Control of underactuated mechanical systems

Brommer, A.

Published: 01/01/2004

Document Version
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
Control of Underactuated Mechanical Systems

Ard Bommer

DCT 2004.34

Traineeship report

Coach(es): dr. M. Reyhanoglu

Supervisor: Prof. dr. H. Nijmeijer

Technische Universiteit Eindhoven
Department Mechanical Engineering
Dynamics and Control Technology Group

Eindhoven, March, 2004
Thanks

I want to thank professor H. Nijmeijer, who made it possible for me to go on a traineeship to Embry-Riddle Aeronautical University in Daytona Beach in Florida, USA. He arranged the contact with Dr. M. Reyhanoglu, who is a professor at the department of Physical Sciences at ERAU.

Second of all I want to thank Dr. M. Reyhanoglu, who was my coach during my traineeship at EMU. He provided me with new control-techniques that enlarged my knowledge of non-linear control. Also he arranged us a stay at professor Nancy Parker.

I want to thank Nancy Parker and her husband Tom. They made my stay in the United States wonderful. I learned a lot of the American culture from them and they made it possible for me to see a lot of the environment of Florida.

Last, but not least, I want to thank Theo Geluk, who was a great companion during my stay there, and my family and girlfriend, who supported me all the way.
Abstract

In this report, different control techniques for underactuated systems are considered. Underactuated systems are systems that have more degrees of freedom to be controlled than independent control inputs. Several control techniques are considered for the point-to-point control of underactuated mechanical systems. After that, a control technique is considered for tracking control of underactuated mechanical systems. A modification of the control technique is designed to avoid singularities, such that a given trajectory can be followed. The control techniques are provided by papers of Dr. Reyhanoglu and other researchers in this area. Simulations, using several examples of underactuated systems, with Matlab are done to provide animations of the controlled system, for use in nonlinear control classes by Dr. Reyhanoglu.
## Contents

1 Introduction ................................. 3  
1.1 Why research in underactuated systems? .......... 3  
1.2 Underactuated systems and second-order nonholonomic constraints .... 3  
1.3 Brockett's condition .......................... 5  
1.4 Outline ....................................... 5  

2 Point-to-point control: the $\sigma$-process method 7  
2.1 General ...................................... 7  
2.1.1 Model formulation .......................... 7  
2.1.2 Transformation to linear form ................. 8  
2.1.3 Construction of a discontinuous controller ...... 9  
2.1.4 Main result ................................ 10  
2.2 The $\sigma$-process applied to a PPR underactuated manipulator 10  
2.2.1 Model formulation .......................... 11  
2.2.2 Controller design ........................... 12  
2.2.3 Simulation .................................. 13  
2.3 The $\sigma$-process applied to a planar manipulator with a unactuated elastically mounted end-effector .... 15  
2.3.1 Model Formulation .......................... 15  
2.3.2 Controller .................................. 17  
2.3.3 Simulation .................................. 18  

3 Point-to-point control: geometric phase method 21  
3.1 General ...................................... 21  
3.1.1 Bang-bang control ........................... 22  
3.2 Geometric phase method applied to an icebot .... 22  
3.2.1 Model formulation .......................... 23  
3.2.2 Controller .................................. 24  
3.2.3 Simulation .................................. 25  
3.3 Geometric phase method applied to a PPR underactuated manipulator .... 28  
3.3.1 Control .................................... 28  
3.3.2 Simulation .................................. 29  

4 Tracking control: cascaded backstepping 32  
4.1 General ...................................... 32  
4.1.1 Cascaded backstepping control ................. 33  
4.2 Cascaded backstepping applied on a PPR underactuated manipulator .... 34  
4.2.1 Control .................................... 34  
4.2.2 Simulation .................................. 35  
4.3 Cascaded backstepping applied on a surface vessel ........... 36
4.3.1 Transformation $Z$ ......................................................... 37
4.3.2 Transformation $X$ ......................................................... 40
4.3.3 Combination .............................................................. 41

5 Conclusions and Recommendations ........................................ 48

A Simulation figures ................................................................ 49
A.1 $\sigma$-process applied to a PPR underactuated manipulator .......... 49
A.2 $\sigma$-process applied to a planar manipulator with a unactuated elastically mounted end-effector ........................................ 50
A.3 Geometric phase applied to an icebot ..................................... 51
A.4 Geometric phase applied to a PPR underactuated manipulator .... 51
A.5 Cascaded backstepping applied on a PPR underactuated manipulator 53
A.6 Cascaded backstepping applied on a surface vessel .................. 54
A.6.1 Transformation $Z$ ......................................................... 54
A.6.2 Transformation $X$ ......................................................... 55
A.6.3 Combination .............................................................. 58
Chapter 1

Introduction

Control researchers have given considerable attention in the last ten years to many examples of control problems associated with underactuated mechanical systems, defined as systems with more degrees of freedom than inputs. As such, underactuated mechanical systems have some degrees of freedom that are unactuated. Examples of underactuated systems are: mobile robots, spacecrafts, space-robots, underwater vehicles, surface vessels and underactuated manipulators. The control problems for underactuated mechanical systems usually require nonlinear techniques. The linear approximation around equilibrium points may, in general, not be controllable and the feedback stabilization problem, in general, can not be transformed into a linear control problem. Therefore for such systems linear control methods can not be used to solve the feedback stabilization problem, not even locally.

1.1 Why research in underactuated systems?

Research in underactuated systems is driven by several reasons. Some motivations for the development of a control system with less actuators then degrees of freedom are:

- **Save fuel.** If less actuators is possible, one can reduce the cost of fuel or any kind of energy needed for any actuator to work.
- **Failure mode management.** If one of the actuators fails, one can still obtain the objective where the system is designed for.
- **Build compact.** If no space is needed for an actuator, space can be saved or used for other purposes.
- **Lighter structure.** If less actuators is possible, one can reduce the weight of the structure. This can be an advantage in positioning the structure.

As seen there are several practical reasons for research in underactuated systems. Theoretical insight of the problem is also needed, so there are also theoretical reasons for research in underactuated systems.

1.2 Underactuated systems and second-order nonholonomic constraints

Many underactuated systems are subject to nonholonomic constraints. There are two types of nonholonomic constraints; the first-order classical nonholonomic constraints and second-order nonholonomic constraints. The first-order nonholonomic constraints are defined as
constraints on the generalized coordinates and velocities of the form $\Phi(q, \dot{q}) = 0$ and can not be written as a constraint of the form $\phi(q) = 0$. They occur in, for example, wheeled mobile robots or wheeled vehicles with trailers. The second order nonholonomic constraints are defined as constraints on the generalized coordinates, velocities, and accelerations of the form $\Psi(q, \dot{q}, \ddot{q}) = 0$, which are non-integrable, i.e., can not be written as the time-derivatives of some functions of the generalized coordinates and velocities given by $\phi(q, \dot{q}) = 0$. They occur in, for example, surface vessels, space robots, and underactuated manipulators.

Consider an underactuated mechanical system and let $q = (q_1, \ldots, q_n)$ denote the set of generalized coordinates. Partition the set of coordinates as $q = (q_a, q_b)$, where $q_a \in \mathbb{R}^m$ denotes the directly actuated part and $q_b \in \mathbb{R}^{n-m}$ denotes the unactuated part. With $u \in \mathbb{R}^m$ as the vector of control variables, the equations of motion for underactuated mechanical system become:

$$\begin{align*}
M_{11}(q)\ddot{q}_a + M_{12}(q)\ddot{q}_b + F_1(q, \dot{q}) &= B(q)u \\
M_{21}(q)\ddot{q}_a + M_{22}(q)\ddot{q}_b + F_2(q, \dot{q}) &= 0
\end{align*} \tag{1.1} \tag{1.2}$$

The equations (1.2) define $n-m$ relations involving the generalized coordinates as well as their first-order and second-order derivatives. If there exists no non-trivial integral, i.e., a smooth function $\sigma(t, q, \dot{q})$ such that $d\sigma/dt = 0$ along all solutions of (1.2), then these $n - m$ relations can be interpreted as second-order nonholonomic constraints. In this report, only underactuated mechanical systems with second-order nonholonomic constraints are considered.

As an example of second order nonholonomic system, consider underactuated vehicles described by the following model:

$$\begin{align*}
M \ddot{v} + C(v)v + D(v)v + g(v) &= \begin{bmatrix} \tau \\ 0 \end{bmatrix} \\
\dot{\theta} &= J(\eta)v \\
\dot{\eta} &= J(\eta)v
\end{align*} \tag{1.3}$$

where $\eta \in \mathbb{R}^n, v \in \mathbb{R}^m, n > m$ and $\tau \in \mathbb{R}^k, k < m$. Here is $M$ the inertia matrix including added mass, $C(v)$ is the Coriolis and centripetal matrix, also including added mass, $D(v)$ is the damping matrix and $g(v)$ is the vector of gravitational and buoyant forces and torques. Let $M_u, C_u(v), D_u(v)$ and $g_u(v)$ denote the last $m - k$ rows of the matrices $M, C(v), D(v)$ and the vector $g(v)$, respectively. The constraint imposed by the unactuated dynamics can be written as

$$M_u \ddot{v} + C_u(v)v + D_u(v)v + g_u(v) = 0 \tag{1.4}$$

The gravitation and buoyancy vector $g(v)$ is important for the stabilizability properties of underactuated vehicles. If the vector $g_u(v)$ corresponding to the unactuated dynamics contains a zero function, then the constraint (1.4) is a second order nonholonomic constraint. It can be shown that underactuated vehicles subject to second-order nonholonomic constraints, do not satisfy Brockett’s necessary condition (see the next section) for asymptotic stabilization using time-invariant continuous state feedback.
1.3 Brockett’s condition

Consider the second-order chained form \[7\] with \( n = 3 \) variables and \( m = 2 \) inputs:

\[
\begin{align*}
\dot{\xi}_1 &= u_1 \\
\dot{\xi}_2 &= u_2 \\
\dot{\xi}_3 &= \xi_2 u_1
\end{align*}
\]  

which is given in state-space form as

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= u_1 \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= u_2 \\
\dot{x}_5 &= x_6 \\
\dot{x}_6 &= x_3 u_1
\end{align*}
\]  

where \( x_{2i-1} = \xi_i, x_{2i} = \dot{\xi}_i, i = 1, 2, 3 \). Define the state vector as \( x = [x_1, \ldots, x_6]^T \). For stabilization the Brockett’s condition can be stated as follows.

**Theorem** Assume that there exists a continuous time-invariant state-feedback \( u : \mathbb{R}^3 \to \mathbb{R}^2 \), that renders the origin of \( \dot{x} = f(x, u) \), with \( x \in \mathbb{R}^3 \) and \( u \in \mathbb{R}^2 \), asymptotically stable. Then the function \( f : \mathbb{R}^3 \times \mathbb{R}^2 \to \mathbb{R}^2 \) is locally surjective, i.e., the function \( f \) maps an arbitrary neighborhood of \((0,0) \in \mathbb{R}^3 \times \mathbb{R}^2 \) onto a neighborhood of \( 0 \) in \( \mathbb{R}^3 \).

Since the image of the mapping \((x, u) \mapsto f(x, u) = (x_2, x_4, x_6, u_1, u_2, x_3 u_1)\) of the second order chained form does not contain any point \((0, 0, 0, 0, 0, \varepsilon) \) for \( \varepsilon \neq 0 \), the system does not satisfy Brockett’s condition. Therefore, the system can not be stabilized by continuous time-invariant feedback. Underactuated systems with a single unactuated degree of freedom can often be transformed into the second order chained form as in (1.5). It is clear that underactuated systems can not be controlled by a continuous time-invariant controller. In this report, we will consider the design of discontinuous time-invariant controllers for point-to-point control.

1.4 Outline

After this introduction to underactuated mechanical systems, two control-techniques for point-to-point control are considered. The goal in point-to-point control is to bring a system from any initial state of the system to a desired state of the system. For point-to-point control the velocities of the state-variables are zero for the system in the initial and desired state.

The first control technique considered for point-to-point control is called the \( \sigma \)-process method. This technique is applied to two different systems. The first system considered is a planar PPR underactuated manipulator with two actuated prismatic joints and an unactuated revolute joint. The second system is an underactuated planar PPR robot manipulator with elastically mounted end-effector. The manipulator operates in a horizontal plane, and consists of two actuated prismatic links, an actuated revolute link, and an unactuated end-effector that is elastically mounted on the third link.

The second control technique considered for point-to-point control is called geometric phase method. A so-called icebot is controlled with this technique. An icebot consists of 3 links which are connected with 2 actuated revolute joints. Also the same three degrees of freedom planar PPR underactuated manipulator is controlled with this technique.
After the demonstration of the above control-techniques for point-to-point control, the tracking control problem for underactuated mechanical systems is considered. The goal in tracking control is to let a system follow a desired trajectory. This tracking control problem is solved using a control technique called cascaded back-stepping. This method is applied to the tracking control of the planar PPR underactuated manipulator described above as well as the tracking control of a surface vessel with two rear thrusters.
Chapter 2

Point-to-point control: the \( \sigma \)-process method

The first control-technique considered for point-to-point control of underactuated systems is the \( \sigma \)-process method. First the method is explained in general. After that it is applied to two systems, the PPR underactuated manipulator and the planar manipulator with an elastically mounted end-effector.

2.1 General

This control (derived in [1]) stabilizes for a special class of underactuated mechanical systems all degrees of freedom, including those that are not directly actuated, through nonlinear coupling. The class of systems are those with several actuated degrees of freedom \( m \geq 2 \) (with \( m \) the number of independent controls) and a single unactuated degree of freedom. The specific form of this nonlinear coupling depends on the physical features of the mechanical system. This control uses state transformations expressed in terms of the scaling variable to simplify the resulting equations of motion for the underactuated mechanical system. This simplification effectively decouples the model into a double integrator model for the scaling variable and its derivative and a control model for the remaining transformed variables that is linear but with coefficients depending on the scaling variable and its derivative. The control design is completed by using phase plane methods for control of the scaling variable and its derivative and piecewise linear control for the remaining transformed variables, this construction is subtle but avoids the singularities introduced by the rational transformations. The resulting controller is then expressed in terms of the original coordinates.

