ILC for functional electrical stimulation of human arms

ILC for Functional Electrical Stimulation of Human Arms

C.H.A. Criens

DCT 2007.008

Traineeship report

Coach(es): Prof. E. Rogers
P.L. Lewin
C. T. Freeman

Supervisor: Prof. dr. ir. M. Steinbuch

Technische Universiteit Eindhoven
Department Mechanical Engineering
Dynamics and Control Technology Group

Eindhoven, January 25, 2007

Samenvatting

In dit verslag wordt de ontwikkeling van een ILC regelaar voor het regelen van een mensenarm uitgelegd. Het doel van dit project is om hemiplegiepatienten te helpen rehabiliteren.

Hemiplegie is een aandoening waarbij patienten (gedeeltelijk) verlamd zijn aan een kant van hun lichaam. Een nieuwe techniek voor rehabilitatie van dergelijke patienten wordt ontwikkeld. Deze techniek maakt gebruik van *functional electrical stimulation* (FES) om de verlamde arm te bewegen. *Functional electrical stimulation* betekent dat spieren elektrisch worden gestimuleerd zodat de resulterende beweging functioneel is. Deze techniek zal worden gebruikt om hemiplegiepatienten te helpen met het doen van bewegingen die ze zelf niet kunnen, zodat ze de oefening krijgen die ze nodig hebben om te rehabiliteren.

Voor de ontwikkeling van dit regelsysteem, is een model van een mensenarm gemaakt. Metingen in het tijddomein laten zien dat een correct model nietlineair is. In het frequentiedomein echter blijkt dat het gedrag van spieren ook gemodelleerd kan worden met een tweede orde kritisch gedempt linear model. Dit model gebruikt de recruitment van de spier als ingang. Dit is een nietlineaire algebraïsche functie van de werkelijke input, de pulsbreedte. Als dit spiermodel wordt gecombineerd met de massatraagheid van de arm ontstaat een vierde-orde model.

De bandbreedte die gehaald kan worden door een feedback regelaar wordt begrens door de snelheid van de spierdynamica. Omdat spieren vrij langzaam reageren, zal ook het geregelde systeem vrij traag zijn. Hierdoor zal een systeem geregeld met alleen een feedback regelaar niet erg nauwkeurig zijn.

Voor deze toepassing zullen de bewegingen steeds herhaald worden. Een regelschema waarbij gebruik wordt gemaakt van *iterative learning control* (ILC) kan daardoor toegepast worden. ILC regelaars maken gebruik van kennis van de vorige keren om de output van de regelaar te verbeteren. Dit maakt het mogelijk om bewegingen met steeds toenemende nauwkeurigheid te regelen. Een P-type ILC regelaar gebruik makend van zero-phase filters zal worden gebruikt. Deze regelaar is zodanig afgesteld dat hij erg robuust is voor veranderingen in de gain van het systeem. Vooral deze veranderingen zijn van belang, omdat deze voorkomen als gevolg van vermoeiing. Een andere bron van afwijkingen zijn verstoringen. ILC regelaars kunnen alleen fouten tegengaan die ook in vorige keren voorkwamen. Daarom moet bij het afstellen van de regelaar er voor worden gezorgd dat hij afwijkingen veroorzaakt door verstoringen niet zal proberen tegen te gaan. Deze twee problemen worden opgelost door de ILC regelaar niet te snel te laten convergeren, de anticausale tijdstap die wordt gebruikt iets te verhogen en de afsnijfrequentie te verlagen.

Het uiteindelijke regelsysteem is getest met een experiment op een toepassing zoals die ook gebruikt gaat worden voor hemiplegiepatienten. De toepassing van de ILC regelaar zorgde voor een vermindering van de volgfout met ongeveer een factor 4 ten opzichte van alleen een feedback regelaar. Dit maakt de toepassing succesvol.

Abstract

In this report the design of an ILC controller suitable for control of a human arm is explained. The goal of this project is to help hemiplegic patients in their rehabilitation.

Hemiplegia is a condition where patients have a (partial) paralysis of one side of the body. A new technique for their rehabilitation will be developed. This technique makes use of functional electrical stimulation (FES) to move a paralysed arm. Functional electrical stimulation means that muscles are being provided with electrical stimulation so that at the very time of stimulation the muscle contraction has a functional purpose. This technique will be applied to assist hemiplegic patients in doing movements they cannot do on their own, giving these patients the practice they need to rehabilitate.

For the purpose of designing this control system, a model of a human arm is made. Measurements in the time domain show that the model of a human arm is in reality highly nonlinear. However, in literature frequency domain measurements show that the frequency domain behaviour of a muscle can be modelled by linear second order critically damped dynamics. These models use the muscle recruitment as input, this is a nonlinear function of the real input variable. Combining the muscle model with the inertia they move gives a fourth order model.

The achievable bandwidth of a feedback controller applied to this system is limited by the speed of the muscle dynamics. Since these dynamics are quite slow, the controlled system will be quite slow as well. A system controlled only using a feedback controller will therefore not be very accurate.

Since for this application repeated movements will be made. A control scheme making use of iterative learning control (ILC) can be applied. ILC makes use of knowledge from previous trials to improve the controller output. This makes it possible to control the movement with increasing accuracy. A P-type ILC controller making use of zero-phase filtering is tuned for this purpose. While tuning the ILC controller, special care has been taken to make the controller very robust for changes in the plant gain. These changes are of great importance, because these are the changes that are expected due to fatigue. Another source of errors are disturbances. ILC controllers cannot counteract errors that were not present in previous trials. Therefore, when tuning the controller, it must be taken into account that the ILC controller leaves out the error caused by disturbances as much as possible. These two problems are counteracted by lowering the convergence speed of the ILC controller, slightly overdoing the noncausal time shift and using a conservative cut-off frequency for the low-pass filter.

The designed control system was tested in practice by doing an experiment is similar to what hemiplegic patients will have to do. The ILC controller combined with the feedback controller gives an improvement in tracking accuracy of about a factor 4 compared with only a feedback controller. This makes the application very successful.

Contents

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 Aims and scope . . . 1
- 1.3 Contents of this report . . . 1

2 Functional Electrical Stimulation **3**

3 Modelling a Human Arm **5**
- 3.1 General description of the test rig . . . 5
- 3.2 Modelling muscles . . . 6

4 Control **11**
- 4.1 Objectives . . . 11
- 4.2 Structure of the controller . . . 11
- 4.3 Optimization . . . 12
  - 4.3.1 Feedback controller . . . 12
  - 4.3.2 ILC controller . . . 13
- 4.4 Robustness . . . 15
- 4.5 Disturbances . . . 17

5 Results **19**
- 5.1 Experiments . . . 19

6 Conclusions and Recommendations **21**

A Extended Model **25**
- A.1 Muscles . . . 25
- A.2 Lagrange . . . 26
  - A.2.1 Kinematics . . . 26
  - A.2.2 Forces . . . 27
  - A.2.3 Lagrange . . . 28
  - A.2.4 Full model . . . 28
- A.3 Identification . . . 28
- A.4 Controller improvement . . . 29
  - A.4.1 Discussion . . . 29

B Arm Geometry **31**
B.1 Elbow orientation 31
B.2 Elbow bend 32

C ILC Convergence **33**
C.1 Derivation 33
C.2 Limit values 34

Chapter 1

Introduction

1.1 Motivation

Improvements in stroke rehabilitation can increase life quality of thousands of people every year. In the United Kingdom, about 100 000 strokes occur every year [6] and in The Netherlands this number is about 41 000. Of the people going into rehabilitation treatment after a stroke, 85 % regain their ability to walk. Whereas at the moment only 14 % of the people going into rehabilitation regain useful control over their upper limbs [7]. A likely cause of this, is that patients are much less forced to keep trying to use their disabled arm compared with the disabled leg.

A new project using functional electrical stimulation (FES) to help hemiplegic patients move their disabled arm is started. This makes it possible for patients to exercise movements they can not yet do on their own. Scientific research provides evidence that this method will lead to an improvement in muscle function for hemiplegic patients. When successful, this method will enhance recovery methods and can improve the life of many people.

1.2 Aims and scope

In this report, the use of iterative learning control (ILC) for controlling a human arm doing repetitive movements using FES is investigated. This method of control makes use of information from previous trials to improve the input in the next trial. This makes it possible to control repeated movements with increasing accuracy.

The goal of ILC is not to get a working system, it is implemented to increase tracking performance. Consequently, for ILC to be successful not only should the controlled system be stable, the error should also decrease significantly.

1.3 Contents of this report

First the basics of FES will be explained. This is followed by a chapter on modelling a human arm. Here, the behaviour of muscles under electrical stimulation will be explained. After that, the control algorithm and its tuning will be explained. It will be explained how changes in the parameters of the controller effect the performance and robustness of the system. The last chapter of the main text will deal with simulations and experimental results.

In the appendix of this report a more advanced model is explained. This will provide some insight in the differences between the simulations and the experiments. When properly identified, this model also allows for the design of a better controller, possibly improving the performance.

