal control in which variability in task irrelevant directions is ignored [218]. The problem in this debate is that all cost functions result in approximately the same smooth motions. Similarly, we expect that all smooth motions that result from such cost functions will perform well in terms of accuracy.

10.6 Conclusion

In this chapter we focused on the question 'does the choice of the feedforward controller influence the accuracy of systems with (limited) feedback?'. The answer to this question is: 'yes, the choice for a certain feedforward controller makes the difference between an accurate and an inaccurate task execution.' The feedforward controller was tuned using the novel Manipulation Sensitivity Norm, which measures the accuracy while taking into account disturbances and model errors. Feedforward controllers that were either minimized or maximized for this norm were implemented on our robotic arm. Results show that for a large range of feedback gains, the error varies between 0.3 and 2.5 cm, depending on the choice for the feedforward controller. Further experiments with alternative feedforward controllers indicated that a trajectory that is either smooth, or approaches the goal position quickly, will be accurate. Therefore, the commonly used trajectory with a trapezoidal velocity profile performs well and is a good choice in terms of accuracy.

Acknowledgement

This work is part of the research programme STW, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO).

11

Discussion, conclusions and future directions

The topic of this thesis is improving the performance of robotic arms by using inspiration from humans. Part I focused on reducing the energy consumption of robots by using elasticity. Section 11.1 discusses the results of the first part of this thesis. Part II focused on increasing the reliability of robots by using feedforward control and its results are discussed in section 11.2. This chapter ends with general conclusions in section 11.3 and interesting future research directions in section 11.4.

11.1 Elasticity in robots

This section discusses the results of the first part of the thesis. It starts with a recapitulation of the results, answering the first three research questions of this thesis. Then, two choices that were made in chapter 1 are revisited in sections 11.1.2 and 11.1.3. First, we chose to limit the research to one and two DOF robotic arms, where in practice, robots have more DOFs. Therefore, section 11.1.2 discusses how the results in the first part of this thesis extend to robots with more DOFs. Secondly, we chose to use locking mechanisms in PEAs as opposed to using continuously variable transmissions. In section 11.1.3, we evaluate this decision. Finally, one of the most critical parts in the development of future CEAs is the design of the locking mechanisms. Therefore, section 11.1.4 discusses the requirements of locking mechanisms in CEAs and evaluates the statically balanced brake concept.

11.1.1 Recapitulation

Regarding the mechanical design of robotic arms, three questions were addressed. The main results on those questions are:

- *What is the best mechanism to reduce the energy consumption of pick-and-place robotic arms?*
  In Chapter 2, we proposed to use a non-linear spring mechanism that reaches a singular position at the pick position and the place position. Moving between the pick position and the place position costs less energy because the spring takes over large part of the torque of the motor. In Chapter 3, we showed that this mechanism can also be seen as a singular locking mechanism. These locking mechanisms have the disadvantage that they can only lock at a limited amount of positions, limiting the amount of possible pick and place positions. Therefore, this non-linear spring mechanism limits the versatility of the arm too much. As opposed to singular locking mechanisms, friction based locking mechanisms are a type of locking mechanisms with an infinite amount of locking positions. Therefore, they are more suitable for this application.

In Chapter 5, we proposed the BIC-PEA, which is a new type of CEA that uses friction based locking mechanisms. Using a differential mechanism and two locking mechanisms, the spring can be disconnected from the joint, locked or connected to the joint at any desired position. Furthermore, at the beginning of a movement, the direction of acceleration can be chosen by selecting which locking mechanism to release first. Implementation of such a mechanism can lead to a reduction in the energy consumption up to 65%. With a differential mechanism with low friction, the reduction in energy consumption might be increased to 80-90% in the future.

In pick-and-place tasks, it is crucial to be able to vary the pick position and the place position on the fly. The BIC-PEA is the only type of EA that allows for such versatility, while still reducing the energy efficiency drastically.

- *What is the best locking mechanism for clutched elastic actuators?*
  The three most important properties for a locking mechanism to be applicable in CEAs, are the ability to unlock under load, have a low energy consumption and lock at many positions. In Chapter 3, we showed that friction based locking mechanisms are the most promising, because they unlock well under load and lock at many positions. However, their disadvantage is that they typically consume much energy. In Chapter 4, we introduced the statically balanced brake. These brakes consume few energy because they do not have to generate a large normal force between the friction surfaces. This means that this type of brake can unlock under load, has a low energy consumption, has infinite locking positions and is controllable. Statically balanced brakes are the only locking mechanisms with these properties.

- *How can the functionality of complex clutched elastic actuators be analyzed?*
  In Chapter 6, we proposed a framework to analyze the functionalities of CEAs. We argued that CEAs consist of springs, differentials and clutches that can be expressed in terms of a stiffness matrix, a constraint matrix and a combination of a diagonal clutch matrix and an incidence matrix. Using this description, the set of possible resulting stiffnesses can be found. Furthermore, it can lead to new CEA designs in which the number of resulting stiffnesses grows exponentially with the number of springs and clutches.

11.1.2 On extending the research to large DOF systems

The research on the mechanical design of robotic arms has mainly focused on one DOF systems. The question that remains is: how well do the results in one DOF

translate to a system with more DOFs? We see three approaches to implement PEAs in systems with multiple DOFs:

The first approach is to place a PEA on every joint, such that all joints are moved separately from their initial to their goal position. However, we expect that this is not optimal in systems where the dynamics of different joints influence each other, because during the motion, energy will be transferred from one joint to another. This will make it hard to start and end every movement with the same amount of potential energy in every PEA.

The second approach is to use one joint as the main DOF. In many pick and place tasks, the variations in the pick positions and place positions are small. Therefore, the arm can be designed such that the main distance of the movement is performed by one joint with a PEA and the variations are performed by the other joints. This approach is explored in Chapter 2, where the first joint is connected to a PEA and the second joint is only actuated by a motor. The results show that the reduction in the energy consumption in the one DOF and two DOF system are comparable, showing that this can be a feasible approach.

The third approach would be to design PEAs that connect to multiple joints and are therefore bi-articular, tri-articular, etc. Using such PEAs, the energy could be distributed over multiple joints at the beginning of the movement and could be recaptured from multiple joints at the end of the movement. Chapter 6 provides a framework to analyze such mechanisms.

Depending on the application, the second or the third approach should be pursued. The second approach is the simplest and should therefore be considered first. However, not all applications allow for one joint to be the main DOF. In such systems, the second approach decreases the versatility of the robot too much. The solution is to pursue the third approach, which will lead to a higher versatility.

11.1.3 On the application of locking mechanisms or continuously variable transmissions

The approach that was explored in this thesis was to use locking mechanisms to control the energy inflow and outflow of the PEA. This section discusses the choice for locking mechanisms in comparison to the possible use of a continuously variable transmission (CVT).

Theoretically, with a perfect CVT, the energy inflow and outflow of the spring can be controlled completely. Such a system would perform at least as good as the PEAs described in this thesis and would in most cases perform better. However, current CVT designs are not applicable yet, because they are typically based on a wheel

rolling on a surface. In order to be able to transfer high forces, the wheel should be pushed against the surface to prevent slip. This results in the desired friction between the wheel and the surface, but also increases the friction in the overall system. This friction causes energy losses, that decrease the efficiency of the CVT. Moreover, it will also induce wear, limiting the life time of the CVT.

A CEA can be seen as having a discretely variable transmission between the joint and the spring. In this thesis, we showed that this functionality can be enough to reduce the energy consumption drastically. Chapter 6 showed that with locking mechanisms, the amount of resulting stiffnesses that can be obtained in CEAs, grows exponentially with the number of springs and locking mechanisms. Therefore, we envision that in the future, CEAs will consist of many small springs, many small locking mechanisms and maybe even many small motors. The resulting mechanism can be seen as the robotic equivalent of human muscles. Such actuators will outperform PEAs with CVTs in terms of compactness, life time and efficiency.

11.1.4 On the design of locking mechanisms

Critical in the design of clutched (parallel) elastic actuators is the design of the locking mechanisms. Chapter 3 provides a list of nine properties that are important in locking mechanisms in general: adjustable locking directions, unlocking while under load, low energy consumption, lockable in any position, compact, lightweight, short switching time, inexpensive and high locking force. Here, we discuss the importance of those properties in CEAs and to what extent statically balanced brakes from Chapter 4 are applicable.

In order to be able to apply multiple locking mechanisms in CEAs, the most important properties are unlocking under load, low energy consumption and inexpensiveness. The unlocking under load is required, because when the locking mechanisms have to unlock, the springs are typically loaded. The low energy consumption is important, because otherwise the energy that is saved by implementing the CEA, will be consumed by the locking mechanisms. And the inexpensiveness is important, especially in future CEAs with many locking mechanisms.

Depending on the exact application, compactness and lightweight might be important. In mobile robots, weight and size usually matter and thus the CEA will only be implemented if it is relatively small and lightweight. On the other hand, in robots that are mounted to the ground, weight and size are usually less important, especially when the CEA can be placed on the ground instead of on the links of the robot.

The last four properties might be less important in general, but will contribute to the capabilities of the CEAs. A short switching time will make the timing of the

energy release and recapture easier. Being lockable at every position results in a low backlash, where a high backlash reduces the efficiency of the CEA. A high locking force allows for higher spring forces and thus a larger amount of energy that can be stored. And finally, in some CEAs, adjustable locking directions are preferred, because this causes some locking mechanisms to be unlocked passively by a changing direction.

In chapter 3, we evaluated all locking mechanisms used in robotics. The best available locking mechanism concept is the statically balanced brake since it has a low energy consumption and unlocks well under load. At the moment, SBBs consist of many parts, making the locking mechanism relatively expensive. However, there is no theoretical lower bound on the number of parts of a statically balanced brake and thus they could become inexpensive in the future. Also the compactness and the weight will depend on the ability to reduce the amount of components. The two best alternatives for statically balanced brakes are piezoelectric brakes [114] and electrostatic clutches [48], the most promising of which are the electrostatic clutches. The main disadvantage of electrostatic clutches is that they require high voltages. If such high voltages are available or the required voltage would be reduced, electrostatic clutches would be a promising concept for CEAs.