2.1.1 Model formulation

To explain this control technique we consider an unactuated mechanical system with a single unactuated degree of freedom. After suitable nonlinear state and control transformations, many examples of underactuated mechanical systems can be described by nonlinear control equations of the form:

\[
\begin{align*}
\ddot{q}_1 &= u \\
\ddot{q}_2 &= J(q_1, q_2)\dot{q}_1 + R(q_1, q_2, \dot{q}_1, \dot{q}_2)
\end{align*}
\]  

(2.1)

where \( q_1 = (q_{11}, \ldots, q_{1m}) \) denotes the configuration variables for the \( m \geq 2 \) directly actuated degrees of freedom, \( u_1 = (u_{11}, \ldots, u_{1m}) \) denotes the transformed control variables for
the directly actuated degrees of freedom, and \( q_2 \) denotes the scalar configuration variable for the unactuated degree of freedom. The functions \( J(q_1, q_2) \) and \( R(q_1, q_2, \dot{q}_1, \dot{q}_2) \), that characterize the coupling from the actuated degrees of freedom to the unactuated degree of freedom, reflect the fundamental physical features of the mechanical system. For simplicity of the control law and without loss of generality, it is assumed that \( R(0, 0, 0, 0) = 0 \). This implies that the origin is an equilibrium of (2.1) corresponding to \( u = 0 \).

### 2.1.2 Transformation to linear form

With the transformation

\[
\begin{align*}
v &= \ddot{q}_1 \\
\omega &= \dot{q}_2 - J(q)\dot{q}
\end{align*}
\]

so that (2.1) can be written in terms of the state variables \((q, v, \omega)\) as

\[
\begin{align*}
\dot{q}_1 &= v \\
\dot{q}_2 &= \omega + J(q)v \\
\dot{\omega} &= S(q, v, \omega) \\
\dot{v} &= u
\end{align*}
\]

where

\[
S(q, v, \omega) = R(q, v, \omega + J(q)v) - \dot{J}(q)v
\]

Consider the partitions

\[
q_1 = \begin{pmatrix} q_s \\ \dot{q}_1 \end{pmatrix}, \quad v = \begin{pmatrix} v_s \\ \dot{v} \end{pmatrix}, \quad u = \begin{pmatrix} u_s \\ \dot{u} \end{pmatrix}
\]

that reflect the selection of \( q_s \) as a scaling variable and introduce the rational transformations

\[
\zeta = \frac{q_2}{q_s}, \quad \eta = \frac{\omega}{q_s}
\]

which are defined so long as \( q_s \neq 0 \). In [1], it is assumed that \( J(q) \) and \( S(q, v, \omega) \) are nonlinear functions of the scaling variable \( q_s \) and its time derivative \( v_s \), and are linear with respect to the remaining variables. So they can be written as:

\[
\begin{align*}
J(q) &= J_{10}(q_s)q_s + J_{11}(q_s)\dot{q}_1 + J_{12}(q_s)q_2 + \dot{J}(q_s) \\
S(q, v, \omega) &= S_0(q_s, v_s) + S_1(q_s, v_s)\dot{q}_1 + S_2(q_s, v_s)\zeta + S_3(q_s, v_s)\eta + \ddot{S}(q_s, v_s)\dot{v}
\end{align*}
\]

Then (2.3) can be expressed as

\[
\begin{align*}
\dot{q}_s &= v_s \\
\dot{v}_s &= u_s
\end{align*}
\]
for the scaling variable \( q_s \) and its time derivative \( v_s \), and as the vector equations

\[
\begin{align*}
\dot{q}_1 &= \tilde{v} \\
\dot{\zeta} &= J_{11}(q_s) \frac{v_s}{q_s} \dot{q}_1 + \left( J_{12}(q_s) v_s - \frac{v_s}{q_s} \right) \zeta + \eta + \tilde{J}(q_s) \tilde{v} + J_{10}(q_s) v_s \\
\dot{\eta} &= S_1(q_s, v_s) \dot{q}_s + S_2(q_s, v_s) \zeta + \left[ S_3(q_s, v_s) - \frac{v_s}{q_s} \right] \eta + \tilde{S}(q_s, v_s) \tilde{v} + \\
\dot{\tilde{v}} &= S_0(q_s, v_s) \tilde{v} + S_0(q_s, v_s) \\
\end{align*}
\]  

Equations (2.8)-(2.9) are nonlinear but they have a very special form. Equations (2.8) are completely decoupled from (2.9), and if \((q_s, v_s)\) are viewed as given time-functions defined according to (2.8), then (2.9) defines a linear time-varying control system, at least so long as \( q_s \neq 0 \).

2.1.3 Construction of a discontinuous controller

A linear controller can be used to stabilize the origin of (2.8), and this controller should only depend on \((q_s, \dot{q}_s)\). By choosing

\[ v_s = -k q_s \]  

(2.10)

The following linear controller

\[ v_s = -K (v_s + k q_s) - kv_s \quad \text{with} \quad K > k > 0 \]  

(2.11)

guarantees that the resulting second-order close loop is linear with eigenvalues at \(-k\) and \(-K\), and the origin is a stable node (proof see [1]). The subset of the phase plane defined by

\[ \Omega = (q_s, v_s) : (q_s > 0 \text{ and } v_s + K q_s \geq 0) \text{ or } (q_s < 0 \text{ and } v_s + K q_s \leq 0) \]  

(2.12)

is an invariant set of (2.8,2.10,2.11) that excludes the line \( q_s = 0 \), and is globally attracting, except for \((q_s, v_s) = (0, 0)\).

Next a controller \( \hat{u} \) is constructed for (2.9), that is valid so long as \((q_s, v_s)\) lies in the set \( \Omega \) so that singularities in the rational transformations (2.6) are avoided. First assume constants exist according to the following limits:

\[
\begin{align*}
\lim_{q_s \to 0} J_{1i}(q_s) &= c_i & i &= 0, 1, 2, \\
\lim_{q_s \to 0} J(q_s) &= \tilde{c} \\
\lim_{q_s \to 0} S_j(q_s, -k q_s) &= s_j & j &= 0, 1, 2, 3 \\
\lim_{q_s \to 0} \tilde{S}(q_s, -k q_s) &= \tilde{s} \\
\end{align*}
\]  

(2.13)

Assume that the matrix pair

\[
A = \begin{pmatrix} 0 & 0 & 0 \\ -k c_1 & k & 1 \\ s_1 & s_2 & s_3 + k \end{pmatrix}, \quad B = \begin{pmatrix} \frac{I}{\tilde{c}} \\ \frac{\tilde{c}}{\tilde{s}} \end{pmatrix}
\]  

(2.14)

is stabilizable (see [1]) so that there exists a constant gain matrix

\[ l = (l_1, l_2, l_3), \ l_1 \in R^{1 \times 1}, \ l_2 \in R^{1 \times 1}, \ l_3 \in R^{1 \times 1} \]
such that the following matrix is Hurwitz:

\[
A_1 = A + Bl = \begin{pmatrix}
-l_1 & -l_2 & -l_3 \\
-(kc_1 + l_2\bar{c}) & k - l_2\bar{c} & 1 - l_3\bar{c} \\
-s_1 - l_1\bar{s} & s_2 - l_2\bar{s} & s_3 - l_3\bar{s} + k
\end{pmatrix}
\] (2.15)

By choosing

\[
\bar{v} = -l_1\bar{q}_1 - l_2\bar{c} - l_3\eta
\] (2.16)

and assuming that \((q_s, v_s) \in \Omega\), the following controller \(\bar{u}\) is constructed, using backstepping, so that the origin of (2.9) is attractive (proof see [1]):

\[
\bar{u} = -L(\bar{v} + l_1\bar{q}_1 + l_2\bar{c} + l_3\eta) + f(q_1, v, \zeta, \eta)
\] (2.17)

where \(L \in \mathbb{R}^{1 \times 1}\) is a constant gain matrix and

\[
f(q_1, v, \zeta, \eta) = -l_1\bar{v} - l_2\eta_q - \zeta v_s + J(q_1, \zeta q_s)v - l_3 S(q_1, \zeta q_s, v, \eta q_s) - \eta v_s
\]

In the original co-coordinates, controller (2.17) takes the form

\[
\bar{u} = -L(\bar{q}_1 + l_1\bar{q}_1 + l_2\frac{q_2}{q_s} + l_3\frac{\dot{q}_2 - J(q)\dot{q}_1}{q_s}) + \bar{f}(q, \dot{q})
\] (2.18)

where

\[
\bar{f}(q, \dot{q}) = -l_1\dot{q}_1 - l_2\frac{q_2\dot{q}_2 - \dot{q}_2 q_2}{q_s} - l_3\frac{q_2(R(q, \dot{q}) - \dot{J}q_1) - \dot{q}_2 q_2 - J(q)\dot{q}_1}{q_s}
\]

For the case where \((q_s, v_s) \notin \Omega\), the controller \(\bar{u}\) can be arbitrary defined, only subject to the constraint that no solution of the closed loop has a finite escape time. A particular choice is

\[
\bar{u} = -L(\bar{q}_1 + l_1\bar{q}_1) - l_1\dot{q}_1
\] (2.19)

### 2.1.4 Main result

The previous controller can be summarized as follows:

\[
u_s = -K(v_s + kq_s) - kv_s \\
\bar{u} = \begin{cases} 
-L(\bar{q}_1 + l_1\bar{q}_1 + l_2\frac{q_2}{q_s} + l_3\frac{\dot{q}_2 - J(q)\dot{q}_1}{q_s}) + \bar{f}(q, \dot{q}) & \text{if } (q_s, v_s) \in \Omega \\
-L(\bar{q}_1 + l_1\bar{q}_1) - l_1\dot{q}_1 & \text{if } (q_s, v_s) \notin \Omega
\end{cases}
\] (2.20)

which guarantees that the origin is a global attractor for all initial states of (2.8-2.9).

#### 2.2 The \(\sigma\)-process applied to a PPR underactuated manipulator

Here a control law, based on the \(\sigma\)-process, is constructed for a three degree of freedom planar PPR underactuated manipulator moving in a horizontal plane. The manipulator has two prismatic joints which are actuated, while the third joint is a revolute passive joint. The controller is designed using a \(\sigma\)-process. By using a coordinate transformation provided by [2] the system is transformed into chained form which makes it suitable for the application of the discontinuous control law. Simulation results are included in section 2.2.3, to demonstrate the effectiveness of the controller.
2.2.1 Model formulation

Consider a 3 link planar manipulator moving in a horizontal plane as shown in figure 2.1. Let \((x, y)\) denote the position of the free joint and \(\theta\) the orientation of the third link. Let \(F_1\) and \(F_2\) be the control inputs to the two prismatic joints. Also, denote \(l\) the distance between the center of mass of the third link and the third joint. The equations of motion [3] of the system are

\[
\begin{align*}
  m_x \ddot{x} - m_3 l^2 \cos \theta - m_3 l \dot{\theta} \sin \theta &= F_1 \\
  m_y \ddot{y} - m_3 l^2 \sin \theta + m_3 l \dot{\theta} \cos \theta &= F_2 \\
  l \ddot{\theta} + m_3 \dot{\theta} \cos \theta - m_3 l \dot{\theta} \sin \theta &= 0
\end{align*}
\]  

where \(m_x = m_1 + m_2 + m_3, m_y = m_2 + m_3\) and \(I = I_3 + m_3 l^2\). The last equation in (2.22) represents a second-order nonholonomic constraint on the system. Due to the absence of gravitational torques the three link manipulator is not linearly controllable.

The chained form of equations for a kinematic system makes it suitable for the application of a \(\sigma\)-process. After using a coordinate and input transformation given in [2], the system (2.22) can be written in the chained form given by

\[
\begin{align*}
  \ddot{x}_1 &= u_1 \\
  \ddot{x}_2 &= u_2 \\
  \ddot{x}_3 &= x_2 u_1
\end{align*}
\]  

where \(\xi = [x_1 \ x_2 \ x_3]^T\) and \(u = [u_1 \ u_2]^T\) are the new coordinates and the new input, respectively. The transformation is well-defined as long as the desired point \(q_d = \)
Following the notation in the section 2.1, the chained form (2.23) can be rewritten as

\[
\begin{align*}
\dot{\xi}_1 &= u_1 \\
\dot{\xi}_2 &= u_2 \\
\dot{\xi}_3 &= J(\xi)u_1
\end{align*}
\]  

(2.24)

where \( J(\xi) = [\xi_2 \ 0] \). Let \( v = [v_1 \ v_2]^T \), then in the new coordinates \((\xi, v, \omega)\) (using transformation (2.2)) equations (2.24) become (similar to (2.3))

\[
\begin{align*}
\dot{\xi}_1 &= v_1 \\
\dot{v}_1 &= u_1 \\
\dot{\xi}_2 &= v_2 \\
\dot{v}_2 &= u_2 \\
\dot{\xi}_3 &= \omega + J(\xi)v \\
\dot{\omega} &= \bar{S}(\xi, v, \omega)
\end{align*}
\]

(2.25)

where

\[
\begin{align*}
J(\xi)v &= \xi_2 v_1, \\
S(\xi, v, \omega) &= -J(\xi)v = -v_1 v_2
\end{align*}
\]

Consider the transformations, with \( \xi_1 \) as the scaling variable (similar to (2.6))

\[
\begin{align*}
z_1 &= \xi_2, \\
z_2 &= \frac{\xi_3}{\xi_1}, \\
z_3 &= \frac{\omega}{\xi_1}
\end{align*}
\]

(2.26)

which are defined so long as \( \xi_1 \neq 0 \). Then

\[
\begin{align*}
J_{10}(\xi_1) &= 0, \\
J_{11}(\xi_1) &= 1, \\
J_{12}(\xi_1) &= 0, \\
J(\xi_1) &= 0, \\
S_j(\xi_1, v_1) &= 0, \\
\bar{S}(\xi_1, v_1) &= -\frac{v_1}{\xi_1}
\end{align*}
\]

in the functions \( J(\xi) \) and \( \frac{S(\xi, v, \omega)}{\xi_1} \) (as in (2.7)). Then (2.25) can be expressed as (similar to (2.8)-(2.9)):

\[
\begin{align*}
\dot{z}_1 &= v_1 \\
\dot{z}_2 &= v_2 \\
\dot{z}_3 &= \frac{v_1}{\xi_1}(z_2 + z_2)
\end{align*}
\]

(2.27)

2.2.2 Controller design

A linear control law, by choosing \( v_1 = -k_2 \xi_1 \), of the form (similar to (2.20))

\[
u_1 = -k_1(v_1 + k_2 \xi_1) - k_2 v_1
\]

(2.28)

with \( k_1 > k_2 > 0 \) constants, is used to stabilize the origin of the first two variables \((\xi_1, v_1)\) of the system (2.27). The above controller places the eigenvalues at \(-k_1\) and \(-k_2\), where \((\xi_1, v_1) = (0, 0)\) is a stable node. The allowable set in which the trajectories originate is given by (similar to (2.12))

\[
\Omega = (\xi_1, v_1): (\xi_1 > 0 \text{ and } v_1 + k_1 \xi_1 \geq 0) \text{ or } (\xi_1 < 0 \text{ and } v_1 + k_1 \xi_1 \leq 0)
\]

(2.29)

which excludes the line \( \xi_1 = 0 \), and is an attractor that excludes the point \((0, 0)\).
Now assume that \((\xi_1, v_1)\) lies in the set \(\Omega\), then the singularities are avoided in the transformations (2.26). Then the following constants can be defined (similar to (2.13)):

\[
\begin{align*}
c_1 &= \lim_{\xi_1 \to 0} J_{11}(\xi_1) = 1 \\
\bar{s} &= \lim_{\xi_1 \to 0} \bar{S}(\xi_1, -k_2 \xi_1) = k_2
\end{align*}
\]

such that the matrix-pair \(A\) and \(B\) (similar to (2.14)) become

\[
A = \begin{pmatrix} 0 & 0 & 0 \\
-k_2 & k_2 & 1 \\
0 & 0 & k_2 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\
0 \\
k_2 \end{pmatrix}
\]  (2.30)

The matrix-pair is controllable (\(\text{rank}[B \ AB \ A^2B] = 3 = n\)) and stabilizable so that there exist a constant gain vector \(l = (l_1, l_2, l_3)\), such that the matrix \(A_1 = A + Bl\) is Hurwitz.

