The modeling and H-infinite tracking control of an autonomous unicycle

Ebus, R.

Published: 01/01/1996

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.
The modeling and $H_\infty$ tracking control of an autonomous unicycle

R.Ebus

WFW report 96.024

Trainee project
Supervisor: A.G. de Jager
System and Control Group
Faculty of Mechanical Engineering
Eindhoven University of Technology
February 1996
Abstract

This report deals with the modeling and tracking control of an unstable autonomous unicycle, using a 2D model. The goal is to design a stabilizing, multivariable $H_\infty$ controller with a good tracking performance and a high level of robustness for both directions.

An accurate non-linear model is derived and linearized for design purposes. The actual design took place using the MHC-toolbox for Matlab. The controller designed has in first instance a so called two degrees-of-freedom structure, because with such a controller it is usually easier to attain the time domain specifications.

The controller designed is used during several simulations. Besides investigations considering tracking and stability of the non-linear system, also differences between the linear and non-linear model are investigated. Robustness is also researched by looking at the influence of parameter changes during the simulations. Lastly, the results are compared to the results when using an one degree-of-freedom controller.

The results are satisfying as far as they concern the tracking and stabilization in the direction of motion. In the direction perpendicular to the direction of motion, the simulations with the non-linear model give unstable results just as the simulations with the linear model. This in spite of the fact that all poles of the closed loop system are situated in the left half plane. The reason is that for design purposes the poles of the linear model have to be moved slightly, because two of them are zero. The closed loop system with this lightly disturbed model is stable, but computing the poles of the closed loop system with the original linear model gives two poles in the right half plane, which of course are the cause of the unstable behavior. The controller designed is apparently not designed well enough to be robust for this kind of model errors.

The robustness for parameter-disturbances is good, except for changes in the parameters which describe the friction. The used model of the model errors is probably not adequate enough to take this kind of errors into consideration well. Lastly, the results of the controller with a two degree-of-freedom structure appear to be almost the same as the results with an one degree-of-freedom structure.
Table of contents

ABSTRACT 2

LIST OF SYMBOLS 4

1. INTRODUCTION 5

2. EQUATIONS OF MOTION 6
   2.1. System description 6
   2.2. Linearization 10
   2.3. Controllability and observability 12

3. CONTROLLER DESIGN 14

4. RESULTS 19

5. CONCLUSIONS AND RECOMMENDATIONS 24

REFERENCES 25

APPENDICES 26
   I. Derivation of the kinetic energy
   II. Simulation files 30
   III: Transfer functions 37
List of symbols

$A$ Work
$COM$ Center-of-mass of the frame, the two motors and the disk
$d$ Sensor noise
$g$ Acceleration of gravity
$G$ Generalized plant
$h$ Distance of the COM to the wheel axis
$i$ Transmission ratio
$J$ Inertia
$K$ Controller
$M$ Mass
$n$ Model error
$P$ Controllability matrix
$P$ Plant
$q$ Column which contains the generalized coordinates
$Q$ Column with the generalized forces
$Q$ Observability matrix
$r$ Reference signal
$R$ Radius
$R$ Rotation matrix
$R$ Control sensitivity
$S$ Matrix with eigenvalues
$S$ Sensitivity
$t$ Time
$T$ Kinetic energy
$T$ Complementary sensitivity
$u$ Control input
$U$ Potential energy
$v$ Column with external inputs
$V$ Weight matrix
$w$ Error vector
$W$ Weight matrix
$x$ The position of the contact point of the wheel and the ground along the $e_{o1}$ axis
$X$ Matrix with eigenvectors
$y$ Observation vector

$\varphi$ The angular rotation of the COM around the local $e_{w2}$ axis of the wheel, as defined in Fig 1 by its rotation vector. This angle is zero when, in the projection of the unicycle on the $e_{o1}/e_{o2}$-plane, the angle $\varphi$ is zero (see also Fig 1).

$\alpha$ The angular rotation of the COM around the $e_{o1}$-axis as defined in Fig 1, relative to the vertical

$\beta$ The angular rotation, with respect to the motor's housing, of the disk fixed on the upper motor axis

$\mu$ Coefficient of the wheel bearing friction
$\tau$ Motor torque
$v$ Coefficient of the viscous rotor friction in the motor
$\omega$ Rotation vector
CHAPTER 1

Introduction

The unicycle, a wheel and a frame with an actuated joint in between, is a motivating example for applications of modern control techniques, because it is unstable in the upright position and it is a non linear system. In future, the unicycle may be built for experimental purposes, but first different classes of controllers can be tested during several student assignments. This report is such an assignment and investigates the possibilities for the implementation of an $H_\infty$ controller.

In previous studies of an unicycle, see [1,2,3,4], unicycles are considered with a setup different from the one used here. Therefore, an accurate model of the unicycle is derived. The design of the controller is based on a linearized version of this model and took place using the MHC-toolbox for Matlab, see [11]. This approach of the $H_\infty$ design method is based on expressing specifications in the frequency domain for several closed loop transfers and is a so-called loop shaping method (see e.g. [10]). The main goal is to find a stabilizing controller with a good tracking performance. Furthermore, the controller has to have a high level of robustness.

CHAPTER 2 contains the derivation of the equations of motion as well as its linearization. Also observability and controllability of the system will be investigated. In CHAPTER 3, one will find a description of the controller design and an explanation of the setup which is used. The results of the simulations will be discussed in CHAPTER 4 and they will lead to the conclusions and recommendations of CHAPTER 5.
CHAPTER 2

Equations of motion

2.1. System description

The dynamic behavior of an unicycle has been studied previously, see [1,2,3,4]. There are however some differences between these studies and the one in this report. In [1] and [2], they consider a system with guiding wheels. Because the purpose is to stabilize the system in both longitudinal (direction of motion) and lateral direction (direction perpendicular to the direction of motion), the system has to be extended in such a way that also the dynamics in the lateral direction can be affected. In [3] they use a system in which a human riding an unicycle is modeled in detail. This, of course, is much more complex in both modeling and building an experimental version of the system. Because one of the intentions, in future, is to build the unicycle with taking into consideration simplicity and costs, the derived model of [3] is not useful here. In [4], a turntable is used for stabilizing the unicycle. This turntable has, especially at high rates, a major influence at the dynamics in both longitudinal and lateral direction. The idea (using rotational inertia's to stabilize) seems very useful, but here there is decided to decouple the dynamics in lateral and longitudinal direction as much as possible, and the principle is only used in lateral direction. This results in the unicycle as shown in FIG 1. It can be considered as a multi-body system which consists of four different parts: a frame, a wheel, a motor with transmission (the lower motor) and a motor with a rotating disk fixed on it (the upper motor).

The friction perpendicular to the direction of motion is considered to be infinite. This assumption can be justified, because under normal conditions it is very hard to move a wheel in the e_o directional by sliding it over the floor. Slipping of the wheel in the e_o-direction will therefore be out of order. Also rotation around the vertical is not taken into consideration. After all, this would imply that the unicycle makes curves, while the goal is to make the unicycle ride in a straight line. Because of the
friction which arises when turning the wheel on the ground and the lack of an special actuator which
influences this turning, the assumption seems justified.

The equations of motion will be derived with the Lagrange-formalism as defined in the following
equation:

\[ \frac{d}{dt}(T) - T + U = Q \]  

(2.1)

Here, \( T \) is the kinetic energy of the system and \( U \) is the potential energy of the system. Furthermore, \( Q \) is a column with the generalized forces except of those which can be derived from a potential, like gravity forces. The elements of \( Q \) can be derived by the principle of virtual work.

The column \( q \) contains the generalized coordinates \( x, \varphi, \alpha \) and \( \beta \). These are defined as follows:

- \( x \): the position of the contact point of the wheel and the ground along the \( e_3 \)-axis
- \( \varphi \): the angular rotation of the COM (center-of-mass of the frame, the two motors and the disk) around the local \( e_2 \)-axis of the wheel, as defined in FIG 1 by its rotation vector. This angle is zero when, in the projection of the unicycle on the \( e_0/e_o \)-plane, the angle \( \varphi^a \) is zero (see also FIG 1).
- \( \alpha \): the angular rotation of the COM around the \( e_2 \)-axis as defined in FIG 1, relative to the vertical
- \( \beta \): the angular rotation, with respect to the motor's housing, of the disk fixed on the upper motor axis