Chapter 2

Functional Electrical Stimulation

In this project use will be made of so called functional electrical stimulation (FES) to actuate a human arm. As this is not a common subject in mechanical engineering, this chapter will deal with some basics on the purpose and application of this technique.

When a person has a stroke, a blood clot blocks a blood vessel inside the brain. As a result of this, a part of the brain will no longer receive fresh blood and the connecting nerve fibres in this part will die. This can result in a (partial) paralysis of one side of the body, in medical terms called hemiplegia. It is impossible for the damaged nerve fibers to re-grow, however it is possible to make new connections inside the brain. These new connections are made all the time. In this way humans learn new skills. To learn these new skills, practice and feedback are essential. For hemiplegic patients it is very hard to get practice, because they have hardly any mobility in their limbs. This is where FES will find its use. The patients will be asked to perform a movement with their arm and will be assisted in this using FES. By themselves they would not be able to do the movement and consequently they would not be able to train their brain. With FES they will be able to do the movement and new nerve fibre connections should be the result of this. The purpose of the project is that hemiplegic patients relearn to use their arm in this way.

Functional electrical stimulation means that muscles are being provided with electrical stimulation so that at the very time of stimulation the muscle contraction has a functional purpose. For this project it means that hemiplegic patients will get electrodes on their upper arm. Using these electrodes a controlled voltage will be supplied to the disabled muscles, in this case the triceps. This voltage serves instead of the voltage supplied by the brain and nerves, making it possible to control the movement of the arm using a computer system to provide the input voltage. Patients that were not able to do certain movements by themselves, because of defective nerve control, can be assisted in this using FES.

The voltage that will be put on the arm will not be a smooth signal. It will consist of a series of pulses with fixed time intervals and having a fixed amplitude. The width of these pulses can be varied. It is up to the controller to control the width of the pulses in such a way that the movement of the arm is as desired. The remainder of this report will explain how such a controller is designed. Before it is possible to design such a controller, a model of a human arm is needed. This model must in some sense give the same behaviour as a real human arm. In the next chapter this model and the process of making this model will be explained.

Chapter 3

Modelling a Human Arm

The first step in making a controller is always making a model of the system. In this case the system to be controlled is a human arm actuated by electric pulses.
The real human arm is expected to give highly nonlinear behaviour, mainly due to the nonlinear nature of the muscle forces. As a result, the easiest method for model identification, frequency response measurement using an input signal with a broad band of frequencies, can not be used to identify the system. Other problems include unpredictable reflex forces and the fact that muscles generate forces in one direction only. A noise input would cause the arm to stretch and after that, no more movement is possible.
In this chapter a rather simple model, useful for control design, is going to be made. In appendix A a more detailed model is explained. This more complex model finds its use in explaining the differences between the simulations done with the model from this chapter and the experiments. Also, when properly identified, this model can be used to improve the controller and performance can be increased.

3.1 General description of the test rig

The main components of the test rig are a robotic arm, a computer, a stimulation device, a projector and a chair. The patient is asked to sit on the chair, keeping his body in a fixed position. His arm is strapped to the robotic arm. The patient is then asked to follow a certain trajectory, typically a stretching movement. The desired movement is projected on a moving plane above the hand, and crosshairs fixed at the hand should follow a red dot moving on the trajectory.

The robotic arm has multiple uses. It makes sure that the patient can move his forearm only in the horizontal plane. Also, the motors on the robotic arm are controlled such that the patient cannot deviate too much from the projected trajectory. It acts as a kinematic constraint, keeping the arm from moving off the trajectory, but it does not help the movement in the direction of the trajectory. A third function of the robotic arm is to provide a resistance force such that the training can be made more difficult when it is becoming too easy. The robot is also used to measure the position of the hand. Goniometers attached to the joints of the robot measure the angle and from this the position of the patients hand can be calculated. Force and torque sensors are attached to grip for the hand. These sensors can measure the force and torque in all six degrees of freedom. The robot is constructed to be as light as possible using carbon fiber as it's main material and aluminium for the joints. Furthermore, electrical motors and a control system are used to actively

neutralise the forces needed to move the robot. This makes the inertia felt when moving the robot feel extremely light.

For moving the patient's arm, a stimulation device is connected to electrodes on the patient's arm. This stimulation device is connected to a computer. This computer also knows the position of the moving red dot and the position of the hand and will be used to calculate the error signal and control the input to the stimulation device.

3.2 Modelling muscles

The main difference between designing a controller for a human arm and a robot is the difference in the actuators. Usually for robot control the actuators are electric motors. A human arm however is actuated by muscles. This difference requires some explanation on the behaviour of muscles.
Usually a human arm is controlled by electrical signals sent by the brain through the nerve system to the muscles. For this project the muscles are stimulated by electrodes connected to a computer system. When patients are not completely paralized, it will be assumed that the human stimulation is intended to help in performing its task better, so no real problems are expected because of this. A muscle model involving only the effects of external stimulation must under this hypothesis be sufficient. All experiments in this report are done on normals, keeping their arm neutral. Since hemiplegic patients don't have much control over their muscles this will be most realistic.

Muscle models can be divided into two classes. Models that use the physical contractile mechanism of muscles as a basis and models that are based on input-output measurements. For control design the second type of model will be sufficient. These models are simpeler and they can be quite accurate. Models of the input-output type are often based on the work of Archibald Vivian Hill [4] and called Hill models.
In this chapter a simplified version of an input-output model will be used. In appendix A a more complicated and more general version will be explained.

The model derived in this chapter is simplified to the essential characteristics needed to design a controller. A Hill model consisting of a series connection of nonlinear algebraic functions and linear dynamics will be used as a basis. The nonlinear function describing the dependency of the output force with varying muscle length and speed will be neglected.
Neglecting all changes caused by varying the arm position makes it possible to identify the complete muscle model isometrically, i.e. without movement. This allows for a fast and relatively easy identification process. All dynamics and the nonlinear variation of output force with varying pulse width will be included in the model.

The muscle model will consist of a nonlinear function describing the steady state muscle force and linear second order critically damped activation dynamics. This will give a rather crude estimation of the real muscle force and substantial errors can be expected. The model will not be used to make accurate predictions of the output force, but rather to design a control system. For this the dynamics are of most importance. The muscle model can mathematically be expressed

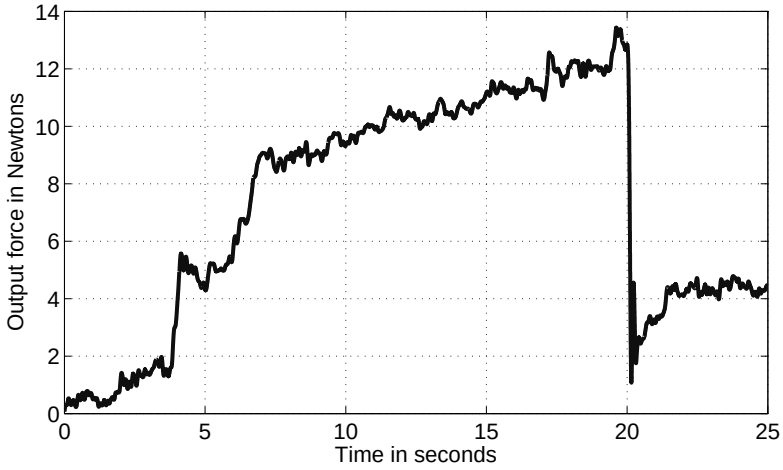

Figure 3.1: Measured isometric output force of a stimulated human arm using a pulse width of 0.250 ms between 4 and 20 seconds and zero stimulation at other times. The force is measured at the hand and the stimulation is applied at the triceps.

as:

$$F(t) = c \cdot t \cdot e^{-\lambda t} \times r(u(t)) \tag{3.1}$$

The $\times$ symbol is used to denote a convolution.
$r(u)$ is the recruitment; c is a constant.

An example of an isometric muscle force measurement is given in figure 3.1. This figure shows a typical step response of a muscle.

The step behaviour consists of a relatively fast increase in force, followed by a drop in force and then a slow increase in force. The drop in force is caused by a reflex contraction of the biceps. If the stimulation would be on for a longer period, a decrease of the force, starting after about 20 to 30 seconds, can be expected. After a minute of stimulation the output force can be expected to be about halved.

