3D Pendulum Swinging Control by an Industrial Robot Manipulator

Wilhelm August\textsuperscript{1}, Jian Ren\textsuperscript{2}, Simon Notheis\textsuperscript{1}, Thomas Haase\textsuperscript{1}, Björn Hein\textsuperscript{1} and Heinz Wörn\textsuperscript{1}
Karlsruhe Institute of Technology, IPR, Karlsruhe, Germany

Abstract

In this paper a pendulum of two degrees of freedom is investigated, i.e. a spherical pendulum which is attached to the end effector of an industrial robot manipulator. The result of the presented research is a controllable transport of the pendulum in all three dimensions. That means a real time control of the pendulum during all the time of the transport and a stable reaction of suddenly changing circumstances. The mathematical model has been derived due to the Lagrangian mechanics without consideration of the external forces. The position of the pendulum is observed with an USB camera which is also attached to the end effector and follows the motion of the manipulator. The analysis of the pictures taken by the USB camera provides an estimation of the current position of the pendulum and builds the input of the controller. For the first the experimental setup, the mathematical model and the controller are explained and then some experimental results are presented.

1 Introduction

On the one hand transportation of suspended loads (e.g. transporting goods by crane on construction sites) is a very common and useful way of transportation. On the other hand it bears the risk of causing injuries to human beings or mechanical damages which might lead to economic disadvantages. An uncontrollable swinging of hanging loads can overbalance the crane. This causes typically a lot of damage in a high density residential area [1]. Another example is the aspect of safety and efficiency of loading and unloading of container ships. It is important to move a huge number of containers as fast as possible [2], [3]. The transportation of heavy objects as a hanging load by a helicopter can get risky by cross wind or/and flight maneuver of the pilot, causing unwanted and dangerous resonant vibrations of the load. To limit the damage, the load typically has to be dropped and is therefore lost [4].

All these transporting mechanisms mentioned above can be modeled as a spherical pendulum. In general the point of suspension can be driven freely in all three directions in space. This movement feeds some energy into the oscillatory system and the pendulum starts swinging. In almost every paper cited in here some derivations of the mathematical model of a spherical pendulum with a flexible point of suspension are presented. Chaturvedi et al. also discuss the problem of control of a spherical pendulum in their works [5] and [6], but regrettably only in theory and evaluate their results in numeric simulations. Chen et al. [7] use the acceleration compensation principle, which was originally developed to minimize undesired shear forces, to plan the trajectory before they transport a spherical pendulum by an industrial robot. There are a lot of publications about the inverted pendulum. The most of them deal with the swing-up process and stabilization of the instable upright position of a 2D inverted pendulum [8], [9], [10]. All the working groups propose different kinds of controllers to drive the cart to which the pendulum is attached. For example the team of Graichen presented a so called “feedforward and linear feedback-controller” for a 2D inverted double pendulum. This group achieved remarkable interesting experimental results. Åström and Furuta propose in their publication a controller which observes the swinging energy of the system in 2D and minimizes it by driving the cart. A spherical inverted pendulum is discussed in the work of Shirieav et al. [9]. They deal with the theory of the swinging-up process.

In all this works cited above the path of the manipulator movement has to be planned and computed before starting the transport. In our paper we present a real time, optical sensor observed (USB camera) transport system which is able to react upon unexpected occurrences, like cross winds etc. or inaccuracies in the model or experimental setup, like an incorrectly measured pendulum length. The controller, which is designed in Matlab/Simulink, is based on the mathematical model which is derived in section 3 due to the Lagrangian mechanics.

2 Experimental Setup

The core of the experiment is the industrial robot manipulator KUKA KR-16 (see Figure 1). It is originally designed for transporting loads up to 16 kg. Because of its six-joint construction it is extremely versatile. In the industry this universal robot is often used for welding, drilling, grinding etc. or transporting small loads. A pendulum of two degrees of freedom, i.e. a spherical pendulum is
Figure 1: KUKA KR-16 robot manipulator transporting a pendulum

attached to the end effector of the manipulator. It is made of an airy (in the model a massless) hairline and an aluminium cylinder of about 0.2 kg. The length of the hairline can be chosen up to 1.5 m and remains constant during the experiment. For our experiment we chose the length of the pendulum to be 0.68 m.

Figure 2: End effector of the KUKA KR16 with USB camera and IR-LED an cooling element.

Figure 3: A drawing of a real time communication between the Host-PC on which a Simulink controller is running and the robot manipulator PC (Robot-PC). A QNX based transceiver supports an online connection to both, the Robot-PC and the Simulink controller and allows a real time connection between them.