The relevant parameters are shown in the table below. Some of the parameters are taken from [2]. The other parameters are derived by using realistic chosen dimensions, hereby using steel as material. When speaking of an inertia around a particular \( e_i \)-axis, there is meant the local \( e_i \)-axis. This \( e_i \)-axis equals the corresponding global \( e_i \)-axis when all the generalized coordinates are zero.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M_F )</td>
<td>Mass of the frame, the two motors and the disk</td>
<td>8.50</td>
<td>kg</td>
</tr>
<tr>
<td>( M_W )</td>
<td>Mass of the wheel</td>
<td>1.50</td>
<td>kg</td>
</tr>
<tr>
<td>( J_{F1} )</td>
<td>Inertia of the frame around the ( e_1 )- and ( e_2 )-axis</td>
<td>0.180</td>
<td>kg m²</td>
</tr>
<tr>
<td>( J_{F2} )</td>
<td>Inertia of the frame around the ( e_3 )-axis</td>
<td>0.045</td>
<td>kg m²</td>
</tr>
<tr>
<td>( J_{W1} )</td>
<td>Inertia of the wheel around the ( e_1 )- and ( e_3 )-axis</td>
<td>0.030</td>
<td>kg m²</td>
</tr>
<tr>
<td>( J_{W2} )</td>
<td>Inertia of the wheel around the ( e_2 )-axis</td>
<td>0.060</td>
<td>kg m²</td>
</tr>
<tr>
<td>( J_{M1} )</td>
<td>Inertia of the lower motor axis around the ( e_1 )- and ( e_3 )-axis</td>
<td>0.000</td>
<td>kg m²</td>
</tr>
<tr>
<td>( J_{M2} )</td>
<td>Inertia of the lower motor axis around the ( e_2 )-axis</td>
<td>0.001</td>
<td>kg m²</td>
</tr>
<tr>
<td>( J_{S2} )</td>
<td>Inertia of the upper motor (and disk) around the ( e_2 )- and ( e_3 )-axis</td>
<td>0.035</td>
<td>kg m²</td>
</tr>
<tr>
<td>( J_{S1} )</td>
<td>Inertia of the upper motor (and disk) around the ( e_1 )-axis</td>
<td>0.070</td>
<td>kg m²</td>
</tr>
<tr>
<td>( \nu_1 )</td>
<td>Coefficient of the viscous rotor friction of the lower motor</td>
<td>6.33</td>
<td>N m s rad⁻¹</td>
</tr>
<tr>
<td>( \nu_2 )</td>
<td>Coefficient of the viscous rotor friction of the upper motor</td>
<td>6.33</td>
<td>N m s rad⁻¹</td>
</tr>
<tr>
<td>( \mu )</td>
<td>Coefficient of the wheel bearing friction</td>
<td>2.12</td>
<td>mN m s rad⁻¹</td>
</tr>
<tr>
<td>( R )</td>
<td>Wheel radius</td>
<td>0.2</td>
<td>m</td>
</tr>
<tr>
<td>( h )</td>
<td>Distance of the COM to the wheel axis</td>
<td>0.4</td>
<td>m</td>
</tr>
<tr>
<td>( R_{Si} )</td>
<td>Inner radius of the disk</td>
<td>0.15</td>
<td>m</td>
</tr>
<tr>
<td>( R_{Su} )</td>
<td>Outer radius of the disk</td>
<td>0.2</td>
<td>m</td>
</tr>
<tr>
<td>( i )</td>
<td>Transmission ratio</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>( g )</td>
<td>Acceleration of gravity</td>
<td>9.81</td>
<td>m s⁻²</td>
</tr>
</tbody>
</table>

**TABLE 1: The model parameters of the unicycle**
Now, the kinetic energy of the system can be derived. For this purpose, the four different bodies are considered separately. The complete derivation can be found in APPENDIX I. From this it follows:

\[ T_{\text{kin}} = T_{\text{rot}} + T_{\text{trans}} \]  

\[ T_{\text{rot}} = \frac{1}{2} \left( J_{w_1} + J_{M_1} + (J_{s_1} + J_{F_1}) \cos^2 \varphi + (J_{s_2} + J_{F_3}) \sin^2 \varphi \right) \dot{\alpha}^2 + \frac{1}{2} \left( J_{F_1} + J_{s_2} + J_{M_1} \left( \frac{r_i}{r_i} \right)^2 \right) \dot{\theta}^2 + \frac{1}{2} \left( J_{w_2} \left( \frac{r_i}{r_i} \right)^2 + J_{M_2} \left( \frac{r_i}{r_i} \right)^2 \right) \dot{\gamma}^2 + \frac{1}{2} J_{s_1} \dot{\beta}^2 + (J_{s_1} \cos \varphi) \dot{\alpha} \dot{\beta} + J_{M_2} \left( \frac{r_i}{r_i} \right) \dot{\gamma} \dot{\phi} \]

\[ T_{\text{trans}} = \frac{1}{2} \left( M_w + M_f \right) \dot{\gamma}^2 + \frac{1}{2} M_f \left( h^2 \cos^2 \alpha \cos^2 \varphi + h^2 \sin^2 \varphi \right) \dot{\phi}^2 + \frac{1}{2} \left( M_w R^2 + M_f \left( h^2 \sin^2 \alpha \sin^2 \varphi + (R + h \cos \varphi)^2 \right) \right) \dot{\alpha}^2 + (M_f \ h \cos \alpha \cos \varphi) \dot{\gamma} \dot{\phi} - (M_f \ h \sin \alpha \sin \varphi) \dot{\alpha} - \left( \frac{1}{4} M_f h^2 \sin(2 \alpha) \sin(2 \phi) \right) \dot{\phi} \]

The potential energy of the system is (using the ground as reference level)

\[ U = M_w g R \cos \alpha + M_f g (R + h \cos \varphi) \cos \alpha \]  

As said before, the column \( Q \) can be derived by using the principle of virtual work. The virtual work for virtual changes of the generalized coordinates is

\[ \delta A = \left( -\left( \frac{\nu_1}{i^2} + \mu \right) \left( \frac{\dot{x}}{R} - \dot{\phi} \right) + \frac{\tau_1}{i} \left( \frac{\delta x}{R} - \delta \varphi \right) + \left( \tau_2 - \nu_2 \dot{\beta} \right) \delta \beta = Q^T \delta \dot{q} \right) \]  

(2.4)

In this, \( \tau_1 \) and \( \tau_2 \) represent the motor torque's of the lower and upper motor and \( \nu_1 \) and \( \nu_2 \) are the coefficients of viscous damping in the bearings of the motor axis. \( \mu \) is the coefficient of viscous damping between the wheel axis and the frame.

Knowing \( T, U \) and \( Q \), equation (2.1) can be used to work out the equations of motion. These can be written as follows

\[ M(q) \ddot{q} + B(q, \dot{q}) \dot{q} + C(q, \dot{q}) = Hu \]  

(2.5)

In this, \( u \) represents a column with \( \tau_1 \) and \( \tau_2 \) in it. The matrices \( M, B, C \) and \( H \) are written out on the next page. The following parameters are introduced in these matrices to simplify the expressions.

\[ J_A = \frac{1}{r_i^2} J_{w_2} + \frac{1}{r_i^2} J_{M_2} \]
\[ J_B = \frac{1}{r_i^2} J_{M_2} \]
\[ J_C = J_{F_1} + J_{s_2} + \left( \frac{r_i}{r_i} \right)^2 J_{M_2} \]
\[ J_D = J_{w_1} + J_{M_1} \]
\[ J_E = J_{F_1} + J_{s_1} \]
\[ J_G = J_{s_2} + J_{F_3} \]  

(2.6)
Because deriving these equations from the kinetic and potential energy is a precise job, it makes sense to verify the correctness of the equations as far as possible. Two methods are used. One to check if the equations are derived well and one to check if the system has no energetic curiosities.

Out of [5] follows that there has to be a certain relationship between the mass matrix $M$ and the matrix $B$. More precisely there has to count that

$$B = (G - \frac{1}{2} G^T)$$

with

$$G = \begin{bmatrix} g_1 & g_2 & g_3 & g_4 \end{bmatrix} \quad \text{and} \quad g_i = \frac{\delta M}{\delta q_i}$$

(2.8)
When using $M$ to calculate $B$ in this way, one will find a more complex matrix as derived previously. Note therefore, that not $B$, but the product $B\dot{q}$ has to be checked, because $B$ is not unique. This is the result of several multiplicative rate terms which appear in the equations of motion. Note also that this way of deriving the equations of motion is only allowed if the considered mechanical system has a finite amount of independent degrees of freedom (generalized coordinates).

It is also very obvious that the total energy of the system has to be constant when the friction coefficients are set to zero and the two motors do not deliver any energy to the system ($u = 0$). If the energy would not be constant in such a case, the equations of motions would not be realistic. Verifying the energy for the derived model indeed results in a constant level of total energy.