Then, by choosing \(v_2 = -l_1 z_1 - l_2 z_2 - l_3 z_3\) and assuming that \((\xi_1, v_1) \in \Omega\), the following controller is constructed so that the origin of the last four variables \((z_1, z_2, z_3, v_2)\) of the system (2.27) is attractive (similar to (2.21)):

\[
u_2 = -L(v_2 + l_1 z_1 + l_2 z_2 + l_3 z_3) - l_1 \dot{z}_1 - l_2 \dot{z}_2 - l_3 \dot{z}_3
\]  (2.31)

with

\[
\begin{align*}
\dot{z}_1 &= v_2 \\
\dot{z}_2 &= z_3 - k_2 z_1 + k_2 z_2 \\
\dot{z}_3 &= k_2 v_2 + k_2 z_3
\end{align*}
\]

In the region where \((\xi_1, v_1) \notin \Omega\) the following controller is chosen:

\[
u_2 = -L(v_2 + l_1 z_1) - l_1 v_2
\]  (2.32)

The origin of the control law defined by (2.28), (2.31) and (2.32) guarantees that the origin is an attractor for all initial states except those that lie on the manifold \((\xi_0, v_0) = (0, 0)\). By using a open loop control \(u_1\) during some small time interval, \(\xi_1\) can be steered to a nonzero value \(\epsilon\), such that no large control effort is required for values close to zero. The set (2.29) can be modified to:

\[
\tilde{\Omega} = \Omega \cap \{(\xi_1, v_1) : |\xi_1| > \alpha, \alpha > 0\}
\]  (2.33)

And the control law (2.28), (2.31) and (2.32) becomes:

\[
\begin{align*}
u_1 &= -k_2(v_1 + (\xi_1 - \epsilon)) \\
u_2 &= -L(v_2 + l_1 \xi_2) - l_1 v_2 \\
u_1 &= -k_1(v_1 + k_2 \xi_1) - k_2 v_1 \\
u_2 &= -L(v_2 + l_1 \xi_1 + l_2 \xi_2 + l_3 v_2) - l_1 v_2 + l_2 (\dot{\xi}_1 - k_2 \xi_2 + k_2 \dot{\xi}_2) - l_3 (k_2 v_2 + k_2 \dot{\xi}_2)
\end{align*}
\]  (2.34)

### 2.2.3 Simulation

Simulations are done using the control given by (2.34) in Matlab, to evaluate the motion of the PPR using the \(\sigma\)-process as control-technique. The PPR parameters are set at \(m_1 = m_2 = m_3 = 1[kg], l = 1[m]\). The control parameters are \(k_1 = 1, k_2 = 2\) and \(L = 1\).
The eigenvalues of the matrix $A_1$ are placed at $\{-1, -2, -3\}$. The constant chosen for the alternative controller is $\epsilon = 0.5$. And the constant $\alpha$ chosen in the set $\Omega$ is 0.5.

The initial rest-situation is $(x, y, \theta) = (0, 0, \frac{\pi}{2})$ in the original coordinates, which means for the transformed coordinates a initial rest situation of $(\xi_1, \xi_2, \xi_3) = (-2.77, 0.27, -0.47)$. The desired rest-situation is $(2, 2, \frac{\pi}{2})$, which means that the transformed coordinates have to be driven to the origin $(0, 0, 0)$.

![Figure 2.2: x, y and $\theta$](image)

As can be seen in figures 2.2 and A.1 the controller (2.34) controls the initial rest state to the desired rest state. Problems occur when the $\theta - \theta_d \geq \frac{\pi}{2}$, because of the singularities of the local transformation in [2]. The chained equations (2.23) are valid only in the domain $S$. More results are shown in the appendix A.1. In figure A.1 the motion of the PPR is illustrated. The prismatic joints are not shown, for the clarity of the figure. Note also that for the same reason the PPR is plotted smaller then its original size used in the simulation. As seen in the figure the PPR first approaches the desired point and then by moving from left to right and approaching slowly, the desired orientation is obtained. In figure A.2, the transformed variables $(\xi_1, v_1, \xi_2, v_2, \xi_3, w)$ are shown. It is clear that the motion of the scale-variable $\xi_1$ is specific for the linear control law (2.28).

If an initial and desired orientation are chosen such that these are close to the boundary of the domain $S$, the control effort increases. If the initial value of the scaling variable $\xi_1$ is zero, then $\xi_1$ is steered to a nonzero value using the first part of controller (2.34). If $|\xi_1(t)| \geq \alpha$, the control is switched back to the second part of controller (2.34). Considerable attention has to be given to the chosen eigenvalues. The controlled system is very sensitive for the placement of the eigenvalues. If these differ too much from $\{-1, -2, -3\}$, then the system is not controlled to its desired rest state.
2.3 The $\sigma$-process applied to a planar manipulator with a unactuated elastically mounted end-effector

Here a control law, based on the $\sigma$-process, is constructed for a three degree of freedom planar manipulator with an unactuated elastically mounted end-effector. The manipulator acts in a horizontal plane and consists of two prismatic joints and one revolute joint which are actuated, while the end-effector is elastically mounted to the third link. This system is almost the same as the system used in section 2.2, but this system has an actuated revolute joint with an elastically mounted end-effector instead of an unactuated revolute joint. Simulation results are included in section 2.3.3 to demonstrate the effectiveness of the controller.

2.3.1 Model Formulation

Consider a planar robot manipulator with three actuated joints and a single unactuated degree of freedom that describes the relative motion of an end-effector that is elastically mounted on the manipulator as shown in figure 2.4. Let $(x, y)$ denote the position of the revolute joint and $\theta$ the orientation of the third link, which is also the orientation of the end-effector. External control forces, $F_1$ and $F_2$, act on the two prismatic joints and an external control moment $T$ acts on the revolute joint. The end-effector is modelled as a single degree of freedom mass that is constrained to move axially with respect to the third link, where the relative position of the end-effector is denoted by $s$. There is an elastic restoring force on the end-effector with respect to the third link. The constant $c$ denotes the length of the third link.
By defining input transformations from \((F_1, F_2, T)\) to new control inputs \((u_1, u_2, u_3)\), the equations of motion can be written in a nonlinear control system from [4]

\[
\begin{align*}
\dot{\theta} &= u_1 \\
\dot{y} &= u_2 \\
\dot{x} &= u_3 \\
\ddot{s} &= -u_2 \sin \theta - u_3 \cos \theta + (c + s)\dot{\theta}^2 - k_s s \\
\end{align*}
\]

(2.35)

where \(k_s > 0\) is the elastic stiffness of the spring connecting the end-effector with the third link. The following design constraint is imposed to achieve the control objective

\[-\dot{y} \sin \theta - \dot{x} \cos \theta + (c + s)\dot{\theta}^2 - k_s s + \lambda^2 s + 2\lambda \dot{s} = 0\]

(2.36)

which implies that

\[\ddot{s} = -\lambda^2 s - 2\lambda \dot{s}\]

(2.37)

Thus the constant \(\lambda > 0\) denotes the desired decay rate of the unactuated relative end-effector dynamics. This constraint \((2.36)\) can be equivalently written as

\[u_3 = -u_2 \tan \theta + [(c + s)\dot{\theta}^2 - k_s s + \lambda^2 s + 2\lambda \dot{s}] \sec \theta\]

(2.38)

so long as the singularities at \(\theta = \pm \frac{\pi}{2}\) are avoided. With \((2.38)\) the equations of motion take the form

\[
\begin{align*}
\dot{\theta} &= u_1 \\
\dot{y} &= u_2 \\
\dot{x} &= -u_2 \tan \theta + [(c + s)\dot{\theta}^2 - k_s s + \lambda^2 s + 2\lambda \dot{s}] \sec \theta \\
\ddot{s} &= -\lambda^2 s - 2\lambda \dot{s} \\
\end{align*}
\]

(2.39)
Let $q = (\theta, u, x, s)$. Equations (2.39) can be used to design feedback controllers $u_1(q, \dot{q})$, $u_2(q, \dot{q})$. And with these controllers, the feedback control $u_3(q, \dot{q})$ can be determined using relation (2.38), thereby obtaining a feedback law $(u_1(q, \dot{q}), u_2(q, \dot{q}), u_3(q, \dot{q}))$ for the system (2.35). Considering the first three equations in (2.39), the following reduced order system is obtained (similarly to (2.3))

\[
\begin{align*}
\dot{\theta} &= v_1 \\
\dot{v}_1 &= u_1 \\
\dot{y} &= v_2 \\
\dot{v}_2 &= u_1 \\
\dot{x} &= w - v_2 \tan \theta \\
\dot{w} &= v_1 v_2 \sec^2 \theta + [(c + s) \dot{\theta}^2 - k_s s + \lambda^2 s + 2 \lambda \dot{s}] \sec \theta
\end{align*}
\] (2.40)

where $w = \dot{x} + y \tan \theta$ (as in transformation (2.2)). Consider the transformations, with $\theta$ as the scaling variable

\[
\eta_1 = y, \quad \eta_2 = \frac{x}{\dot{\theta}}, \quad \eta_3 = \frac{w}{\dot{\theta}}
\] (2.41)

which are defined so long as $\theta \neq 0$. Then (2.40) can be expressed as (similar to (2.8)-(2.9))

\[
\begin{align*}
\dot{\theta} &= v_1 \\
\dot{v}_1 &= u_1 \\
\dot{\eta}_1 &= v_2 \\
\dot{\eta}_2 &= \eta_3 - \frac{v_1}{\theta} \eta_2 - \frac{\tan \theta}{\theta} v_2 \\
\dot{\eta}_3 &= -\frac{v_1}{\theta} \eta_3 + [(c + s) v_1^2 - k_s s + \lambda^2 s + 2 \lambda \dot{s}] \sec \theta + \frac{v_1 v_2}{\theta} \sec^2 \theta \\
\dot{v}_2 &= u_2
\end{align*}
\] (2.42)

which is defined so long as $\theta \neq 0$ and $-\frac{\pi}{2} < \theta < \frac{\pi}{2}$.

### 2.3.2 Controller

A linear control law, by choosing $v_1 = -k \theta$, of the form (similar to (2.20))

\[
u_1 = -K_1 (v_1 + k \theta) - kv_1
\] (2.43)

with constants $K_1 > k$ and $\lambda > k > 0$, is used to stabilize the origin of the first two variables $(\theta, v_1)$ of the system (2.42). The above controller places the eigenvalues at $-K_1$ and $-k$, where $(\theta, v_1) = (0, 0)$ is a stable node. While the following set has to be avoided

\[
N = \{(q, \dot{q}) \in \mathbb{R}^4 | \theta = 0, (q, \dot{q}) \neq 0\}
\] (2.44)

to avoid singularities in transformation (2.41) and to avoid the case that $\theta = 0$ while the whole state $(\theta, y, x, w, \dot{\theta}, \dot{y}, \dot{x}, \dot{w}) \neq 0$. Because it is not wanted that $\theta$ is zero unless the whole state (including the time-derivatives) is zero.

Now assume that the set $N$ is avoided. Then the following constants can be defined (similar to (2.13))

\[
\begin{align*}
\bar{c} &= \lim_{\theta \to 0} \frac{\tan \theta}{\theta} = 1 \\
\bar{s} &= \lim_{\theta \to 0} \frac{\sec^2 \theta}{\theta} = 1
\end{align*}
\] (2.45)
And the limit \( \lim_{\theta \to 0} \frac{\sec \theta}{\theta} = 0 \) will cause the term \([c + s] \dot{\theta}^2 - k_2 s + \lambda^2 s + 2 \lambda \delta \frac{\sec \theta}{\theta}\) to go to zero as \( t \to \infty \). Then the matrix pair \( A \) and \( B \) (similar to 2.14) become

\[
A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & k & 1 \\ 0 & 0 & k \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ -1 \\ -k \end{pmatrix}
\] (2.46)

The matrix-pair is controllable (\( \text{rank}[B \ AB \ A^2B] = 3 = n \)) and stabilizable so that there exists a constant gain vector \( L = (l_1, l_2, l_3) \), such that the matrix \( A_1 = A + BL \) is Hurwitz.

Then, by choosing \( \nu_2 = -l_1 \eta_1 - l_2 \eta_2 - l_3 \eta_3 \) and assuming that \( (\theta, v_1) \) are not in the set \( N \), the following controller is constructed so that the origin of the last four variables \( (\eta_1, \eta_2, \eta_3, \nu_2) \) of the system (2.42) is attractive (similar to (2.21)):

\[
u_2 = -K_2(\nu_2 + l_1 \eta_1 + l_2 \eta_2 + l_3 \eta_3) - l_1 \dot{\eta}_1 - l_2 \dot{\eta}_2 - l_3 \dot{\eta}_3
\] (2.47)

with

\[
\dot{\eta}_1 = \nu_2 \\
\dot{\eta}_2 = \eta_3 - k \nu_2 + \frac{\tan \theta}{\theta} \nu_2 \\
\dot{\eta}_3 = k \nu_3 + [k^2 \theta^2 (c + s) - k_2 s + \lambda^2 s + 2 \lambda \delta] \frac{\sec \theta}{\theta} + k \sec^2 \theta \nu_2
\]

In [4] it is proven that the feedback controllers (2.43), (2.47) and (2.38) guarantee that if \( \theta_0 \neq 0 \) and \( \theta_0 (\theta_0 + k \theta_0) \geq 0 \), then \( \theta(t) \neq 0, \forall t \in [0, \infty) \). Thus, for all initial conditions satisfying \( \theta_0 \neq 0, -\frac{\pi}{2} < \theta_0 < \frac{\pi}{2} \) and \( \theta_0 (\theta_0 + k \theta_0) \geq 0 \), the feedback control laws (2.43), (2.47) and (2.38) are well-defined for all \( t \geq 0 \). If the initial conditions are not satisfied then the following controller is chosen

\[
u_1 = -k(\nu_1 + (\theta - \epsilon_0)) \\
u_2 = \nu_3 = 0
\] (2.48)

to transfer the system to a state in finite time satisfying the above conditions.