11.2 Feedforward control in robots

This section discusses the results of the second part of the thesis. It starts with a recapitulation of the results, answering the last four research questions of this thesis. Then two topics are discussed that have played an important role throughout the thesis. Section 11.2.2 discusses the applications of pure feedforward control. And section 11.2.3 compares human feedforward control and robotic feedforward control.

11.2.1 Recapitulation

Regarding open loop control in robotic arms, three questions were addressed. The main results on those questions are:

- *How can the effect of model inaccuracies be eliminated?*
  This question is addressed in Chapter 7, where feedforward controllers are optimized such that the sensitivity to inaccuracies in the friction model are minimized. The results showed that the sensitivity to all the used friction parameters could be eliminated. This resulted in an arm that could move from one position to another, independent of the exact amount of friction. The same technique can be used to reduce the sensitivity to other model parameters, such

as the motor constant and motor resistance [174]. In a one DOF system, the sensitivity to the inertia or the initial position cannot be reduced. However, preliminary results on a two DOF system show that in a system with non-linear dynamics, sensitivities to those parameters can be reduced.

- *How can disturbances be rejected?*
  This question is addressed in Chapter 8, where the repetitive motions are analyzed using limit cycle theory. Small disturbances diminish over time when the motion that the arm performs is a stable limit cycle. Results show that such cycles can be found when there is a parallel spring on the first joint. This spring stabilizes the first joint, while other dynamic effects, such as Coriolis forces, stabilize the other joints. The results in Chapter 8 also show that although it is possible to perform open loop stable cycles on robotic arms, the exact cycle the arm performs can be very sensitive to model inaccuracies.

- *How can both model inaccuracies and disturbances be handled at the same time?*
  This question is addressed in Chapter 9, where the robotic arms learns to perform cycles that are open loop stable, even when the model of the arm is inaccurate. This means that before the cycle is performed without feedback, the arm needs a period of learning that requires feedback. The learning was performed using repetitive control, which allows for feedback with one cycle delay, making camera feedback feasible. The results show that the accuracy is improved with respect to the control setup in Chapter 8. At the pick and the place positions, the arm has a maximum position error of 1-2.5 cm, which is accurate enough for coarse pick-and-place tasks.

- *Are feedforward techniques still useful when a small amount of feedback is available?*
  In Chapter 10, the accuracy of a robotic arms is studied as function of the feedback gain, while the motion is optimized for sensitivity to disturbances. The results show that for all gains, minimizing the sensitivity results in a better accuracy than maximizing the sensitivity. Therefore, pure feedforward techniques can still be useful when (partial) feedback is available.

11.2.2 On the application of feedforward control

The research on feedforward control started because of two reasons. The first reason was that humans use feedforward control and we expected that a more fundamental understanding of the capabilities of feedforward control, would lead to better performance of robots. The second reason was that we had an academic interest to

what is still possible when no feedback is available. Soon after starting the research, we realized that the first reason was not that strong and that the applications were unclear. As stated in chapter 1, we now believe that pure feedforward techniques are useful in applications that have to be very reliable and have to work even when sensory feedback fails.

Next to reliability, we foresee seven reasons for using open loop control. First, *sensors increase the price of a robot.* Although sensors are currently not the most expensive components of robots, future applications might require very cheap and possibly even disposable robots. In such applications, it is preferable to use cheaper, less accurate or no sensors at all. Secondly, *sensor signals might be disturbed* by (for instance) radiation. This might cause the system with feedback to become unstable. Places with a large amount of radiation include nuclear reactors and space. A third reason for using open loop control is because *sensors consume energy.* Although sensors are currently not the largest energy consumers in robots, future applications might require very low energy consuming robots. Therefore, sensorless control will be beneficial in such applications. There are also applications in which *sensors could be too large.* This occurs especially in miniature robots. An example of such an application has been researched by Becker et al. [19], who research miniature drug delivering agents within the human body. A fifth reason for using open loop control is because *processing all signals might become infeasible.* This can be the case when many robots have to be controlled with one computer. Processing all the sensor signals or computing all the control signals might become infeasible. For this reason, Becker and Bretl. [17, 18] researched the use of one control signal to control multiple robotic agents. Next, *feedback is per definition too late.* Since feedback always responds to an error, there will be a tracking error. In theory, feedforward control does not have this tracking error and by using feedforward control, the accuracy could improve. And finally, *humans use feedforward control.* By studying feedforward control in robotic arms, new insights might be obtained regarding how humans control their body.

11.2.3 On human and robotic feedforward control

The last reason mentioned in the previous section for studying feedforward control is because this might lead to new insights into human motion control. This section discusses the differences and similarities between human and robotic feedforward control.

The reasons for using feedforward control in humans and robots are different. Humans use feedforward control because their nervous system introduces large time delays of typically 150 ms [44, 216]. Therefore, high feedback gains will destabilize

the body and humans must partially rely on feedforward control [47]. This would not be a problem if humans had a (nearly) perfect model of their body. However, the internal model of humans is often inaccurate [101], resulting in motions that deviate from the planned motions, which is a problem in tasks where accuracy is important. Therefore, they have to use special techniques to compensate for their slow feedback and inaccurate model.

The techniques used by humans and the techniques used in this thesis are similar: both exploit a task redundancy to increase the performance. In the game of darts for instance, there are infinite ways to hit a certain target. This means that there is a manifold of position-velocity combinations at which the dart can be released such that it will hit the target. When moving along this manifold, the timing of the release is not important and can be performed with feedforward control [37]. Similar motions can be observed in the game of skittles [146, 147]. Another example of a redundant task is hammering, where the only thing that matters is that the hammer hits the nail vertically. Therefore, the path towards the nail can be chosen and variability in that path is allowed. Humans have shown the ability to exploit dynamics in order to lower the variability in the part of the task where accuracy is required [148]. The techniques used in this thesis also exploit task redundancy. When the task is to move from one position to another within a certain time, the trajectories in between can be chosen freely. The research in this thesis shows that the choice for a certain trajectory influences the stability and accuracy.

Chapter 10 shows an interesting parallel between human motion control and robot feedforward control: smooth motions perform well. In research on human motion control, there is an ongoing debate about the cost function humans optimize their motions for. Suggested cost functions are the maximum jerk [59, 92], change of torque [227] and sensitivity to motor noise [81]. Other researchers suggest that humans perform some kind of stochastic optimal control in which variability in task irrelevant directions is ignored [218]. What makes this debate difficult is the fact that all those cost functions result in smooth motions that are approximately similar [81]. This shows similarities with the smooth motions in the feedback-feedforward controlled system in Chapter 10. All smooth motions seem to perform approximately equal when it comes to accuracy.

11.3 General conclusions

The main conclusions of this research are:

- The implementation of a clutched elastic actuator in parallel with the motor can reduce the energy consumption of robots with 65%. The setup with a differential and two locking mechanisms causes the versatility to remain high.
- Statically balanced brakes solve the problem of regular friction based locking mechanisms that a large actuation force is needed. The concept of statically balanced brakes allows for friction based locking mechanisms with an actuation force that is reduced with 95-97% in comparison to regular friction based locking mechanisms, while being relatively small.
- Feedforward control on robotic arms is possible when there are disturbances, model inaccuracies or both.

11.4 Future directions

This final section discusses two directions for future research.

11.4.1 Clutched Elastic Actuators

The CEAs in this thesis have snown the ability of CEAs to reduce the energy consumption of robots drastically. A reduction of 65% was reached and with more efficient components, even higher reductions will be obtained. The most important limitation of the current design is that it is only suited for robots with rest-to-rest motions. In order to reduce the energy consumption of all robotic devices, it is crucial that the versatility of CEAs is increased further. This will have a large impact on the field of mobile robots, since a reduction of 80% means that the uptime of the robot is increased with a factor 5. One of the key aspects of more versatile CEAs is the question how to increase the number of springs and locking mechanisms while having a compact and lightweight design. Especially the design of small and lightweight locking mechanisms is a challenging research topic. Next to statically balanced brakes, other actuation principles such as electrostatic actuators should be investigated.

11.4.2 Gearboxes

One of the main contributors to energy losses in robots are the gearboxes. Most robots use either planetary gears or harmonic drives as gearbox in between the motors and the joints. Both suffer from friction and sometimes non-backdrivability, especially when the transfer ratios are large. On the other hand, gearbox ratios cannot be taken too small, because this will increase the copper losses in the motors. In transferring linear motions into rotational motion or vice versa, this problem has mostly been

solved by the introduction of spindle drives. The efficiency of such spindles are high because they rely on rolling contacts. An interesting direction for future research would be to develop a gearbox that is based on rolling contacts.

A

Division of work in shared first authorships

Chapters 8, 9 and 10 are papers that were written in close collaboration with Wouter Wolfslag, leading to shared first authorships. The fact that both authors contributed equally to those chapters does not mean that both authors contributed equally to all parts of the chapters. The goal of this appendix is to describe the task division as clearly as possible. Note however that in reality the task division has been less clear, because all ideas were discussed extensively between the two first authors.

A.1 Chapter 8: Feedforward control and stability

The main idea for this chapter came from a discussion between Wouter Wolfslag, Martijn Wisse and Michiel Plooij. Michiel Plooij performed the initial (one DOF) simulations for this paper, showing the feasibility of the approach. Wouter Wolfslag performed the optimizations as presented in the chapter. Michiel Plooij then implemented the results of the optimizations on the two DOF robotic arm and Wouter Wolfslag performed the hardware experiments on the pendulum setup. Finally, Michiel Plooij analyzed the basin of attraction in simulation.