### 2.2. Linearization

The system can be linearized in an analytical way or numerically. Using the analytical method implies (according to [6]):

$$
\frac{\delta f}{\delta \dot{q}} \delta \dot{q} + \frac{\delta f}{\delta \ddot{q}} \delta \ddot{q} + \frac{\delta f}{\delta q} \delta q + \frac{\delta f}{\delta u} \delta u = 0
$$

(2.9)

Hereby, the equations have to be written as $f(\dot{q}, \ddot{q}, q, u) = 0$. When the system is linearized around its vertical position with $q_0 = [0 \ 0 \ 0]^T$ and $u_0 = [0 \ 0]^T$, the following set of linear equations will be found.

$$
M_L \ddot{q} + B_L \dot{q} + C_L q = H_L u
$$

(2.10)

in which

$$
M_L = \begin{bmatrix}
M_w + M_f + J_A & M_f h + J_B & 0 & 0 \\
M_f h + J_B & M_f h^2 + J_C & 0 & 0 \\
0 & 0 & M_f (h+R)^2 + M_w R^2 + J_k + J_d & J_{s1} \\
0 & 0 & J_{s1} & J_{s1}
\end{bmatrix}
$$

$$
B_L = \begin{bmatrix}
\left(\frac{1}{h}\right)^2 \left(\frac{1}{h} v_1 + \mu\right) & -\frac{1}{h} \left(\frac{1}{h} v_1 + \mu\right) & 0 & 0 \\
-\frac{1}{h} \left(\frac{1}{h} v_1 + \mu\right) & \left(\frac{1}{h} v_1 + \mu\right) & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & v_2
\end{bmatrix}
$$

$$
C_L = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & -M_f g h & 0 & 0 \\
0 & 0 & -M_w g R - M_f g (R+h) & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
$$

$$
H_L = \begin{bmatrix}
\frac{1}{h} & 0 \\
-\frac{1}{t} & 0 \\
0 & 0 \\
0 & 1
\end{bmatrix}
$$
Note that both $B_L$ and $C_L$ are the result of the linearization of matrix $C$ in (2.7) and that the terms of matrix $B$ all cancel out during the linearization.

In state space representation with the parameters of TABLE I, (2.10) gives

$$\dot{z} = Az + Bu$$  \hspace{1cm} (2.11)$$

with

$$A = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & -17.2035 & 0 & 0 & -54.1535 & 10.8307 & 0 & 0 \\
0 & 58.3149 & 0 & 0 & 137.0048 & -27.4010 & 0 & 0 \\
0 & 0 & 15.9081 & 0 & 0 & 0 & 0 & 1.9009 \\
0 & 0 & -15.9081 & 0 & 0 & 0 & 0 & -92.3295
\end{bmatrix}$$

$$B = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 1.7104 \\
0 & -4.3273 \\
0 & 0 \\
0 & -0.3003 \\
0 & 14.5860
\end{bmatrix}$$

$$z = [x \ \phi \ \alpha \ \beta \ \dot{x} \ \dot{\phi} \ \dot{\alpha} \ \dot{\beta}]^T$$

$$u = [\tau_1 \ \tau_2]^T$$  \hspace{1cm} (2.12)$$

Frequency plots of the transfer functions from $u$ to the elements of $q$, can be found in APPENDIX III.

Note that the linearization could also be done numerically with the use of the Matlab-routine Jacobian. As one will find out when doing it, this results in the same matrices as printed above.

The matrix $A$ contains two zero-columns which correspond to the variables $x$ and $\beta$. The rows in $B$ which correspond to these two variables contain also only zeros. The only relationships which are described, are $\dot{x} = \dot{x}$ and $\dot{\beta} = \dot{\beta}$. One could therefore conclude that the state-space variables $x$ and $\beta$ are redundant and can be eliminated in the state-space representation. There are however two reasons to contain the above representation. First, $x$ is the variable which has to track a reference-signal, so it is not logical to eliminate it from the equations. Secondly, $\beta$ is an angle which can be measured relatively easy. When eliminating this variable, as will appear in the next paragraph, it is needed to measure $\alpha$, $\dot{\alpha}$ or $\dot{\beta}$. This will give more problems than not eliminating $\beta$, so also $\beta$ will be a part of the state vector.
2.3. Controllability and observability

As could be expected, the system is unstable by itself. Two of the poles are in the right half plane and two poles are at the origin. Stabilizability and detectability should therefore be analyzed.

Except from calculating the controllability matrix \( P \) and the observability matrix \( Q \) as pointed out in [9], this can be done by determining the eigenvalue decomposition of the system matrix \( A \). The system can then be decoupled, which leads to more insight in the variables needed to make the system observable (and thus detectable).

The eigenvalue decomposition has the following form:

\[
A = X S X^{-1}
\]

(2.13)

The matrix \( S \) contains all the eigenvalues on the diagonal. The matrix \( X \) contains the corresponding eigenvectors.
Adapting the system

\[
\dot{z} = Az + Bu
\]

\[
y = Cz
\]

using equation (2.13), leads to

\[
\dot{c} = Sc + (X^{-1}B)u
\]

\[
y = (CX)c
\]

with

\[
c = X^{-1}z
\]

(2.14)

(2.15)

When the matrix \( X^{-1}B \) does not contain any zero-rows, the system will be controllable. To get the system observable, the matrix \( C \) should be chosen in such a way that the matrix \( CX \) does not contain any zero-columns. Note that when two eigenvalues are the same, the two corresponding rows of \( X^{-1}B \) have to be different to guarantee controllability. Likewise, the two corresponding columns of \( CX \) have to be different to guarantee observability.

Calculating \( X \) gives

\[
X = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & -0.0045 & 0.0281 & -0.1038 \\
0 & 0 & 0 & 0 & 0 & 0.0113 & 0.2783 & -0.3154 \\
0 & 0 & 0.0002 & -0.2453 & -0.2454 & 0 & 0 & 0 \\
0 & 1 & -0.0108 & 0.0103 & -0.0112 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.3669 & 0.0965 & 0.2949 \\
0 & 0 & 0 & 0 & 0 & -0.9302 & 0.9552 & 0.8960 \\
0 & 0 & -0.0206 & -0.9686 & 0.9683 & 0 & 0 & 0 \\
0 & 0 & 0.9997 & 0.0405 & 0.0442 & 0 & 0 & 0
\end{bmatrix}
\]

(2.16)

Calculating \( X^{-1}B \) results in a matrix with no zero-rows and with two different rows corresponding to the double eigenvalue 0. Therefore, the system is fully controllable. The two eigenvalues of 0 correspond to the variables \( x \) and \( \beta \). This can be explained by the fact that the dynamic behavior of the system is independent of these two variables. This did already become clear when looking at the state-
space representation of the system on page 11. The starting position at a step response, for example, may be 4, 23.7 or 64.1 (m), the response will still be the same. When looking at $\dot{\beta}$, the angle itself is not important for the dynamic behavior, but its time-derivative is.

Knowing that $CX$ should not contain any zero-columns leads to the conclusion that at least $x$ and $\dot{\beta}$ have to be measured. Because the intention is to let follow $x$ a certain reference-signal $r$, measuring $x$ is certainly a very logical action, but it can give some problems in practice. It is not very easy to measure $x$ without having the influence of the angle $\varphi$ in it. This problem could be solved by a independent measurement of $\varphi$, because $i\varphi = \gamma_r - \varphi$ with $\varphi$ the rotor-rotation of the lower motor with respect to its housing. This angle $\varphi$ is relatively easy to measure. In [1] they use a camera to measure the value of $x$, but this does not seem to be a very cheap and easy option. In the rest of this report however, the assumption is made that the value of $x$ is measurable and thus known. Additional measuring of $\varphi$ does also take place, mainly because measuring this angle does not encounter big problems.

All this leads to the following matrix $C$

$$
C = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & -5 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

(2.17)

Note that the analyzed controllability and observability only applies to the linearized system, but that in all likelihood it also applies to the non linear system when not drifting to much from the vertical position.

Checking the observability double can be done by calculating the matrix

$$
Q = \begin{bmatrix}
C \\
CA \\
\vdots \\
CA^5
\end{bmatrix}
$$

(2.18)

This matrix should have a rank of 8 to guarantee observability. Computing the rank with Matlab results in a matrix $Q$ with a rank of 8. Observability is thus, as proved before, guaranteed by choosing $C$ as printed in (2.17).

Checking controllability by computing

$$
P = \begin{bmatrix}
B & AB & \ldots & A^6B
\end{bmatrix}
$$

(2.19)

gives a rank of 8, so, as also proved before, the system is fully controllable.
CHAPTER 3

Controller design

The controller design is based on the theory presented in [10] and took place using the MHC-toolbox, see [11]. For this, the setup should therefore be brought in the standard form of Fig 2.

![Diagram of controller design](image)

**FIGURE 2: The augmented plant**

$G$ is the generalized plant and includes all the weight functions, $K$ is the controller. The column $v$ contains all the external inputs (like reference-signals, disturbances and sensor noise) and the column $w$ contains all the variables which have to be minimized. Furthermore, $y$ is the observation vector and $u$ is the control input.

The goal is to minimize the transfer from $v$ to $w$. For that purpose, a control structure has to be defined before it is possible to adjust several weights and to compute the $H_\infty$ norm of the problem. Using the MHC-toolbox in this process has the advantage that this only has to be done once. When this is done, one can adjust the weights iteratively.