### 2.3.3 Simulation

Simulations are made using the control given in section 2.3.2, to evaluate the motion of the planar manipulator with a unactuated elastically mounted end-effector. The objective in this simulation is to move the manipulator to a specified equilibrium position, while not exciting the unactuated end-effector dynamics.

The end-effector stiffness is \( k_2 = 0.5 \). The length of the third link is assumed to be \( c = 1 \). The control parameters are \( k = 1, \lambda = 3, K_1 = 5, K_2 = 5 \). The eigenvalues of the matrix \( A_1 \) are placed at \(-1, -3, -2\). The constant chosen for controller (2.48) is \( \epsilon_0 = \frac{\pi}{4} \).

The initial state is given by \((\theta, \nu, \dot{\theta}, \nu, \dot{\nu}) = (\pi/4, 1, 1, 0.2, 0, 0, 0, 0, 0)\) corresponding to a nontrivial initial motion of the end-effector. The desired equilibrium state is given by \((q, \dot{q}) = (0, 0, 0, 0, 0, 0, 0, 0, 0)\).

The results of the simulations are shown in figures 2.5 and 2.6:
Figure 2.5: $\theta$, $y$, $x$ and $s$

Figure 2.6: $u_1$, $u_2$ and $u_3$
As can be seen in figure 2.5, the initially perturbed relative end-effector motion converges to the origin while the other states converge to the desired values. If an initial condition is chosen such that $\theta_0$ is close to zero, then controller (2.48) drives the system to a nonzero value of $\theta$. If $\theta$ is greater or equal to a certain "switch" value of $\theta$ (here $\theta_{\text{switch}} = 0.3$ is used), then the control switches to controllers (2.43), (2.47) and (2.38). In appendix A.2 the motion plot is given for this simulation. Here the relative position of the end-effector is scaled up, for the clarity of figure A.3. Again the two prismatic joints are left out. While bringing the orientation $\theta$ to the desired value ($=0$), the actuator moves in the $x$ and $y$-direction to the desired position such that the end-effector dynamics are not excited. The same holds for this simulation as for the simulation with the PPR in section 2.2.3, considerable attention for choosing the eigenvalues has to be given. If i.e. the second eigenvalue is chosen lower than $-3$, the movement of the manipulator in the $y$-direction is more changing in direction (positive/negative) and has more overshoot than if the second eigenvalue is $-3$. If these differ too much from $\{-1, -3, -2\}$, the system is not controlled to its desired state.
Chapter 3

Point-to-point control: geometric phase method

Another technique, which is able to control any initial condition to the origin, is considered in this section. In the literature, the geometric phase method is widely applied to mobile robots (see e.g. [5]) and other systems with first-order nonholonomic constraints. In this section, this control technique is applied to underactuated systems with second-order nonholonomic constraints. After briefly describing the geometric phase concept, it is applied to the icebot and PPR underactuated manipulator.

3.1 General

To explain the geometric phase concept, consider the control of the 3-dimensional first-order chained form given by

\[\begin{align*}
\dot{z}_1 &= u_1 \\
\dot{z}_2 &= u_2 \\
\dot{z}_3 &= z_2 u_1
\end{align*}\] (3.1)

The \((z_1, z_2)\) variables, which are directly actuated, are called the base variables. The unactuated variable \(z_3\) corresponds to the fiber variable that can be controlled using the geometric phase technique.

Let \((z_1^0, z_2^0, z_3^0)\) denote the initial state. The following steps transfer the system to the origin \((0, 0, 0)\).

1. Drive \((z_1^0, z_2^0, z_3^0)\) to \((0, 0, z_3^0)\) in finite time

2. Traverse a closed path (or a concatenation of a series of closed paths) which yields the so-called geometric phase given by integrating the last equation

\[\begin{align*}
\int_0^T \dot{z}_3 dt &= \int_0^T z_2 \dot{z}_1 dt \\
\Delta z_3 &= \int z_2 dx_1
\end{align*}\] (3.2)

with \(T\) the time needed to traverse one closed path. So a path in the base-variable-space (or base-space) has to be chosen such that the path-integral of this path is equal
to $\Delta z_3$. This will drive the $z_3$-variable to go to a desired value in steps of $\Delta z_3$ (see figure 3.1). The number of steps is equal to the number of traversed closed paths.

![Figure 3.1: geometric phase](image)

To follow the path in the base-space a controller is needed.

### 3.1.1 Bang-bang control

This controller [9] is used to follow the path in the base space. This control will drive the base-variables to the desired point in the shortest possible time. Consider a double integrator plant given by:

$$\dot{z} = u$$

and assume that the control has to satisfy $|u| < k$, with $k$ the control gain. Then the time-optimal control law $u$ which drives the system to $(z, \dot{z}) = (z_d, 0)$ (with $z_d$ the desired value of $z$) is given by:

$$u = \begin{cases} 
-k \cdot \text{sign}(s) & \text{if } |s| > \epsilon \\
-k \cdot \text{sign}(\dot{z}) & \text{if } |s| \leq \epsilon 
\end{cases}$$

(3.4)

where $s = (z - z_d) + \frac{1}{2k} \dot{z} \dot{z}$ the switching function and $\epsilon$ a value close to zero. This control first controls the function $s$ to zero. If $s$ is, or is close to, zero then the controller switches to $-k \cdot \text{sign}(\dot{z})$ to control the system to $(z, \dot{z}) = (z_d, 0)$ while $s$ stays around zero. It is called "Bang Bang" because it drives the system by accelerating with the maximum amount of control ($= +k \rightarrow "Bang") and then maximum decelerating ($= -k \rightarrow "Bang")$ the system to the desired point in the shortest possible time.

### 3.2 Geometric phase method applied to an icebot

Here a control law, based on the geometric phase-concept, for a planar multibody system with joint torque inputs is considered. The icebot is an interconnection of three bodies using two torque inputs at the two joint connections. The controller uses the geometric phase concept described above. By using a coordinate transformation provided by [6] the system is transformed into a form which makes it suitable for the application of the discontinuous control law. Simulation results are made to demonstrate this controller.
3.2.1 Model formulation

Consider a interconnection of three bodies using two torque inputs at the two joints, a so-called icebot as shown in figure 3.2. Let $\theta$ denote the absolute angle of the first link and $(\psi_1, \psi_2)$ denote the joint angles (relative angles). Here we consider a three link model, which consists of three bars of equal length, mass and body moment of inertia. The equations of motion are as follows (6):

\[ \begin{align*}
\dot{\theta} &= s_1(\psi_1, \psi_2)u_1 + s_2(\psi_1, \psi_2)u_2 \\
\end{align*} \]

\[ \begin{align*}
\dot{\psi}_1 &= u_1 \\
\dot{\psi}_2 &= u_2 \\
\end{align*} \]  

(3.5)

with

\[ u = -J_s^{-1}(\psi)F_s(\psi, \dot{\psi}) + J_s^{-1}(\psi)T \]

where $u = (u_1, u_2)^T, T = (T_1, T_2)^T, \dot{\psi} = (\psi_1, \psi_2)^T$ and $J_s(\psi_1, \psi_2)$ is a $2 \times 2$ matrix with entries

\[ \begin{align*}
J_{s_11}(\psi) &= 12 + 6 \cos \psi_2 - N_1^2(\psi)/D(\psi) \\
J_{s_12}(\psi) &= 4 + 3 \cos \psi_2 - N_2^2(\psi)/D(\psi) \\
J_{s_21}(\psi) &= 4 + 3 \cos \psi_2 - N_1^2(\psi)/D(\psi) \\
J_{s_22}(\psi) &= 4 - N_2^2(\psi)/D(\psi) \\
\end{align*} \]

where

\[ \begin{align*}
N_1(\psi) &= 12 + 3 \cos \psi_1 + 6 \cos \psi_2 + \cos(\psi_1 + \psi_2) \\
N_2(\psi) &= 4 + 3 \cos \psi_2 + \cos(\psi_1 + \psi_2) \\
D(\psi) &= 16 + 6 \cos \psi_1 + 6 \cos \psi_2 + 2 \cos(\psi_1 + \psi_2) \\
\end{align*} \]
And the vector function $F_s(\psi, \dot{\psi})$ is given by

$$F_s(\psi, \dot{\psi}) = \frac{d}{dt} [J_s(\psi)] \dot{\psi} - \frac{1}{2} \frac{\partial}{\partial \psi} \left( \psi^2 J_s(\psi) \dot{\psi} \right)$$  \hspace{1cm} (3.6)

And the functions $s_1(\psi), s_2(\psi)$, determined from the angular momentum expression, are given as

$$s_i(\psi) = -\frac{N_i(\psi)}{D(\psi)} : i = 1, 2$$  \hspace{1cm} (3.7)

such that

$$s_1 = \frac{12 + 3 \cos \psi_1 + 6 \cos \psi_2 + \cos(\psi_1 + \psi_2)}{16 + 6 \cos \psi_1 + 6 \cos \psi_2 + 2 \cos(\psi_1 + \psi_2)}$$

$$s_2 = \frac{4 + 3 \cos \psi_2 + \cos(\psi_1 + \psi_2)}{16 + 6 \cos \psi_1 + 6 \cos \psi_2 + 2 \cos(\psi_1 + \psi_2)}$$  \hspace{1cm} (3.8)

Note that $\psi_1$ and $\psi_2$ are the base-variables and $\theta$ is the fiber-variable.

### 3.2.2 Controller

Consider a situation where it is needed to control every initial condition $(\psi_1^0, \psi_2^0, \theta^0)$ to the origin $(0, 0, 0)$. The following steps will transfer the system from the initial state to the origin $(0, 0, 0)$:

1. Drive $(\psi_1^0, \psi_2^0, \theta^0)$ to $(0, 0, \theta^1)$ in finite time using a bang-bang controller as described in section 3.1.1.

2. Traverse a square path defined by four points $(0, 0), (z, 0), (z, z), (0, z)$ using again a bang-bang controller. The geometric phase obtained by traversing this path is given by(similar to (3.2)):

$$\Delta \theta = \oint s_1(\psi_1, \psi_2)d\psi_1 + s_2(\psi_1, \psi_2)d\psi_2$$

$$= \int_{0}^{z} s_1(\psi_1, 0)d\psi_1 + \int_{0}^{z} s_2(z, \psi_2)d\psi_2 + \int_{z}^{0} s_1(\psi_1, z)d\psi_1 + \int_{z}^{0} s_2(0, \psi_2)d\psi_2$$

$$= \alpha(z)$$  \hspace{1cm} (3.9)

Here the solution for the path-integral ($= \alpha(z)$) is known but very complex, so it is not expanded here. Because of the complexity of the function $\alpha(z)$, it is not possible to come up with a function $\beta$ such that $z = \beta(\Delta \theta)$. If this function $\beta$ was known, one could easily determine $z$, which is the length of a side of the square path defined above, as a function of the desired $\Delta \theta$. This function $\alpha(z)$ is well-defined for $-\pi < z < \pi$ and is shown in figure 3.3.
As can be seen in figure 3.3 \( \alpha(z) \) has a maximum and a minimum of \( \alpha(-\pi) \approx 1.46 \) rad and \( \alpha(\pi) \approx -1.46 \) rad, respectively. This means that the change in \( \theta \) after each traversed square path in the base-space has a maximum of about 1.46 rad. If \( \theta^1 \) (after the first step) is greater than this maximum, then another closed path (or a concatenation of a series of closed paths) has to be traversed to bring the fiber-variable \( \theta \) to zero.

Now the maximum length of a side of the square path is \( \pi \). In practice it may be needed to restrict the maximum angle between two links. This means that the maximum for \( \psi_1 \) and \( \psi_2 \) is restricted, which also means a restriction of the length of a side of the square path. Let \( \psi_{\text{max}} \) denote the maximum allowable joint angle and \( \Delta\theta_{\text{max}} \) denotes the corresponding geometric phase. If \( \theta^1 \) is smaller than \( \Delta\theta_{\text{max}} \), a single square path can be determined such that \( \Delta\theta = -\theta^1 \). Otherwise, more than one square paths are needed to produce the desired geometric phase.

### 3.2.3 Simulation

Matlab simulations are done using the control-strategy given in the previous section, to evaluate the motion of the icebot. A bang-bang controller with control-gain \( k = 5 \) is used. For the maximum joint angle \( \psi_{\text{max}} = \pi \) is used. The initial state is \((\psi_1, \psi_2, \theta) = (1, 1, \frac{\pi}{3})\).

Here we also consider the center of mass of the icebot to be constant at one point. That point is chosen to be in the origin, so \((CG_x, CG_y) = (0, 0)\). This means for the \(x\)- and \(y\)-coordinates of the center of mass of the three bodies, with equal mass, that:

\[
\begin{align*}
x_1 + x_2 + x_3 &= 0 \\
y_1 + y_2 + y_3 &= 0
\end{align*}
\]

where \(x_i\) and \(y_i\) is the position of the \(i^{th}\)-body in \(x\)- and \(y\)-coordinates and note that the three bodies have the same mass. This results in the following equation for the \(x\)- and \(y\)-coordinate for point A (as in figure 3.2):

\[
\begin{align*}
A_x &= -\frac{5l}{6} \cos \theta - \frac{l}{2} \cos(\theta + \psi_1) - \frac{l}{6} \cos(\theta + \psi_1 + \psi_2) \\
A_y &= -\frac{5l}{6} \sin \theta - \frac{l}{2} \sin(\theta + \psi_1) - \frac{l}{6} \sin(\theta + \psi_1 + \psi_2)
\end{align*}
\]

\[\text{(3.10)}\]
If the position of point A is known then the position of the joints are known. The results are shown here:

![Graphs showing \(\psi_1, \psi_2\) and \(\theta\)](image)

Figure 3.4: \(\psi_1, \psi_2\) and \(\theta\)

As can be seen in figures 3.4 and 3.5 the initial state is driven to the origin. In figure 3.4 one can see that first the 2 base-variables \((\psi_1, \psi_2)\) are driven to the origin in 0.95 sec, which makes \(\theta = 2.57\) [rad]. In figure 3.6 one can see that in these first 0.95 sec both \(u_1\) and \(u_2\) first decelerate the base-variables with the maximum amount of control \((u_1 = u_2 = -5)\) and then switches after 0.45 sec, such that the base-variables are accelerated with the maximum amount of control \((u_1 = u_2 = +5)\) to the origin. This is typical the discontinuous behavior of the BangBang-controller described in section 3.1.1.