It is impossible to accurately describe this behaviour with a second order critically damped linear model. In literature [1], the frequency response is measured by stimulating isolated muscles in anesthetized cats with pulses of which the widths vary in time according to a sinusoid, while measuring the output force. Doing this while varying the frequencies makes it possible to directly measure the frequency response. From these experiments on various muscles it is clear that a linear second order critically damped system, i.e. two poles at the same location, can quite accurately describe the variations in amplitude and phase with varying frequency.
Doing this experiment ourselves is impractical. Problems with fatigue and measurement accuracy would cause such a measurement to take a very long time and probably still be much less accurate. For the measurements on cats it was possible to anesthetize them cut some nerves to be able to stimulate a specific muscles. Therefore, rather than measuring this, it is assumed that the frequency response of a human triceps can also be modelled in this way. The location of the

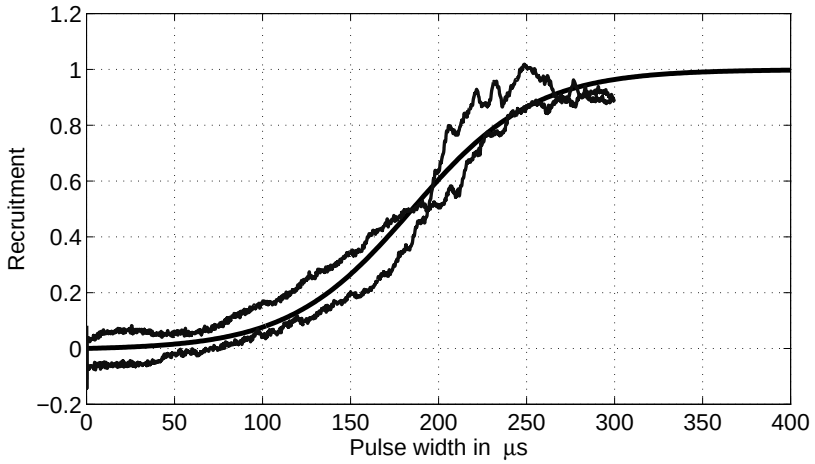

Figure 3.2: Recruitment curve measured using a two way ramp response and a time shift. The smooth line is a fit to the measurement.

pole will be estimated in a different way.

A function describing the steady state isometric muscle force, normalised at one for the maximum force, for varying pulse widths is called a recruitment curve (figure 3.2). Knowing this function is essential when making a good controller. The standard method for measuring this function is to apply a series of step responses varying in intensity, wait for a fixed time to let the dynamics settle, and then average the output force over some time [2]. This is a very simple method, but because this function should be measured before each session a faster method is proposed.
A triangular input signal ranging from zero pulse width to 0.3 ms will be applied to the triceps. While doing this, the arm is held in a fixed position and the force is measured at the hand. To account for the dynamics, rather than doing a deconvolution with the second order system, a simple time shift is used such that the measured force when increasing the pulse width equals the force when decreasing the pulse width. A real deconvolution is mathematically more accurate, but is not practical. Because the muscle dynamics resemble a low pass filter, the inverted dynamics will amplify a lot of noise. Filtering out all this distortion gave a less accurate representation of the recruitment curve than when only a simple time shift is used. This method makes it possible to accurately measure the recruitment curve using only one 10 second measurement. The time shift used also characterizes the pole location of the dynamics.
This method makes a great improvement over the conventional method. It is much faster, and probably more accurate as well. Inaccuracies caused by patients shifting position and fatigue give much less difficulties when this method is applied.

For making figure 3.2 a time shift of 0.85 seconds was used. This implies that the pole location of the frequency response can be expected to be around 1.17 Hz. This is slightly slower than the values found for cat muscles [1]. To be on the safe side when designing the controller, for this the pole location is put at 1 Hz.

The force provided by the muscle is used to move a system of masses. Accurately modelling

this is very difficult. If done properly, the robot, its control system, the upper arm and the brain's control system for the upper arm should be included. To simplify this, it is assumed that the inertias of the robotic arm are accurately neutralised and are therefore practically nonexistent in the input output behaviour.
The upper arm is still a problem. Modelling the brain's control system would be impractical. Therefore it is assumed that this system is perfect, i.e. all effects of movement of the fore arm are counteracted in such a way that it has no net effect on the movement of the upper arm. This makes the location of the elbow a simple time varying signal. The intended rotational speeds are relatively small. This makes it possible to neglect centrifugal forces and model the complete inertia as a simple $\frac{1}{ms^2}$. The movement of the upper arm will now act as a time varying disturbance.

Combining the transfer function for the muscle dynamics with the mass transfer function gives a fourth order system.

$$H(s) = \frac{c}{ms^2\left(\frac{1}{\lambda}s+1\right)^2} \tag{3.2}$$

The input for this system is the muscle recruitment which is a nonlinear function of the pulse width.

Fatigue is not implemented in this model. The reason for this is that the fatigue-dynamics act on a larger time scale than a single experiment. Hence, the amount of fatigue is almost constant during an experiment. When experiments last longer than 20 seconds fatigue will play a role. This is observed as a decreased muscle force when the same stimulation is applied. Fatigue can thus be modelled by adapting the gain of the transfer function. Fatigue is hard to predict because it is not known what happens before and in between experiments. Therefore it should be made sure that the controller is robust for changes in the gain of the plant.

Chapter 4

Control

4.1 Objectives

From a practical point of view, the control objectives are relatively easy to satisfy. There is no real need for very accurate tracking to get the desired training of the hemiplegic patients. Any movement that resembles the intended movement will do for this purpose. However, if the movement is more accurate, more advanced applications can be used.

In this report the use of an ILC controller is investigated. This seems to be an obvious choice since the reference trajectories are being repeated numerous times for the training. In practice this can make the designing process much more difficult. Some difficulties can be especially apparent when making an ILC controller. For example the nonlinearity of the human arm makes it much more difficult to use a lot of ILC algorithms. Because of fatigue, the arm will not reset to the original state (the amount of fatigue is also a state). The model of the arm is time varying in an unknown way, i.e. repeating the same measurement can give substantial variations in output. The errors can also be expected to have a substantial nonrepeatable part. Variations in the muscle model due to fatigue and human interaction limit the accuracy achievable by an ILC controller. The main objective is to design a control system making use of ILC that works when practically applied.

4.2 Structure of the controller

The controller will consist of two parts, a feedback controller and a learning feedforward controller. This setup should make it possible to give reasonable tracking in the first trials, and increasingly accurate tracking when the learning controller learns the correct input signal. This makes it possible to use a relatively low bandwidth for the feedback controller without sacrificing a lot of tracking accuracy.

Key in designing a successful controller will be robustness. Muscle behaviour is not very predictable. The muscle model can be expected to differ from trial to trial. Also, the control system has to be robust for unmodelled nonlinearities in the muscle behaviour and fatigue. The learning controller should be able to cope well with nonrepeatable disturbances.

Inside the feedback loop the inverse of the recruitment curve is used. This will make the

combination of the inverse recruitment curve and the real arm a much more linear system. When the mass of the systems for convenience is assumed to be equal to c, the transfer function equals:

$$H(s) = \frac{1}{s^2(\frac{1}{2\pi}s+1)^2} \tag{4.1}$$

Off course, in practice the mass will not equal 1. A simple change in the gain of the feedback controller can make up for this. Hence, this difference will, mutatis mutandis, not influence the accuracy of the designed control system. A system like this is suitable to be controlled using a PD controller combined with a low-pass filter. The achievable bandwidth is limited by the pole location of the muscle dynamics. Since these are located around 1 Hz, the bandwidth will be well under 1 Hz.

The ILC controller will directly change the reference signal. This, as will be shown, makes it possible to make a robust, simple and fast converging ILC controller. The model that will be used will be far from accurate. Therefore strict model based ILC algorithms will not be suitable.

4.3 Optimization

4.3.1 Feedback controller

For the feedback loop, a PD controller in combination with a low-pass filter will be used. The speed of the muscle dynamics will be a limit for the achievable bandwidth of such a controller, the open loop phase of the plant will drop here from -180 to -360. Since the muscle dynamics have a timescale varying with the amount of stimulation [5], it seems sensible to make sure that the bandwidth of the controller is well below the slowest possible timescale of the muscle dynamics. Failing to satisfy this criterion can destabilize the controller.

Tuning a PD+low-pass feedback controller is relatively easy. In this case the steps to be taken are: choose a safe bandwidth, choose the PD zero location below this bandwidth, choose the low-pass pole location above this bandwidth, tune the controller gain to get the chosen bandwidth.

It is attempted to put the bandwidth at 0.2 Hz. A higher bandwidth can be achieved, but this will go at the expense of robustness. Since the trajectory will be rather slow varying and the controller will be combined with an ILC controller, a higher bandwidth will hardly improve the tracking accuracy.

The next step is choosing the pole location of the controller. Usually, putting the controller gain a factor 3 below the bandwidth gives a good tradeoff between noise amplification and phase margin. In this case, it is attempted to make a controller that is very robust for changes in the gain of the plant, so the controller pole is placed at 0.04 Hz. This extra robustness is needed because of inaccurate modelling of the nonlinearities and fatigue.

In practice, noise is not much of a problem, a low pass-cut-off frequency of 3 Hz will do. This is well above the attempted bandwidth, a broad region of phase margin is created. This makes the controlled system very stable for changes in open-loop gain.