The generalized plant can be written as (see also [10])

$$
\begin{bmatrix}
  w \\
  y
\end{bmatrix} =
\begin{bmatrix}
  G_{11} & G_{12} \\
  G_{21} & G_{22}
\end{bmatrix}
\begin{bmatrix}
  v \\
  u
\end{bmatrix}
\text{with } u = Ky
$$

(3.1)

From this, a direct relationship between $z$ and $w$ can be derived, namely

$$
w = M(K) v \quad \text{with} \quad M = G_{11} + G_{12} K (I - G_{22} K)^{-1} G_{21}
$$

(3.2)

The objective of the $H_\infty$ design method is to find a stabilizing controller which minimizes the $H_\infty$ norm of $M$. In formula

$$
\min_{K \text{ Stabilizing}} \|M(K)\|_{\infty} = \min_{K \text{ Stabilizing}} \sup_{\|w\|_2, \|v\|_2} \|w\|_2
$$

(3.3)

The controller has to satisfy four requirements:
- tracking of the reference signal $r$ by $x$
- reduction of the influence of model errors
- sensor noise rejection
- avoiding actuator saturation

This leads to a mixed sensitivity problem as described in [10], in which a compromise has to be found between two competitive groups. The first group concerns the first two requirements and is centered around the sensitivity, while the second group, concerning actuator saturation and sensor noise...
rejection, is centered around the complementary sensitivity. The $H_\infty$ based design method is now used to loop shape several (complementary) sensitivity functions of the closed loop plant by defining weights on the inputs and outputs.

When choosing the weight functions, several things have to be kept in mind. The order of the controller depends on the order of the weight functions, so this order should not be chosen too high to avoid high order controllers. It will also result in less numerical problems with the MHC-toolbox. This is also the main reason to chose the order of the denominator the same as the order of the numerator for the weights. Furthermore, the weight functions have to be stable by itself.

The extended plant ($P_e$) is shown in Fig 3 and 4

![Figure 3: The nominal plant $P_n$ with model-errors](image)

![Figure 4: The extended plant $P_0$](image)

Two different kinds of external inputs are introduced next to the reference signal $r$. The column $n$ will contain the sensor noises, while the column $d$ will account for model errors. The indices $d$ and $n$ at a variable indicate therefore whether one of these disturbances has been brought into account for the considered variable.

The sequence of the elements in the columns $d$ and $n$ is as follows

$$d = [d_x \ d_\phi \ d_\alpha \ d_\beta \ d_{\varphi_{uv}}]^T$$

$$n = [n_x \ n_\beta \ n_{\varphi_{uv}}]^T$$

(3.4)

The model errors are introduced for design purposes and have to charge for errors which are the result of the linearization and errors due to parameter uncertainties.
When looking at the tracking performance for the reference position $r$, the sensitivity function $S_{e_d r}$ has to be shaped. This function describes the transfer from $r$ to $e_d$ and should ideally be small for all frequencies. As stated, this is not possible due to the specifications concerning actuator saturation and sensor noise rejection. Therefore, the minimization of the sensitivity takes mainly place at the low frequencies, taking into consideration the bandwidth of the system. The specification to be met is

$$\left\| W_{e_d} S_{e_d r} V_r \right\|_{\infty} < 1$$  \hspace{1cm} (3.5)

This means that the combination of the weights $W_{e_d}$ and $V_r$ has to weight heavy for the lower frequencies. Note that the intention of $V_r$ is to characterize $r$ and that the requirements for minimization have to be fulfilled by determining $W_{e_d}$.

Another goal is to limit the torque's which are exerted by the motors. Especially the lower motor is taken into consideration, because when looking at the tracking requirement, this motor has to deliver the biggest achievements. Therefore, the transfer from $r$ to $u_1$ should be minimized. This transfer is the control sensitivity $R_{u_1 r}$, which can be considered as a weighted version of the complementary sensitivity with the system transfer $P$ as a 'fixed' weighting (see also [10]). Because the most actuators have a frequency-characteristic which makes clear that they can not deliver big achievements at high frequencies, the transfer should be made small for especially high frequencies. This means that the combination of the weights $W_{u_1}$ and $V_r$ should weight 'heavy' for high frequencies to meet the specification

$$\left\| W_{u_1} R_{u_1 r} V_r \right\|_{\infty} < 1$$  \hspace{1cm} (3.6)

As mentioned, one of the goals is to reduce the influence of model errors due to linearization and inaccuracy of the system parameters. Considering these errors as multiplicative feedback errors on the output, the requirements lead to minimization of the sensitivity functions $S_{e_d \theta}$, $S_{\dot{e}_d \theta}$, and $S_{\ddot{e}_d \theta}$. These transfers should be small for low frequencies. High weights for low frequencies could therefore be expected. However, when looking at the linearized model of (2.10), one can find out that the transfers $S_{\dot{e}_d \theta}$ and $S_{\ddot{e}_d \theta}$ are fixed at low frequencies and that the controller can not influence them. This becomes clear when adding $R$ times the first differential equation to the second one. The result is then an equation with the following form

$$c_1 \ddot{x} + c_2 \theta - \varphi = 0$$  \hspace{1cm} (3.7)

In this, $c_1$ and $c_2$ are positive constant values. From this, it can easy be derived that for $\omega \rightarrow 0$, the transfer $S_{\dot{e}_d \theta} \rightarrow 1$. Also can be seen in (2.10) that the third differential equation has a similar structure, namely

$$c_3 \ddot{\alpha} + c_4 \dot{\beta} - \alpha = 0$$  \hspace{1cm} (3.8)

So also $S_{\dot{e}_d \alpha} \rightarrow 1$ when $\omega \rightarrow 0$. Now it is clear that it makes no sense to set heavy weights on these two transfers for low frequencies. Therefore, two constant weights will be applied to these transfers. Minimization of the sensitivity function $S_{e_d \alpha}$ for low frequencies does already take place by the implementation of a heavy weight on $e_d$ for low frequencies. This was necessary to meet the tracking requirement.

The influence of the sensor noises on $e_d$, $\varphi_d$ or $\alpha_d$, can be represented in terms of complementary sensitivities and the influence on $u_1$ and $u_2$ in terms of the control sensitivities. However, the most important weights on the outputs have already been characterized and some of them take implicitly
the rejection of sensor noise into account. The transfers will therefore not be analyzed separately and the influence of the sensor noise will be considered small in relation to other errors, especially when measuring $\beta$ and $\varphi_m$. Their real influence will be investigated during the simulations.

A schematic representation of the implementation of the extended plant of FIG. 4 and the weight matrices $V$ and $W$ in the MHC-toolbox, can be seen in FIG. 5.

![Diagram](image)

FIGURE 5: Implementation of $P_o$ in MHC

In this, the choice of the observation vector $y$ partially determines the complexity of the controller. At least $x$ and $\beta$ have to be observed (see paragraph 2.3), but one could decide to extend $y$ by other elements. In this case, the vector will be extended with $\varphi_m$, because it is relative easy to measure this angle in practice. Also the choice has to be made whether to couple back the tracking-error $e=x-r$ (one degree-of-freedom controller) or the elements $x$ and $r$ separately (two degree-of-freedom controller). Both situations will be compared later, but in first instance there is decided to work with the latter situation, mainly because this gives better results in general. The observation vector $y$ therefore contains the elements $x_{dn}$, $\beta_{dn}$, $\varphi_{m,\alpha}$, and $r$.

As can be seen in FIG. 5, the generalized plant consists of a combination of the extended plant $P_o$ and the weight matrices $V$ and $W$. These weight matrices will only have elements on the diagonal. The elements of $V$ characterize the input signals while the elements of $W$ are used to meet the specifications as worked out on the previous page. Problem is now, that the weight functions have to be determined in the frequency domain, while the most important specifications will be set in the time domain.

The input-signals which have to be characterized are the reference signal $r$ and the elements of $n$ and $d$. As stated, the order of the chosen functions should be as low as possible for as long as this gives good results. The weights on the elements of $d$ are therefore all set to an equal constant value. The weights on the elements of $n$ are also chosen constant, but because the two angles $\beta$ and $\varphi_m$ can be measured more accurately as $x$, they get a smaller weight. Because the reference signal $r$ will be a step, one could consider a weight like $1/s$ (the representation of $r$ in the frequency domain). Here however, $r$ will be weighted with (again) a constant value. Such a weight corresponds to the one in [2] and it gives good results as will be shown in CHAPTER 4.

As stated, the controller should meet some specifications in the time domain. The goal is to meet the following objectives as a result on the dynamic response on a stepwise change in $r$:
- rise time of 4 s
- no overshoot
- values of $u_1$ and $u_2$ beneath 75 Nm.
- small deviations of the vertical position to avoid large model errors, because the design is based on a linearized model.

These specifications are based on the specifications which are used in [2].
When designing the elements of $W$ and $V$, the start was mostly based upon the functions out of [2]. Because of the many differences between the system in [2] and the system here, this did not give good results. Responses in time domain and the closed loop transfers of the system were the results on which a lot of adjustments took place, and the final choice of the weight functions ended up as in the following matrices:

$$W = \begin{bmatrix}
\frac{s+.001}{s+20} & 0 & 0 & 0 \\
0 & \frac{1}{100} & 0 & 0 \\
0 & 0 & \frac{2s+5}{s+.001} & 0 \\
0 & 0 & 0 & \frac{1}{4}
\end{bmatrix}$$

$$V = \begin{bmatrix}
\frac{1}{100} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{100} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{100} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{100} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{100} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{1000} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{1000} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2}{3}
\end{bmatrix}$$

(3.9)

(3.10)

When computing the controller with the MHC-toolbox, using the presented linearized system of (2.12) and the above matrices $V$ and $W$, an error message will occur. The reason for this is that two poles of the linearized system are zero. It is therefore necessary to slightly move these poles to the left half plane. This is done by setting the $(5,1)$-and $(8,4)$-entries of the system matrix $A$ to $10^{-3}$. This leads to a shifting of all the poles by even less than $10^{-3}$.