A.2 Chapter 9: Robust open loop stable manipulation

The main idea of designing a robust open loop stable cycle in simulation and learning it on the hardware setup came from Wouter Wolfslag. Wouter Wolfslag also designed the algorithm to find those cycles. Michiel Plooij designed the Repetitive Controller and performed the hardware experiments.

A.3 Chapter 10: Feedforward with low gain feedback

The main idea for this chapter came from Michiel Plooij. Wouter Wolfslag then proposed to use a norm, similar to the Gait Sensitivity Norm and designed the algorithm to calculate the MSN. Finally, Michiel Plooij performed the hardware experiments.

References

[1] Ajoudani, A., Gabiccini, M., Tsagarakis, N., and Bicchi, A. (2013). *"Human-like impedance and minimum effort control for natural and efficient manipulation"*. IEEE International Conference Robotics and Automation (ICRA), pp. 4499–4505.

[2] Akella, S. and Mason, M.T. (1992). *"Posing polygonal objects in the plane by pushing"*. Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on, pp. 2255–2262.

[3] Akinfiev, T. (1985). *"Resonance mechanical hand"*.

[4] Akinfiev, T. (1992). *"Resonance drive"*.

[5] Akinfiev, T., Fernandez, R., and Armada, M. (2006). *"Nontraditional drives for walking robots"*. Climbing and Walking Robots, pp. 727–734.

[6] Alexander, R. and Bennet-Clark, H. (1977). *"Storage of elastic strain energy in muscle and other tissues"*. Nature, 265(5590), pp. 114–117.

[7] Almen, J. and Laszlo, A. (1936). *"The uniform-section disk spring"*. Trans. ASME, 58, pp. 305–314.

[8] Ansari, A. and Murphey, T. (2013). *"Minimal parametric sensitivity trajectories for nonlinear systems"*. American Controls Conference (ACC).

[9] Arisumi, H., Miossec, S., Chardonnet, J.R., and Yokoi, K. (2008). *"Dynamic lifting by whole body motion of humanoid robots"*. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 668–675.

[10] Arnold, J.H., DeWitt, M.D., Perisho, R.J., and Soucie, W.L. (1993). *"Electronically controlled parking brake system"*. US Patent 5,180,038.

[11] Au, S. and Herr, H. (2008). *"Powered ankle-foot prosthesis"*. Robotics & Automation Magazine, IEEE, 15(3), pp. 52–59.

[12] Au, S., Herr, H., Weber, J., and Martinez-Villalpando, E. (2007). *"Powered ankle-foot prosthesis for the improvement of amputee ambulation"*. Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE, pp. 3020–3026.

[13] Babitsky, V.I. and Shipilov, A. (2003). *"Resonant robotic systems"*. Springer.

[14] Baca, J., Hossain, S., Dasgupta, P., Nelson, C.A., and Dutta, A. (2014). *"Modred: Hardware design and reconfiguration planning for a high dexterity*

modular self-reconfigurable robot for extra-terrestrial exploration". Robotics and Autonomous Systems, 62(7), pp. 1002 – 1015.

[15] Banaszuk, A. and Hauser, J. (1995). *"Feedback linearization of transverse dynamics for periodic orbits"*. Systems & control letters, 26(2), pp. 95–105.

[16] Baser, O. and Konukseven, E.I. (2010). *"Theoretical and experimental determination of capstan drive slip error"*. Mechanism and Machine Theory, 45(6), pp. 815 – 827.

[17] Becker, A. and Bretl, T. (2012). *"Approximate steering of a plate-ball system under bounded model perturbation using ensemble control"*. Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on, pp. 5353–5359.

[18] Becker, A. and Bretl, T. (2012). *"Approximate steering of a unicycle under bounded model perturbation using ensemble control"*. IEEE Transactions on Robotics, 28(3), pp. 580–591.

[19] Becker, A., Felfoul, O., and Dupont, P.E. (2014). *"Simultaneously powering and controlling many actuators with a clinical mri scanner"*. Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, pp. 2017–2023.

[20] Belov Vladimir, B. (1992). *"Resonance manipulator modulus"*.

[21] Berger, E. (2002). *"Friction modeling for dynamic system simulation"*. Applied Mechanics Reviews, 55(6), pp. 535–577.

[22] Bickford, J.H. (1972). *"Mechanisms for intermittent motion"*. Industrial Press New York.

[23] Bliman, P. and Sorine, M. (1993). *"Friction modeling by hysteresis operators. application to dahl, sticktion and stribeck effects"*. Pitman Research Notes in Mathematics Series, pp. 10–10.

[24] Boyd, S.P., Barratt, C.H., Boyd, S.P., and Boyd, S.P. (1991). *"Linear controller design: limits of performance"*. Prentice Hall Englewood Cliffs, NJ.

[25] Brackx, B., Van Damme, M., Matthys, A., Vanderborght, B., and Lefeber, D. (2013). *"Passive ankle-foot prosthesis prototype with extended push-off"*. Int J Adv Robotic Sy, 10(101).

[26] Brown, E., Rodenberg, N., Amend, J., Mozeika, A., Steltz, E., Zakin, M.R., Lipson, H., and Jaeger, H.M. (2010). *"Universal robotic gripper based on the jamming of granular material"*. Proceedings of the National Academy of Sciences, 107(44), pp. 18809–18814.

[27] Brown, W.R. and Ulsoy, A.G. (2011). *"A passive-assist design approach for improved reliability and efficiency of robot arms"*. Robotics and Automation (ICRA), 2011 IEEE International Conference on, pp. 4927–4934.

[28] Burden, S., Revzen, S., and Sastry, S. (2011). *"Dimension reduction near periodic orbits of hybrid systems"*. Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on, pp. 6116–6121.

[29] Chen, J. and Liao, W. (2010). *"Design, testing and control of a magnetorheological actuator for assistive knee braces"*. Smart Materials and Structures, 19(3), p. 035029.

[30] Cherelle, P., Grosu, V., Beyl, P., Mathys, A., Van Ham, R., Van Damme, M., Vanderborght, B., and Lefeber, D. (2010). *"The maccepa actuation system as torque actuator in the gait rehabilitation robot altacro"*. IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), pp. 27–32.

[31] Cherelle, P., Grosu, V., Cestari, M., Vanderborght, B., and Lefeber, D. (2014). *"The amp-foot 3 - new generation propulsive prosthetic feet with explosive motion characteristics - design and validation"*. Under review.

[32] Cherelle, P., Grosu, V., Matthys, A., Vanderborght, B., and Lefeber, D. (2014). *"Design and validation of the ankle mimicking prosthetic (amp-) foot 2.0"*. Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 22(1), pp. 138–148.

[33] Cherelle, P., Junius, K., Grosu, V., Cuypers, H., Vanderborght, B., and Lefeber, D. (2014). *"The amp-foot 2.1: actuator design, control and experiments with an amputee"*. Robotica, 32(08), pp. 1347–1361.

[34] Cherry, M.S., Kota, S., and Ferris, D.P. (2009). *"An elastic exoskeleton for assisting human running"*. ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 727–738.

[35] Chesi, G. (2011). *"Domain of attraction: Analysis and control via sos programming"*. Springer.

[36] Chou, J., Yu, K., and Wu, M. (2012). *"Electrothermally actuated lens scanner and latching brake for free-space board-to-board optical interconnects"*. Microelectromechanical Systems, Journal of, 21(5), pp. 1107–1116.

[37] Cohen, R.G. and Sternad, D. (2012). *"State space analysis of timing: exploit-*

ing task redundancy to reduce sensitivity to timing". Journal of Neurophysiology, 107(2), pp. 618–627.

[38] Collins, S. and Ruina, A. (2005). *"A bipedal walking robot with efficient and human-like gait"*. Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on, pp. 1983–1988.

[39] Collins, S., Ruina, A., Tedrake, R., and Wisse, M. (2005). *"Efficient bipedal robots based on passive-dynamic walkers"*. Science, 307(5712), pp. 1082–1085.

[40] Collins, S.H. and Kuo, A.D. (2010). *"Recycling energy to restore impaired ankle function during human walking"*. PLoS one, 5(2), p. e9307.

[41] Collins, S.H., Wiggin, M.B., and Sawicki, G.S. (2015). *"Reducing the energy cost of human walking using an unpowered exoskeleton"*. Nature.

[42] Controzzi, M., Cipriani, C., and Carrozza, M.C. (2010). *"Miniaturized non-back-drivable mechanism for robotic applications"*. Mechanism and Machine Theory, 45(10), pp. 1395 – 1406.

[43] Cordo, P., Carlton, L., Bevan, L., Carlton, M., and Kerr, G.K. (1994). *"Proprioceptive coordination of movement sequences: role of velocity and position information"*. Journal of Neurophysiology, 71(5), pp. 1848–1861.

[44] Cordo, P., Carlton, L., Bevan, L., Carlton, M., and Kerr, G.K. (1994). *"Proprioceptive coordination of movement sequences: role of velocity and position information"*. Journal of Neurophysiology, 71(5), pp. 1848–1861.

[45] Cortez, M. and Forner-Cordero, A. (2015). *"On the study of a clutch device for exoskeletons and robot joints: Energetic efficiency study and mechanism concept"*. XVII International Symposium on Dynamic Problems of Mechanics.

[46] Dahl, P. (1968). *"A solid friction model"*. Technical report, DTIC Document.

[47] Desmurget, M. and Grafton, S. (2000). *"Forward modeling allows feedback control for fast reaching movements"*. Trends in cognitive sciences, 4(11), pp. 423–431.

[48] Diller, S., Majidi, C., and Collins, S. (2015). *"Clutching with high force, high bandwidth, low mass, and low energy consumption using electrostatics"*. Dynamic walking.

[49] Dollar, A.M. and Herr, H. (2008). *"Lower extremity exoskeletons and active orthoses: challenges and state-of-the-art"*. Robotics, IEEE Transactions on, 24(1), pp. 144–158.