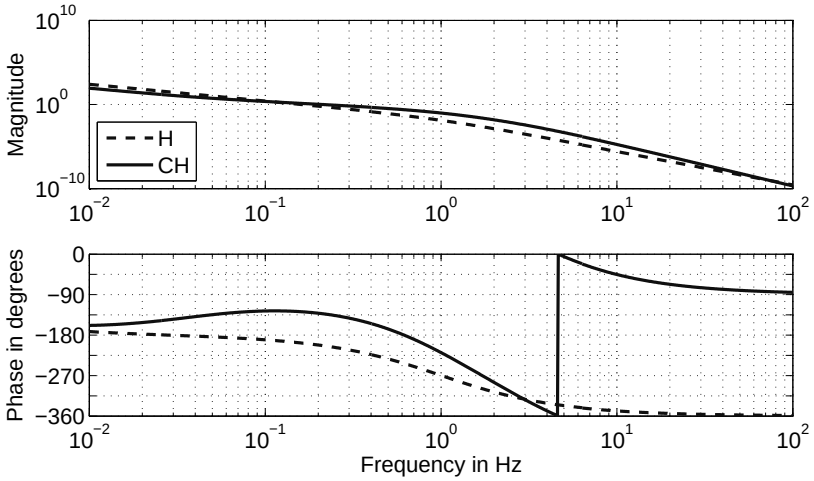

Figure 4.1: The open loop response of the plant and the controlled plant.

The last step in tuning the controller is setting the gain. In figure 4.1 the frequency response of both the original system and the controlled system are plotted.

4.3.2 ILC controller

A P-type iterative learning controller with a noncausal time shift and zero phase low-pass filtering will be implemented in the control design. For this, three values have to be tuned.
A good ILC controller will converge fast, is highly robust for plant changes and will also work well in the presence of nonrepeatable disturbances and noise. Keeping these three aspects in mind, an ILC controller will be tuned.

The ILC output will be added to the error signal that goes into the feedback controller. This setup is very suitable for P-type ILC controllers. In the part of the bode plot where the closed loop response is close to 1 (0 dB), a P-type ILC controller with an amplification of 1, $ILC_{k+1} = ILC_k + e_k$, will approximate deadbeat control.
This makes it possible to get very good convergence rates using a very simple ILC law. However, such a simple ILC law will not be stable. The error only decreases where the closed loop sensitivity is under 0 dB and increases where the sensitivity is over 0 dB. From the bode sensitivity integral it is known that the area of the sensitivity under 0 dB equals the area where the sensitivity is over 0 dB. For ILC it is not necessary to comply with the Bode sensitivity integral. An ILC law does not need to be causal.

Two changes to this simple ILC law are proposed. The first is to add a simple time shift. The second change is to lower the P-gain of the ILC controller. The new ILC law becomes: $ILC_{k+1}(t) = ILC_k(t) + \alpha e_k(t+T)$

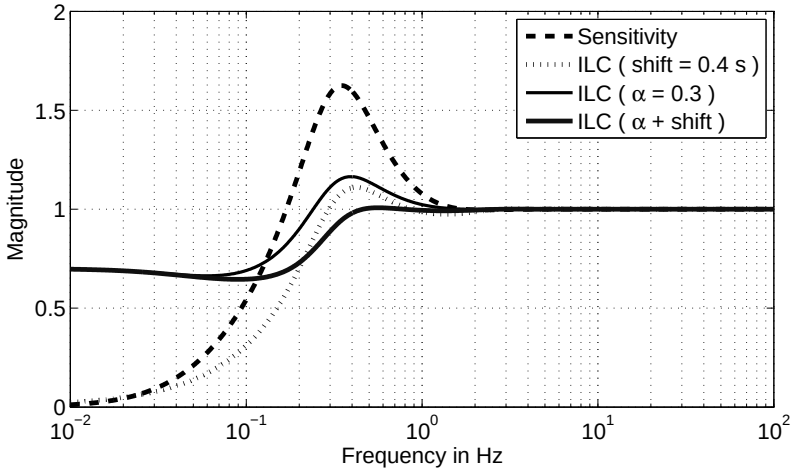

Figure 4.2: Convergence of ILC controllers

The convergence criterion can be derived in the following way.

$$\begin{aligned} e_{k+1} &= r - y_{k+1} \\ &= r - \tfrac{CH}{1+CH} \times (r + ILC_{k+1}) \\ &= r - \tfrac{CH}{1+CH} \times (r + ILC_k + \alpha e_k\,(t+T)) \\ &= r - \tfrac{CH}{1+CH} \times (r + ILC_k) - \tfrac{CH}{1+CH} \times \alpha e_k\,(t+T) \\ &= e_k - \tfrac{CH}{1+CH}\alpha e^{j\omega T} \times e_k\,(t) \\ &= \left(1 - \tfrac{CH}{1+CH}\alpha e^{j\omega T}\right) \times e_k\,(t) \end{aligned} \tag{4.2}$$

The $\times$ symbol is used to denote a convolution.
The symbol e is used both for denoting the mathematical constant (when used with a power) and the error signal (when used with a subscript).

From this, it can be concluded that the error decreases where $\left|1 - \frac{CH}{1+CH}\alpha e^{j\omega T}\right| < 1$. A plot of this criterion can be found in figure 4.2, this plot uses the plant and controller from figure 4.1.

Obviously, the small time shift makes it possible to get convergence up to a higher frequency and decrease the diverging peak. The bode sensitivity integral is no longer limiting. Changing the gain will decrease the amount of convergence for low frequencies and also increases the frequency region for which the ILC controller converges. A combination of the two almost eliminates divergence and gives convergence up to a high frequency.

The ILC law is not yet stable. Though hard to see in figure 4.2, the controller still diverges slightly for high frequencies. The obvious solution is to filter out all high frequency components. This way they can't diverge. This measure also reduces the amount of noise and disturbances that end up in the ILC controller output. The new ILC law becomes:

$$ILC_{k+1} = Q \times (ILC_k + \alpha e_k\,(t+T)) \tag{4.3}$$

The $\times$ symbol is again used to denote a convolution.

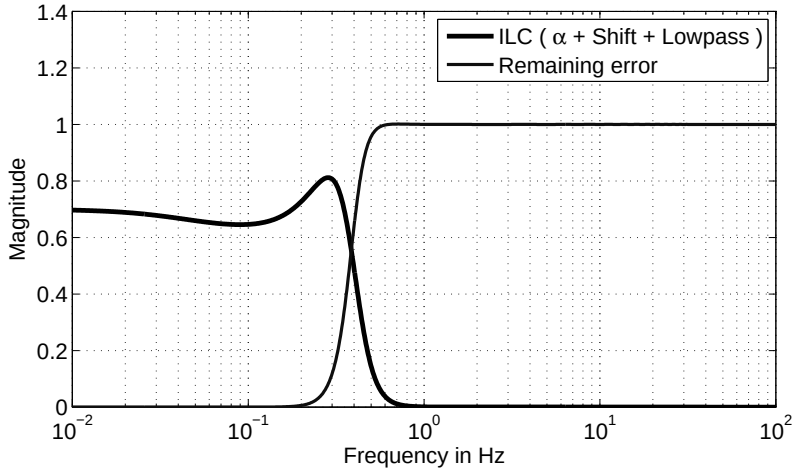

Figure 4.3: Convergence of the ILC controller. The thick line is the convergence criterion in the frequency domain, values greater than 1 will cause instabilities. The thin line gives the remaining part of the error, a value of 0 means that the error for that frequency will eventually completely disappear, a value of one means that there is no change in error at all for those frequencies.

Adding a low-pass filter Q changes the stability criterion to: $\left|Q - Q\frac{CH}{1+CH}\alpha e^{j\omega T}\right| < 1$ (proof can be found in appendix C).

The low-pass filter will be noncausal making it possible to filter out frequencies without causing phase lag by filtering in two directions. A fourth order zero-phase butterworth filter with a cut-off frequency of 0.4 Hz is used. The convergence criterion can be found in figure 4.3.

The ILC law is now stable. The thin line in figure 4.3 shows how much of the error remains once the ILC controller is fully converged. For low frequencies this error is virtually nonexisting, for high frequencies the error completely remains.
Some characteristics of this control system will be explained in the next sections.

4.4 Robustness

A very important aspect for this ILC controller is that it should be robust for changes in the system. Increasing the amount of fatigue is equivalent to lowering the gain. It is particularly hard to predict the amount of fatigue. Hence the control system should be robust for changes in the plant gain.

For the feedback controller a change in the system gain can cause instability in one way only. Lowering the system gain, i.e. increasing fatigue, will not destabilize the feedback loop. D action in the feedback controller and damping in the arm will make sure that there always is some phase margin at low frequencies.
Increasing the gain of the system can destabilize the feedback loop (figure 4.4). This increase in system gain can for example be caused by tuning a controller on a fatigued arm, taking a

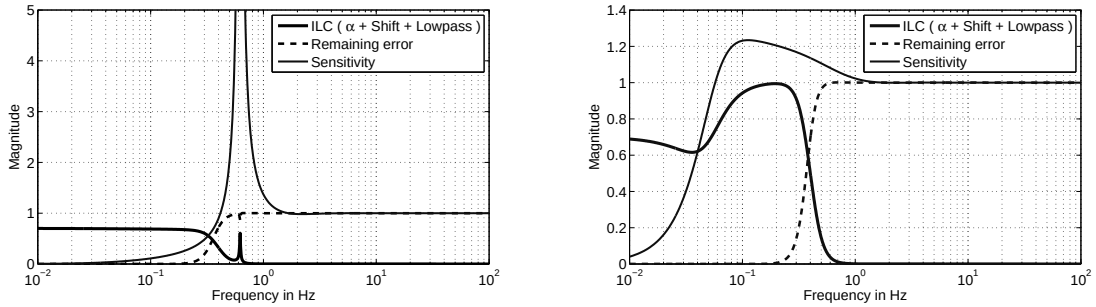

Figure 4.4: Convergence of the ILC controller. The left figure shows what can happen if the open loop gain is higher than expected. The peak of the sensitivity is very high, but because the low-pass filter filters out this frequency, the ILC controller is still stable. The performance of the feedback controller would be very bad. Because of this high peak there would be a lot of vibrations. The right figure shows what happens if the open loop gain is lower than expected. The peak of the sensitivity decreases, but shifted below the cut-off frequency of the low-pass filter. This gives an ILC controller that is on the verge of stability while having a very stable (but slow) feedback loop.

break and using the controller on a fresh arm. Another way for this to happen is by changing the position of the electrodes to a better place where the muscle force is greater with the same stimulation. The obvious solution is to decrease the gain of the controller, this will once again give a stable controller. To make sure the negative consequences of a bad controller are limited, the maximum pulse width should be limited.