As can be seen in [10], the calculation of the controller is based on solving Riccati-equations. Solving the equations in this case results very often in numerical problems. Small changes in the weight functions can even lead to non-stabilizing controllers. Balancing the plant, which is an option in the MHC-toolbox, does not solve the problems. Also additional balancing of $V$ and $W$ does not stop the warnings concerning numerical accuracy. One should therefore be aware of these problems when adjusting the several parameters.

As stated before, the closed-loop transfers of the system (with controller) have also useful information for design purposes. Because the matrices $V$ and $W$ contain only diagonal elements, the MHC-toolbox plots the closed-loop transfers with the bounds. These bounds depend on the several weights. Once a controller is designed, one can see which contribution a certain transfer between an input and an output has in the $H_\infty$-norm. This implies how sensitive the design procedure is for changes of individual weight functions at certain frequencies. Based on this information, decisions can be made whether it is relevant to adjust specific parameters.
CHAPTER 4

Results

The calculated controller is based on the linearized model. As stated, the simulations in the time domain will be executed using the non-linear model of the unicycle. The step-amplitude for $r$ is set on 1 (m). Simulating with no disturbances, using the files of APPENDIX II, results in

As can be seen, all the specifications are met. As could be expected when studying the physical behavior of the system, the response is an inverse one. In first instance, the unicycle will move in negative direction, which results in a positive angle $q$. One could compare this to a broomstick balancing on a hand. When moving the hand to e.g. the left direction, the broomstick will tend to fall into the right direction.

Comparing the results with the results in [2] gives a distorted picture. The parameters (mainly the inertia's) are different and the system is extended with the rotating disk. Furthermore, they use in [2] a step amplitude of 5 (m), while here a step amplitude of 1 (m) is used. When upgrading this amplitude here, the limit is approximately 2 (m) before the unicycle becomes unstable.

Changing the parameters of the system gives different results for the individual parameters. The controller is very robust for changes in the inertia's, which can be up to 50 % for the wheel and up to 100 % for the frame before the unicycle becomes unstable. The exact position of the center-of-mass, represented by $h$, has also a margin of about 50 %. Changes in mass can be made up to 15 % for the

\[ \text{FIGURE 6: Step response results; No disturbances; } r=1 \text{ (m)} \]
frame and even higher for the wheel. Bottleneck are the coefficients of friction, $\mu_t$ and $\mu_r$. Even slight changes of 2% give problems and unstable behavior. This is a little bit annoying result, because determine these parameters accurately in practice is difficult and the used friction model is only an approximation.

This result on robustness does raise the question whether an $H_\infty$-controller is the right controller for an unicycle. It is very obvious that the difficulties with friction are not considered enough during the design of the controller. One could decide to define extra inputs and outputs to the general plant to include errors in the friction model more accurate. This is possible, but makes the design more complicated and it then seems a logical step to change over to $\mu$-analysis/synthesis (see also [10]). This way of designing is based on the same robustness concept, but can be considered as an extended version in which the $H_\infty$ design is a (good) first step. Improvement is not guaranteed, but in most cases it will give better results, because the individual model errors are taken into account more accurately.

Applying the controller to the linear system does not give responses very different from the non-linear case, at least for a step amplitude of 1 (m), see Fig. 7. Increasing the amplitude to 2 (m) however leads to responses which differ more from each other (Fig. 8). Increasing the step amplitude even more, gives unstable responses for the non-linear model. This is the result of the big model errors which occur because of the differences between the non-linear model used for simulations and the linear model used for controller design.

![Figure 7: Step-responses for the linear (dotted) and the non-linear case; $r = 1$ (m)](image1)

![Figure 8: Step-responses for the linear (dashed) and the non-linear case; $r = 2$ (m)](image2)

As stated is the intention not only to let $x$ track $r$ and to stabilize the controller in longitudinal direction, but also to stabilize the controller in lateral direction. To investigate the possibilities of the controller, a force of 40 (N) is applied to the system during .01 (s). The direction of the force is in the
e_2-direction and it is exercised on the center-of-mass of the frame. In file V of APPENDIX II can be seen how the force is implemented in the non linear model. In first instance, the controller seems to stabilize the system adequately. The actuator does not get saturated, which of course is the result of the chosen force-amplitude and the grasping time. The results can be seen in FIG 9.

Problem is that it is not clear whether u_2 is stabilizing itself or that it starts to oscillate. To investigate its behavior on longer term, another simulation is carried out. Note that the time during which the force grasps on the unicycle is doubled because of simulation purposes. This results in a larger amplitude of the involved variables. The results however are fully comparable qualitatively. When not doubling the grasping-time, the routine ode45 in Matlab will 'miss' the push, because its initial stepwidth depends on the simulation time and becomes too big.

FIGURE 9: Simulation results for a lateral 'push' to the unicycle of 40 (N) during .01 (s)

FIGURE 10: Simulation results for a lateral 'push' to the unicycle of 40 (N) during .02 (s)
Now can be seen that the unicycle is unstable on long term. Attempts to get the diverging oscillating behavior away by using 'heavier' weights on the relevant frequencies, encounter numerical problems during the design with MHC or result in big concessions to the tracking of $r$.

Simulations with the linear model as presented in (2.12) give the same, unstable result, which is very strange in first instance, because all the poles of the closed loop system are situated in the left half plane and the system is proved to be fully controllable. Looking at the poles, the poles nearest to the imaginary axis are $-10^{-4} \pm 2*10^{6}i$.

Reason for the unstable behavior could be the fact that design took place based on the presented linear system in (2.12) with a slight disturbance of two entries in $A$. This led to a small movement of the system poles with less than $10^{2}$, which seemed neglectible. Computing the poles of the closed loop system with the controller designed and the linear system of (2.12), gives two poles in the right half plane (namely $0.1205 \pm 0.5746i$) and explains the unstable behavior. Apparently, the slight changes in the poles of the system have more impact as would be expected. The controller is not robust enough to deal with this problem and the design method or structure should therefore be changed.

A simulation with the adjusted linear system indeed gives a stable respons as can be seen in Fig 11.

![Figure 11: Simulation results with the adjusted linear system](image)

Note that although the angle $\alpha$ only makes a small change, the motor-torque has already passed its limit. Using a transmission between disk and upper motor axis is a possibility to increase the dynamic influence of the disk within the same torque-bounds and thus to deal with bigger deviations of the upright position in lateral direction.

Adding sensor noise to the system gives the responses of Fig 12 (see next page). The sensor noise is considered as a white noise with a maximum amplitude of 0.01 for $x$ and a maximum amplitude of 0.005 for the angles $\varphi$, $\psi$, and $\beta$. As one can see, the implementation of the noise does not give undesirable problems for the responses of $x$, $\varphi$ or $\alpha$. The biggest influence of the sensor noises can be seen at the angle $\alpha$ with values up to 0.04 rad. The deviations of the obtained position of $x$ stay within 1% of its value.

Lastly, the results for a controller with an one degree-of-freedom structure are compared to the results for the controller with a two degree-of-freedom structure. Of course, the same weight functions are used. Adjusting the generalized plant is very trivial and will not be worked out. Similar to the situation in [2], it does not seem to matter which structure is used. The results are the same as far one should notice the plots and they are therefore not presented in this report. A clear explanation is not found. There is for example no closed loop transfer towards $u_I$ really dominant for certain frequencies.
This would have implied that there is hardly any room for profit when expanding to a two degree-of-freedom structure.

FIGURE 12: Simulation results with sensor noise
CHAPTER 5

Conclusions and recommendations

Conclusions

The unicycle as printed in FIG 1 is fully controllable and the rotating disk should therefore be suited for stabilization of the unicycle in the lateral direction. Using a $H_\infty$ design procedure encounters numerical problems, at least when using the MHC-toolbox. The specifications set in time domain for a step response are met, but disturbances in lateral direction lead to unstable behavior. This is the result of a slight movement of the poles of the linearized system which has to be made for design purposes. Robustness when changing the system parameters is good, except for changes in the coefficients of friction. Because robustness of these coefficients is very important, one could conclude that the designed controller is not robust enough. Lastly, there is hardly any difference between a controller based on an one degree-of-freedom structure and a controller based on a two degree-of-freedom structure.