The behavior of the pendulum is observed by an USB camera which is also attached to the end effector of the robot and follows the motion of the manipulator (see Figure 2). The analysis of the pictures taken by the USB camera provides the current position of the pendulum body. Collecting the information of the position at different points in time allows an approximation of the velocity. These data are used as input for the controller which is implemented in Matlab/Simulink. It is running on the Host-PC (Windows XP), beside the image processing (see Figure 3). The controller computes the behavior of the pendulum due to the mathematical model and the current position and adjusts the acceleration of the point of suspension in such a way that the end effector reaches its destination with the required boundary conditions. Usually it is a stationary position.

The spherical pendulum is a very sensitive system. To move the robot manipulator to the desired position with the required boundary conditions a good real time connection between the controller Host-PC and the robot manipulator PC (Robot-PC) is needed. Therefore the KUKA.Ethernet RSI XML technology is used (see [11]).

The pendulum application development system bases on Matlab/Simulink. A special Simulink KUKA library which uses the Real Time Windows Target (RTWT) was developed to build, compile and run user defined control systems. The library submits different KUKA robots simultaneously and therewith concurrent operations and the integration of additional informations within the RSI XML interface is slightly possible. The RTWT is able to run Simulink models with a sample time $t_s$ of less than $t_s \leq 12\text{ms}$ which is the sample time of the RSI interface. Therewith it is possible to build up a control system which runs faster than the real data exchange with the KUKA. Alternatively, an intelligent combination of input and output UDP blocks establishes the desired synchronized data
exchange with the RSI interface. Based on the fact, that the RSI interface changes its local UDP port each time the connection respectively the data interchange has stopped and the UDP port in Simulink is defined as a static port it is tricky to maintain a direct connection between Simulink and the KUKA RSI itself.

The solution was to use a small transceiver module to exchange the data between the KUKA RSI and Simulink RTWT interface. This QNX based transceiver module simplifies the transfer to other supported real time systems (XPC, VXWorks).

All in all the communication system described and pictured (Figure 3) above allows a real time exchange of the current position data (robot side) and the new desired position (Simulink controller side). In this application the robot control PC receives new position data and sends his current position every 12 ms. Therefore a real time control is possible.

3 Mathematical Model

The derivation of the mathematical model the controller is based on is due to the Lagrangian mechanics without consideration of the external forces. As shown in Figure 4 our model implies a big mass \( M \) with the position vector \( \mathbf{x} \) in the absolute coordinate system \((x, y, z)\) to which a small sphere \( m \) is attached. The position vector of the sphere \( m \) is in the absolute coordinate system \( \mathbf{x}' \). There is a further coordinate system stuck to the bigger mass \( M \). It is called the relative coordinate system \((\alpha, \beta, \gamma)\).

The kinetic energy of the complete system is

\[
T = \frac{1}{2} M \dot{\mathbf{x}}^2 + \frac{1}{2} m \dot{\mathbf{x}}'^2 \tag{1}
\]

\[
= \frac{1}{2} M \left( \dot{x}^2 + \dot{y}^2 + \dot{z}^2 \right)
+ \frac{1}{2} m \left( \dot{x}'^2 + \dot{y}'^2 + \dot{z}'^2 \right) \tag{2}
\]

with

\[
\dot{\gamma} := \sqrt{l^2 - \dot{\alpha}^2 - \dot{\beta}^2} = \frac{\dot{\alpha} \dot{\beta} + \dot{\beta}^2}{\sqrt{l^2 - \dot{\alpha}^2 - \dot{\beta}^2}}. \tag{3}
\]

and \( l \) the length of the pendulum. With the definition of \( \gamma \) introduced in (3) we only observe the lower half space underneath the point of suspension (POS), because it is always positive.

The potential energy is defined by

\[
V = Mgz + mgz' = Mgz + mg(z - \gamma), \tag{4}
\]

where \( g \) is the acceleration due to gravity. Now we derive the Lagrangian equations from the Lagrangian function \( L = T - V \) due to

\[
\frac{\partial L}{\partial \dot{\alpha}} = \frac{d}{dt} \frac{\partial L}{\partial \dot{\alpha}} = \frac{d}{dt} \frac{\partial L}{\partial \dot{\alpha}}, \tag{5}
\]

for \( \beta \) respectively.