The ILC controller can be destabilized as well (figure 4.4). However, an increase in system gain is not likely to be the cause of this. Increasing the gain will lower the sensitivity for low frequencies and shift its peak to a higher frequency. This means that the low-pass filter will still filter out the unstable frequencies. On the verge of stability, the peak of the sensitivity has a maximum of infinity. No low-pass filter can counteract this and in theory this can destabilize the ILC loop. In practice such a feedback controller will never be used. It will give very bad performance and a decrease in the feedback controller gain will solve this problem. Because a very steep low-pass filter is used, any reasonable high bandwidth feedback system will work.
A more likely cause of instability of the ILC controller is a decrease in open loop gain. This will cause the sensitivity at low frequencies to increase. The peak will probably be lower in magnitude, but more worrying, it will shift to a lower frequency. If the decrease in gain is sufficient, the peak can be sufficiently below the cut-off frequency of the low-pass filter. The time shift needed to properly decrease the peak at low frequencies is larger. All this amounts to the possibility of an unstable ILC controller when the achieved bandwidth is much lower than intended. The ILC controller can be made more robust for this by lowering the cut-off frequency and by using a higher than optimal time-shift. Both measures can deteriorate the nominal performance of the ILC law. Obviously, when tuning the feedback controller, care should be taken that the bandwidth is sufficiently high. When this is done, no problems with the ILC controller are expected.

Off course, changes in the pole location of the muscle dynamics can also destabilize the control system. However, no evidence is found that the pole location will change in time. The pole

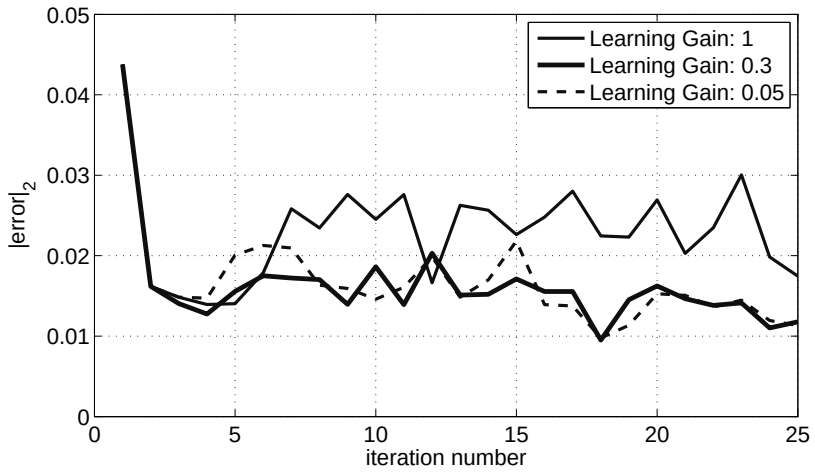

Figure 4.5: Simulated RMS errors using different ILC controllers. Thin line: an ILC controller using maximum convergence throughout; Thick line: the learning gain is changed from 1 to 0.3 after the first trial; Dashed line: the learning gain is changed from 1 to 0.05 after the first trial The noise added is about 1 % of the total movement and the slow disturbance about 50 %.

location will change as a function of the amount of stimulation. Higher stimulation levels will cause the pole to shift to a lower frequency. Using a bandwidth well below the lowest pole location to be expected makes the control system robust for this. Because the pole location doesn't change in time, gain changes are the most likely cause of instability for a control system that used to work. The ratio in plant gain between a high-gain system and a low-gain system that are both on the verge of stability is about a factor 20. This difference is much greater than can be expected to occur in practice.

4.5 Disturbances

A sizeable part of the error is expected to be caused by changing disturbances. When patients are also actively tracking the path themselves, differences in their activity can be expected. Also, variations in the position of the shoulder will cause disturbances.

Usually, a very fast converging ILC controller will be favourable, but when the error mainly consists of disturbances that will not be present in the next iteration, this is no longer the case. Deadbeat ILC control will try to fully counteract the error from the last iteration. When disturbances causing this error are no longer present, such an ILC controller will give an increase in error. Therefore, a lower convergence rate can give better performance. For the first iteration, the error is expected to be mainly caused by the tracking of the reference. Therefore, maximum convergence speed will be used in this trial. Afterwards, increasing amounts of fatigue will cause repeatable changes in the error profile, but disturbances are expected to form a large part of the error.

In figure 4.5 the difference in error is shown between a fast converging controller, a slower converging controller and a very slow converging controller. Obviously, the slower converging

controllers give the smallest error. The difference between the two slower controllers is not as obvious. Using the very low convergence rate doesn't seem sensible. Similar improvements can be expected by using the medium convergence rate. Increasing the convergence rate makes it possible for the ILC controller to counteract fatigue at greater speed. The ILC controller starting with 1 and afterwards using a learning gain of 0.3 seems to be a safe choice. After the main convergence in the first step it is still able to converge in a few steps if the system changes. The greater part of the nonrepeatable disturbance will not be counteracted by the ILC controller, giving close to optimal disturbance rejection for an ILC controller.

Also important for disturbance rejection is the low-pass filter. High frequency noise and other fast disturbances will almost completely be filtered out, leaving only low frequency disturbances to disturb the ILC controller. This is particularly important when the ILC controller is combined with a PD feedback controller. Such a controller will amplify noise which makes the controller less usable when a lot of noise will be provided at the input by the ILC controller. The low-pass is very steep and its cut-off frequency is located at 0.4 Hz making noise coming from the ILC controller virtually nonexisting.

Chapter 5

Results

In theory, the ILC algorithm has very convenient properties. The real test will be the performance in practice. From simulations without disturbances and fatigue, it is clear that the ILC controller can potentially increase the performance of the feedback loop by a factor 100. This would in practice leave an error that is too small to accurately measure. Simulations with disturbances show that disturbances have little influence on the convergence of the ILC controller, but since ILC is only intended to counteract repeatable errors, the error caused by the disturbances still remains. In these simulations some nonlinear properties of the system are not involved in the computations. This makes it particularly interesting if the algorithm will still perform as predicted.

5.1 Experiments

The final test for the ILC controller is a practical experiment doing an exercise similar to what hemiplegic patients will have to do. The test subject in this case was not hemiplegic. He was attempting to not actively move his arm. To make even more sure that the patient is not influencing the results, experiments are done with and without the patient looking at the movement. The attempted movement is a slow stretching movement of the arm. Only the triceps are being stimulated. To keep the person on the track the robot is proving forces as necessary in normal direction. The result of such an experiment is depicted in figure 5.1.

Obviously the ILC controller gives a significant improvement over the feedback controller (the first trial shows the performance of the feedback controller). The sudden increase in error in the third trial cannot be explained from the simulation results. But knowing that one of the electrodes came loose partly during this trial gives sufficient explanation for this. After the fifth trial this problem is solved and the ILC controller again converges to a very small error. Since the convergence rate is lowered after the first trial, the convergence is quite a bit slower now. Also interesting to see is the change of the amount of stimulation between trials. In figure 5.1 the RMS of the stimulation pulse width is depicted. It is clear that an increasing amount of stimulation is needed to keep following the trajectory. This clearly indicates a significant increase in fatigue during the experiment, especially when the influence of this change on the muscle recruitment is taken into account.

The main difference between this experiment and the practical application is that the patient

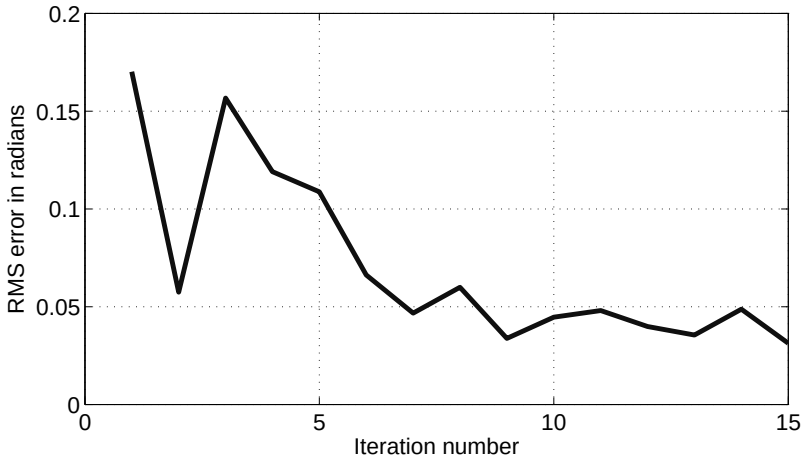

Figure 5.1: The average error as recorded during an experiment.