Recommendations

In future research, more attention should be given to model errors due to friction. For this purpose, a robust controller based on the $\mu$ analysis/synthesis-theory or an adaptive controller seems more suitable. When still using an $H_\infty$ design method, one should search for a way to avoid the problems due to the slight changes of the poles. Because measuring the position of the wheel-axis may give problems in practice, ways should be investigated to avoid this problem. Lastly, one could possibly look for a way to stabilize the unicycle in lateral direction based on moving the center-of-mass of a body. This will give a more complex model, but can perhaps give better lateral stabilization performance with a reduction of the total mass of the unicycle.
References

[1] F. Takemori and Y. Okuyama  
Stabilizing control of a mono-cycle based on $H_\infty$ control theory,  
Proc. of the Asian Control Conf., Vol. 1  

Multivariable $H_\infty$ tracking control of a mono-cycle,  
Faculty of Mechanical Engineering, Eindhoven University of Technology,  
November 24, 1995

Study on the Stability and Motion Control of a unicycle,  

Dynamics and nonlinear adaptive control of an autonomous unicycle: theory and experiment  
Proc. of the 29th IEEE Conference on Decision and Control, Vol. 1  
Hawaii, December 1990, pp. 182-187

Course on advanced control, Lecture notes,  
Faculty of Mechanical Engineering, Eindhoven University of Technology, 1993  
Nr. 004708-219951

[6] Trom, Jeffrey Dean  
Automated linearization of nonlinear coupled differential and algebraic equations,  
Thesis Iowa State University, 1990  
U.M.I. Dissertation Service, Order Number 9101382

[7] A. Sauren  
Multibody dynamica, Lecture notes,  
Faculty of Mechanical Engineering, Eindhoven University of Technology, 1991  
Nr. 004659-300005

[8] O.L.R. Jacobs  
Introduction to Control Theory  
Oxford University Press 1993

Werktuigkundige regeltechniek II, Lecture notes  
Faculty of Mechanical Engineering, Eindhoven University of Technology, 1990  
Nr. 004494-200002

[10] Ad Damen and Siep Weiland  
Robust Control, Lecture notes  
Department of Electrical Engineering, Eindhoven University of Technology, 1995

Multivariable $H_\infty$ control design toolbox: user manual  
Department of Electrical Engineering, Eindhoven University of Technology, 1994  
EUT-report; 94-E-282

[12] Anton Stoorvogel  
The $H_\infty$ control problem, A state space approach  
APPENDIX I

Derivation of the kinetic energy

The kinetic energy of the system can be divided into two parts, translational and rotational.

I-a: The translational part

This part of the kinetic energy describes the energy as result of the movement of the center-of-mass of a part. As stated, the unicycle consists of four bodies: the wheel, the frame and two motors with a disk fixed on the upper one. When looking at the movements of these bodies, it is clear that the frame and the two motors can be considered as a whole.

The translational part of the kinetic energy can be calculated by

\[ T_{\text{trans}} = \frac{1}{2} M \ddot{r}_M^T \ddot{r}_M \] (I-1)

In this, \( r_M \) is the position vector of the center-of-mass of the considered body. Looking at the unicycle as in Fig 1, the relevant positions can be deduced from it.

This leads to (in column-representation)

\[
\begin{align*}
\dot{r}_W &= \begin{bmatrix} x \\ R \sin \alpha \\ R \cos \alpha \end{bmatrix} \\
\dot{r}_F &= \begin{bmatrix} x + h \cos \alpha \sin \phi \\ (R + h \cos \phi) \sin \alpha \\ (R + h \cos \phi) \cos \alpha \end{bmatrix}
\end{align*}
\] (I-2)

Here, \( W \) indicates the wheel and \( F \) indicates the frame with the two motors.

Combining (I-1) and (I-2) gives

\[
T_{\text{trans}, W} = \frac{1}{2} M_W \left( \dot{x}^2 + R^2 \dot{\alpha}^2 \right)
\] (I-3)

\[
T_{\text{trans}, F} = \frac{1}{2} M_F \left\{ \begin{array}{l}
\dot{x}^2 + \left( h^2 \cos^2 \alpha \cos^2 \phi + h^2 \sin^2 \phi \right) \dot{\phi}^2 + \\
\left( h^2 \sin^2 \alpha \sin^2 \phi + (R + h \cos \phi)^2 \right) \dot{\alpha}^2 + \\
(2h \cos \alpha \cos \phi) \dot{x} \dot{\phi} - (2h \sin \alpha \sin \phi) \dot{\alpha} \dot{\phi} - \\
\left( \frac{1}{2} h^2 \sin(2\alpha) \sin(2\phi) \right) \dot{\alpha} \dot{\phi}
\end{array} \right\}
\] (I-4)
I-b: The rotational part

When looking at the rotations of the centers-of-mass of the several bodies, all the bodies have to be considered separately. It applies (according to [7])

\[ T_{\text{rot}} = \frac{1}{2} \vec{\omega}^T \cdot J \cdot \vec{\omega} \]  

(I-5)

In this, \( \vec{\omega} \) is the angle-velocity vector and \( J \) is the inertia-tensor, which can be written as

\[ J = \vec{e}_l^T J \vec{e}_l \]  

(I-6)

Here, \( \vec{e}_l \) is the local vector-basis of the considered body and turns with the body. \( J \) is a matrix which contains all the inertia's around the local axis on the diagonal.

The angle-velocity vector can be determined using the following relationships.

\[ \vec{e}_l = R \vec{e}_o \]
\[ \dot{\vec{e}}_l = \dot{R} R^T \vec{e}_l = \vec{\omega} \times \vec{e}_l, \quad \text{with} \quad \vec{\omega} = \vec{e}_l^T \vec{\omega} \]  

(I-7)

Now the energy can be computed for the several bodies.

The wheel

\[ R_w = R_2 R_1 \]
\[ R_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix}, \quad R_2 = \begin{bmatrix} \cos \left( \frac{\alpha}{2} \right) & 0 & -\sin \left( \frac{\alpha}{2} \right) \\ 0 & 1 & 0 \\ \sin \left( \frac{\alpha}{2} \right) & 0 & \cos \left( \frac{\alpha}{2} \right) \end{bmatrix} \]

\[ \vec{\omega}_w = \begin{bmatrix} \dot{x} \\ -\frac{\dot{x}}{R} \\ \dot{\alpha} \sin \left( \frac{\alpha}{2} \right) \end{bmatrix} \quad \text{and} \quad J_w = \begin{bmatrix} J_{w1} & 0 & 0 \\ 0 & J_{w2} & 0 \\ 0 & 0 & J_{w1} \end{bmatrix} \]

This results in

\[ T_{\text{rot, w}} = \frac{1}{2} J_{w1} \alpha^2 + \frac{1}{2} J_{w2} \left( \frac{\dot{x}}{R} \right)^2 \]  

(I-8)
The frame

\[
R_F = R_3 R_1 = \begin{bmatrix}
\cos \varphi & 0 & -\sin \varphi \\
0 & 1 & 0 \\
\sin \varphi & 0 & \cos \varphi
\end{bmatrix}
\]

\[
R_3 = \begin{bmatrix}
\dot{\alpha} \cos \varphi \\
-\dot{\phi} \\
\dot{\alpha} \sin \varphi
\end{bmatrix}
\text{ and } J_F = \begin{bmatrix}
J_{F1} & 0 & 0 \\
0 & J_{F1} & 0 \\
0 & 0 & J_{F2}
\end{bmatrix}
\]

This results in

\[
T_{rot,F} = \frac{1}{2} J_{F1} \left( \dot{\alpha}^2 \cos^2 \varphi + \dot{\phi}^2 \right) + \frac{1}{2} J_{F3} \dot{\alpha}^2 \sin^2 \varphi
\]

(I-9)

The disk and the upper motor axis

\[
R_S = R_4 R_F
\]

\[
R_4 = \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \beta & -\sin \beta \\
0 & \sin \beta & \cos \beta
\end{bmatrix}
\]

\[
\omega_S = \begin{bmatrix}
\dot{\beta} + \ddot{\alpha} \cos \varphi \\
-\dot{\phi} \cos \beta - \ddot{\alpha} \sin \varphi \sin \beta \\
-\dot{\phi} \sin \beta - \ddot{\alpha} \sin \varphi \cos \beta
\end{bmatrix}
\text{ and } J_S = \begin{bmatrix}
J_{S1} & 0 & 0 \\
0 & J_{S2} & 0 \\
0 & 0 & J_{S3}
\end{bmatrix}
\]

This results in

\[
T_{rot,S} = \frac{1}{2} J_{S1} \left( \dot{\beta}^2 + 2 \dot{\beta} \ddot{\alpha} \cos \varphi + \ddot{\alpha}^2 \cos^2 \varphi \right) + \frac{1}{2} J_{S2} \left( \dot{\phi}^2 + \ddot{\alpha}^2 \sin^2 \varphi \right)
\]

(I-10)

The lower motor-axis

\[
R_M = R_5 R_F
\]

\[
R_5 = \begin{bmatrix}
\cos \varphi_M & 0 & -\sin \varphi_M \\
0 & 1 & 0 \\
\sin \varphi_M & 0 & \cos \varphi_M
\end{bmatrix}
\]

\[
\omega_M = \begin{bmatrix}
\dot{\alpha} \cos \varphi \cos \varphi_M - \ddot{\alpha} \sin \varphi \sin \varphi_M \\
-\dot{\phi} - \dot{\phi}_M \\
\dot{\alpha} \sin \varphi \cos \varphi_M + \ddot{\alpha} \cos \varphi \sin \varphi_M
\end{bmatrix}
\text{ and } J_M = \begin{bmatrix}
J_{M1} & 0 & 0 \\
0 & J_{M2} & 0 \\
0 & 0 & J_{M3}
\end{bmatrix}
\]