After that, in figure 3.5 one can see that 2 square paths are driven. First a square path with length of a side of \(\pi\), which is the maximum joint angle, is traversed. This results in a maximum change of \(\theta\) of \(-1.46\) [rad], such that \(\theta = 1.11\) [rad]. Then, after about 7.76 sec, a second path is traversed with length of a side of 2.79. This results in a change in \(\theta\) of 1.11 [rad] such that \(\theta\) becomes the desired value 0.

More results are shown in appendix A.3. In figure A.4 it can be seen how, using the geometric phase concept, the initial state of the icebot is driven to the origin. The state of the icebot is plotted in time. The points in time which are plotted are the states of the icebot after every traversed side of the square paths in the base-space, except for the first point \((t = 0.95\) sec\) where it is the time needed to bring the base-variables to the origin.

If the maximum joint angle \(\psi_{max}\) is decreased, then the maximum length of a square path is decreased. This causes \(\Delta\theta_{max}\) to decrease, as seen in figure 3.3. This implies that the number of square paths needed to produce the desired geometric phase increases. If the control gain \(k\) is increased, then there is less time needed to traverse the desired paths in the base-space.
Figure 3.5: Geometric phase for fiber-variable $\theta$

Figure 3.6: $u_1$ and $u_2$
3.3 Geometric phase method applied to a PPR underactuated manipulator

Here a control law, based on the geometric phase-concept, for a three degree of freedom planar PPR underactuated manipulator moving in a horizontal plane (as in section 2.2) is considered. To apply the geometric phase concept, the equations of motion in the chained form (2.23) are transformed into the following form

\[
\begin{align*}
\dot{\xi}_1 &= v_1 \\
\dot{v}_1 &= u_1 \\
\dot{\xi}_2 &= v_2 \\
\dot{v}_2 &= u_2 \\
\dot{\xi}_3 &= w + \xi_2 \dot{\xi}_1 \\
\dot{w} &= -\xi_1 \dot{\xi}_2
\end{align*}
\tag{3.12}
\]

using \(w = \xi_3 - \xi_2 \dot{\xi}_1\). The base-variables are \((\xi_1, \xi_2)\) and the fiber-variable is \(\xi_3\).

3.3.1 Control

Consider the situation where it is wanted to control every initial rest situation \((x_0, y_0, \theta_0)\) to a desired rest situation \((x_d, y_d, \theta_d)\) of the PPR manipulator. It's a rest-to-rest manoeuvre, that means that there's no initial motion of the joints. This means that initially \((\bar{x}, \bar{y}, \bar{\theta})\) are zero and thus \((\dot{\xi}_1, \dot{\xi}_2, \dot{\xi}_3)\) are initially zero. With initially \(\dot{\xi}_2\) and \(\dot{\xi}_3\) zero, it can be seen in equations (3.12) that this will cause \(w\) and \(\dot{w}\) to be initially zero also. The desired rest situation \((x_d, y_d, \theta_d)\) is in the transformed coordinates \((\xi_1^d, \xi_2^d, \xi_3^d) = (0, 0, 0)\), the origin.

The following steps will transfer system (3.12) from the initial rest situation to the origin of the transformed coordinates:

1. Drive \((\xi_2^0, v_1^0, \xi_2^0, v_2^0, \xi_3^0, w^0) = (\xi_2^0, 0, \xi_2^0, 0, \xi_3^0, 0)\) to \((0, 0, 0, 0, 0, 0, 0)\) in finite time using a bang-bang controller, as described in section 3.3.1, for the first base-variable \(\xi_1\) only. And no control-input for the other base-variable \(\xi_2\), which means \(u_2 = 0\) for this step. Then \(\dot{\xi}_2\) stays zero and \(\xi_2\) doesn't change in this step. And if \(\dot{\xi}_2\) stays zero, then \(\dot{w} = -\xi_1 \dot{\xi}_2 = 0\), so \(w\) stays zero for this step.

2. Drive \((0, 0, \xi_2^0, 0, \xi_3^0, 0)\) to \((0, 0, 0, 0, \xi_3^0, 0)\) in finite time using a bang-bang controller for the second base-variable \(\xi_2\) only. And zero input for the other base-variable \(\xi_1\), which means \(u_1 = 0\) for this step. This also means that \(\xi_3\) stays the same \((= \xi_3^0)\), because \(w\) again stays zero and \(\xi_1\) stays zero.

3. Traverse a rectangle path defined by four points \((0, 0), (a, 0), (a, b), (0, b)\) using a bang-bang controller. The geometric phase obtained by traversing this path is given by:

\[
\int_{t_1}^{T} \xi_3 dt = \int_{t_1}^{T} w dt + \int_{t_1}^{T} \xi_2 \dot{\xi}_1 dt
\]

\[
-\xi_3^1 = w^1 (T - t_1) + \int_{a}^{b} d\xi_1 = -ab
\]

where \(w^1 = w(t_1) = 0\), which is the variable \(w\) after the first 2 steps. It is very important to have \(w\) initially zero and to keep \(w\) zero, by using one input \((u_1\) or \(u_2)\) at the time. Otherwise, if \(w\) is nonzero, there would be no solution for equation (3.13).
The path-integral $J_2 \int d\xi_1$ is only nonzero for the side of the rectangle path connected by the two points $(a, b)$ and $(0, b)$. The parameters $a$ and $b$ for the rectangle path can be chosen arbitrarily, so long as the constraint $\xi_1^0 = ab$ is satisfied.

### 3.3.2 Simulation

Matlab simulations are done using the control-strategy given in the previous section, to evaluate the motion of the PPR. A bang-bang controller with control-gain $k = 1$ is used. The same PPR-parameters are used as in the simulation with the PPR using $\sigma$-process, section 2.2.3. In this simulation the geometric phase parameters $(a, b)$ are chosen to be

$$a = 1 \quad b = \frac{\xi_3^1}{a}$$

(3.14)

The initial rest-situation is $(x, y, \theta) = (1, 1, \frac{\pi}{4})$ in the original coordinates, which means for the transformed coordinates a initial rest situation of $(\xi_1, \xi_2, \xi_3) = (-2.77, 0.27, -0.47)$. The desired rest-situation is $(2, 2, \frac{\pi}{4})$, which means that the transformed coordinates are driven to the origin $(0, 0, 0)$.

The results are shown in figure 3.7.

![Figure 3.7: x, y and \(\theta\)](image)

As can be seen in figure 3.7 and 3.8 the initial rest state is driven to the desired rest state. The same as for the simulation with $\sigma$-process as control in section 2.2.3 holds: Problems occur when the $\theta - \theta_d \geq \frac{\pi}{2}$, because of the singularities of the local transformation in $[2]$. The chained equations 2.23 are valid only in the domain $S$.

In figure 3.8 (and A.6 in appendix A.4) one can see that the 2 base-variables $(\xi_1, \xi_2)$ are driven to the origin in 4.52 sec, such that the fiber-variable $\xi_3 = 0.27$. This means in the original coordinates that $x$ and $y$ are driven from the origin to 1.87 and 2.23, respectively, and $\theta$ from $\frac{\pi}{4}$ to $\theta = \frac{\pi}{2}[rad]$ (see figure 3.7).

After that, in figure 3.8 one can see that a rectangle path is followed in the base-space. With $a = 1$ and $\xi_3^1 = 0.27$, the parameter $b$ becomes 0.27. The rectangle path is given by the
Figure 3.8: geometric phase for fiber-variable $\theta$

Figure 3.9: $F_1$ and $F_2$
points (0, 0), (1, 0), (1, 0, 27), (0, 0, 27) in the base-space. This results in a change in the fiber-variable $\xi_3$ such that $\xi_3 = 0$ after 11.95 sec. This means in the original coordinates that $(x, y)$ are driven to $(2, 2)$ and $\theta$ to $\theta = \frac{\pi}{8}$ [rad] (see figure 3.7). This motion is also illustrated in figure A.5, in appendix A.4. Where again the prismatic joints are not shown and the PPR is plotted smaller then its original size used in the simulation.

In figure 3.9, for the original inputs $F_1$ and $F_2$, and in figure A.7, for the transformed inputs $u_1$ and $u_2$, you can see the discontinuous behavior of the bang-bang controller. In figure A.7 one see that only one control ($u_1$ or $u_2$) at the time is active, which is needed to keep $w$ and $\dot{w}$ zero.

Comparing the geometric phase for the PPR with the $\sigma$-process, one can say that for simulations with the same parameters for the PPR with these initial states the geometric phase works better. In these simulations the system is controlled faster to its desired state and needs less control-effort (one can see it if figure 2.3 and 3.9 are compared, where one can see that with the $\sigma$-process much higher values of the forces $F_1$ and $F_2$ are needed then with the geometric phase). Also in these simulations with the geometric phase less motion is needed to control the orientation then with the $\sigma$-process. But for the $\sigma$-process it holds that the movement can be influenced by the choice of the control gains. So by fine-tuning the control gains, the performance of the $\sigma$-process can be improved. So at this point nothing can be said comparing these 2 control-techniques for point-to-point control.
Chapter 4

Tracking control: cascaded backstepping

In this chapter tracking control of underactuated systems is considered. There exists many control-techniques for the tracking problem of underactuated systems. The control-techniques, in general, use a local transformation to transform the underactuated mechanical system into the second-order chained form (i.e. equation (1.5)). Then, in general, a globally stable control-technique is applied to this chained form. Due to the local transformation this controlled system is only capable of tracking simple paths, such as a line or a circle (i.e. $[10]$). With these control-techniques in combination with a local transformation, problems arise when the orientation (or the change in the orientation) has to be controlled. An orientation will introduce goniometric functions, i.e. a tangent-term. And these terms are causing singularities, i.e. in a coordinate transformation or in the equations of motion of the system, since the transformation is local. Other control-techniques, which have no singularities since the transformation is global or which do not use a transformation at all, are developed, but these techniques yield very complex controllers (i.e. $[11]$, where a controller is used which exists of over 2500 terms).

In this report, we consider a modification of the tracking controller designed in $[7]$. This controller is based on a cascaded backstepping technique, where the tracking error dynamics are treated as two separate subsystems. A switching tracking controller is developed to avoid the singularities associated with the desired trajectories caused by the local coordinate transformation. After a short introduction of the ideas, the technique is applied to a PPR underactuated manipulator and a surface vessel.

4.1 General

Consider the second-order chain-form (1.5). Suppose it is needed to follow a predefined path, i.e. we want the state $x = [x_1, \ldots, x_6]^T$ to follow a prescribed path $x_d = [x_{d1}, \ldots, x_{d6}]^T$. This reference trajectory $x_d$ thus satisfies

$$
\begin{align*}
\dot{x}_{1d} &= x_{2d} \\
\dot{x}_{2d} &= u_{1d} \\
\dot{x}_{3d} &= x_{4d} \\
\dot{x}_{4d} &= u_{2d} \\
\dot{x}_{5d} &= x_{6d} \\
\dot{x}_{6d} &= u_{1d}x_{3d}
\end{align*}
$$

(4.1)
using $x_{ie} = x_i - x_{id}$, $i = 1, 2, ..., 6$, the tracking error dynamics become:

\[
\begin{align*}
\dot{x}_{1e} &= x_{2e} & \dot{x}_{2e} &= u_1 - u_{1d} \\
\dot{x}_{3e} &= x_{4e} & \dot{x}_{4e} &= u_2 - u_{2d} \\
\dot{x}_{5e} &= x_{6e} & \dot{x}_{6e} &= x_{3e}u_{1d} + x_3(u_1 - u_{1d}) \\
\end{align*}
\]

(4.2)

In [7], a linear time-varying controller is developed that globally asymptotically stabilizes the second-order chained form system to a reference trajectory. This reference trajectory can not be chosen arbitrarily, but has to satisfy a so-called 'persistence of excitation' condition. This persistence of excitation condition implies that the reference trajectory is not allowed to converge to a point. A short explanation of the controller is given in the next section.

4.1.1 Cascaded backstepping control

In this section a cascade design, designed in [7], is given to stabilize the equilibrium $x = 0$ of the error-dynamics (4.2). The tracking dynamics can be written in a more convenient form:

\[
\begin{align*}
\Delta_1 &= \begin{cases} 
\dot{x}_{31} = x_{32} \\
\dot{x}_{32} = x_{21}u_{1d} + (x_{21} + \xi_{2d})(u_1 - u_{1d}) 
\end{cases} \\
\Delta_2 &= \begin{cases} 
\dot{x}_{21} = x_{22} \\
\dot{x}_{22} = u_2 - u_{2d} 
\end{cases} \\
\Delta_3 &= \begin{cases} 
\dot{x}_{11} = x_{12} \\
\dot{x}_{11} = u_1 - u_{1d} 
\end{cases} 
\end{align*}
\]

(4.3)

where $\xi_{2d}$ denotes the reference trajectory of the state $\xi_2$ in (1.5). Suppose that the $\Delta_3$ subsystem has been stabilized to the origin $(x_{11}, x_{12}) = (0, 0)$ by choosing the following controller

\[
u_1 = u_{1d} - k_1x_{11} - k_2x_{12}, \quad k_1 > 0, k_2 > 0
\]

(4.4)

where the polynomial $p(\lambda) = \lambda^2 + k_1\lambda + k_2$ is Hurwitz. Then since $x_{12} \equiv 0$ it also holds that $u_1 - u_{1d} \equiv 0$ such that the time-varying subsystem $\Delta_1$ can be written as

\[
\begin{align*}
\dot{x}_{31} &= x_{32} \\
\dot{x}_{32} &= x_{21}u_{1d} 
\end{align*}
\]

(4.5)

A stabilizing feedback for subsystem (4.5) is designed using backstepping in which $x_{21}$ is a virtual input.

Consider the first equation ($\dot{x}_{31} = x_{32}$) of the subsystem (4.5) and assume that $x_{32}$ is the virtual input. A stabilizing function $x_{32} = \alpha_1(x_{31})$ for the first equation is

\[
\alpha_1 = -c_1u_{1d}^2x_{31}
\]

(4.6)

where $c_1 > 0$. The choice to take the square of $u_{1d}$ is done to guarantee that the $(\Delta_1, \Delta_2)$ subsystem can be stabilized by a backstepping procedure in which no divisions by $u_{1d}$ occur. Define $\dot{x}_{32} = x_{32} - \alpha_1 = x_{32} + c_1u_{1d}^2x_{31}$ and consider the second equation of subsystem (4.5), then $\dot{x}_{32}$ becomes

\[
\dot{x}_{32} = x_{21}u_{1d} + c_1u_{1d}^2x_{32} + 2c_1u_{1d}\dot{u}_{1d}x_{31}
\]

(4.7)

Suppose that $x_{21}$ is the virtual input. A stabilizing function $x_{21} = \alpha_2(\dot{u}_{1d}, x_{31}, x_{32})$, with $\dot{u}_{1d} = (u_{1d}, \dot{u}_{1d}, ..., u_{1d}^{(3)})$, for the $\dot{x}_{32}$-subsystem is then given by

\[
\alpha_2 = -c_1u_{1d}x_{32} - 2c_1\dot{u}_{1d}x_{31} - c_2u_{1d}\dot{x}_{32} \\
= -(c_1c_2u_{1d}^3 + 2c_1u_{1d})x_{31} - (c_1u_{1d} + c_2u_{1d})x_{32}
\]

(4.8)
where $c_2 > 0$ and the relation $\dot{x}_{32} = x_{32} + c_1 u_{id}^2 x_{31}$ has been substituted.

