An Active Vertical-Direction Gravity Compensation System

Project of the Control Systems Course
Prof. Stefano Battilotti
Original Paper Authors: Gregory C. White and Yangsheng Xu
Candidate: Francesco Valente - Student ID: 1706964

Gravity Compensation Solutions and issues

Gravity Compensation Systems are very important for accurate experiments of Space Robotics to be conducted on Earth, in order to simulate environments with different values of gravitational acceleration. There are several systems to simulate microgravity (passive counterweight, active motor control, underwater experiments), but they lack of accuracy since they introduce additional variables and complexity. The authors of the analyzed paper cite a passive control solution (shown in Fig.1) realized in the Robotics Insitute of Carnegie Mellon University, which however shows the following weaknesses:

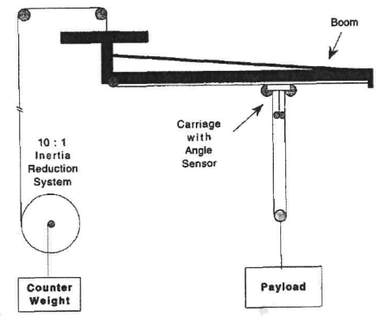

Figure 1: Passive Control System realized at CMU

- Effects of the static friction are not negligible, since there is a complex cable routing;
- Additional inertial mass;
- The mass of the counterweight is chosen by trial and error, this could lead to a not complete compensation of the gravitational acceleration.

In order to improve these weaknesses, the authors of the paper propose an active compensation system, combining the passive solution with an active servo motor control to modulate tension in the cable acting on one of the pulleys. This solution allows to reduce the negative effects of passive counterweight control solutions and to avoid fatal accidents to the robot in case of break of the DC motor, thanks to the presence of a counterweight mass.

Modeling of System Dynamics

In order to obtain an easy control design, the initial system is reduced to an easier one composed by one pulley and a rigid cable which supports two equal masses (see Fig.2)

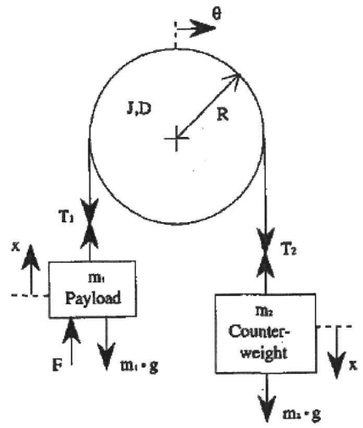

Figure 2: Simplified system for dynamics modeling

Moreover, the following assumptions are made in order to neglect high-order dynamics in this very first model to simplify it (but they will be taken into account further):

- Mass of the cable much smaller than the counterweight and payload ones;
- Negligible contribution of longitudinal compliance of the cable to the response;
- No slippage between the cable and the pulley;
- Electrical time constant of the DC servo motor much smaller than its physical time constant.

From Fig.2 it is very easy to calculate the dynamic equations:

$$T_1 = m_1 \cdot (g + \ddot{x}) - F \tag{1}$$

$$T_2 = m_2 \cdot (g - \ddot{x}) \tag{2}$$

$$J\ddot{\theta} + D\dot{\theta} = \Gamma_{ms} + R \cdot (T_2 - T_1) \tag{3}$$

where T_1 and T_2 represent respectively the tensions of the robot and counterweight cables; m_1 and m_2 the masses of the robot and counterweight; J, D and R are the inertial moment, the viscous damping and the radius of the pulley; Γ_{ms} is the motor torque, F the externally applied force; x and θ the vertical and angular position.