[50] Dunning, A.G., Tolou, N., and Herder, J.L. (2013). *"A compact low-stiffness six degrees of freedom compliant precision stage"*. Precision Engineering, 37(2), pp. 380 – 388.

[51] Dunning, A.G., Tolou, N., Pluimers, P., Kluit, L., and Herder, J.L. (2012). *"Bistable compliant mechanisms: Corrected finite element modeling for stiffness tuning and preloading incorporation"*. Journal of Mechanical Design, 134(8), p. 084502.

[52] Dupont, P.E. (1992). *"The effect of coulomb friction on the existence and uniqueness of the forward dynamics problem"*. Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on, pp. 1442–1447.

[53] Elliott, G., Marecki, A., and Herr, H. (2014). *"Design of a clutch–spring knee exoskeleton for running"*. Journal of Medical Devices, 8(3), p. 031002.

[54] Elliott, G., Sawicki, G.S., Marecki, A., and Herr, H. (2013). *"The biomechanics and energetics of human running using an elastic knee exoskeleton"*. IEEE International Conference on Rehabilitation Robotics (ICORR).

[55] Endo, K. and Herr, H. (2009). *"A model of muscle-tendon function in human walking"*. Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, pp. 1909–1915.

[56] Endo, K., Paluska, D., and Herr, H. (2006). *"A quasi-passive model of human leg function in level-ground walking"*. Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on, pp. 4935–4939.

[57] Eslamy, M., Grimmer, M., and Seyfarth, A. (2012). *"Effects of unidirectional parallel springs on required peak power and energy in powered prosthetic ankles: Comparison between different active actuation concepts"*. Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on, pp. 2406–2412.

[58] Ferris, D.P., Louie, M., and Farley, C.T. (1998). *"Running in the real world: adjusting leg stiffness for different surfaces"*. Proceedings of the Royal Society of London. Series B: Biological Sciences, 265(1400), pp. 989–994.

[59] Flash, T. and Hogan, N. (1985). *"The coordination of arm movements: an experimentally confirmed mathematical model"*. The journal of Neuroscience, 5(7), pp. 1688–1703.

[60] Flynn, L., Geeroms, J., Jimenez-Fabian, R., Vanderborght, B., Vitiello, N., and Lefeber, D. (2014). *"Ankle–knee prosthesis with active ankle and en-*

ergy transfer: Development of the cyberlegs alpha-prosthesis". Robotics and Autonomous Systems.

[61] Friedland, B. (2012). *"Control system design: an introduction to state-space methods"*. Courier Corporation.

[62] Fukuda, T., Nakagawa, S., Kawauchi, Y., and Buss, M. (1989). *"Structure decision method for self organising robots based on cell structures-cebot"*. IEEE/ASME Proceedings International Conference on Robotics and Automation (ICRA), volume 2, pp. 695–700.

[63] Gantmacher, F.R. (1960). *"The theory of matrices"*, volume 2. Taylor & Francis US.

[64] Geeroms, J., Flynn, L., Jimenez-Fabian, R., Vanderborght, B., and Lefeber, D. (2013). *"Ankle-knee prosthesis with powered ankle and energy transfer for cyberlegs α-prototype"*. IEEE International Conference on Rehabilitation Robotics (ICORR), pp. 1–6.

[65] Gerelli, O., Carloni, R., and Stramigioli, S. (2009). *"Port-based modeling and optimal control for a new very versatile energy efficient actuator"*. 9th IFAC Symposium on Robot Control.

[66] Gilbert, J. and Abu Hassan, A. (1998). *"Design and control of a multi-mode drive system"*. Advanced Motion Control, 1998. AMC '98-Coimbra., 1998 5th International Workshop on, pp. 611–616.

[67] Gilpin, K., Kotay, K., Rus, D., and Vasilescu, I. (2008). *"Miche: Modular shape formation by self-disassembly"*. The International Journal of Robotics Research, 27(3-4), pp. 345–372.

[68] Gilpin, K. and Rus, D. (2010). *"Modular robot systems"*. Robotics & Automation Magazine, IEEE, 17(3), pp. 38–55.

[69] Goris, K., Saldien, J., Vanderborght, B., and Lefeber, D. (2011). *"How to achieve the huggable behavior of the social robot probo? a reflection on the actuators"*. Mechatronics, 21(3), pp. 490–500.

[70] Gosline, A. and Hayward, V. (2008). *"Eddy Current Brakes for Haptic Interfaces: Design, Identification, and Control"*. IEEE/ASME Transactions on Mechatronics, 13(6), pp. 669 –677.

[71] Gosselin, C. and Angeles, J. (1990). *"Singularity analysis of closed-loop kinematic chains"*. Robotics and Automation, IEEE Transactions on, 6(3), pp. 281–290.

[72] Goswami, A., Espiau, B., and Keramane, A. (1996). *"Limit cycles and their stability in a passive bipedal gait"*. Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on, volume 1, pp. 246–251.

[73] Goswami, A., Espiau, B., and Keramane, A. (1996). *"Limit cycles and their stability in a passive bipedal gait"*. Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on, volume 1, pp. 246–251.

[74] Grez, J., Gueble, J., Wood, J., Nelson, D., Miller, K., Hanela, C., Taber, B., Taylor, R., Hall, S., Rens, P., et al. (2010). *"Method for tuning a spring element used in a resonant driving system for an appliance which includes a work piece"*. EP Patent App. EP20,100,178,883.

[75] Gu, H., Ceccarelli, M., and Carbone, G. (2009). *"An experimental characterization of a 1-dof anthropomorphic arm for humanoid robots"*. 13th WSEAS International Conference on Computers, pp. 92–99.

[76] Guarneri, P., Mastinu, G., Gobbi, M., Cantoni, C., and Sicigliano, R. (2014). *"Brake energy efficiency"*. Journal of Mechanical Design.

[77] Haeufle, D.F.B., Taylor, M.D., Schmitt, S., and Geyer, H. (2012). *"A clutched parallel elastic actuator concept: Towards energy efficient powered legs in prosthetics and robotics"*. IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), pp. 1614–1619.

[78] Ham, R., Sugar, T., Vanderborght, B., Hollander, K., and Lefeber, D. (2009). *"Compliant actuator designs"*. Robotics Automation Magazine, IEEE, 16(3), pp. 81–94.

[79] Ham, R.v., Sugar, T., Vanderborght, B., Hollander, K., and Lefeber, D. (2009). *"Compliant actuator designs"*. Robotics & Automation Magazine, IEEE, 16(3), pp. 81–94.

[80] Hanley, M.G., Caliendo, G.P., and Anderson, D.B. (1999). *"Actuator having piezoelectric braking element"*. US Patent 5,986,369.

[81] Harris, C.M. and Wolpert, D.M. (1998). *"Signal-dependent noise determines motor planning"*. Nature, 394(6695), pp. 780–784.

[82] Harris, C.M. and Wolpert, D.M. (1998). *"Signal-dependent noise determines motor planning"*. Nature, 394(6695), pp. 780–784.

[83] Herder, J. (2001). *"Energy-free systems: Theory, conception and design of statically balanced spring mechanisms"*.

[84] Herder, J.L. (2005). *"Development of a statically balanced arm support: Ar-*

mon". Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on, pp. 281–286.

[85] Herr, H. and Wilkenfeld, A. (2003). *"User-adaptive control of a magnetorheological prosthetic knee"*. Industrial Robot: An International Journal, 30(1), pp. 42–55.

[86] Hild, M., Siedel, T., and Geppert, T. (2011). *"Design of a passive, bidirectional overrunning clutch for rotary joints of autonomous robots"*. Intelligent Robotics and Applications, volume 7101 of *Lecture Notes in Computer Science*, pp. 397–405.

[87] Hirzinger, G., Albu-Schaffer, A., Hahnle, M., Schaefer, I., and Sporer, N. (2001). *"On a new generation of torque controlled light-weight robots"*. IEEE International Robotics and Automation (ICRA), volume 4, pp. 3356–3363.

[88] Hirzinger, G., Fischer, M., Brunner, B., Koeppe, R., Otter, M., Grebenstein, M., and Schäfer, I. (1999). *"Advances in robotics: The dlr experience"*. The International Journal of Robotics Research, 18(11), pp. 1064–1087.

[89] Hobbelen, D. (2008). *"Limit cycle walking, chapter 6"*.

[90] Hobbelen, D. and Wisse, M. (2007). *"A disturbance rejection measure for limit cycle walkers: The gait sensitivity norm"*. Robotics, IEEE Transactions on, 23(6), pp. 1213–1224.

[91] Hobbelen, D.G.E. and Wisse, M. (2007). *"A disturbance rejection measure for limit cycle walkers: The gait sensitivity norm"*. Robotics, IEEE Transactions on, 23(6), pp. 1213–1224.

[92] Hogan, N. (1984). *"An organizing principle for a class of voluntary movements"*. The Journal of Neuroscience, 4(11), pp. 2745–2754.

[93] Howell, L.L., Magleby, S.P., and Olsen, B.M. (2013). *"Handbook of compliant mechanisms"*. Wiley Online Library.

[94] Hürmüzlü, Y. and Moskowitz, G. (1986). *"The role of impact in the stability of bipedal locomotion"*. Dynamics and Stability of Systems, 1(3), pp. 217–234.

[95] Hürmüzlü, Y. and Moskowitz, G. (1986). *"The role of impact in the stability of bipedal locomotion"*. Dynamics and Stability of Systems, 1(3), pp. 217–234.

[96] Hurst, J. (2004). *"An actuator with mechanically adjustable series compliance"*.

[97] Ingen Schenau, G. van (1984). *"An alternative view of the concept of utilisation of elastic energy in human movement"*. Human Movement Science, 3(4), pp.

301–336.

[98] Ishikawa, M., Komi, P.V., Grey, M.J., Lepola, V., and Bruggemann, G.P. (2005). *"Muscle-tendon interaction and elastic energy usage in human walking"*. Journal of applied physiology, 99(2), pp. 603–608.