Define $\ddot{x}_{21} = x_{21} - \alpha_2(\bar{u}_{id}, x_{31}, x_{32})$ and consider the first equation of $\Delta_2$-subsystem (see equations (4.3)), then $\ddot{x}_{21}$ becomes

$$\ddot{x}_{21} = x_{22} - \frac{d}{dt}[\alpha_2(\bar{u}_{id}, x_{31}, x_{32})]$$  \hspace{1cm} (4.9)

where $x_{22}$ denotes a virtual input. For the clarity of the derivations, the time-derivative of $\alpha_2$ is not expanded and written as $\frac{d}{dt}[\alpha_2(\bar{u}_{id}, x_{31}, x_{32})]$. A stabilizing function $x_{22} = \alpha_3(\bar{u}_{id}, \bar{x}_{21}, x_{31}, x_{32})$ for the $x_{21}$-subsystem is given by

$$\alpha_3 = -c_3 \bar{x}_{21} + \frac{d}{dt}[\alpha_2(\bar{u}_{id}, x_{31}, x_{32})]$$  \hspace{1cm} (4.10)

where $c_3 > 0$. Define $\bar{x}_{22} = x_{22} - \alpha_3(\bar{u}_{id}, \bar{x}_{21}, x_{31}, x_{32})$ and consider the second equation of $\Delta_2$-subsystem (see equations (4.3)), then $\ddot{x}_{22}$ becomes

$$\ddot{x}_{22} = (u_2 - u_{2d}) - \frac{d}{dt}[\alpha_3(\bar{u}_{id}, \bar{x}_{21}, x_{31}, x_{32})]$$  \hspace{1cm} (4.11)

This $\ddot{x}_{22}$-subsystem can be stabilized by choosing the input $u_2$ as

$$u_2 = u_{2d} - c_4 \bar{x}_{22} + \frac{d}{dt}[\alpha_3(\bar{u}_{id}, \bar{x}_{21}, x_{31}, x_{32})]$$

$$= -c_3 c_4 \bar{x}_{21} - (c_3 + c_4)x_{22} + c_3 c_4 \alpha_2(\bar{u}_{id}, x_{31}, x_{32})$$

$$+ (c_3 + c_4) \frac{d}{dt}[\alpha_2(\bar{u}_{id}, x_{31}, x_{32})] + \frac{d^2}{dt^2}[\alpha_2(\bar{u}_{id}, x_{31}, x_{32})]$$  \hspace{1cm} (4.12)

It is proven in [7] that with controls (4.4) and (4.12) the complete system is asymptotically stable, because of the exponential stability of the two separate subsystems ($\Delta_1, \Delta_2$) and ($\Delta_3$).

### 4.2 Cascaded backstepping applied on a PPR underactuated manipulator

Here the tracking problem for a PPR underactuated manipulator is considered. The manipulator used is described in section 2.2. The controller used is described in the previous section. Using this controller the manipulator can follow a desired path, the simulation results are made in 4.2.2 to demonstrate the controller.

#### 4.2.1 Control

Consider the 2nd-order chained form for the underactuated manipulator, see equations (2.23). The error-dynamics are the same as expressed in equations (4.2). Using cascaded integrator backstepping (as in section 4.1) the first part ($\dot{x}_{11}, \dot{x}_{12}$) of the error-dynamics can be controlled by choosing $u_1$ to be

$$u_1 = u_{id} - k_1 x_{1e} - k_2 x_{2e}$$  \hspace{1cm} (4.13)

such that the polynomial $p(\lambda) = \lambda^2 + k_1 \lambda + k_2$ is Hurwitz, with $k_1$ and $k_2$ the control gains. Now suppose that $u_1 - u_{id} = 0$ after a certain time, then we only have to consider
\[ \dot{x}_{6e} = x_{3e}u_{1d} \] for the last equations of the error-dynamics. Then we can write the second part (\( \dot{x}_{3e}, \dot{x}_{4e}, \dot{x}_{5e}, \dot{x}_{6e} \)) as:

\[
\begin{align*}
\dot{x}_{6e} &= x_{3e}u_{1d} \\
\dot{x}_{3e} &= x_{4e} \\
\dot{x}_{4e} &= u_2 - u_{2d}
\end{align*}
\]  

(4.14)

then it is possible to apply integrator backstepping to determine \( u_2 \) as in [7]:

\[ u_2 = u_{2d} - G_3(t)x_{3e} - G_4(t)x_{4e} - G_5(t)x_{5e} - G_6(t)x_{6e} \]  

(4.15)

with

\[
\begin{align*}
G_3(t) &= c_3c_4u_{1d}^2 + (c_1 + c_2)(c_3 + c_4)u_{1d}^2 + (c_3 + c_4)\ddot{u}_{1d}u_{1d} + c_1c_2 \\
G_4(t) &= (c_3 + c_4)u_{1d}^2 + (c_1 + c_2) \\
G_5(t) &= c_1c_2c_3c_4u_{1d}^2 + 2c_3\dddot{u}_{1d} + (3c_3c_4u_{1d}^2 + 2c_3(c_1 + c_2))\ddot{u}_{1d} \\
&+ (3c_3c_4(c_2 + c_1)u_{1d}^2 + 2c_3c_1c_2)\dot{u}_{1d} + 6c_3c_4u_{1d}\dddot{u}_{1d} \\
G_6(t) &= c_3c_4(c_2 + c_1)u_{1d}^2 + (c_3 + c_4)c_1c_2u_{1d} + (5c_3 + c_4)\dddot{u}_{1d} \\
&+ (6c_3c_4u_{1d}^2 + (c_1 + c_2)(3c_3 + c_4))\dot{u}_{1d}
\end{align*}
\]

with \( u_{1d}, \dddot{u}_{1d}, \ddot{u}_{1d}, \dot{u}_{1d} \) dependent on the desired trajectory and \( c_1, ..., c_4 \) the control gains.

4.2.2 Simulation

A simulation is made using controls (4.13) and (4.12). Suppose that the PPR has to follow the desired path:

\[
\begin{align*}
x_d &= r \cos(\omega t) \\
\theta_d &= 0 \\
y_d &= 0
\end{align*}
\]  

(4.16)

with constants \( r = 1 \) and \( \omega = 1 \). This desired path corresponds to the PPR moving back and forward between the points \((-1, 0)\) and \((1, 0)\) and keeping the orientation \( \theta \) of the third link zero. Where the desired trajectory begins (at time \( t = 0 \)) at point \((-1, 0)\). The desired variables \((x_d, \theta_d, y_d)\) satisfy equations (2.22) such that the tracking of the desired variables is realizable. The system parameters used are the same as in section 2.2.3. The control gains used are:

\[ k_1 = 4, \ k_2 = 2\sqrt{2}, \ c_1 = 3, \ c_2 = 3, \ c_3 = 0.5, \ c_4 = 5 \]  

(4.17)

The initial position is \((x, y, \theta) = (0, 1, 0)\) in the original coordinates. The results are shown in figure 4.1.

35
Figure 4.1: $x, y$ and $\theta$

As can be seen in figure 4.1 the trajectory is successfully tracked after approximately 15 sec. The same as for the simulation in section 2.2.3 holds: Problems occur when the $\theta - \theta_d \geq \frac{\pi}{2}$, because of the singularities of the local transformation in [2]. The chained equations (2.23) are valid only in the domain $S$.

One can see that first the error in the $x$-direction is brought to zero and then the error of the remaining variables $y$ and $\theta$ are brought to zero. This can also be seen in section 4.2.1 where the first subsystem of the error-dynamics is controlled, such that a term can be neglected in the second subsystem.

The inputs are shown in figure 4.2. As can be seen the force $F_1$ becomes a sinusoidal-function to move the PPR between the two points $(-1,0)$ and $(1,0)$ and the force $F_2$ is brought to zero. This is due to the fact that if the error in the $y$ and $\theta$ is zero no control-effort of $F_2$ is needed.

In appendix A.5 the motion plots of the simulation are given in figure A.8 till A.13. Here again, the 2 prismatic joints are not shown and the PPR is plotted smaller than its original size during the simulation. Also the desired path is plotted as triangles, where the "direction" of the triangles represents the direction in which the desired trajectory moves in the $x$-direction ($>$ means the desired point moves to the right and $<$ means that the desired point moves to the left).

4.3 Cascaded backstepping applied on a surface vessel

Consider a surface vessel which has 2 actuators or thrusts in the forward direction. The direction sideways is hereby unactuated (See figure 4.3). Examples of this are submarines, jetski's or unmanned deep-sea-vessels.

The equations of motion are:

\[
F_x = m(\dot{x}\cos \theta + \dot{y}\sin \theta) + c_x(\dot{x}\cos \theta + \dot{y}\sin \theta)
\]

\[
T_x = I\ddot{\theta} + c_\theta \dot{\theta}
\]

\[
0 = m(-\dot{x}\sin \theta + \dot{y}\cos \theta) + c_y(-\dot{x}\sin \theta + \dot{y}\cos \theta)
\]

Where the zero in the third equation is related to the unactuated direction of the surface vessel. The actuator-forces or thrusts $F_1$ and $F_2$ are related to $F_x$ and $T_x$ in the following way:

\[
F_x = F_1 + F_2, \quad T_x = (F_1 - F_2) \frac{d}{2}
\]
where $d$ is the distance between the two actuators. In [12] a global solution of the tracking control problem for a surface vessel is presented.

### 4.3.1 Transformation $Z$

In this section a local transformation is used. The surface vessel is able to track a desired trajectory so long as the transformation is well-defined. The following transformation is used (from [8]):

$$
\begin{align*}
    z_1 &= x \\
    z_2 &= \dot{x} \\
    z_3 &= \tan \theta \\
    z_4 &= \frac{1}{\cos^2 \theta} \dot{\theta} \\
    z_5 &= y \\
    z_6 &= \dot{y}
\end{align*}
$$

which is well-defined so long as $\cos \theta \neq 0$. It transforms (4.18) to

$$
\begin{align*}
    \dot{z}_1 &= z_2 \\
    \dot{z}_2 &= u_1 \\
    \dot{z}_3 &= z_4 \\
    \dot{z}_4 &= u_2 \\
    \dot{z}_5 &= z_6 \\
    \dot{z}_6 &= u_1 z_2 + \frac{c_y}{m} (\dot{z}_1 z_2 - \dot{z}_3)
\end{align*}
$$

with

$$
\begin{align*}
    u_1 &= \frac{F_z}{m} \cos \theta - \frac{f_1}{m} \cos \theta + \frac{f_2}{m} \sin \theta \\
    u_2 &= \left(\frac{2 \sin \theta}{\cos^3 \theta}\right) \dot{\theta}^2 + \frac{1}{\cos^2 \theta} \left(\frac{T_z}{I} - c_2 \dot{\theta}\right)
\end{align*}
$$
where
\[ f_1 = c_x (\dot{x} \cos \theta + \dot{y} \sin \theta) \]
\[ f_2 = c_y (\dot{y} \sin \theta + \dot{y} \cos \theta) \] (4.22)

With this transformation the accelerations \((\ddot{x}, \ddot{y}, \ddot{\theta})\) become:
\[ \ddot{x} = u_1 \]
\[ \ddot{y} = u_1 \tan \theta + c_y / m (\dot{x} \tan \theta - \dot{y}) \] (4.23)
\[ \ddot{\theta} = (u_2 - 2 \sin \theta / \cos^2 \theta \dot{\theta}^2 \cos \theta) \cos \theta \]

Assuming that \(\theta = (-\pi, \pi)\), one can see that the transformation has singularities if \(\theta\) becomes, or is close to, \(\pm 3 \pi / 2\) because of the tan-term in \(\ddot{y}\) and the \(1 / \cos^2 \theta\)-term in \(\ddot{\theta}\) in equation (4.23), which means that this transformation is a local transformation.

The desired variables, for a certain desired trajectory in the \(z\)-variables, are
\[
\begin{align*}
\dot{z}_{1d} &= z_{2d} \\
\dot{z}_{2d} &= z_{4d} \\
\dot{z}_{3d} &= z_{4d} \\
\dot{z}_{4d} &= u_{1d} + c_y (\dot{x}_d x_{2d} - \dot{y}_d) \\
\end{align*}
\] (4.24)

The desired variables in the original variables \((x, y, \theta)\) have to satisfy equations (4.24). The last equation of (4.24) can be written as:
\[ \ddot{y}_d = \ddot{x}_d \tan \theta_d + c_y / m (\dot{x}_d \tan \theta_d - \dot{y}_d) \] (4.25)

this means that the desired variables \((x_d, y_d, \theta_d)\) can not be chosen arbitrary, but have to satisfy the above equation. Consequently, \(z_{3d} = \tan \theta_d\), for a certain given path in \(x = \tan \theta_d\) and \(y = \tan \theta_d\), becomes:
\[ z_{3d} = \frac{\ddot{y}_d + c_y / m \dot{y}_d}{\ddot{x}_d + c_y / m \dot{x}_d} \] (4.26)
\[ \frac{\dot{y}}{\dot{x}} \]

and (with \(z_{4d} = \dot{z}_{3d}\))
\[ z_{4d} = \frac{\dot{y} \dot{x} - \dot{x} \dot{y} \dot{x}^2}{\dot{x}^2} \] (4.27)

The desired inputs become:
\[ u_{1d} = \ddot{x}_d \]
\[ u_{2d} = \frac{(\ddot{y} \dot{x} - \dot{x} \dot{y}) \dot{x}^2 - 2 \dot{x} (\ddot{x} - \dot{x} \dot{y})}{\dot{x}^4} \] (4.28)

where the derivatives of \(\dot{x}\) and \(\dot{y}\) (from (4.26)) are:
\[ \dot{x} = \ddot{x}_d + c_y \ddot{x}_d \]
\[ \dot{y} = \ddot{y}_d + c_y \ddot{y}_d \] (4.29)
Now, all the desired variables \((z_{1d}, \ldots, z_{6d})\) are known if a desired trajectory is generated in the \(x, y\)-space.