This results in

\[
T_{rot,M} = \frac{1}{2} J_{M1} \ddot{\alpha}^2 + \frac{1}{2} J_{M2} \left( \dot{\phi} + \dot{\phi}_M \right)^2
\]

(I-11)
In (I-11), \( \varphi_M \) is the rotation of the motor axis with respect to its housing. Because

\[ i \varphi_M = \frac{x}{R} - \varphi \quad \text{(I-12)} \]

equation (I-11) can be written as

\[ T_{rot,M} = \frac{1}{2} J_{M1} \dot{\alpha}^2 + \frac{1}{2} J_{M2} \left( \left( \frac{1}{\tau_h} \right) \dot{\phi} + \left( \frac{1}{\tau_h} \right) \dot{\chi} \right)^2 \quad \text{(I-13)} \]

Now the total kinetic energy of the unicycle can be written as

\[
\begin{align*}
T_{\text{kin}} &= T_{\text{rot}} + T_{\text{trans}} \\
T_{\text{rot}} &= T_{\text{rot},W} + T_{\text{rot},F} + T_{\text{rot},X} + T_{\text{rot},M} \\
T_{\text{trans}} &= T_{\text{trans},W} + T_{\text{trans},F}
\end{align*}
\quad \text{(I-14)}
\]
APPENDIX II

Simulation files

File I:

% This file is a simulation file for simulating the non linear, dynamic behavior of an unicycle

%--------Starting conditions

ll=length(Ac(:,1));
z0=[0;0;0;0;0;0;0;zeros(ll,1)];
global Ac Bc Cc ll

%--------Simulation

[T,Z]=ode45('system',0,8,z0,1e-3);
U=(Cc*(Z(:,9:(ll+8))'))'
File II:

% This file is the function file which describes the non-linear, dynamic behavior of an unicycle
% in the following structure
%  %
%  zdot=function(t,z,u)
%  %
%  The equations of motion derived for this purpose are
%  %
%  M*qpp+B*qp+C=H*u
%  %
%  In this, qp is a column with the time-derivatives of the generalize coordinates,
%  and qpp is the time-derivative of qp.
%  u is a column with the actuator entries of the system. Its value depends on the calculated
%  controller, which is represented in a state-space representation with Ac, Bc, Cc and Dc.

function zdot=systeem(t,z,u);

global Ac Cc Bc ll

%-------The reference signal r
r=1;
t

%-------The parameters of the system
mw=1.5; mf=8.5; R=.2; h=4;
i=1; g=9.81; nu1=6.33; nu2=6.33;
jw1=0.030; jw2=0.060; jm1=0; jm2=0.001;
jst=0.070; jst1=0.035; mu=2.12*10^(-3);
jf1=0.180; jf2=0.180; jf3=0.045;

%-------Defining the assisting parameters
ja=jw2/(R^2+jm2/(R^2*i^2);
jb=(i-1)/(R*i^2)*jm2;
je=jf2+js2+(i-1)/i^2*jm2;
jd=jw1+jm1;
jg=js2+jf3;
jh=jf1+jf3-js1+js2;

%-------Set the elements of the state-space vector to variables
x=z(1); p=z(2); a=z(3); b=z(4);
xp=z(5); pp=z(6); ap=z(7); bp=z(8);

%-------Calculation of y and u
y=[x;b;5*x-p;r];
u=Cc*z(9:(i+8));

%-------Calculation of the system matrices M, B, C and H
m11=mw+mf+ja;
m21=jb+mf*h*cos(a)*cos(p);
m31=-m*h*sin(a)*sin(p);
\[
m_{22} = 1 + mf^*h^2*(1 - \sin(a)^2*\cos(p)^2);
\]
\[
m_{32} = -2*mf^*h^2*\sin(2*a)*\sin(2*p);
\]
\[
m_{33} = mf^*(h_2^2 + R^2) + 2*mf^*h_2^2*\sin(p)^2 + jd + je*\cos(p)^2 + jg*\sin(p)^2;
\]
\[
m_{43} = jl^*\cos(p);
\]
\[
m_{44} = js^1;
\]
\[
M = [m_{11}, m_{21}, m_{31}, 0; m_{21}, m_{22}, m_{32}, 0; m_{31}, m_{32}, m_{33}, 0, 0, m_{43}, m_{44}];
\]
\[
c_{2} = -mf^*g*\cos(a)*\sin(p);
\]
\[
c_{3} = -mw*R^*g*\sin(a) - mf^*(R + h^*\cos(a))^*\sin(a)*g;
\]
\[
C = [0; c_2; c_3];
\]
\[
b_{11} = 1/R^2*(nu_1/l^2 + mu);
\]
\[
b_{21} = 1/R*(nu_1/l^2 + mu);
\]
\[
b_{43} = js^1*\sin(p)*pp;
\]
\[
b_{12} = mf^*h^*\cos(a)*\sin(p)*pp + b_{21};
\]
\[
b_{22} = 5*mf^*h^2*\sin(2*p)*\sin(a)^2*pp - b_{21}*R;
\]
\[
b_{32} = 5*hf^2*mf^*h^2*\sin(a)*\sin(p)^2*pp;
\]
\[
b_{13} = -mf^*h^*\cos(a)*\sin(p)*ap - 2*mf^*h^*\sin(a)*\cos(p)*pp;
\]
\[
b_{23} = (mf^*R^*h^*\sin(p) + 5*mf^*h^2*\sin(a)^2*\sin(2*p) - 5*\sin(2*p)*h)*ap + (-mf^*h^2*\sin(2*a)*\cos(p)^2)*pp;
\]
\[
b_{33} = 5*mf^*h^2*\sin(2*a)*\sin(p)^2*ap + (mf^*h^2*\sin(a)^2*\sin(2*p)^2 - 2*mf^*(R + h^*\cos(p))^*h^*\sin(p) + (g - je)^*\sin(2*p))*pp;
\]
\[
b_{24} = js^1*\sin(p)*ap;
\]
\[
b_{34} = js^1*\sin(p)*pp;
\]
\[
b_{44} = nu_2;
\]
\[
B = [b_{11}, b_{12}, b_{13}, 0; b_{21}, b_{22}, b_{23}, b_{24}, 0, b_{32}, b_{33}, b_{34}, 0, 0, b_{43}, b_{44}];
\]
\[
H = [1/(i*R), 0; -1/i, 0, 0, 0, 0, i];
\]
\%
%----------Set the equations of motion to the correct structure
\%
M_{ster} = [eye(4), zeros(4, 4); zeros(4, 4), M];
\]
\[
B_{ster} = [zeros(4, 4), -eye(4); zeros(4, 4), B];
\]
\[
G_{ster} = [zeros(4, 1); C];
\]
\[
H_{ster} = [zeros(4, 2); H];
\]
\[
As = inv(M_{ster})*(-B_{ster});
\]
\[
Bs = inv(M_{ster})*(H_{ster});
\]
\[
Gs = inv(M_{ster})*(-G_{ster});
\]
\[
zdot = [As zeros(8, ll); zeros(ll, 8) Ac]*z + [Bs zeros(8, 4); zeros(4, ll, 2) Bc]*[u; y] + [Gs; zeros(ll, 1)];
\]
File III:
% This file computes the potential, kinetic and total energy of the unicycle after
% having simulated its dynamic behaviour

% -----------Set the elements of the state-space vector to variables
x=Z(:,1);  p=Z(:,2);  a=Z(:,3);  b=Z(:,4);
xp=Z(:,5);  pp=Z(:,6);  ap=Z(:,7);  bp=Z(:,8);

%--------The parameters of the system
mw=1.5;  mf=8.5;  R=2;  h=4;
i=1;  g=9.81;  nu1=6.33;  nu2=6.33;
jw1=0.030;  jw2=0.060;  jm1=0;  jm2=0.001;
js1=0.070;  js2=0.035;  mu=2.12e-3;
jf1=0.180;  jf2=0.180;  jf3=0.045;
Rv=R*ones(length(xp),1);

%---------Calculating the rotational part of the kinetic energy
Trot1=.5*(jw1*ap.^2+jw2*(xp/R).^2+ja1*(ap.^2).*a.+2+ja2*pp.^2+ja3*(ap.^2).*a.+2+
+ jsl*(bp+ap.*cos(p)).^2);
Trot2=.5*(js2*(ap.^2).*a.(xp.^2+pp.^2)+jm1*ap.^2+jm2*(xp/(i*R)+((i-1)/i)*pp).^2);
Trot=Trot1+Trot2;

%---------Calculating the translational part of the kinetic energy
Ttrans1=.5*mw*(xp.^2+R^2*ap.^2)+.5*mf*(xp.^2+(h^2*(cos(p).^2).*a.+2+(sin(p).^2).*a.+2);
Ttrans2=.5*mf*((h^2*(sin(a).^2).*a.(pp.^2)+(Rv+h*cos(p)).^2).*a.+2+(.-5*h^2*sin(2*a).*a.sin(2*p)).*a.*pp);
Ttrans3=.5*mf*((2*h*cos(a).*cos(p)).*pp.*xp+(-2*h*sin(a).*sin(p)).*a.*xp);
Ttrans=Ttrans1+Ttrans2+Ttrans3;

%--------Calculating the potential energy
Tp=R*cos(a)*mw*g+(Rv+h*cos(p)).*cos(a)*mf*g;

%--------Calculating the total energy of the system
Ttot=Trot+Ttrans+Tpot;
% This file calculates the linear state space representation of the non linear system of an unicycle using system matrices calculated by analytical linearization.