[99] Karssen, D.J. and Wisse, M. (2012). *"Running robot phides"*. Dynamic Walking Conference.

[100] Karssen, J.D. and Wisse, M. (2011). *"Running with improved disturbance rejection by using non-linear leg springs"*. The International Journal of Robotics Research, 30(13), pp. 1585–1595.

[101] Kawato, M. (1999). *"Internal models for motor control and trajectory planning"*. Current Opinion in Neurobiology, 9(6), pp. 718 – 727.

[102] Kern, N.I., Triolo, R.J., Kobetic, R., Quinn, R.D., and Majewski, T.J. (2009). *"A locking compliant device inspired by the anatomy of the spine"*. Journal of Mechanical Design, 131(1), pp. 014501–1–014501–3.

[103] Khoramshahi, M., Parsa, A., Ijspeert, A., and Ahmadabadi, M.N. (2014). *"Natural dynamics modification for energy efficiency: A data-driven parallel compliance design method"*. Proceedings of 2014 IEEE International Conference on Robotics and Automation, pp. 2412–2417.

[104] Kikuchi, T., Tanida, S., Otsuki, K., Yasuda, T., and Furusho, J. (2010). *"Development of third-generation intelligently controllable ankle-foot orthosis with compact mr fluid brake"*. IEEE International Conference on Robotics and Automation (ICRA), pp. 2209–2214.

[105] Kim, C.H., Yonekura, K., Tsujino, H., and Sugano, S. (2011). *"Physical control of the rotation of a flexible objectâĂŤrope turning with a humanoid robot"*. Advanced robotics, 25(3-4), pp. 491–506.

[106] Kim, J. and Choi, S. (2011). *"Design and modeling of a clutch actuator system with self-energizing mechanism"*. Mechatronics, IEEE/ASME Transactions on, 16(5), pp. 953–966.

[107] Klawuhn, M. (1983). *"Oscillating drive for small electrical apparatuses"*.

[108] Kossett, A. (2013). *"Design principles for miniature rotary-wing hybrid-locomotion robots"*. Master's thesis, University of Minnisota.

[109] Kossett, A. and Papanikolopoulos, N. (2011). *"A robust miniature robot design for land/air hybrid locomotion"*. IEEE International Conference on Robotics and Automation (ICRA), pp. 4595–4600.

[110] Kragten, G.A. and Herder, J.L. (2010). *"The ability of underactuated hands to grasp and hold objects"*. Mechanism and Machine Theory, 45(3), pp. 408–425.

[111] Kubo, K., Kawakami, Y., and Fukunaga, T. (1999). *"Influence of elastic properties of tendon structures on jump performance in humans"*. Journal of Applied Physiology, 87(6), pp. 2090–2096.

[112] Kuo, A. (2002). *"The relative roles of feedforward and feedback in the control of rhythmic movements."* Motor control, 6(2), pp. 129–145.

[113] Kurichh, S.L. and Acre, L.R. (1975). *"Air operated spring brake"*. US Patent 3,926,094.

[114] Laffranchi, M., Tsagarakis, N., and Caldwell, D. (2010). *"A variable physical damping actuator (vpda) for compliant robotic joints"*. IEEE International Conference on Robotics and Automation (ICRA), pp. 1668–1674.

[115] Lapeyre, M., Rouanet, P., and Oudeyer, P.Y. (2013). *"The Poppy Humanoid Robot: Leg Design for Biped Locomotion"*. IEEE/RSJ International Conference on Intelligent Robots and Systems.

[116] Leach, D., Gunther, F., Maheshwari, N., and Iida, F. (2014). *"Linear multimodal actuation through discrete coupling"*. Mechatronics, IEEE/ASME Transactions on, 19(3), pp. 827–839.

[117] Li, B., Deng, Q., and Liu, Z. (2009). *"A spherical hopping robot for exploration in complex environments"*. IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 402–407.

[118] Li, Q., Naing, V., Hoffer, J., Weber, D., Kuo, A., and Donelan, J.M. (2008). *"Biomechanical energy harvesting: Apparatus and method"*. IEEE International Conference on Robotics and Automation (ICRA), pp. 3672–3677.

[119] Limpert, R. (1992). *"Brake design and safety"*, volume 120.

[120] Linde, R. van der and Schwab, A. (1997). *"Lecture notes on multibody dynamics b, wb1413"*. Delft University of Technology.

[121] Linde, R. van der and Schwab, A. (1997/1998). *"Lecture notes on multibody dynamics b, wb1413"*. Delft University of Technology.

[122] Lohmiller, W. and Slotine, J. (1999). *"Contraction analysis of nonlinear systems"*. Ph.D. thesis, Massachusetts Institute of Technology, Dept. of Mechanical Engineering.

[123] Longman, R.W. (2000). *"Iterative learning control and repetitive control for*

engineering practice". International Journal of Control, 73(10), pp. 930–954.

[124] Mahmoud Tavakoli, L.M. and Almeida, A.T. de (2013). *"Flexirigid, a novel two phase flexible gripper"*. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

[125] Majumdar, A. (2013). *"Robust online motion planning with reachable sets"*. Ph.D. thesis, Massachusetts Institute of Technology.

[126] Manchester, I. (2010). *"Transverse dynamics and regions of stability for nonlinear hybrid limit cycles"*. arXiv preprint arXiv:1010.2241.

[127] Manchester, I. and Slotine, J. (2012). *"Contraction criteria for existence, stability, and robustness of a limit cycle"*. arXiv preprint arXiv:1209.4433.

[128] Martin, J. (1974). *"Mécanismes à mouvements intermittents"*. Dunod.

[129] Martinez-Villalpando, E.C. and Herr, H. (2009). *"Agonist-antagonist active knee prosthesis: A preliminary study in level-ground walking"*. J. Rehabil. Res. Dev, 46(3), pp. 361–374.

[130] Mathijssen, G., Cherelle, P., Lefeber, D., and Vanderborght, B. (2013). *"Concept of a series-parallel elastic actuator for a powered transtibial prosthesis"*. Actuators, volume 2, pp. 59–73.

[131] Mathijssen, G., Lefeber, D., and Vanderborght, B. (2015). *"Variable recruitment of parallel elastic elements: Series–parallel elastic actuators (spea) with dephased mutilated gears"*. Mechatronics, IEEE/ASME Transactions on, 20(2), pp. 594–602.

[132] Mauch, H.A. (1968). *"Stance control for above-knee artificial legs–design considerations in the sns knee"*. Bulletin of Prosthetics Research, (10), pp. 61–72.

[133] McGeer, T. (1990). *"Passive dynamic walking"*. the international journal of robotics research, 9(2), pp. 62–82.

[134] McGeer, T. (1990). *"Passive dynamic walking"*. The International Journal of Robotics Research, 9(2), pp. 62–82.

[135] Mei, Y., Lu, Y.H., Hu, Y., and Lee, C.S.G. (2004). *"Energy-efficient motion planning for mobile robots"*. IEEE/ASME Proceedings International Conference on Robotics and Automation (ICRA) Conference on, volume 5, pp. 4344–4349 Vol.5.

[136] Mettin, U., La Hera, P.X., Freidovich, L.B., and Shiriaev, A.S. (2009). *"Parallel*

elastic actuators as control tool for preplanned trajectories of underactuated mechanical systems". The international journal of robotics research.

[137] Mitsui, K., Ozawa, R., and Kou, T. (2013). *"An under-actuated robotic hand for multiple grasps"*. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

[138] Mombaur, K., Longman, R., Bock, H., and Schlöder, J. (2005). *"Open-loop stable running"*. Robotica, 23(1), pp. 21–33.

[139] Mombaur, K.D., Bock, H.G., Schlöder, J.P., and Longman, R.W. (2005). *"Open-loop stable solutions of periodic optimal control problems in robotics"*. ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik, 85(7), pp. 499–515.

[140] Mombaur, K.D., Bock, H.G., Schlöder, J.P., and Longman, R.W. (2005). *"Open-loop stable solutions of periodic optimal control problems in robotics"*. ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik, 85(7), pp. 499–515.

[141] Mombaur, K.D., Longman, R.W., Bock, H.G., and Schlöder, J.P. (2005). *"Open-loop stable running"*. Robotica, 23(1), pp. 21–33.

[142] Mombaur, K.D., Longman, R.W., Bock, H.G., and Schlöder, J.P. (2005). *"Open-loop stable running"*. Robotica, 23(1), pp. 21–33.

[143] Morita, T. and Sugano, S. (1995). *"Design and development of a new robot joint using a mechanical impedance adjuster"*. IEEE International Conference on Robotics and Automation (ICRA), volume 3, pp. 2469–2475.

[144] Morita, T. and Sugano, S. (1995). *"Development of one-dof robot arm equipped with mechanical impedance adjuster"*. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), volume 1, pp. 407–412.

[145] Morrone, R. (2010). *"Arrangement and process for mounting a resonant spring in a refrigeration compressor"*. URL http://www.google.co.in/patents/US20100310396. US Patent App. 12/809,121.

[146] Müller, H. and Loosch, E. (1999). *"Functional variability and an equifinal path of movement during targeted throwing"*. Journal of Human Movement Studies, 36, pp. 103–126.

[147] Müller, H. and Sternad, D. (2004). *"Decomposition of variability in the execution of goal-oriented tasks: Three components of skill improvement."* Journal of Experimental Psychology: Human Perception and Performance, 30(1), pp.

212–233.

[148] Müller, H. and Sternad, D. (2009). *"Motor learning: changes in the structure of variability in a redundant task"*. Progress in motor control, pp. 439–456.

[149] Munir, S., Tognetti, L., and Book, W. (1999). *"Experimental evaluation of a new braking system for use in passive haptic displays"*. American Control Conference, volume 6, pp. 4456–4460 vol.6.