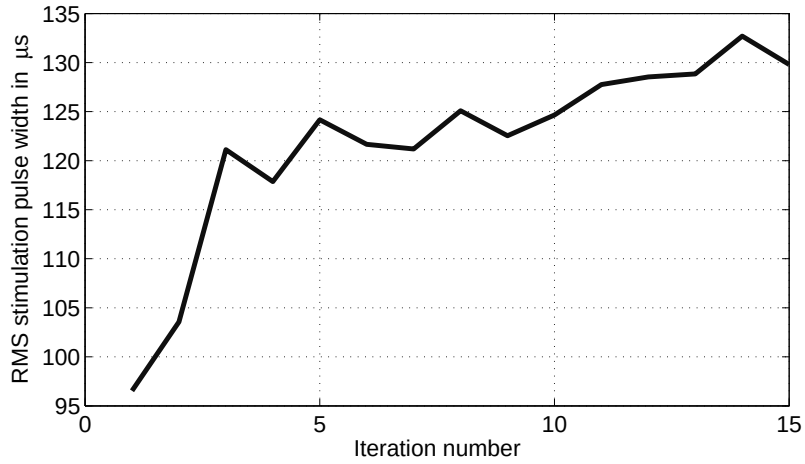

Figure 5.2: The average stimulation used to follow the trajectory.

performing the test had full use of his arm and was not using it. When a disabled person will have to do the test, he is asked to actively try to follow the trajectory. When the patients help changes significantly between trials, this can even have a negative influence on the performance of the ILC controller.

Chapter 6

Conclusions and Recommendations

A new project making use of FES to control the movement of paralysed human arms is started. For the control of this movement the use of an ILC controller to control the repeated movements was investigated.

A human arm was modelled with fourth order linear dynamics. When making a feedback controller for this system, the reaction speed of the muscles is limiting for the achievable bandwidth. Hence, the accuracy of this system is limited by the slow muscle dynamics.

A hybrid ILC-feedback control system has been designed using a PD + low-pass feedback controller and a P-type ILC controller. The ILC controller uses the measured error from the last trial to update the trajectory in such a way that the next error is expected to be smaller. A non-causal time shift of the error and a low-pass filter were used to improve stability and performance.

The stability of the control system for variations in the system due to mainly muscle fatigue is investigated and the controller was designed to be robust for these changes. It was found that excessive fatigue can destabilize the ILC controller. Lowering the cut-off frequency of the low-pass filter and increasing the non-causal time shift increases robustness for this.
Also analysed was the effect of disturbances. A fast converging ILC controller amplifies disturbances. Therefore, a trade-off needs to be made.

When the ILC controller was put into practice, a significant decrease in tracking error was achieved, making ILC successful. At first, the tracking error was a few centimeters when tracking a dot moving on a plane. After convergence of the ILC controller, the tracking error is only a few millimeters.

It seems sensible to focus future research on making a better model of the arm, making it possible to make a nonlinear feedback controller. Tuning the controller on a model taking into account the variations of output force with different positions presumedly gives an increase in overall performance. Simulations show that the current ILC controller makes it possible to virtually eliminate the complete repeatable part of the error. It is therefore not likely that a more complex ILC law will give a substantial increase in performance. Some improvement in tracking might be possible by adapting the feedback controller to the current amount of fatigue. This can for example be done by using an adaptive feedback controller, or by updating the gain of the controller in between trials.

Appendices

Appendix A

Extended Model

A more accurate model can make it possible to make a better control system. Also, knowledge of the differences between simulations and reality makes it possible to understand and predict where and why the outcome differs. In general, the more complicated a model is, the more complicated and time consuming it is to identify and to tune a controller for it. Therefore, if it is possible to get satisfying performance using a simple model, this is preferable. However, only proper knowledge of the physical situation makes it possible to choose the most suitable model.

A.1 Muscles

In this chapter a generalized version of a Hill model will be explained. Also, a more complex geometry will be taken into account. A method to identify this system will also be explained. Finally, a way to improve the controller using this knowledge will be suggested. A trade-off between identification speed and accuracy must be made. The accuracy achieved in the simulations is sufficient to use for the intended rehabilitation training. More accurate control can make it possible to use this technology for more complicated purposes.

The muscle model in the main text is a so called isometric model. It is the same regardless of the current geometry. A Hill model does take the changing geometry into account. It can give a force dependent on the muscle length and muscle contraction speed. These forces can be described by nonlinear functions, i.e. they are equivalent to nonlinear springs and dampers. The force of the active part can change with position as well. The force the muscle provides due to the stimulation depends not only on the level of stimulation, but also on the position and the speed of contraction.
In a standard Hill model, the passive force is described by nonlinear springs and dampers. The active force is described by a multiplication of the isometric model and two nonlinear functions. One is a function of muscle length and the other of muscle contraction speed.
The muscle length and speed are unknown quantities in our setup. It will be assumed that the length of the triceps can be estimated by a function with of only the muscle angle. When this relation is nonlinear, the speed of contraction will, according to the chain rule, be a function of both the angular velocity of the elbow and the joint angle. Therefore, when this relationship is unknown, it doesn't make much sense to treat these functions separately. The nonlinear spring and damper will be combined into one nonlinear function and the active muscle length and

muscle velocity functions will also be combined into one function. This gives a more general version of the Hill model, since the assumption that these functions were respectively parallel and in series no longer has to be satisfied. The identification process will be more difficult now, because instead of one-dimensional unknown functions, two-dimensional functions have to be identified.
In a mathematical format, the muscle force can be expressed as:

$$F(t) = f_p\left(\theta, \dot{\theta}\right) + f_a\left(\theta, \dot{\theta}\right) \cdot \left(te^{-\lambda t} \times r\left(u\left(t\right)\right)\right) \tag{A.1}$$

The $\times$ symbol is used to denote a convolution.
$r(r)$ is the muscle recruitment;
f_p and f_a are nonlinear functions describing respectively the active and passive muscle forces.

A.2 Lagrange

Not only the muscle model used was very simple. The mass geometry was simplified to a single point mass. In this section, using a Lagrange approach, more complicated dynamic equations will be derived.

A.2.1 Kinematics

Before deriving the dynamic equations, a kinematic description must be made. When the forearm is properly constrained, i.e. the arm and elbow can only move in the horizontal plane and the rotation of the hand is fixed as well, and the shoulder position is fixed, only two degrees of freedom are left. As coordinates describing this geometry, the rotation of the upper arm around the vertical axis and the difference in rotation between the forearm and upper arm around the vertical axis are used. These coordinates will respectively be called θ_1 and θ_2.

Using the Lagrange technique, the first step in deriving the dynamic equations is to calculate the kinetic energy. Due to the constraints, the arm can be seen as a two-link planar arm. Assuming the centers of mass are located on the axes of the forearm and upper arm, these centers won't move due to rotation around these axes. The full mass distribution of the arm can now be described using: the mass of the upper arm m_u with its center at distance r_u from the vertical axis through the shoulder; the rotational inertia I_u of the upper arm around the center of mass; the mass of the forearm m_f with its center at distance r_f from the elbow; the rotational inertia of the forearm I_f around the vertical axis and the center of mass; and finally the rotational inertia I_e associated with rotating (not bending, difference between α and β in figure A.1) the elbow. The kinetic energy can, using these symbols, be expressed as:

$$\begin{aligned} T &= \tfrac{1}{2}m_u\mathbf{v}_{cm,u}^2 + \tfrac{1}{2}\omega_u\mathbf{I}_u\omega_u + \tfrac{1}{2}m_f\mathbf{v}_{cm,f}^2 + \tfrac{1}{2}\omega_f\mathbf{I}_f\omega_f + \tfrac{1}{2}\omega_e\mathbf{I}_e\omega_e \\ &= \tfrac{1}{2}\left(I_u + m_u r_u^2 + I_f + m_f r_f^2 + m_f\left(L_u^2 + 2L_u r_f\cos\theta_2\right)\right)\dot{\theta}_1^2 + \\ &\quad \tfrac{1}{2}\left(I_f + m_f r_f^2\right)\dot{\theta}_2^2 + \left(I_f + m_f\left(r_f^2 + L_u r_f\cos\theta_2\right)\right)\dot{\theta}_1\dot{\theta}_2 + \tfrac{1}{2}I_e\dot{\alpha}^2 \end{aligned} \tag{A.2}$$

α, The rotation of the elbow, must be a function of the coordinates and is in particular a function of θ_2. This function can be found in appendix B.1.