This results in the error-dynamics (using \(z_e = z - z_d\)):

\[
\begin{align*}
\dot{z}_{1e} &= z_{2e} \\
\dot{z}_{2e} &= u_1 - u_{1d} \\
\dot{z}_{3e} &= z_{4e} \\
\dot{z}_{4e} &= u_2 - u_{2d} \\
\dot{z}_{6e} &= z_{6e} = z_{3e} \bar{u}_{1d} - \frac{c_y}{m} z_{6e} + z_2 (\frac{c_y}{m} \dot{z}_{2e} + u_1 - u_{1d}) 
\end{align*}
\]

with \(\bar{u}_{1d} = u_{1d} + \frac{c_y}{m} z_2\)

**Controller**

Using cascaded integrator backstepping (as in section 4.1) the first part \((\dot{z}_{1e}, \dot{z}_{2e})\) of the system (4.30) can be controlled by choosing \(u_1\) to be:

\[
u_1 = u_{1d} - k_1 z_{1e} - k_2 z_{2e}
\]

such that the polynomial \(p(\lambda) = \lambda^2 + k_1 \lambda_1 + k_2\) is Hurwitz, with \(k_1\) and \(k_2\) the control gains. Now suppose that \(u_1 = u_{1d} = 0\) after a certain time and that the \(-\frac{c_y}{m} z_{6e}\)-term will help us so that we only have to consider \(\dot{z}_{6e} = z_{3e} \bar{u}_{1d}\) for the last equation of (4.30). Then we can write the second part \((\dot{z}_{3e}, \dot{z}_{4e}, \dot{z}_{6e})\) of system (4.30) as:

\[
\begin{align*}
\dot{z}_{5e} &= z_{6e} \\
\dot{z}_{6e} &= z_{3e} \bar{u}_{1d} \\
\dot{z}_{6e} &= z_{4e} \\
\dot{z}_{4e} &= u_2 - u_{2d}
\end{align*}
\]

then it is possible (assuming that \(\theta \neq \pm \pi\)) to apply integrator backstepping to determine \(u_2\) as in [7]:

\[
u_2 = u_{2d} - G_3(t) z_{3e} - G_4(t) z_{4e} - G_5(t) z_{5e} - G_6(t) z_{6e}
\]

with

\[
\begin{align*}
G_3(t) &= c_3 c_4 \bar{u}_{1d}^2 + (c_1 + c_2)(c_3 + c_4) \bar{u}_{1d} + (5c_3 + 3c_4) \bar{u}_{1d} + c_1 c_2 \\
G_4(t) &= (c_3 + c_4) \bar{u}_{1d}^2 + (c_1 + c_2) \\
G_5(t) &= c_1 c_2 c_3 c_4 \bar{u}_{1d}^2 + 2c_3 \bar{u}_{1d} + (3c_3 c_4 \bar{u}_{1d}^2 + 2c_3 (c_1 + c_2)) \bar{u}_{1d} \\
&\quad + (6c_3 c_4 (c_2 + c_1) \bar{u}_{1d}^2 + 2c_3 c_1 c_2) \bar{u}_{1d} + 6c_3 c_4 \bar{u}_{1d} \bar{u}_{1d} \\
G_6(t) &= c_3 c_4 (c_2 + c_1) \bar{u}_{1d}^2 + (c_3 + c_4) c_1 c_2 \bar{u}_{1d} + (5c_3 + c_4) \bar{u}_{1d} \\
&\quad + (6c_3 c_4 \bar{u}_{1d}^2 + (c_1 + c_2) (3c_3 + c_4)) \bar{u}_{1d}
\end{align*}
\]

with \(\bar{u}_{1d}, \bar{u}_{1d}, \bar{u}_{1d}, \bar{u}_{1d}\) dependent on the desired trajectory and \(c_1, \ldots, c_4\) the control gains. The derivatives of \(\bar{u}_{1d}\), needed for control-law (4.33), are:

\[
\begin{align*}
\bar{u}_{1d} &= u_{1d} + \frac{c_y}{m} \dot{x}_d \\
\dot{\bar{u}}_{1d} &= \dot{u}_{1d} + \frac{c_y}{m} \ddot{x}_d \\
\ddot{\bar{u}}_{1d} &= \ddot{u}_{1d} + \frac{c_y}{m} \dddot{x}_d \\
\dddot{\bar{u}}_{1d} &= \dddot{u}_{1d} + \frac{c_y}{m} \ddddot{x}_d
\end{align*}
\]
with \( u_{id}^{(k)} \) the \( k \)-th derivative of \( u_{1d} \).

The singularities occur if \( \theta \) becomes, or is close to, \( \pm \frac{\pi}{2} \). Therefore system (4.18), with this controller will become unstable if \( \theta \) becomes, or is close to, \( \pm \frac{\pi}{2} \). A simulation is made to test transformation \( Z \), without having any singularities. The simulation is shown in appendix A.6.1. From this simulation can be seen that if the desired trajectory is chosen such that \( -\frac{\pi}{2} < \theta_d < \frac{\pi}{2} \) and the initial orientation is chosen, such that \( -\frac{\pi}{2} < \theta_0 < \frac{\pi}{2} \), then the orientation stays \( -\frac{\pi}{2} < \theta < \frac{\pi}{2} \). This means that during this simulation of the controlled system no singularity occurred for tracking a sin in the \( x \)-direction. Obviously this could occur if a circle is tracked, where the orientation \( \theta \) becomes, or is close to, \( \pm \frac{\pi}{2} \). A solution for this is to use another transformation in the case that \( \theta \) is close to \( \pm \frac{\pi}{2} \).

### 4.3.2 Transformation \( X \)

The following transformation is found that has no singularities at \( \pm \frac{\pi}{2} \):

\[
\begin{align*}
    x_1 &= y \\
    x_2 &= \dot{y} \\
    x_3 &= \cot \theta \\
    x_4 &= -\frac{1}{\sin^2 \theta} \dot{\theta} \\
    x_5 &= x \\
    x_6 &= \dot{x} 
\end{align*}
\]  

(4.34)

which is well-defined so long as \( \sin \theta \neq 0 \). With this transformation the original system (4.18) can be transformed to:

\[
\begin{align*}
    \dot{x}_1 &= x_2 \\
    \dot{x}_2 &= v_1 \\
    \dot{x}_3 &= x_4 \\
    \dot{x}_4 &= v_2 \\
    \dot{x}_5 &= x_6 \\
    \dot{x}_6 &= v_1 x_2 + \frac{c_y}{m} (x_1 x_2 - \dot{x}_3) 
\end{align*}
\]  

(4.35)

In appendix A.6.2 the same mathematical steps are shown as in section 4.3.1. This will give the following error-dynamics (using \( x_e = x - x_d \)):

\[
\begin{align*}
    \dot{x}_{1e} &= x_{2e} \\
    \dot{x}_{2e} &= v_1 - v_{1d} \\
    \dot{x}_{3e} &= x_{4e} \\
    \dot{x}_{4e} &= v_2 - v_{2d} \\
    \dot{x}_{6e} &= x_{6e} + \frac{c_y}{m} x_{6e} + x_2 (\frac{c_y}{m} x_{2e} + v_1 - v_{1d}) 
\end{align*}
\]  

(4.36)

with \( \ddot{x}_{1d} = v_{1d} + \frac{c_y}{m} x_2 \).

It is obvious that the error-dynamics (4.36) for system (4.35) are the same as the error-dynamics (4.30) for system (4.21). That means that this system can be controlled with the same kind of controller as in (4.31) and (4.33). This means for this transformation \( X \) that \( v_1 \) is equal to \( u_1 \) and \( v_2 \) is equal to \( u_2 \) only with gains different for this transformation and \( v_{1d} = \ddot{v}_{1d} = \dot{v}_{1d} = v_{1d}, \ddot{v}_{1d} = \dot{v}_{1d}, \dot{v}_{1d} = v_{1d} \), which are depending on the desired trajectory. The derivatives of \( \dddot{v}_{1d} \), needed for controller \( v_2 \) are:

\[
\begin{align*}
    \dddot{v}_{1d} &= \dddot{y}_d \\
    \dddot{v}_{1d} &= y_d^{(4)} \\
    \dddot{v}_{1d} &= y_d^{(5)} 
\end{align*}
\]  

(4.37)
and for the derivatives of $\bar{u}_{3d}$, needed for control-law $v_2$:

\[
\begin{align*}
\ddot{\bar{u}}_{1d} &= \dot{v}_{1d} + \frac{c_2}{m} \ddot{y}_d \\
\ddot{\bar{v}}_{1d} &= \ddot{v}_{1d} + \frac{c_2}{m} \dddot{y}_d \\
\ddot{\bar{u}}_{1d} &= \dot{v}_{1d} + \frac{c_2}{m} \ddot{y}_d \\
\ddot{\bar{v}}_{1d} &= \ddot{v}_{1d} + \frac{c_2}{m} \dddot{y}_d
\end{align*}
\] (4.38)

The singularities occur if $\theta$ becomes, or is close to, 0 or $\pi$. Therefore system (4.18), with this controller, will become unstable for the case $\theta$ is, or is close to, 0 or $\pi$. But there are no singularities in the case that $\theta$ is close to $\pm \frac{\pi}{2}$. A simulation is made to test transformation $X$, without having any singularities. The simulation is shown in appendix A.6.2. From this simulation it can be seen that if the desired trajectory is chosen such that $0 < \theta_d < \pi$ and the initial orientation is chosen such that $0 < \theta_0 < \pi$, then the orientation stays $0 < \theta < \pi$. This means that during the simulation of the controlled system no singularity occurred for tracking a sin in the $y$-direction.

### 4.3.3 Combination

Transformation $Z$ has no singularities if $\theta$ is close to 0 or $\pi$ and transformation $X$ has no singularities if $\theta$ is close to $\pm \frac{\pi}{2}$. To combine these advantages of the 2 transformations it is necessary to come up with some discontinuous control that makes it possible to switch between the 2 transformations ($Z$ and $X$) with their controllers (respectively $u_1$, $u_2$ and $\dot{z}_1$, $\dot{z}_2$).

**The switch at $\frac{1}{2} \pi$**

The first attempt in this discontinuous control is to switch between the two transformations if $\theta$ is at $\pm \frac{1}{2} \pi$ or $\pm \frac{3}{4} \pi$. This means that transformation $Z$, with control $u_1$ and $u_2$, is used for the case that $|\theta| \leq \frac{1}{4} \pi$ or $|\theta| \geq \frac{3}{4} \pi$. Transformation $X$, with control $v_1$ and $v_2$, is used for the case that $\frac{1}{4} \pi < |\theta| < \frac{3}{4} \pi$. See figure 4.4.

Figure 4.4: switch at $\frac{\pi}{4}$
To test this switch at $\frac{1}{4}\pi$ a simulation is made with the following desired $x$ and $y$:

$$
\begin{align*}
x_d &= r_1 \sin(\omega_1 t + \phi_1) \\
\dot{x}_d &= \omega_1 r_1 \cos(\omega_1 t + \phi_1) \\
\ddot{x}_d &= -\omega_1^2 r_1 \sin(\omega_1 t + \phi_1) \\
\dot{y}_d &= -r_2 \cos(\omega_2 t + \phi_2) \\
\ddot{y}_d &= \omega_2 r_2 \sin(\omega_2 t + \phi_2) \\
\dddot{x}_d &= \omega_1^3 r_1 \sin(\omega_1 t + \phi) \\
\dddot{y}_d &= -\omega_2^3 r_2 \sin(\omega_2 t + \phi_2)
\end{align*}
$$

And with the following parameters:

$$
r_1 = 2, \quad r_2 = 2, \quad \omega_1 = \omega_2 = 0.4, \quad \phi_1 = 0, \quad \phi_2 = 0
$$

a circle is obtained which has to be tracked by the surface vessel. With the desired orientation of the surface vessel satisfying equation (4.25) for transformation $Z$ and equation (A.6) for transformation $X$. The initial position of the surface vessel is $(x, \theta, y) = (3, -\frac{\pi}{4}, 0)$. Where the desired trajectory begins (at time $t = 0$) with $(x, \theta, y) = (0, 0.46, -1)$. The results obtained with this switch at $\frac{1}{4}\pi$ as discontinuous control for the surface vessel tracking a circle are shown below:

Figure 4.5: $x, \dot{x}, \theta, \dot{\theta}, y, \dot{y}$ with switch at $\frac{\pi}{4}$  

Figure 4.6: errors with switch at $\frac{\pi}{4}$
As can be seen from the figures, the errors are driven to zero. At the moment that there is a switch between the $z$ transformations an error is introduced. If, for example, there is an error in the $x$-variable, then transformation $X$ with control $u_1$ and $u_2$ will try in a different way to reduce this error then transformation $Z$ with control $v_1$ and $v_2$. This is due to the fact that for transformation $Z$ the $x$-coordinate is controlled by controller (4.31) and for transformation $X$ the $x$-coordinate is controlled by a controller such as (4.33), but then with different coefficients as depicted in section 4.3.2. As seen in the figures this bump in the error decays to zero.

A problem with the hard switch is that when $Q$ is around $\pi$ or $\pi$ rapidly switching between controllers can occur. This rapidly switching between controllers can also be seen in the real inputs $F_1$ and $F_2$ and influences the motion of the surface vessel negatively. This can especially be seen in figure 4.5 in $\theta$, where after approximately 1 second there is a lot of chattering. In appendix A.6.3 the motion of the surface vessel is plotted in figure A.18. Here the bars at the end of the surface vessel correspond to the forces $F_1$ and $F_2$.

To prevent this rapidly switching, another combination of the different controls can be developed.