[150] Naidu, D.S. (2002). *"Optimal control systems"*, volume 2. CRC press.

[151] Ollero, A. and Merino, L. (2004). *"Control and perception techniques for aerial robotics"*. Annual reviews in Control, 28(2), pp. 167–178.

[152] Olsson, H., Åström, K.J., Wit, C. Canudas de, Gäfvert, M., and Lischinsky, P. (1998). *"Friction models and friction compensation"*. European journal of control, 4(3), pp. 176–195.

[153] Oort, G. van, Carloni, R., Borgerink, D.J., and Stramigioli, S. (2011). *"An energy efficient knee locking mechanism for a dynamically walking robot"*. IEEE International Conference on Robotics and Automation (ICRA), 2011, pp. 2003–2008.

[154] Orthwein, W.C. (2004). *"Clutches and brakes: design and selection"*. CRC Press.

[155] Ottobock (2014). *"Ottobock"*. http://professionals.ottobockus.com/. [Online; accessed 10-December-2014].

[156] Palpacelli, M., Carbonari, L., and Palmieri, G. (2014). *"A lockable spherical joint for robotic applications"*. Mechatronic and Embedded Systems and Applications (MESA), 2014 IEEE/ASME 10th International Conference on, pp. 1–6.

[157] Parietti, F., Baud-Bovy, G., Gatti, E., Riener, R., Guzzella, L., and Vallery, H. (2011). *"Series Viscoelastic Actuators Can Match Human Force Perception"*. IEEE Transactions on Mechatronics, 16(5), pp. 853–860.

[158] Park, F. and Kim, J.W. (1998). *"Manipulability and singularity analysis of multiple robot systems: a geometric approach"*. Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on, volume 2, pp. 1032–1037 vol.2.

[159] Parmerlee, J.K. et al. (1973). *"Bi-stable brake"*. US Patent 3,741,353.

[160] Parrott, C., Dodd, T., and Gross, R. (2014). *"Higen: A high-speed gender-*

less mechanical connection mechanism with single-sided disconnect for self-reconfigurable modular robots". Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, pp. 3926–3932.

[161] Patterson, M.A. and Rao, A.V. (2013). *"Gpops- ii: A matlab software for solving multiple-phase optimal control problems using hp–adaptive gaussian quadrature collocation methods and sparse nonlinear programming"*. ACM Transactions on Mathematical Software, 39(3), pp. 1–41.

[162] Peabody, C.G. and Roberts, R.W. (1936). *"Automatic parking brake"*. US Patent 2,031,062.

[163] Peerdeman, B., Pieterse, G., Stramigioli, S., Rietman, H., Hekman, E., Brouwer, D., and Misra, S. (2012). *"Design of joint locks for underactuated fingers"*. Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS EMBS International Conference on, pp. 488–493.

[164] Peerdeman, B., Stramigioli, S., Hekman, E.E., Brouwer, D.M., and Misra, S. (2013). *"Development of underactuated prosthetic fingers with joint locking and electromyographic control"*. Mechanical Engineering Research, 3(1), p. p130.

[165] Pfeifer, S., Pagel, A., Riener, R., and Vallery, H. (2014). *"Actuator with angle-dependent elasticity for biomimetic transfemoral prostheses"*. Mechatronics, IEEE/ASME Transactions on, PP(99), pp. 1–11.

[166] Picasso, B. and Colaneri, P. (2008). *"A factorization approach for the l_∞-gain of discrete-time linear systems"*. Proc. of the 17-th IFAC world congress, Seoul, ROK, pp. 1299–1304.

[167] Plooij, M., De Vries, M., Wolfslag, W., and Wisse, M. (2013). *"Optimization of feedforward controllers to minimize sensitivity to model inaccuracies"*. Intelligent Robots and Systems (IROS), Proceedings of the IEEE/RSJ International Conference on.

[168] Plooij, M., Dunning, A., and Wisse, M. (2015). *"Statically balanced brakes"*. Under review.

[169] Plooij, M., Mathijssen, G., Cherelle, P., Lefeber, D., and Vanderborght, B. (2015). *"Lock your robot: A review of locking devices in robotics"*. Robotics Automation Magazine, IEEE, 22(1), pp. 106–117.

[170] Plooij, M., Van Nunspeet, M., Wisse, M., and Vallery, H. (2015). *"Design of the bi-directional clutched parallel elastic actuator (bic-pea)"*. Robotics and

Automation (ICRA), IEEE International Conference on, pp. 1002–1009.

[171] Plooij, M. and Wisse, M. (2012). *"A novel spring mechanism to reduce energy consumption of robotic arms"*. Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, pp. 2901 –2908.

[172] Plooij, M., Wolfslag, W., and Wisse, M. (2014). *"Open loop stable control in repetitive manipulation tasks"*. Robotics and Automation (ICRA), 2014 IEEE/RSJ International Conference on.

[173] Plooij, M., Wolfslag, W., and Wisse, M. (2015). *"Clutched elastic actuators"*. Under review.

[174] Plooij, M., Wolfslag, W., and Wisse, M. (2015). *"Robust feedforward control of robotic arms with friction model uncertainty"*. Robotics and Autonomous Systems, 70(0), pp. 83 – 91.

[175] Pratt, G.A. and Williamson, M.M. (1995). *"Series elastic actuators"*. Intelligent Robots and Systems 95.'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on, volume 1, pp. 399–406.

[176] Pratt, J.E. and Krupp, B.T. (2004). *"Series elastic actuators for legged robots"*. Defense and Security, pp. 135–144.

[177] Pratt, J.E., Krupp, B.T., Morse, C.J., and Collins, S.H. (2004). *"The roboknee: an exoskeleton for enhancing strength and endurance during walking"*. IEEE/ASME Proceedings International Conference on Robotics and Automation (ICRA), volume 3, pp. 2430–2435.

[178] Quigley, M., Asbeck, A., and Ng, A. (2011). *"A low-cost compliant 7-dof robotic manipulator"*. IEEE International Conference on Robotics and Automation (ICRA), pp. 6051–6058.

[179] Rao, A., Benson, D., Darby, C., Patterson, M., Francolin, C., Sanders, I., and Huntington, G. (2010). *"Algorithm 902: Gpops, a matlab software for solving multiple-phase optimal control problems using the gauss pseudospectral method"*. ACM Transactions on Mathematical Software, 37(2), pp. 1–39.

[180] Realmuto, J., Klute, G., and Devasia, S. (2015). *"Nonlinear passive cam-based springs for powered ankle prostheses"*. Journal of Medical Devices, 9(1), p. 011007.

[181] Reed, M. and Book, W. (2004). *"Modeling and control of an improved dissipative passive haptic display"*. IEEE International Conference on Robotics and

Automation (ICRA), volume 1, pp. 311–318 Vol.1.

[182] Reuleaux, F. (1893). *"The constructor"*. Philadelphia, H.H. Suplee.

[183] Revzen, S. and Guckenheimer, J. (2012). *"Finding the dimension of slow dynamics in a rhythmic system"*. Journal of The Royal Society Interface, 9(70), pp. 957–971.

[184] Rome, L.C., Flynn, L., and Yoo, T.D. (2006). *"Biomechanics: Rubber bands reduce the cost of carrying loads"*. Nature, 444(7122), pp. 1023–1024.

[185] Rouse, E.J., Mooney, L.M., and Herr, H.M. (2014). *"Clutchable series-elastic actuator: Implications for prosthetic knee design"*. The International Journal of Robotics Research.

[186] Rouse, E.J., Mooney, L.M., Martinez-Villalpando, E.C., and Herr, H.M. (2013). *"Clutchable series-elastic actuator: Design of a robotic knee prosthesis for minimum energy consumption"*. Proceedings of IEEE International Conference on Rehabilitation Robotics.

[187] Sarakoglou, I., Tsagarakis, N., and Caldwell, D. (2014). *"Development of a hybrid actuator with controllable mechanical damping"*. Robotics and Automation (ICRA), 2014 IEEE International Conference on, pp. 1078–1083.

[188] Schaal, S. and Atkeson, C. (1993). *"Open loop stable control strategies for robot juggling"*. Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on, pp. 913 –918 vol.3.

[189] Schaal, S. and Atkeson, C. (1993). *"Open loop stable control strategies for robot juggling"*. Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on, pp. 913 –918 vol.3.

[190] Scherer, C. and Weiland, S. (2000). *"Linear matrix inequalities in control"*. Lecture Notes, Dutch Institute for Systems and Control, Delft, The Netherlands.

[191] Sclater, N. (2011). *"Mechanisms and mechanical devices sourcebook"*. McGraw-Hill Professional.

[192] Seok, S., Wang, A., Chuah, M.Y., Otten, D., Lang, J., and Kim, S. (2013). *"Design principles for highly efficient quadrupeds and implementation on the mit cheetah robot"*. Robotics and Automation (ICRA), 2013 IEEE International Conference on, pp. 3307–3312.

[193] Sergej, M. (1990). *"Resonance robot"*.

[194] Serov Evegnij, P. (1990). *"Resonance manipulator"*.

[195] Seyfarth, A., Geyer, H., and Herr, H. (2003). *"Swing-leg retraction: a simple control model for stable running"*. Journal of Experimental Biology, 206(15), pp. 2547–2555.

[196] Seyfarth, A., Geyer, H., and Herr, H. (2003). *"Swing-leg retraction: a simple control model for stable running"*. Journal of Experimental Biology, 206(15), pp. 2547–2555.

[197] Shamaei, K., Cenciarini, M., Adams, A., Gregorczyk, K., Schiffman, J., and Dollar, A. (2015). *"Biomechanical effects of stiffness in parallel with the knee joint during walking"*. Biomedical Engineering, IEEE Transactions on, PP(99), pp. 1–1.

[198] Shamaei, K., Napolitano, P., and Dollar, A. (2014). *"Design and functional evaluation of a quasi-passive compliant stance control knee-ankle-foot orthosis"*. Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 22(2), pp. 258–268.