To obtain the transfer functions, we can apply the time derivative property for the Laplace Transform. From eq.1:

$$T_1(s) = m_1 s^2 X(s) - m_1 s x(0) - m_1 \dot{x}(0) + W_1(s) - F(s) \tag{4}$$

while from eq.2

$$T_2(s) = -m_2 s^2 X(s) + m_2 s x(0) + m_2 \dot{x}(0) + W_2(s) \tag{5}$$

then, since $x = R\theta$,

$$T_2(s) - T_1(s) = -R(m_1 + m_2)[s^2\Theta(s) - s\theta(0) - \dot{\theta}(0)] + W_2(s) - W_1(s) + F(s) \tag{6}$$

From eq.3 we can obtain with the same properties, defining $\alpha = J + (m_1 + m_2)R^2$:

$$(\alpha s^2 + Ds)\Theta(s) - (D + \alpha s)\theta(0) - \alpha\dot{\theta}(0) = \Gamma_{ms}(s) + R(W_2(s) - W_1(s)) + RF(s) \tag{7}$$

from which we get

$$\Theta(s) = \frac{\alpha s + D}{(\alpha s + D)s}\theta(0) + \frac{\alpha}{(\alpha s + D)s}\dot{\theta}(0) + \frac{\Gamma_{ms}}{(\alpha s + D)s} + \frac{R(W_2(s) - W_1(s))}{(\alpha s + D)s} + \frac{RF(s)}{(\alpha s + D)s} \tag{8}$$

Replacing eq.8 in eq.4, considering as initial conditions $\theta(0) = 0$ and $\dot{\theta}(0) = 0$ and assuming $m_2 = m_1$ we get:

$$T_1(s) = W_1(s) + \frac{m_1 R s^2}{(\alpha s + D)s}\Gamma_{ms}(s) - \frac{(\alpha - m_1 R^2)s^2 + Ds}{(\alpha s + D)s}F(s) \tag{9}$$

Thus, we get the following transfer functions (represented in Fig.3)

$$H(s) = (\frac{\alpha - m_1 R^2}{\alpha})\frac{s + \frac{D}{\alpha - m_1 R^2}}{s + \frac{D}{\alpha}} \tag{10}$$

$$G(s) = (\frac{m_1 R}{\alpha})\frac{s}{s + \frac{D}{\alpha}} \tag{11}$$

Combining equations 1,2,3 the following expression for Γ_{ms} can be also obtained:

$$\Gamma_{ms} = (\frac{2R^2 m_1 + J}{R})\ddot{x} + \frac{D}{R}\dot{x} \tag{12}$$

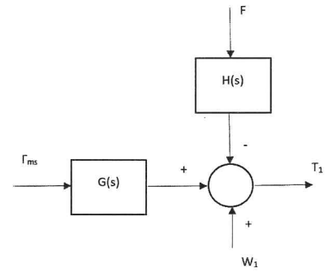

Figure 3: Block diagram

Controller Design and Simulation

We can describe for any point in time the position of the centre of mass of our body through the following state vector:

$$\vec{x} = \begin{bmatrix} \ddot{x} & \dot{x} & x \end{bmatrix}^T \tag{13}$$

In a zero-gravity environment, using an inertial frame, we obtain that the acceleration over a body is given by $\frac{F_0}{m}$, where F_0 is the sum of the outside forces and m is the mass of the body. The state vector, assuming zero initial conditions, is, thus, equal to:

$$\vec{x} = \frac{1}{m}\begin{bmatrix} F_0 & \int F_0 dt & \int\int F_0 dt^2 \end{bmatrix}^T \tag{14}$$

When the tension of the cable gets far from the ideal zero-gravity condition by a term δT, we have a deviation in the state vector equal to:

$$\vec{\delta x} = \frac{1}{m}\begin{bmatrix} \delta T & \int \delta T dt & \int\int \delta T dt^2 \end{bmatrix}^T \tag{15}$$