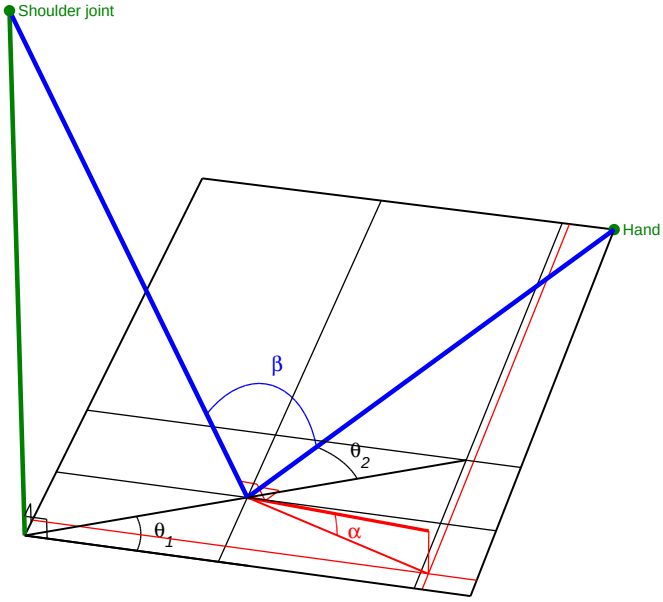

Figure A.1: Geometry of the arm. α: elbow orientation angle; β: elbow bend angle

A.2.2 Forces

The next step is defining the muscle forces. On the shoulder there will be a force due to its rotation and rotational speed, this is the so called passive muscle force. This force will be in the direction of θ_1. On the elbow there will also be a passive muscle force and in addition an active muscle force. It is possible to define these forces as being in the direction of θ_2 and using the constraints to counteract the force in the other directions. It will be assumed that it is advantageous to take this force in the direction of the elbow bend β, this replicates the physical situation better. It is hoped, that by doing this, the identified functions will be simpeler. Appendix B.2 deals with calculating this angle using θ_2. The forces applied by the robotic arm at the hand can be measured and will also be included in the dynamic equations. F_x And F_y are the forces at the hand in respectively the right and forward direction.

A.2.3 Lagrange

All forces and inertias are now defined. Using the fundamental holonomic form of Lagrange's equation [3], dynamic equations can be derived. Doing this gives the following equations.

$$\text{first}: \quad \begin{cases} \left(I_u + m_u r_u^2 + I_f + m_f r_f^2 + m_f L_u^2 + 2 m_f L_u r_f \cos(\theta_2)\right) \ddot{\theta}_1 \\ + \left(I_f + m_f r_f^2 + m_f L_u r_f \cos(\theta_2)\right) \ddot{\theta}_2 - m_f L_u r_f \sin(\theta_2) \left(2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2\right) \\ = \left(-L_u \sin\theta_1 - L_f \sin(\theta_1 + \theta_2)\right) F_x + \left(-L_u \cos\theta_1 - L_f \cos(\theta_1 + \theta_2)\right) F_y + F_{\theta_1} \end{cases}$$

$$\text{second}: \quad \begin{cases} \left(I_f + m_f r_f^2 + m_f L_u r_f \cos\theta_2\right) \ddot{\theta}_1 + \left(I_f + m_f r_f^2 + I_e \left(\frac{\partial \alpha(\theta_2)}{\partial \theta_2}\right)^2\right) \ddot{\theta}_2 \\ + \left(m_f L_u r_f \sin(\theta_2)\right) \dot{\theta}_1^2 + \left(I_e \frac{\partial \alpha(\theta_2)}{\partial \theta_2} \cdot \frac{\partial^2 \alpha(\theta_2)}{\partial \theta_2^2}\right) \dot{\theta}_2^2 \\ = L_f \sin(\theta_1 + \theta_2) F_x + L_f \cos(\theta_1 + \theta_2) F_y - \sqrt{\frac{\left(1 - \left(\frac{h_s}{L_u}\right)^2\right) \sin^2\theta_2}{1 - \left(1 - \left(\frac{h_s}{L_u}\right)^2\right) \cos^2\theta_2}} F_\beta \end{cases} \tag{A.3}$$

In these equations, F_{θ_1} and F_β represent the force due to the shoulder and elbow muscles. F_{θ_1} Will not be controlled and does not involve additional dynamics, the force is simply a function of the shoulder rotation θ_1 and the rotational velocity. F_β On the other hand is constituted of a passive part and an active part, having dynamics depending on the stimulation input.

A.2.4 Full model

The complete model consists of a combination of A.1 and A.3. Substituting A.1 for F_β and the passive part of A.1 for F_{θ_1} gives a fourth order nonlinear system. When all mass-parameters and functions describing the muscle behaviour are identified, this model can be used to predict the motion of the arm for any given stimulation input.

A.3 Identification

Even though the format of the dynamic equations is known, the identification is not a straightforward process. It is possible to use the model from the main text to simplify the identification process. The model in the main text already provides a recruitment curve and linear dynamics for the muscle. Using this, only the mass parameters and the nonlinear functions f_p (for the shoulder and the elbow) and f_a (for the elbow) have to be identified. Approximating these functions by determining its value at fixed grid-points and estimating the values in between these points by bilinear interpolation makes it possible to describe nearly any function with reasonable accuracy. Doing this, gives a model that, though having nonlinear dynamics, is completely linear in its parameters.

This characteristic makes it possible to efficiently fit these dynamic equations to measurements using a least squares fit. Doing this involves solving a set of linear equations which is a simple task using MATLAB. Moving the arm around while stimulating the muscle and measuring the force at the hand gives measurement data. Trying to make the dynamic equations fit at every measurement point gives an overdetermined system of equations. The solution giving the least squared error, gives the solution of maximum likelyhood. A MATLAB function doing

this was written but not yet successfully tested in practice. This code reduces noise and finds gridpoints where no measurement data is a available by minimizing a combination of the least squares error and the squared difference between adjoining gridpoints. This should give a more smooth and better usable estimation of the nonlinear muscle functions.

A.4 Controller improvement

The model from this chapter can directly be used to improve on the controller used in the main text. The nonlinear muscle function f_a describes the change of output force for changing positions and speeds. When it is known how the output force changes, the reciproque of this function can be inserted in the closed loop. This should have as a result that a certain error will result in the same output force at all positions. This makes it possible to tune a better feedback controller because the stability is much less depending on the position. Improvements in the feedback loop directly improve the performance of the ILC controller as well. It is now also possible to tune a new ILC law. When the bandwidth of the feedback controller increases, the cutoff frequency of the ILC low-pass can be increased. The time shift needed for optimal convergence will be lower. Obviously, overall performance should increase.

A.4.1 Discussion

Though the performance of the control system is expected to improve, it is not clear whether this means a practical improvement. To arrive at the increased performance, more measurements have to be performed every time the system is used. This brings about a less convenient usage of the system. Moreover, when the identified functions turn out to be inaccurate, due to e.g. disturbances during the identification process, the performance can actually deteriorate.
Since the simple control system already gave good performance, there is, from a practical point of view, no real need for a more complicated better performing control system.
Another use for this model can be to explain peculiarities of the system. Knowing where the differences between simulation and practice arise makes it possible to explain these differences and helps to draw sensible conclusions from simulation results.

Appendix B

Arm Geometry

B.1 Elbow orientation

To calculate the elbow orientation angle (α in figure A.1), the axis around which the elbow bends must first be calculated. This axis is perpendicular to both the forearm and the upper arm. If the directions of the forearm and the upper arm are known, the direction of the axis can be calculated using a vector cross product.

$$\begin{aligned}
\mathbf{a} &= \mathbf{f} \times \mathbf{u} = \det \begin{bmatrix} \mathbf{x} & \mathbf{y} & \mathbf{z} \\ f_x & f_y & f_z \\ u_x & u_y & u_z \end{bmatrix} \\
&= \det \begin{bmatrix} \mathbf{x} & \mathbf{y} & \mathbf{z} \\ \cos(\theta_1+\theta_2) & \sin(\theta_1+\theta_2) & 0 \\ -r_e\cos(\theta_1) & -r_e\sin(\theta_1) & h_s \end{bmatrix} \\
&= h_s \sin(\theta_1+\theta_2)\,\mathbf{x} - h_s\cos(\theta_1+\theta_2)\,\mathbf{y} + \\
&\quad -r_e\sin(\theta_1)\cos(\theta_1+\theta_2) + r_e\cos(\theta_1)\sin(\theta_1+\theta_2)\,\mathbf{z} \\
&= h_s \sin(\theta_1+\theta_2)\,\mathbf{x} - h_s\cos(\theta_1+\theta_2)\,\mathbf{y} + r_e\sin(\theta_2)\,\mathbf{z}
\end{aligned} \tag{B.1}$$

To get the required angle α from this, the length of the vector will be adjusted to unit length. After this, the inner product with the vertical unit vector will be calculated. This will give the sine of the required angle.

$$\begin{aligned}
\hat{\mathbf{a}} &= \frac{h_s\sin(\theta_1+\theta_2)\mathbf{x} - h_s\cos(\theta_1+\theta_2)\mathbf{y} + r_e\sin(\theta_2)\mathbf{z}}{\sqrt{(h_s\sin(\theta_1+\theta_2))^2 + (-h_s\cos(\theta_1+\theta_2))^2 + (r_e\sin(\theta_2))^2}} \\
&= \frac{h_s\sin(\theta_1+\theta_2)\mathbf{x} - h_s\cos(\theta_1+\theta_2)\mathbf{y} + r_e\sin(\theta_2)\mathbf{z}}{\sqrt{h_s^2 + r_e^2\sin^2(\theta_2)}}
\end{aligned} \tag{B.2}$$

$$\alpha = \arcsin(\hat{\mathbf{a}}\cdot\mathbf{z}) = \arcsin\left(\frac{r_e\sin(\theta_2)}{\sqrt{h_s^2 + r_e^2\sin^2(\theta_2)}}\right) \tag{B.3}$$

From the geometry of the arm, it could already be expected that the angle would only depend on θ_2. A rotation around the shoulder joint does not change the angle of the elbow.