Combining the equations (2), (4) and (5) yields the following equation of motion

\[
\ddot{\alpha} = -\frac{1}{\gamma^2(\dot{\gamma}^2 + \dot{\alpha}^2)} \left( \gamma^4 \ddot{y} + \gamma^3 \alpha (\ddot{z} + g) + \gamma^2 \dot{\alpha} \dot{\beta} \right.
+ \alpha \dot{\dot{\alpha}} (\gamma^2 + \dot{\alpha}^2) + \dot{\alpha} \dot{\beta}^2 (\gamma^2 + \dot{\beta}^2) + 2 \dot{\alpha} \dot{\beta} \dot{\alpha}^2 \dot{\beta}^2 \right), \tag{6}
\]

with \( \gamma^2 = l^2 - \dot{\alpha}^2 - \dot{\beta}^2 \) as imposed in (3).

The same expression results for \( \dot{\beta} \) if \( \dot{\alpha}, \ddot{\alpha} \) and \( \alpha \) is replaced by \( \dot{\beta}, \ddot{\beta} \) and \( \beta \), and \( \ddot{y} \) by \( \ddot{x} \) respectively, \( \ddot{x} \) remains.

Remember that \( \mathbf{x} = (\dot{x}, \ddot{y}, \ddot{z}) \) is the acceleration of the POS. Due to the equation of motion (6) an acceleration of POS in the direction of \( y \) apparently causes a (swinging) acceleration in the direction of \( \alpha \), \( \ddot{\alpha} = \ddot{\alpha}(\ddot{y}, \ddot{z}, ...) \), which is comprehensible.

4 Control

In this part we concentrate on the control of the system, which includes not only the controller design but also the acquisition of important system parameters. Furthermore we give a short overview over the optical sensor system and the image processing which provides the input of the controller.

4.1 Sensor System and Image Processing

As already mentioned in section 2 and pictured in Figure 2 the optical sensor system consists of an uEye USB camera with a monochrome 1/3” CCD sensor (1024x768),
an IR-LED on a cooling element (electric power about $p = 0.8-0.9\,W$) and a flat reflector on the pendulum body. The camera and LED, which is attached next to the camera look downward along the hairline of the pendulum. The aperture angle of the camera is about $90^\circ$ in the diameter. The LED which is used to simplify the image processing emits light with the maximum intensity at $\lambda_{peak} = 850\,nm$. This wavelength was chosen because the camera is eminently sensitive in this range and the unwanted daylight can be blocked by a filter in front of the camera chip. Using this system it does not matter if there are some other sources of light (not IR). A flat reflector is stuck to the upper side of the pendulum body. It reflects the light of the LED and is therefore distinctively visible in the frame. Figure 5 shows a frame taken by the camera before the image processing. The bright mark on the dark background is the reflector. It is quite easy to find the reflector and its center of mass in the picture by using the threshold method. After some further image processing operation, like distortion etc., it is possible to estimate the position ($\alpha, \beta$) and with the knowledge of the time interval between two frames (about 20-30 fps) it is also possible to estimate the velocity ($\dot{\alpha}, \dot{\beta}$) of the pendulum body. This two values compose the input of the controller which is described in the following section.

![Figure 5: Camera view: reflector on the upper side of the pendulum body reflects the light of the IR-LED, which is attached next to the camera (see Figure 2)](image)

### 4.2 Swinging and Robot Manipulator Control

#### 4.2.1 Linearisation of the pendulum model

Through analysis of the system there are ten essential states, called the state vector $\mathbf{x} = [\alpha, \dot{\alpha}, \beta, \dot{\beta}, x, \dot{x}, y, \dot{y}, z, \dot{z}]$, which will be observed and controlled by the controller. Remember $\alpha$ and $\beta$ are the coordinates in the relative coordinate system and $x, \dot{x}, y, \dot{y}, z$ and $\dot{z}$ are the position and velocity of the POS in the absolute coordinate system (see Figure 4). With (6) the whole system can be described as a ten-state-space-equation system (see also Figure 6)

$$\dot{x} = f(x) + b(x)u$$  \hspace{1cm} (7)

$$y = Cx,$$  \hspace{1cm} (8)

where $u$ is the acceleration of the POS and therefore the control signal.

The relationship between the elements of the state vector (7) are non-linear differential equations. Before designing the controller a linearisation of the model is needed. According to [13], partial derivative of the equations will be performed and evaluated at the operating point (rest position of the pendulum).

$$A = \frac{\partial f(x)}{\partial x}|_{x=x_R}$$   \hspace{1cm} (9)

$$B = b(x)|_{x=x_R}$$   \hspace{1cm} (10)

with $x_R = 0$ and $u_R = 0$. With this conversion the equations turns into a standard form

$$\dot{x} = Ax + Bu$$  \hspace{1cm} (11)

$$y = Cx$$  \hspace{1cm} (12)

Now the controller due to [14] can be designed.