A smooth Switch

Another combination of the controller is to use both controllers at the same time. The "hard" switch (as in section 4.3.3) is not wanted. The idea is to use only transformation $Z$ if $\theta$ is close to zero or $\pi$ and to use only transformation $X$ if $\theta$ is close to $\pm \frac{\pi}{2}$. In the area in between an overlap of the controls is used to produce a smooth, without chattering, output to apply on the system. The new forces are calculated with the parameters $a$ and $b$ (which are a function of $\theta$). See figure 4.9.
The parameters $a$ and $b$ are related with $\theta$ as follows:

\[
a = \begin{cases} 
1 - \frac{\theta \cdot \text{sign}\theta + (\frac{\pi}{2} - \epsilon)}{\frac{\pi}{2} - 2\epsilon} & \text{if } |\theta| \leq \epsilon \text{ or } |\theta| \geq \pi - \epsilon \\
0 & \text{if } \epsilon < |\theta| < \frac{\pi}{2} - \epsilon \\
\frac{\theta \cdot \text{sign}\theta - (\frac{\pi}{2} + \epsilon)}{\frac{\pi}{2} - 2\epsilon} & \text{if } \frac{\pi}{2} - \epsilon \leq |\theta| \leq \frac{\pi}{2} + \epsilon \\
1 - \frac{\theta \cdot \text{sign}\theta + (\pi - \epsilon)}{\frac{\pi}{2} - 2\epsilon} & \text{if } \frac{\pi}{2} + \epsilon < |\theta| < \pi - \epsilon
\end{cases}
\]  

\[b = \begin{cases} 
0 & \text{if } |\theta| \leq \epsilon \text{ or } |\theta| \geq \pi - \epsilon \\
1 - \frac{\theta \cdot \text{sign}\theta - \epsilon}{\frac{\pi}{2} - 2\epsilon} & \text{if } \epsilon < |\theta| < \frac{\pi}{2} - \epsilon \\
\frac{\theta \cdot \text{sign}\theta + (\pi - \epsilon)}{\frac{\pi}{2} - 2\epsilon} & \text{if } \frac{\pi}{2} - \epsilon \leq |\theta| \leq \frac{\pi}{2} + \epsilon \\
1 & \text{if } \frac{\pi}{2} + \epsilon < |\theta| < \pi - \epsilon
\end{cases} \]  

(4.41)

This combination uses both the forces $F^v_x, T^v_x$, which are determined using transformation $Z$ and corresponding controller, and the forces $F^w_x, T^w_x$, which are determined using transformation $X$ and corresponding controller. Using the original forces gives a better physical idea of the combination. The new forces $F^*_x$ and $T^*_x$, which are applied on the system, are determined in the following way:

\[
F^*_x = aF^v_x + bF^w_x \\
T^*_x = aT^v_x + bT^w_x
\]

(4.42)

This gives the following result for the surface vessel tracking a circle (with the desired variables as in (4.39)). With the same parameters as used for the simulation in the previous section. The value of $\epsilon$ is 0.15. The initial condition is $(x, \theta, y) = (3, -\frac{\pi}{2}, 0)$. 

Figure 4.9: Parameters $a$ and $b$ as a function of $\theta$
Figure 4.10: $x, \dot{x}, \theta, \dot{\theta}, y, \dot{y}$ with smooth switching controller

Figure 4.11: errors with smooth switching controller
As can be seen from the figures, the errors are also driven to zero. In appendix A.6.3 the motion of the surface vessel is plotted in figure A.19. To compare the results of both the combinations the following plots are made:
As can be seen in figure 4.14, where only the first 5 seconds are plotted for comparison, the actual forces are smoother if the smooth switching controller is used. Also in figure 4.15, where only the first 5 seconds are plotted for comparison, one can see that especially the error in $\dot{\theta}$ is smoother with the smooth switch then for the switch at $\frac{\pi}{2}$. And as one can see in figure 4.15 and 4.16, there is no difference in the time needed to control the errors to zero and the time needed to exactly follow the desired trajectory.

This smooth switching controller works for the simulation with the system as described above. It is not proven yet that this smooth switching controller always results in stable tracking. The proof of stability of this smooth switching controller is an interesting point for future research.
Chapter 5

Conclusions and Recommendations

At the end of this report it can be concluded that for point-to-point control of underactuated systems 2 control-techniques are presented, the $\sigma$-process and the geometric phase. First the $\sigma$-process is able to bring the PPR underactuated manipulator from an initial state to a desired state. The $\sigma$-process is also able to bring the planar manipulator with an unactuated elastically mounted end-effector to a specified equilibrium position, while the unactuated end-effector dynamics are not excited. Second the geometric phase, which can be applied to systems with first order nonholonomic constraints, is applied to underactuated systems with second order nonholonomic constraints. The geometric phase also brings the initial state of the icebot to the origin, which is the desired state. The geometric phase also brings the PPR underactuated manipulator from an initial rest state to a desired rest state. These 2 control-techniques for point-to-point control of underactuated systems can not be compared at this point. This is because the control gains have a lot of influence such that probably the $\sigma$-process can be improved, comparing to the geometric phase, by fine-tuning of the control gains.

For the tracking-control problem of underactuated systems it can be concluded that the existing controller works well for the tracking control problem of the PPR underactuated manipulator.

If this existing controller is applied to a surface vessel, problems arise due the singularities in the local coordinate transformation. For the tracking control problem of the surface vessel a discontinuous controller is developed, which can cope with singularities in the local transformation. The problems that occurred while using the switching controller at $\frac{\pi}{4}$ are solved using a smooth switching controller. This smooth switching controller produced smooth outputs, without chattering, to apply on the system.

Some future research should be done in finding another transformation (or another combination of transformations), such that both the transformations reduce the error in the same way, thereby using the same kind of controller for the same variable. This will prevent the problems with the switch at $\frac{\pi}{4}$, due to rapidly switching between the transformations. Another point of consideration is that the stability for the smooth switching controller is not proven yet. This is an interesting point for future research. Some other recommendations can be made to obtain more realistic results in the simulations. First simulations have to be done where damping is taken into account. Also noise on the measurements and an observer for the unmeasured parameters has to be dealt with. And finally some experiments with real systems should be done.
Appendix A

Simulation figures

A.1 $\sigma$-process applied to a PPR underactuated manipulator

Figure A.1: Motion-plot of the PPR
A.2 \( \sigma \)-process applied to a planar manipulator with a unactuated elastically mounted end-effector
A.3 Geometric phase applied to an icebot

Figure A.4: Motion-plot of the icebot

A.4 Geometric phase applied to a PPR underactuated manipulator

Figure A.5: Motion-plot of the PPR
Figure A.6: $\xi_1$, $v_1$, $\xi_2$, $v_2$ and $\xi_1$

Figure A.7: $u_1$ and $u_2$
A.5 Cascaded backstepping applied on a PPR underactuated manipulator

Figure A.8: motion-plot of the PPR (1)  Figure A.9: motion-plot of the PPR (2)

Figure A.10: motion-plot of the PPR (3)  Figure A.11: motion-plot of the PPR (4)
A.6 Cascaded backstepping applied on a surface vessel

A.6.1 Transformation $Z$

Simulation

A simulation is made to test the transformation $Z$, without having any singularities. A desired trajectory is made such that $-\frac{\pi}{2} < \theta_d < \frac{\pi}{2}$. An example of such trajectory is a $\sin$ in the $x$-direction. The initial orientation of the surface vessel is chosen such that $-\frac{\pi}{2} < \theta_0 < \frac{\pi}{2}$. For this trajectory the desired variables are:

$$
\begin{align*}
    x_d &= \omega_1 t \\
    y_d &= -r_2 \cos(\omega_2 t + \phi_2) \\
    \dot{x}_d &= \omega_1 \\
    \dot{y}_d &= \omega_2 r_2 \sin(\omega_2 t + \phi_2) \\
    \ddot{x}_d &= 0 \\
    \ddot{y}_d &= -\omega_2^2 r_2 \cos(\omega_2 t + \phi_2) \\
    \dot{x}^{(4)}_d &= 0 \\
    \dot{y}^{(4)}_d &= -\omega_2^4 r_2 \cos(\omega_2 t + \phi_2) \\
    x^{(5)}_d &= 0 \\
    y^{(5)}_d &= \omega_2^5 r_2 \sin(\omega_2 t + \phi_2)
\end{align*}
$$

(A.1)

The simulation is done with the following constants

$$r_2 = 1, \; \omega_1 = 2, \; \omega_2 = 1, \; \phi_2 = 0$$

for the desired trajectory. And the system parameters are chosen as

$$c_y = 1, \; c_z = 1, \; g_z = 1, \; m = 1, \; I = 1, \; d = 1$$

The initial conditions are $(x, \theta, y) = (0, 0, 0)$ The results are shown in figure A.14. As seen in the figure stays $-\frac{\pi}{2} < \theta < \frac{\pi}{2}$ during the simulation of the controlled system, such that no singularities occur. The motion of the surface vessel is also plotted in figure A.15. This motion is only plotted for the first 6 seconds.
A.6.2 Transformation $X$

The following transformation is found that has no singularities at $\pm \frac{\pi}{2}$:

\[
\begin{align*}
    x_1 &= y \quad x_2 = \dot{y} \\
    x_3 &= \cot \theta \quad x_4 = -\frac{1}{\sin^2 \theta} \dot{\theta} \\
    x_5 &= x \quad x_6 = \dot{x}
\end{align*}
\]  

(A.2)

which is well-defined so long as $\sin \theta \neq 0$. With this transformation the original system (4.18) can be transformed to:

\[
\begin{align*}
    \dot{x}_1 &= x_2 \\
    \dot{x}_3 &= x_4 \\
    \dot{x}_5 &= x_6 \\
    \dot{x}_6 &= v_1 x_2 + \frac{c_y}{m} (x_1 x_2 - \dot{x}_3)
\end{align*}
\]  

(A.3)

which is the same as (4.21) but now with

\[
\begin{align*}
    v_1 &= \frac{F_z}{m} \sin \theta - \frac{f_1}{m} \sin \theta - \frac{f_2}{m} \cos \theta \\
    v_2 &= (2 \frac{\cos \theta}{\sin^3 \theta}) \dot{\theta}^2 - \frac{1}{\sin^2 \theta} \left( \frac{T_z}{I} - \frac{c_y}{I} \dot{\theta} \right)
\end{align*}
\]

with $f_1$ and $f_2$ as in (4.22). With this transformation the accelerations $(\ddot{x}, \ddot{y}, \ddot{\theta})$ become:

\[
\begin{align*}
    \ddot{y} &= v_1 \\
    \ddot{x} &= v_1 \cot \theta + c_y/m(\dot{y} \cot \theta - \ddot{x}) \\
    \ddot{\theta} &= -(v_2 - 2 \frac{\cos \theta}{\sin^3 \theta} \dot{\theta}^2) \sin^2 \theta
\end{align*}
\]  

(A.4)
Assuming that \( \theta = (-\pi, \pi) \), you can see that the transformation has singularities if \( \theta \) becomes, or is close to, 0 or \( \pi \) because of the cot-term in \( \ddot{x} \) and the \( \frac{1}{\sin^3 \theta} \)-term in \( \dot{\theta} \) in equation (A.4).

The desired variables, for a certain desired trajectory in the \( x \)-variables, are

\[
\begin{align*}
\dot{x}_{1d} &= x_{2d} \\
\dot{x}_{2d} &= v_{1d} \\
\dot{x}_{3d} &= x_{4d} \\
\dot{x}_{4d} &= v_{2d} \\
\dot{x}_{5d} &= x_{6d} \\
\dot{x}_{6d} &= v_{1d}x_{2d} + \frac{c_y}{m}(\dot{x}_{1d}x_{2d} - \dot{x}_{3d})
\end{align*}
\] (A.5)

The desired variables in the original variables \( (x, y, \theta) \) have to satisfy equations (A.5). The last equation of (A.5) can be written as:

\[
\ddot{x}_d = \ddot{y}_d \cot \theta_d + \frac{c_y}{m}(\dot{y}_d \cot \theta_d - \dot{x}_d)
\] (A.6)

This means that the desired variables can not be chosen arbitrarily, but have to satisfy the above equation. This means that \( x_{3d} = \cot \theta_d \), for a certain given path in \( x (= x_{5d}) \) and \( y (= x_{1d}) \), becomes

\[
x_{3d} = \frac{\ddot{x}_d + \frac{c_y}{m}\dot{x}_d}{\ddot{y}_d + \frac{c_y}{m}\dot{y}_d}
\] (A.7)

and (with \( x_{4d} = \ddot{x}_d \))

\[
x_{4d} = \frac{\dot{\ddot{y}}_d - \dot{\ddot{x}}_d}{\dot{y}_d^2}
\] (A.8)

The desired inputs become:

\[
\begin{align*}
v_{1d} &= \ddot{y}_d \\
v_{2d} &= \frac{(\ddot{x}_d - \dot{y}_d)\dot{y}_d^2 - 2\dot{x}_d\dot{y}_d(\ddot{x}_d - \dot{y}_d)}{\dot{y}_d^2}
\end{align*}
\] (A.9)

where the derivatives of \( \ddot{x} \) and \( \dot{y} \) are given as in (4.29). All the desired variables are known if a desired trajectory is generated in the \( x, y \)-space.

This results in the error-dynamics (using \( x_e = x - x_d \)):

\[
\begin{align*}
\dot{x}_{1e} &= x_{2e} \\
\dot{x}_{2e} &= v_1 - v_{1d} \\
\dot{x}_{3e} &= x_{4e} \\
\dot{x}_{4e} &= v_2 - v_{2d} \\
\dot{x}_{5e} &= x_{6e} \\
\dot{x}_{6e} &= x_{3e}v_{1d} - \frac{c_y}{m}x_{6e} + x_2\left(\frac{c_y}{m}x_{2e} + v_1 - v_{1d}\right)
\end{align*}
\] (A.10)

with \( v_{1d} = v_{1d} + \frac{c_y}{m}x_2 \).

**Simulation**

A simulation is made to test the transformation \( X \), without having any singularities. A desired trajectory is made such that \( 0 < \theta_d < \pi \). An example of such trajectory is a sin in
the $y$-direction. The initial orientation of the surface vessel is chosen such that $0 < \theta_0 < \pi$. For this trajectory the desired variables are:

$$
\begin{align*}
    x_d &= r_1 \sin(\omega_1 t + \phi_1) \\
    y_d &= \omega_2 t \\
    \dot{x}_d &= \omega_1 r_1 \cos(\omega_1 t + \phi_1) \\
    \dot{y}_d &= \omega_2 \\
    \ddot{x}_d &= -\omega_1^2 r_1 \sin(\omega_1 t + \phi_1) \\
    \ddot{y}_d &= 0 \\
    \dot{x}_d^{(4)} &= \omega_1^4 r_1 \sin(\omega_1 t + \phi) \\
    \dot{y}_d^{(4)} &= 0 \\
    \dot{x}_d^{(5)} &= \omega_1^5 r_1 \cos(\omega_1 t + \phi) \\
    \dot{y}_d^{(5)} &= 0
\end{align*}
$$

(A.11)

The simulation is done with the following constants for the desired trajectory:

$$
    r_2 = 1, \quad \omega_1 = 1, \quad \omega_2 = 2, \quad \phi_2 = 0
$$

Here the system parameters are chosen as:

$$
    c_y = 1, \quad c_x = 1, \quad c_z = 1, \quad m = 1, \quad I = 1, \quad d = 1
$$

The initial conditions are $(x, \dot{x}, \theta, \dot{\theta}) = (0, \frac{\pi}{2}, 0)$. The results are shown here in figure A.16. As seen in the figure stays $0 < \theta < \pi$ during the simulation of the controlled system, such

![Figure A.16: $x$, $\dot{x}$, $\theta$, $\dot{\theta}$, $y$, $\dot{y}$](image)

that no singularities occur. The motion of the surface vessel is also plotted in figure A.17. This motion is only plotted for the first 6 seconds.

![Figure A.17: Motion-plot of the surface vessel (only the first 6 seconds)](image)
A.6.3 Combination

Figure A.18: Motion-plot with switch at $\frac{\pi}{4}$

Figure A.19: Motion-plot with smooth switch
Bibliography