[199] Shin, D., Yeh, X., and Khatib, O. (2014). *"A new hybrid actuation scheme with artificial pneumatic muscles and a magnetic particle brake for safe human-robot collaboration"*. The International Journal of Robotics Research.

[200] Shirata, S., Konno, A., and Uchiyama, M. (2007). *"Design and evaluation of a gravity compensation mechanism for a humanoid robot"*. Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on, pp. 3635–3640.

[201] Shiriaev, A., Freidovich, L., and Gusev, S. (2010). *"Transverse linearization for controlled mechanical systems with several passive degrees of freedom"*. Automatic Control, IEEE Transactions on, 55(4), pp. 893–906.

[202] Siedel, T., Lukac, D., Geppert, T., Benckendorff, C., and Hild, M. (2011). *"Operating characteristics of a passive, bidirectional overrunning clutch for rotary joints of robots"*. XXIII International Symposium on Information Communication and Automation Technologies (ICAT), pp. 1–7.

[203] Singer, N.C. and Seering, W.P. (1990). *"Preshaping command inputs to reduce system vibration"*. Journal of Dynamic Systems, Measurement, and Control, 112, pp. 76–82.

[204] Singhose, W., Seering, W., and Singer, N. (1994). *"Residual vibration reduction using vector diagrams to generate shaped inputs"*. Journal of Mechanical

Design, 116, pp. 654–659.

[205] Sledd, A. and O'Malley, M. (2006). *"Performance enhancement of a haptic arm exoskeleton"*. Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, pp. 375–381.

[206] Smeets, J., Frens, M., and Brenner, E. (2002). *"Throwing darts: timing is not the limiting factor"*. Experimental Brain Research, 144, pp. 268–274.

[207] Stanway, R. (2004). *"Smart fluids: current and future developments"*. Materials science and technology, 20(8), pp. 931–939.

[208] Stellin, G., Cappiello, G., Roccella, S., Carrozza, M., Dario, P., Metta, G., Sandini, G., and Becchi, F. (2006). *"Preliminary design of an anthropomorphic dexterous hand for a 2-years-old humanoid: towards cognition"*. IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), pp. 290–295.

[209] Stirling, T., Wischmann, S., and Floreano, D. (2010). *"Energy-efficient indoor search by swarms of simulated flying robots without global information"*. Swarm Intelligence, 4(2), pp. 117–143.

[210] Stramigioli, S., Oort, G. van, and Dertien, E. (2008). *"A concept for a new energy efficient actuator"*. Advanced Intelligent Mechatronics, 2008. AIM 2008. IEEE/ASME International Conference on, pp. 671–675.

[211] Strogatz, S. (2001). *"Nonlinear dynamics and chaos: with applications to physics, biology, chemistry and engineering"*.

[212] Sugahara, Y., Endo, T., Lim, H. ok, and Takanishi, A. (2002). *"Design of a battery-powered multi-purpose bipedal locomotor with parallel mechanism"*. IEEE/RSJ International Conference on Intelligent Robots and Systems, volume 3, pp. 2658–2663.

[213] Sugar, T.G., Hollander, K.W., and Hitt, J.K. (2011). *"Walking with springs"*.

[214] Sup, F., Bohara, A., and Goldfarb, M. (2008). *"Design and control of a powered transfemoral prosthesis"*. The International Journal of Robotics Research, 27(2), pp. 263–273.

[215] Thorpe, S., Fize, D., and Marlot, C. (1996). *"Speed of processing in the human visual system"*. Nature, 381(6582), pp. 520–522.

[216] Thorpe, S., Fize, D., and Marlot, C. (1996). *"Speed of processing in the human visual system"*. Nature, 381(6582), pp. 520–522.

[217] Tobenkin, M., Manchester, I., and Tedrake, R. (2011). *"Invariant funnels around trajectories using sum-of-squares programming"*. Proceedings of the 18th IFAC World Congress, extended version available online: arXiv:1010.3013.

[218] Todorov, E. and Jordan, M.I. (2002). *"Optimal feedback control as a theory of motor coordination"*. Nature neuroscience, 5(11), pp. 1226–1235.

[219] Tolou, N., Henneken, V.A., and Herder, J.L. (2010). *"Statically balanced compliant micro mechanisms (sb-mems): Concepts and simulation"*. ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 447–454.

[220] Tolou, N., Smit, G., Nikooyan, A.A., Plettenburg, D.H., and Herder, J.L. (2012). *"Stiffness compensation mechanism for body powered hand prostheses with cosmetic covering"*. Journal of medical devices, 6(1).

[221] Tsagarakis, N., Dallali, H., Negrello, F., Roozing, W., Medrano-Cerda, G., and Caldwell, D. (2014). *"Compliant antagonistic joint tuning for gravitational load cancellation and improved efficient mobility"*. Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on, pp. 924–929.

[222] Tsagarakis, N., Morfey, S., Dallali, H., Medrano-Cerda, G., and Caldwell, D. (2013). *"An asymmetric compliant antagonistic joint design for high performance mobility"*. Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on, pp. 5512–5517.

[223] Tsagarakis, N., Sardellitti, I., and Caldwell, D. (2011). *"A new variable stiffness actuator (compact-vsa): Design and modelling"*. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 378–383.

[224] Tsay, D. and Lin, B. (1996). *"Profile determination of planar and spatial cams with cylindrical roller-followers"*. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 210(6), pp. 565–574.

[225] Tuijthof, G.J. and Herder, J.L. (2000). *"Design, actuation and control of an anthropomorphic robot arm"*. Mechanism and machine theory, 35(7), pp. 945–962.

[226] Unal, R., Behrens, S., Carloni, R., Hekman, E., Stramigioli, S., and Koopman, H. (2010). *"Prototype design and realization of an innovative energy efficient transfemoral prosthesis"*. 3rd IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), pp. 191–196.

[227] Uno, Y., Kawato, M., and Suzuki, R. (1989). *"Formation and control of optimal trajectory in human multijoint arm movement"*. Biological cybernetics, 61(2), pp. 89–101.

[228] Vanderborght, B., Albu-Schaeffer, A., Bicchi, A., Burdet, E., Caldwell, D., Carloni, R., Catalano, M., Eiberger, O., Friedl, W., Ganesh, G., et al. (2013). *"Variable impedance actuators: a review"*. Robotics and Autonomous Systems, 61(12), pp. 1601–1614.

[229] Vanderborght, B., Van Ham, R., Lefeber, D., Sugar, T.G., and Hollander, K.W. (2009). *"Comparison of mechanical design and energy consumption of adaptable, passive-compliant actuators"*. The International Journal of Robotics Research, 28(1), pp. 90–103.

[230] Vanderniepen, I., Van Ham, R., Van Damme, M., and Lefeber, D. (2008). *"Design of a powered elbow orthosis for orthopaedic rehabilitation using compliant actuation"*. IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), pp. 801–806.

[231] Vermeulen, M. and Wisse, M. (2010). *"Intrinsically safe robot arm: Adjustable static balancing and low power actuation"*. International Journal of Social Robotics, 2(3), pp. 275–288.

[232] Vladimir, G. (1990). *"Resonance drive"*.

[233] Wang, J. and Meng, G. (2001). *"Magnetorheological fluid devices: principles, characteristics and applications in mechanical engineering"*. Institution of Mechanical Engineers, Part L: Journal of Materials Design and Applications, 215(3), pp. 165–174.

[234] Wang, X. and Zhu, W. (2014). *"Design, modeling, and simulation of a geared infinitely variable transmission"*. Journal of Mechanical Design, 136(7), p. 071011.

[235] Weinberg, B., Nikitczuk, J., Patel, S., Patritti, B., Mavroidis, C., Bonato, P., and Canavan, P. (2007). *"Design, control and human testing of an active knee rehabilitation orthotic device"*. IEEE International Conference on Robotics and Automation (ICRA), pp. 4126–4133.

[236] Werkmeister, J. and Slocum, A. (2007). *"Theoretical and experimental determination of capstan drive stiffness"*. precision Engineering, 31(1), pp. 55–67.

[237] Westervelt, E., Grizzle, J., and Koditschek, D.E. (2003). *"Hybrid zero dynamics of planar biped walkers"*. Automatic Control, IEEE Transactions on, 48(1),

pp. 42–56.

[238] Wiggin, M.B., Sawicki, G.S., and Collins, S.H. (2011). *"An exoskeleton using controlled energy storage and release to aid ankle propulsion"*. IEEE International Conference on Rehabilitation Robotics (ICORR), pp. 1–5.

[239] Williamson, M. (1998). *"Rhythmic robot arm control using oscillators"*. Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on, volume 1, pp. 77–83 vol.1.

[240] Williamson, M. (1999). *"Designing rhythmic motions using neural oscillators"*. Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on, volume 1, pp. 494–500 vol.1.

[241] Williamson, M.M. (1998). *"Neural control of rhythmic arm movements"*. Neural Networks, 11(7âĂŞ8), pp. 1379 – 1394.

[242] Williamson, M.M. (1999). *"Robot arm control exploiting natural dynamics"*. Ph.D. thesis, Massachusetts Institute of Technology.

[243] Wisse, M., Feliksdal, G., Van Frankkenhuyzen, J., and Moyer, B. (2007). *"Passive-based walking robot"*. Robotics Automation Magazine, IEEE, 14(2), pp. 52–62.

[244] Wolfslag, W., Plooij, M., Babuska, R., and Wisse, M. (2015). *"Learning robustly stable open-loop motions for robotic manipulation"*. Robotics and Autonomous Systems, Available Online.

[245] Wright, C., Buchan, A., Brown, B., Geist, J., Schwerin, M., Rollinson, D., Tesch, M., and Choset, H. (2012). *"Design and architecture of the unified modular snake robot"*. IEEE International Conference on Robotics and Automation (ICRA), pp. 4347–4354.