#### 4.2.2 Adaptation of the Control Signals

As has been discussed above the control signal of the pendulum model is the acceleration of the POS (see 4.2). However, because of the intern mechanism of the robot, we cannot use the acceleration for the control. The RSI-XML Technology (see 2) from KUKA allows us to assign a desired angle step for each joint every 12 ms, which means, we can control the rotation speed $\dot{q}$ of every joint. By means of Jacobian matrix the speed of the end effector $v$ can be transformed into joint space $\dot{q}$ and in this way be directly controlled [12], [15]:

$$\dot{q} = J^{-1}(q)v$$  \hspace{1cm} (13)

But in fact, the joint speed $\dot{q}$ cannot be reached instantly. There is always a manifest latency. To simplify the control design we just approximate it with a first order inert element. Its Laplace transformation is

$$\frac{v}{u} = \frac{1}{1 + Ts}$$  \hspace{1cm} (14)

where $v$ is the actual and $u$ the desired speed which later will be used as the control signal. $T$ is the time constant which was determined by experiment to $T = 0.1\,s$.

We can also notice that the control signal acceleration $a$ is the derivative of $v$

$$a = \dot{v} = \frac{1}{T}(u - v)$$  \hspace{1cm} (15)

This equation will be substituted as the control signal $u$ into the equation system (11). It becomes a new but still a linear state space equation system.
4.2.3 Controller Design

With all the elements discussed above, this control of the Pendulum has turned into a traditional control problem. According to the criterion of Kalman (see [14]) the original system is controllable. To control it we use a commonly used state feedback controller with the set point input. By using this controller it is possible not only an anti-disturbance function i.e. stabilization of the load at a stationary point but also a transportation of the load in space. The complete model of the controlled system is shown in Figure 6.

![Model of the controlled system (see [14])](image)

The system matrix of the controlled system is now \((A - BK)\) and this stabilized system will converge to zero after finite time. The matrix \(M\) (see Figure 6) which is responsible for the static performance of the system is determined according to the design principle (see [14])

\[
M = (C(-A - BK))^{-1}B^{-1}. \quad (16)
\]

Otherwise the feedback matrix \(K\) of the controller has a great effect on the system. It will be determined by solving the Riccati equation

\[
A^T P + PA - PBK^T P + Q = 0 \quad (17)
\]

which will be an optimal solution for the criterion (see [14])

\[
J = \int_0^\infty [x^T(t)Qx(t) + u^T(t)Ru(t)]dt. \quad (18)
\]

Therein \(Q\) is the weighting matrix for states \(x\) and \(R\) the weighting matrix for control signal \(u\).

We use a function lqr\((A, B, Q, R)\) to solve the Riccati equation (17) optimization problem in Matlab. The result of the numerical computation is the matrix \(K\), the feedback matrix. By using and optimizing (by experiment) the weighting factors in \(Q\) and \(R\) the corresponding states can be weighted in the desired way. In the following section 5 we will present two basic cases: a stabilization and a transportation case. For the case of stabilization, an anti-disturbance performance, the weighting factors of \(x, \dot{x}, y, \dot{y}, z\) and \(\dot{z}\) are chosen quite large. In the case of transportation, on the contrary, the position and the speed of POS will be focused more and the weight for \(\alpha, \alpha, \beta\) and \(\beta\) will be lifted.

5 Experimental Results

In this section we present the experimental results of our work. We have designed a system for transporting suspended loads with an industrial manipulator. As already mentioned in section 2 we have experimented with a pendulum with a length of about \(l = 0.68\ \text{m}\) and with a pendulum body weight of about \(m = 0.2\ \text{kg}\).

We have investigated two different kinds of experiment. In the first place of course we can transport the pendulum in space from a starting point to an end point, with the constraint that it is swinging controllable during the transportation and does not swing at the end point. The initial condition of the pendulum, if it is swinging or not, is non-efficient. Secondly we can control and stabilize a suspended load at a stationary point, if the pendulum i.e. is unbalanced by some kind of crosswind or collision with obstacles. This process can of course be considered as a stabilized transport in which the staring point is coextensive with the end point.

A second part of our work was to simulate the transporting processes described above and compare the experimental and simulated data. For the simulation the mathematical model (see 3) and the controller (see 4) were modeled as a Simulink model.