The objective is to minimize this state deviation. To reach this goal, PI action can be used in order to reject impulsive disturbances and track step disturbances with a finite error. Since the proposed control system will be implemented through a software, a discrete model must be considered. In the paper, the matched pole-zero approach is used to discretize the system. Starting from the analogic transfer function, all poles and zeros are mapped from s-plane to z-plane locations $z = e^{sT}$, where T is the sampling period (equal to $1/f_s$, with the setting of $f_s = 230Hz$ by the authors of the paper). In general, if we have the following analogic transfer function:

$$H(s) = k_a \frac{\prod_{i=1}^{M}(s - z_i)}{\prod_{i=1}^{N}(s - p_i)} \tag{16}$$

we get the following digital transfer function

$$H(z) = k_d \frac{\prod_{i=1}^{M}(1 - e^{z_i T} z^{-1})}{\prod_{i=1}^{N}(1 - e^{p_i T} z^{-1})} \tag{17}$$

where k_d is set to match the analogic filter gain to the digital filter one (putting $s = 0$ and $z = 1$ and then solving for k_d).

In our case, we get the following digital transfer function applying equations 16, 17 to 10 and 11:

$$G(z) = (\frac{m_1 R}{\alpha}) \frac{z - 1}{z - e^{-\frac{DT}{\alpha}}} \tag{18}$$

$$H(z) = \frac{(1 - e^{-\frac{DT}{\alpha}})(z - e^{-\frac{DT}{\alpha - R^2 m_1}})}{(1 - e^{\frac{-DT}{\alpha - R^2 m_1}})(z - e^{-\frac{DT}{\alpha}})} \tag{19}$$

Furthermore, $MATLAB©$ can be used to obtain our digital model through the function "$c2d$", as shown in the following code lines

```
%Continuous transfer functions

Gcont = tf([m1*R 0],[alpha D]);
Hcont = tf([(alpha-m1*R^2) D],[alpha D]);

%Discrete time transfer functions

Gdis = c2d(Gcont,T,'matched')
Hdis = c2d(Hcont,T,'matched')
```

where the following parameters given by the experimental setup description on the paper have been used:

m_1	m_2	D	R	J	T
$0.2615kg$	$0.3272kg$	$6.5 \cdot 10^{-6} Nms$	$0.0095m$	$65 \cdot 10^{-6} Nms^2$	$\frac{1}{230}s$

From the $MATLAB©$ simulation, the following transfer functions can be then obtained:

- Continuous time:

$$G(s) = \frac{21s}{s + 0.055} \tag{20}$$

$$H(s) = 0.8\frac{s + 0.0688}{s + 0.055} \tag{21}$$

- Discrete time:

$$G(z) = 21\frac{z - 1}{z - 0.9998} \tag{22}$$

$$H(z) = 0.8\frac{z - 1}{z - 0.9998} \tag{23}$$

As far as the digital control design is concerned, to compensate a step-type disturbance an integral action before $G(s)$ is needed, which corresponds to a pole in $z = 1$ in our digital control. Furthermore, since there is a delay of a period between sensing and actuating, a pole in $z = 0$ must be introduced, too, since it represents a delay in the digital control. Thus:

$$C(z) = K\frac{z - b}{z(z - 1)} \tag{24}$$

where K and b must be chosen using the discrete time root locus (Fig.4), remembering that poles must be inside the unitary circumference. It is important to remark that these values must be checked experimentally, in order to verify whether neglected behaviour compromise stability or not. Chosing, as the authors of the paper, $b = 0.83$ and using the following $MATLAB^{©}$ code, the root locus is obtained:

```
b=0.83;
Cdis = tf([1 -b],[1 -1 0],T);
rlocus(Cdis*Gdis);
```

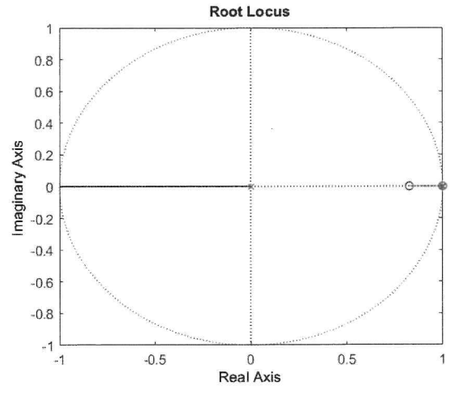

Figure 4: z-plane root locus of $C(z)G(z)$

From Fig.4, we can see that, choosing a small value of K (as the authors of the paper $K = 2.756 \cdot 10^{-4}$), we are inside the unitary circumference, then the system is asymptotically stable. The digital control is:

$$C(z) = 2.756 \cdot 10^{-4}\frac{z - 0.83}{z(z - 1)} \tag{25}$$

Implementing the discrete time model on $Simulink^{©}$, the behaviour of the tension error in time shown in Fig.5 is obtained, which demonstrates that the chosen controller guarantees that the error goes to zero

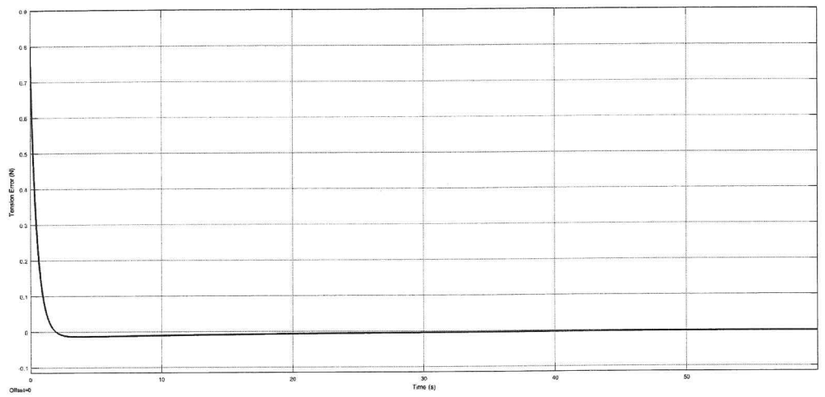

Figure 5: Error trend

considering a step disturbance.

In the paper, the simulation has been carried out rewriting the controller in time thorugh the zero order hold method. The pole in $z = 0$, which represents the delay between sensing and actuation, cannot be transformed back in time using the zero order hold method. Thus, the Padé approximant has been used in order to approximate the pole in zero, and the general trend obtained in this case is very similar to the one obtained with *Simulink*©.

Experimental Setup

The control system outlined in the previous chapter has been obtained under several simplifying hypotheses. In order to check the correct operativeness and to improve the model description, experimental trials have been carried out by the authors of the paper. In particular, high frequency behaviour may show unpredicted effects. In this part of the report, the experimental results obtained by the original authors will be outlined and commented.
The first problem underlined by experiments is that the gain selected for the controller leads to instability due to higher frequency effects. The gain has been then lowered in order to eliminate instability. After that, the Bode and step response analysis of $G(s)$ have been carried out in order to better understand the real system behaviour and improve the final control system.

As far as the Bode analysis is concerned, it has been carried out using a sinusoidal input torque with frequencies between 2 and $300Hz$ and recording the tension values. Two behaviours have been investigated: one for single-wrapped pulley (in which some slippage is possible) and one for double-wrapped pulleys (in this case no slippage is possible). The results are shown in Fig.6. Some noticeable behaviours should be underlined. The initial slope of the plot for double-wrapped pulleys is $+20dB/dec$, which suggests the presence of a zero more than poles at low frequencies. Furthermore, there are two couples of complex conjugate poles which are located at frequency 18 and $28Hz$. In particular, these two sets of complex conjugate poles are bounded to non-linear behaviour due to support cable length, and they change as the robot moves in the vertical workspace, but a linear approximation is still a good solution to improve the model. From this experimental trials, an improvement of the model can be obtained adding the following experimentally de-

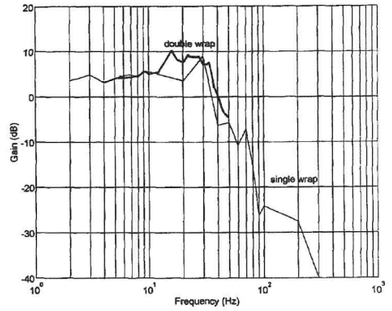

Figure 6: Bode plot of $G(s)$ for single and double wrapped pulley

termined poles and zeros

Poles	Zeros
-1.70	-1.36
$-3.5 \pm 58j$	-8.5
$-1.3 \pm 90j$	

Comparison between the actual step response and the ones gotten using the corrected model as before is shown in Fig.7 and Fig.8.