To calculate the dynamics of the arm, the first and second derivative of the angle α with respect to θ_2 are needed as well.

$$\frac{d\alpha}{d\theta_2} = \frac{r_e h_s\cos(\theta_2)}{h_s^2 + r_e^2\sin^2(\theta_2)} \tag{B.4}$$

and

$$\frac{d^2\alpha}{d\theta_2^2} = -r_e h_s \sin(\theta_2) \frac{h_s^2 + r_e^2 + r_e^2 \cos^2(\theta_2)}{\left(h_s^2 + r_e^2 \sin^2(\theta_2)\right)^2} \tag{B.5}$$

and their multiple

$$\frac{d\alpha}{d\theta_2}\frac{d^2\alpha}{d\theta_2^2} = -\tfrac{1}{2} r_e^2 h_s^2 \sin(2\theta_2) \frac{h_s^2 + r_e^2 + r_e^2 \cos^2(\theta_2)}{\left(h_s^2 + r_e^2 \sin^2(\theta_2)\right)^3} \tag{B.6}$$

B.2 Elbow bend

Calculating the elbow bend angle β in figure A.1 can be done quite fast. The shoulder, elbow and hand form three points of a triangle. All lengths of this triangle are known: the length of the upper arm L_u, the length of the forearm, L_f and the distance between the shoulder and the hand which is a function of the generalized coordinates (B.7).

$$L_{hand} = \sqrt{L_u^2 + L_f^2 + 2L_f\sqrt{L_u^2 - h_s^2}\cos\theta_2} \tag{B.7}$$

Now, using the cosine rule, the following function for the elbow bend angle β can be derived.

$$\beta = \arccos\left(-\sqrt{1 - \left(\frac{h_s}{L_u}\right)^2}\cos\theta_2\right) \tag{B.8}$$

For deriving the dynamic equations, the derivative of β with respect to θ_2 will be used as well.

$$\frac{\partial\beta}{\partial\theta_2} = -\sqrt{\frac{\left(1 - \left(\frac{h_s}{L_u}\right)^2\right)\sin^2\theta_2}{1 - \left(1 - \left(\frac{h_s}{L_u}\right)^2\right)\cos^2\theta_2}} \tag{B.9}$$

Appendix C

ILC Convergence

Convergence to small error is the main reason ILC is used. It is therefore very important to know if the ILC controller converges, i.e. if the ILC controller is stable, and to what error it will converge.

The following is used as a starting point for a derivation of the convergence criterion for the ILC controller.

$$\begin{aligned} ILC_{k+1}(t) &= Q \times ILC_k + Q \times \alpha \cdot error_k(t+T) \\ ILC_{k+1}(\omega) &= Q \cdot ILC_k + Q \cdot \alpha \cdot e^{j\omega T} error_k \end{aligned}$$

$$y_k = \frac{CH}{1+CH} ILC_k + \frac{CH}{1+CH} r$$

$$error_k = r - \frac{CH}{1+CH} ILC_k - \frac{CH}{1+CH} r$$

C.1 Derivation

The derivation of the convergence criterion consists mainly of two steps. First a few iterations of the ILC controller are elaborated. Then, from this result, a general formula is found. For this general formula a convergence criterion is found. The derivation is done for a P-type ILC controller with time shift T and filter Q, but can easily be generalised to any linear ILC law satisfying the following format: $ILC_{k+1} = T_{ILC} ILC_k + T_{error} error_k$ with linear filters T_{ILC} and T_{error}.

The first step is:

$$\begin{aligned}
ILC_{k+1} &= QILC_k + Qe^{j\omega T}\left(r - \left(\tfrac{CH}{1+CH}ILC_k + \tfrac{CH}{1+CH}r\right)\right) \\
&= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)ILC_k + Qe^{j\omega T}\tfrac{1}{1+CH}r \\
&= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)\left(\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)ILC_{k-1} + Qe^{j\omega T}\tfrac{1}{1+CH}r\right) + \\
&\quad Qe^{j\omega T}\tfrac{1}{1+CH}r \\
&= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^2 ILC_{k-1} + Qe^{j\omega T}\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH} + 1\right)\tfrac{1}{1+CH}r \\
&= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^2\left(\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)ILC_k + Qe^{j\omega T}\tfrac{1}{1+CH}r\right) + \\
&\quad Qe^{j\omega T}\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH} + 1\right)\tfrac{1}{1+CH}r \\
&= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^3 ILC_{k-2} + \\
&\quad Qe^{j\omega T}\left(\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^2 + \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right) + 1\right)\tfrac{1}{1+CH}r \\
&= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^3\left(\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)ILC_k + Qe^{j\omega T}\tfrac{1}{1+CH}r\right) + \\
&\quad Qe^{j\omega T}\left(\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^2 + \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right) + 1\right)\tfrac{1}{1+CH}r \\
&= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^4 ILC_{k-3} + \\
&\quad Qe^{j\omega T}\left(\begin{array}{l}\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^3 + \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^2 + \\ \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right) + 1\end{array}\right)\tfrac{1}{1+CH}r
\end{aligned} \tag{C.1}$$

This can be generalised as follows:

$$\begin{aligned}
ILC_n &= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^n ILC_0 + \sum_{i=0}^{n-1}\left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^i Qe^{j\omega T}\tfrac{1}{1+CH}r \\
&= \left(Q - Qe^{j\omega T}\tfrac{CH}{1+CH}\right)^n ILC_0 + \frac{1-\left(Q-Qe^{j\omega T}\frac{CH}{1+CH}\right)^n}{1-\left(Q-Qe^{j\omega T}\frac{CH}{1+CH}\right)}Qe^{j\omega T}\tfrac{1}{1+CH}r
\end{aligned} \tag{C.2}$$

The equation C.2 converges to a fixed value when:

$$\left|Q\left(1 - e^{j\omega T}\frac{CH}{1+CH}\right)\right| < 1 \tag{C.3}$$

Hence, the ILC controller is stable if criterion C.3 is satisfied.

C.2 Limit values

Stability is not the only criterion the ILC controller has to satisfy. It is also interesting to see what performance the ILC controller can achieve. For this, the final error is calculated. To do this, first the final value of the ILC output is calculated.

$$ILC_\infty = \lim_{n\to\infty} ILC_n = \frac{1}{1 - \left(Q - Q\alpha e^{j\omega T}\frac{CH}{1+CH}\right)}Q\alpha e^{j\omega T}r - \frac{CH}{1+CH}Q\alpha e^{j\omega T}r \tag{C.4}$$

Now the following final error follows:

$$error_\infty = \left(1 - \frac{\frac{CH}{1+CH}Q\alpha e^{j\omega T}}{1 - Q + \frac{CH}{1+CH}Q\alpha e^{j\omega T}}\right)r + \frac{CH}{1+CH}\left(\frac{CH}{1+CH}Q\alpha e^{j\omega T} - 1\right)r \tag{C.5}$$

Bibliography

[1] R. BARATTA and M. SOLOMONOW. The Dynamic Response Model of Nine Different Skeletal Muscles.

[2] W.K. Durfee, K.E. Maclean, and I. INTRODUCTION. Methods for Estimating Isometric Recruitment Curves of Electrically Stimulated Muscle. *Biomedical Engineering, IEEE Transactions on*, 36(7), 1989.

[3] D.T. Greenwood. *Advanced Dynamics*. Cambridge University Press, 2003.

[4] AV Hill. The Heat of Shortening and the Dynamic Constants of Muscle. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 126(843):136–195, 1938.

[5] K.J. Hunt, M. Munih, N.N. Donaldson, and F.M.D. Barr. Investigation of the Hammerstein Hypothesis in the Modeling of Electrically Stimulated Muscle. *Biomedical Engineering, IEEE Transactions on*, 45(8), 1998.

[6] Royal College of Physicians. Stroke: Towards better management. *Royal College of Physicians*, 1989.

[7] Wood VA Skilbeck CE Wade DT, Langton-Hawer R and Ismail HM. The hemplegic arm after stroke: measurement and recovery. *J Neurol Neurosurg Psychiatry*, 46:521–524, 1983.