![Figure 7: Stabilization case](image)

Figure 7 shows the case of stabilization of the pendulum at a stationary point \(x = (0, 0, 0)\). To demonstrate how the system reacts to disturbances the pendulum was deflected by a long stick for security reasons. The black dotted curve shows the result of the experiment. The first two diagrams show \(x, y, \dot{y}, z\) and \(\dot{z}\), i.e. the deflection in the relative coordinate system. The last three demonstrate \(x, y\) and \(z\), i.e. the movement of the POS in the absolute. At the beginning the pendulum and therefore the POS is in the position of rest. At \(t_{hit} \approx 0.8\ \text{s}\) it is hit by the stick. The answere of the system comes almost coincidently. The POS starts moving. After about two seconds the amplitude of the pendulum swinging is negligible. The pendulum is in rest again.

The dashed red curve in Figure 7 stands for the result of the Simulink simulation. We started the simulation at the point in time where \(\alpha\) reaches its maximum to be assured that the pendulum has no contact with the stick.

Tendentially the amplitude and the cycle duration of the simulated pendulum are slightly larger. This can be explained by missing the air friction in the simulation. All in all there is a very good correlation of the simulation and experimental results.

The second part of the experiment, a transportation of the pendulum in space is shown in Figure 8. The movement begins here in \(x_{start} = (-0.103, -0.352, -0.103)\) and
Stabilization of the pendulum at a stationary point after a disturbance, a hit with a stick. The motion of the POS begins and ends in the same point \( x = (0, 0, 0) \). The whole process lasts about two seconds. \( \alpha \) and \( \beta \) is the deflection in the relative pendulum coordinate system. \( x, y, \) and \( z \) is the motion of the POS. The black dotted curve demonstrates the experimental part, the dashed red line the simulation results. The simulation starts at the point in time where \( \alpha \) reaches its maximum to be assured that the pendulum has no contact with the stick. Notice, all values are given in meters. The weighting matrices are \( Q = \text{diag}[100, 10, 100, 10, 100, 10, 10, 10, 100, 10] \) and \( R = \text{diag}[10, 10, 10] \).

ends in \( x_{\text{end}} = -1 \cdot x_{\text{start}} \). Initially and at the end of transportation the pendulum is at rest. The experiment and the simulation begin simultaneously because the initial conditions of this transport for both are easy to enforce. The color code of the curves is the same as in Figure 7. The weighting matrices are \( Q = \text{diag}[100, 10, 100, 10, 100, 10, 10, 10, 100, 10] \) and \( R = \text{diag}[10, 10, 10] \).

control signal \( u = [\ddot{x}, \ddot{y}, \ddot{z}] \) may be weighted differently depending on what kind of transport is desired. For the transportation the weighting matrix \( Q \) was chosen to

\[
Q = \text{diag}[50, 5, 50, 5, 30, 20, 30, 20, 30, 20]
\]

\[
R = \text{diag}[10, 10, 10].
\] (19)

for the anti-disturbance performance. This values were determined by experiments.

Let’s start with the matrices (20): the values were determined by experiment. The position is weighted more than the first derivation in time, i.e. the velocity. It was more for us relevant to reach the desired position than to delete the

\[
Q = \text{diag}[100, 10, 100, 10, 100, 10, 100, 10, 100, 10]
\]

\[
R = \text{diag}[10, 10, 10].
\] (20)
swinging completely. That means that small oscillations of the load around the center of mass were neglected. However in the case of transportation the weight of the elements $\dot{\alpha}, \alpha, \dot{\beta}$ and $\beta$ is defined relatively bigger than $x, \dot{x}, y, \dot{y}, z$ and $\dot{z}$. In this case we wanted to force a small amplitude of swinging during the transportation. The consequence is of course the increase of the transport duration.

6 Conclusion

We have successfully designed and presented a complete transportation and stabilization system for suspended loads, consisting of an industrial robot manipulator, an optical sensor system and a controller. The suspended load is represented by a spherical pendulum. We have explained the experimental setup, the derivation of the mathematical model and the designing of the controller and the adaption of the control signal to the robot control.

Two kinds of experiments were performed: transportation and stabilization of suspended loads. The first difference between them is that in the case of anti-disturbance performance (stabilization) the end point of the motion is co-extensive with the starting point. Secondly the weighting matrix $Q$ was chosen differently. The experiments has also been simulated by a Simulink model. The compliance between the experiment and simulation results unveils that the mathematical model in association with controller describe the real system in a very convincing way.

References


[10] Åström, K.J.; Furuta, K.: Swinging up a pendulum by energy control, IFAC 13th World Congress, San Francisco, California, 1996