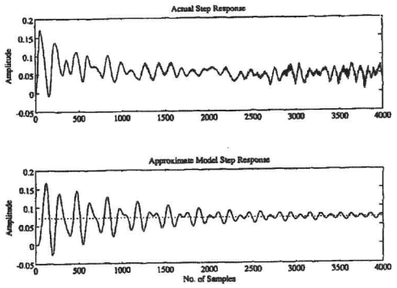

Figure 7: Step response comparison between actual and approximate model

Since the dynamics appear to be only lightly damped, a high feedback could lead the system to instability (due to highest order effects), going beyond the stability boundaries discussed in previous chapter. Experimentally, the digital controller is then modified to be:

$$C(z) = 8.3 \cdot 10^{-4} \frac{z - 0.3}{z(z-1)} \tag{26}$$

The effect of these dynamics can be also lowered by the use of a lowpass filter, in order to eliminate the higher frequency oscillations and only pass the ones needed for the control. The author of the paper choose to use a digital Butterworth filter with a cutoff frequency of $f_s/20$, which experimentally gives a good

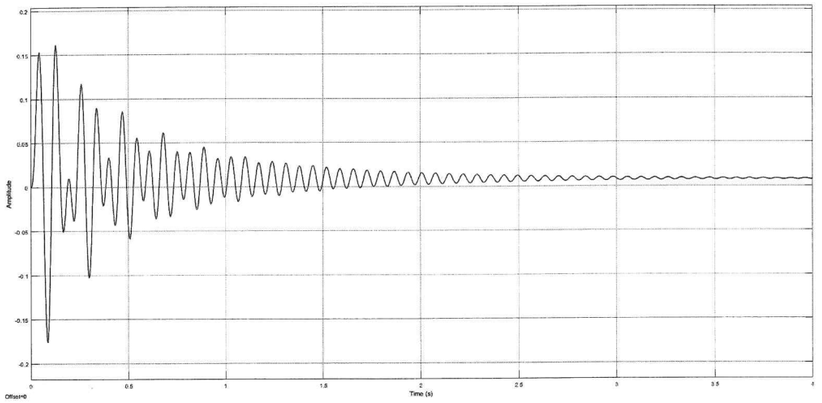

Figure 8: Step response of the approximate model simulated on *Simulink*©

contribution to highest frequency effects attenuation.

Friction Compensation and Offset Adaptation

A further problem to deal with is the static friction. All the previously described model works quite well when the system is already moving, but when a force is applied on the system at rest, static friction effect is not negligible and may lead to no effective torque actions until the applied torque goes beyond a certain threshold. Furthermore, also during the system motion there is still the effect of a Coulomb kinetic friction to take into consideration (in general there is a blend of Coulomb and static friction to be dealt with).

However, precise determination of friction might be very difficult. Thus, a better solution might be to use a fuzzy algorithm in order to compensate friction, associated with PI controller designed before to deal with disturbances. A fuzzy logic must be then designed. As far as inputs of this algorithm are concerned, motor velocity (to understand whether it is stationary or not) and a combination of PI and tension imbalance torque must be sensed (the first through the first difference of θ, the second ones by tension measures). The following membership functions can be stated for each input and output:

- Input:
  - Input torque: negative big, negative small, near zero, positive small, positive big
  - Velocity: negative, near zero, positive
- Output torque: negative big, negative small, near zero, positive small, positive big

On the basis of these functions, the following rules can be set:

- If motor velocity is negative or positive, the system is already moving, thus only a small torque is needed, and it must be applied in the same direction of velocity, since the friction is opposite to velocity;
- If motor velocity is near zero, a big torque is needed in the same direction of the input torque due to PI control and tension imbalance. However, the application of a big torque when a small torque is already applied by the PI control experimentally resulted to largely compensate friction. Thus, a small output torque is applied when the resulting torque from PI controller is small.