%----------The parameters of the system

mw=1.5; mf=8.5; R=.2; h=.4;
i=1; g=9.81; nu1=6.33; nu2=6.33;
jw1=0.030; jw2=0.060; jm1=0; jm2=0.001;
js1=0.070; js2=0.035; nu=2.12*10^(-3);
jf1=0.180; jf2=0.180; jf3=0.045;

%----------Defining the assisting parameters

ja=jw2/R^2+jm2/(R^2*i^2);
jb=(i-1)/(R*i^2)*jm2;
jc=jf2+js2+((i-1)/i)^2*jm2;
jd=jw1+jm1;
je=jf1+js1;
jg=js2+jf3;
jb=jf1+jf3/js1+js2;
ep=nu1/i^2+mu;

%----------Calculating the linearized system matrices

mster=[mw+mf+ja,mf*h+jb,0,0;mf*h+jb,mf*h^2+jc,0,0,0,0,mf*h*(h+R)^2+jd+je,js1;0,0,js1,js1];
bster=[ep*R^2,-ep/R,0,0,-ep/R,ep,0,0,0,0,0,0,0,0,0,0,0,0,0,nu2];
kster=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
hster=[-1/(i*R),0,-1/i,0,0,0,0,-1];

%----------Set the structure to a state-space representation

a0=[eye(4),zeros(4,4);zeros(4,4),mster];
b0=[zeros(4,4),-eye(4);kster,bster];
h0=[zeros(4,2);-hster];

Ap1=-(inv(a0)*b0);
Bp1=inv(a0)*h0;
File V:

This file is practically the same as File II, but implements a lateral push to the system instead of introducing a reference signal $r$. The adjustments to File II are printed bold.

% This file is the function file which describes the non linear, dynamic behavior of an unicycle
% in the following structure
% zdot=function(t,z,u)
% The equations of motion derived for this purpose are
% \[ M*qpp+B*qp+C=H*u \]
% In this, $qp$ is a column with the time-derivatives of the generalize coordinates,
% and $qpp$ is the time-derivative of $qp$.
% $u$ is a column with the actuator entries of the system. Its value depends on the calculated
% controller, which is represented in a state-space representation with $Ac$, $Bc$, $Cc$ and $Dc$.

function zdot=systeem(t,z,u);
    global Ac Cc Bc
    r=0;
    t
    mw=1.5; mf=8.5; R=.2; h=.4;
i=1; g=9.81; nu1=6.33; nu2=6.33;
jw1=0.020; jw2=0.060; jm1=0; jm2=0.001;
js1=0.070; js2=0.035; mu=2.12*10^(-3);
jf1=0.180; jf2=0.180; jf3=0.045;

%---------Defining the assisting parameters
ja=jw2/R^2+jm2/(R^2*i^2);
jb=(i-1)/(R*iA2)*jm2;
jc=jf2+js2+((i-1)/i)^2*jm2;
jd=jw1+jm1;
jf=jf1+js1;
jj=js2+jf3;
jh=jf1+jf3-jf1+js2;

%---------Set the elements of the state-space vector to variables
x=z(1); p=z(2); a=z(3); b=z(4);
xp=z(5); pp=z(6); ap=z(7); bp=z(8);

%------Calculation of $y$ and $u$
y=[x;b;5*x-p;r];
u=Cc*z(9:(ll+8));
%----------Calculation of the system matrices M, B, C and H

\[ m_{11} = m_w + m_f + j_a; \]
\[ m_{12} = j_b + m_f h \cos(a) \cos(p); \]
\[ m_{13} = -m_f h \sin(a) \sin(p); \]
\[ m_{22} = j_c + m_f h^2 (1 - \sin(a)^2 \cos(p)^2); \]
\[ m_{23} = -2.5 m_f h^2 \sin(2a) \sin(2p); \]
\[ m_{33} = -m_f (h^2 + R^2) + 2 m_f R \cos(p) + m_w R^2 - m_f h^2 \sin(p)^2 \cos(a)^2 + j_d + j_e \cos(p)^2 + j_g \sin(p)^2; \]
\[ m_{43} = j_s \cos(p); \]
\[ m_{44} = j_s 1; \]

\[ M = \begin{bmatrix} m_{11} & m_{21} & m_{31} & 0; \\
                      m_{21} & m_{22} & m_{32} & 0; \\
                      m_{31} & m_{32} & m_{33} & m_{43}; \\
                      0 & 0 & m_{43} & m_{44} \end{bmatrix}; \]

if \( t < 1 \)
\[ FF = 0; \]
else
\[ if \ t < 1.01; \]
\[ FF = 24; \]
\[ else \]
\[ FF = 0; \]
\[ end \]
\[ end \]

\[ c_2 = -m_f g^2 h \cos(a) \sin(p); \]
\[ c_3 = -m_w R^2 g^2 \sin(a) \sin(p); \]
\[ c_4 = m_f (R + h \cos(p)) \sin(a)^2 \sin(2p) + FF; \]

\[ C = \begin{bmatrix} 0; c_2; c_3; 0 \end{bmatrix}; \]

\[ b_{11} = 1/R^2 (nu1/i^2 + mu); \]
\[ b_{21} = 1/R^2 (nu1/i^2 + mu); \]
\[ b_{43} = j_s \sin(p) \sin(p); \]
\[ b_{12} = -m_f h \cos(a) \sin(p)^2 \sin(p) + b_{21}; \]
\[ b_{22} = 5 m_f h^2 \sin(2p) \sin(a)^2 \sin(p)^2 \sin(p) - b_{21} R; \]
\[ b_{32} = 5 h^2 m_f h^2 \sin(2a) \sin(p)^2 \sin(p); \]
\[ b_{13} = -m_f h \cos(a) \sin(p) \sin(p) - 2 m_f h \sin(a) \cos(p)^2 \sin(p); \]
\[ b_{23} = (m_f R \sin(p) + 5 m_f h^2 \sin(a) \sin(2p)^2 \sin(p) h^2 \sin(a) \sin(2p)^2 \sin(p) + 2 m_f R \cos(p) \sin(p) + (j_g - j_e)^2 \sin(2p)^2 \sin(p) + b_{24} = j_s \sin(p) \sin(p); \]
\[ b_{34} = j_s \sin(p) \sin(p); \]
\[ b_{44} = nu2; \]

\[ B = \begin{bmatrix} b_{11}, b_{12}, b_{13}, 0; b_{21}, b_{22}, b_{23}, b_{24}, 0, b_{32}, b_{33}, b_{34}, 0, 0, b_{43}, b_{44} \end{bmatrix}; \]

\[ H = \begin{bmatrix} 1/(i R), 0; -i, 0, 0, 0, 0 \end{bmatrix}; \]

%----------Set the equations of motion to the correct structure

\[ M_{\text{ster}} = [\text{eye}(4), \text{zeros}(4, 4); \text{zeros}(4, 4), M]; \]
\[ B_{\text{ster}} = [\text{zeros}(4, 4), -\text{eye}(4); \text{zeros}(4, 4), B]; \]
\[ G_{\text{ster}} = [\text{zeros}(4, 1), C]; \]
\[ H_{\text{ster}} = [\text{zeros}(4, 2), H]; \]

\[ A = \text{inv}(M_{\text{ster}}) * (-B_{\text{ster}}); \]
\[ B = \text{inv}(M_{\text{ster}}) * H_{\text{ster}}; \]
\[ G = \text{inv}(M_{\text{ster}}) * (-G_{\text{ster}}); \]

\[ \text{zdot} = [A \text{ zeros}(8, 11); \text{zeros}(11, 8) \text{ Ac}] * z + [B \text{ zeros}(8, 4); \text{zeros}(11, 2) \text{ Bc}] * [u; y] + [G \text{ zeros}(11, 1)]; \]
APPENDIX III

Transfer functions

The transfer functions from $u_1$ and $u_2$ to the elements of $q$ are printed below for the linearized model of (2.12). The four transfers which are not printed, equal zero.

![Transfer functions graphs]

- Transfer from $u_1$ to $x$
- Transfer from $u_2$ to $\phi$
- Transfer from $u_2$ to $\alpha$
- Transfer from $u_2$ to $\beta$