[246] Yamatoh, K., Ogura, M., Kanbe, K., and Isogai, Y. (1989). *"Piezoelectric brake device"*. US Patent 4,854,424.

[247] Yoon, S.S., Kang, S., Yun, S.k., Kim, S.J., Kim, Y.H., and Kim, M. (2005). *"Safe arm design with mr-based passive compliant joints and viscoŮelastic covering for service robot applications"*. Journal of mechanical science and technology, 19(10), pp. 1835–1845.

[248] Zlatanov, D., Fenton, R., and Benhabib, B. (1998). *"Identification and classification of the singular configurations of mechanisms"*. Mechanism and Machine Theory, 33(6), pp. 743 – 760.

Acknowledgements

The research in this thesis would not have been possible without the help of others.

I would like to start by thanking my advisor Martijn Wisse for giving me the opportunity to do this research. Thank you for offering me a PhD position and to hold the position for five months when I denied your offer. I appreciate the faith you have in me, your encouragements to get the best out of myself and the freedom you gave to choose my own research directions. Although more distantly involved, I would also like to thank Frans van der Helm for guiding me through the last year.

The last four years, I have had the privilege to have a fellow PhD researcher in Wouter Wolfslag. Wouter, I am sure it is safe to say that those years would not have been the same without you. I think that our cooperation was so successful because we are different enough to complement each other and similar enough to share many interests. Thank you for always wanting to dig deeper where I want to move on and to challenge my mathematical skills. And thank you for sharing a feeling for bad humor, a knowledge of bad music and a taste for beer

As visible throughout this thesis, this research would not have been possible without many co-authors. In particular, I would like to thank Wouter Caarls, Heike Vallery, Gabriel Lopes, Glenn Mathijssen, Bram Vanderborght, Gerard Dunning and Robert Babuska. Without your help, discussions, comments and insights, I would not have been able reach the results in this thesis. I also had the pleasure of working with five master students: Michiel de Vries, Timo Walvoort, Jan Warnaars, Tom van der Hoeven and Marijn van de Wijdeven. Thank you for letting me be a part of your final year at Delft University of technology. I would also like to thank the other students that were involved in the RAP-meetings.

Part of what has made the last four years enjoyable is the great atmosphere in the Delft Biorobotics Lab, caused by the great peoples involved. I love how we had spontaneous lab outings like paintballing, climbing and go-karting and of course the Friday afternoon drinks where we often played games such as chess, go, FIFA and kubb. For all the in-depth discussion in the lab, I especially thank Daniël Karssen, Wietse van Dijk, Shiqian Wang, Tim Vercruyssen, Cor Meijneke, Berk Çalli, Mukunda Bharatheesha, Jeff van Egmond and Daniel Lemus.

In order to really let the robots and prototypes work, I had help from several peoples. Jan van Frankenhuyzen, thank you for always giving me valuable advise on the mechanical design. Guus Liqui Lung, thank you for helping me with the part of robots I knew little about when I started: the electronics. I thank the peoples from the

machine shop and in particular Andries Oort and Nisse Linskens, for always providing qualitatively good parts within a short time. I also had the pleasure to work with two students from the Leidse instrumentmakers school: Marvin van Nunspeet and Rik Zomer. Thank you for translating my ideas into working prototypes.

Finally, I would like to thank my family and friends for their support. And above all other peoples, I thank my lovely wife Ineke. Without you, I would probably never have considered doing a PhD and never have started brewing beer. Thank you for you unconditional support. My life is better with you in it.

About the author

January 21, 1987
Born in Zoetermeer, The Netherlands.

1999-2005
VWO at Orange Nassau College in Zoetermeer. Profile: Natuur en Techniek

2005-2008
Bachelor of Science at Delft University of Technology. Final project: *The geometry of the speed skating blade*

2009-2011
Master of Science at Delft Unviersity of Technology. Final project: *Using a resonant mechanism to reduce energy consumption in robotic arms*. Part of the Master education was an internship in the Experimental BioMechatronics Lab of Steve Collins at Carnegie Mellon University in the USA. The topic of the internship was *Push-off strategies for foot prostheses*.

2011-2015
PhD researcher at Delft University of Technology, of which this thesis presents the final results. The topic of the research was *Exploiting Dynamics, in robotic arms with repetitive tasks*.

List of publications

Journal papers

Statically Balanced Brakes
Michiel Plooij, Tom van der Hoeven, Gerard Dunning and Martijn Wisse
To appear in: *Precision Engineering*

Robust feedforward control of robotic arms with friction model uncertainty
Michiel Plooij, Wouter Wolfslag and Martijn Wisse
Robotics and Autonomous Systems vol 70, August 2015

Lock Your Robot: A Review of Locking Devices in Robotics
Michiel Plooij, Glenn Matthijssen, Pierre Cherelle, Dirk Lefeber and Bram Vanderborght
IEEE Robotics and Automation Magazine vol.22, no.1, March 2015

Learning robustly stable open-loop motions for robotic manipulation
Wouter Wolfslag*, Michiel Plooij*, Robert Babuska and Martijn Wisse
Robotics and Autonomous Systems vol 66, April 2015

Dissipatively actuated manipulation
Wouter Wolfslag, Michiel Plooij, Wouter Caarls, Sander van Weperen and Gabriel Lopes
Control Engineering Practice Vol 34, Jan 2015

Conference papers

The effect of the choice of feedforward controllers on the accuracy of low gain controlled robots
Michiel Plooij*, Wouter Wolfslag* and Martijn Wisse
International Conference on Intelligent Robots and Systems (IROS) 2015

Design and evaluation of the Bi-directional Clutched Parallel Elastic Actuator (BIC-PEA)
Michiel Plooij, Marvin van Nunspeet, Martijn Wisse and Heike Vallery
International Conference on Robotics and Automation (ICRA) 2015

Open Loop Stable Control in Repetitive Manipulation Tasks
Michiel Plooij*, Wouter Wolfslag* and Martijn Wisse
International Conference on Robotics and Automation (ICRA) 2014

* Shared first authorship

Optimization of feedforward controllers to minimize sensitivity to model inaccuracies
Michiel Plooij, Michiel de Vries, Wouter Wolfslag and Martijn Wisse
International Conference on Intelligent Robots and Systems (IROS) 2013

Learning while preventing mechanical failure due to random motions
Hendrik Meijdam, Michiel Plooij and Wouter Caarls
International Conference on Intelligent Robots and Systems (IROS) 2013

A Novel Spring Mechanism to Reduce Energy Consumption of Robotic Arms
Michiel Plooij and Martijn Wisse
International Conference on Intelligent Robots and Systems (IROS) 2012

Conference abstracts

Incorporating tasks in the dynamics of robotic arms
Wouter Wolfslag, Michiel Plooij, Irene Staal and Martijn Wisse
ICRA 2014 workshop on Task-based Optimal Design of Robots

Towards open loop stable manipulators
Wouter Wolfslag*, Michiel Plooij* and Martijn Wisse
Dynamic Walking Annual Meeting 2013

A spring mechanism for resonant robotic arms
Michiel Plooij and Martijn Wisse
Workshop on Human Friendly Robotics 2011

Submitted papers

Clutched Elastic Actuators
Michiel Plooij, Wouter Wolfslag and Martijn Wisse
Transactions on Mechatronics

Reducing the energy consumption using the Bi-directional Clutched Parallel Elastic Actuator
Michiel Plooij, Martijn Wisse, Heike Vallery
Transactions on Robotics

Sensor free manipulation: Extending robustly stable feedforward control to hybrid systems

* Shared first authorship

Wouter Wolfslag, Michiel Plooij and Martijn Wisse
Conference on Decision and Control (CDC) 2015

Propositions

1. Locking mechanisms outperform continuously variable transmissions as a tool to control the energy into springs (this thesis)

2. Springs in parallel to the motor outperform springs in series with the motor in lowering the energy consumption, peak torque and peak power (this thesis)

3. Currently used friction models in robotics are inadequate for both energy optimal control and feedforward control

4. Robotics research should be less focussed on making cool robots and more on generic methods

5. The replacement of caregivers by robots should be compensated by extra social activities for patients

6. All engineering is optimization; the challenge is to find the right cost function, parameterization and solver

7. Working hard towards a vague goal is better than working hard to make the goal clearer

8. A strong belief in the reachability of all goals through hard work and determination does not do justice to peoples who are less 'successful'

9. Making the energy supply sustainable is more important than reducing the energy consumption

10. The world is not fundamentally causal

These propositions are regarded as opposable and defendable, and have been approved as such by the supervisor prof. dr. F.C.T. van der Helm.

Stellingen

1. Locking mechanismes presteren beter dan continu variabele transmissies om de energie in veren te regelen (dit proefschrift)

2. Veren parallel aan de motor zijn beter dan veren in series met de motor in het reduceren van de energieconsumptie, het piekkoppel en het piekvermogen (dit proefschrift)

3. Wrijvingsmodellen die gebruikt worden in de robotica zijn ontoereikend voor zowel energieoptimale- als sensorloze aansturing

4. Robotonderzoek zou minder moeten focussen op het maken van coole robots en meer op het ontwikkelen van generieke methodes

5. De vervanging van zorgverleners door robots moet gecompenseerd worden door extra sociale activiteiten voor patienten

6. Ingenieurswerk is optimalisatie; de uitdaging is het vinden van de juist kost-functie, parametrisatie en oplossingsalgorithme

7. Hard werkend vaag doel nastreven is beter dan hard werken om het doel te verduidelijken

8. Een sterk geloof in de maakbaarheid van de wereld doet geen recht aan mensen die minder 'succesvol' zijn

9. De energievoorziening verduurzamen is belangrijker dan het verminderen van de energieconsumptie

10. De wereld is niet fundamenteel causaal

Deze stellingen worden opponeerbaar en verdedigbaar geacht en zijn als zodanig goedgekeurd door de promotor prof. dr. F.C.T. van der Helm.

ISBN 978-94-6186-576-2