We can summarize the described logic in this table:

	PI neg. big	PI neg. small	PI near zero	PI pos. small	PI pos.big
Vel. negative	Out neg. small	Out neg. small	Out neg. small	Out neg. small	Out neg. small
Vel. near zero	Out neg. big	Out neg. small	Out near zero	Out pos. small	Out pos. big
Vel. positive	Out pos. small	Out pos. small	Out pos. small	Out pos. small	Out pos. small

It must be underlined that this algorithm is designed under the ideal hypothesis that if the system is at rest, we have no output of the PI control. However, this does not happen because of little unbalance between the mass of the robot and the counterweight or offset bias in the servo, which results in PI control action even when system is at rest. This DC offset represents a sort of noise (as it happens in various electronic applications, such as audio recording) to be dealt with in order to make the fuzzy system work correctly. In order to solve this issue, an offset buffer is added. Its initial value is an experimentally found value due to mass imbalance and changes when a different DC offset term is detected from the output of PI control. In particular, the detection scheme is based on evaluation of the PI torque over 100 cycles. If during these cycles the output of the fuzzy controller is zero, the average value of the PI command torque over them is added to the buffer and removed from the PI output.

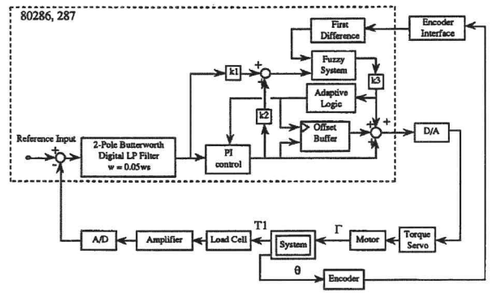

Figure 9: Complete system

Final Experimental Results and Conclusion

The results for acceleration, velocity and position error shown in Fig.10 have been experimentally obtained to evaluate the system response to a step disturbance force equal to $0.105kg \cdot 9.81m/s^2 = 1.03N$. For each state error component, passive control, PI control and PI control plus fuzzy friction compensation are compared. Passive control, as expected, shows the worst behaviour. With only PI action, acceleration error is very high at the beginning, and then remain finite after some time. Velocity error at the beginning follows the behaviour of passive control case, as expected because the applied torque is not enough to compensate static friction (and this determines also the initial behaviour of acceleration error). Position error is zero at the beginning and starts diverging after some time. Adding fuzzy control to PI action improves all performance. In fact, acceleration error remains always near to zero, velocity error is always near zero as well as position error. However, an oscillatory behaviour can be still seen in the acceleration error, due to remaining higher frequency effect which can be more damped through a better choose of gain.

To conclude, the proposed active gravity compensation system shows to overcome issues of passive ones, such as increased inertial mass, viscous effects, errors in the counterweight mass, friction. Furthermore, even

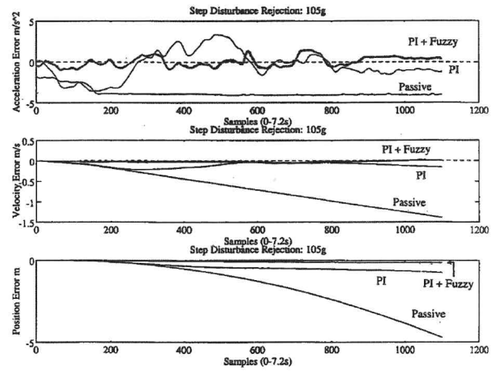

Figure 10: Experimental results

though the proposed scheme is not able to deal with dynamics due to higher frequency behaviour which are not linear, it shows a quite satisfying behaviour thanks to low pass filtering.