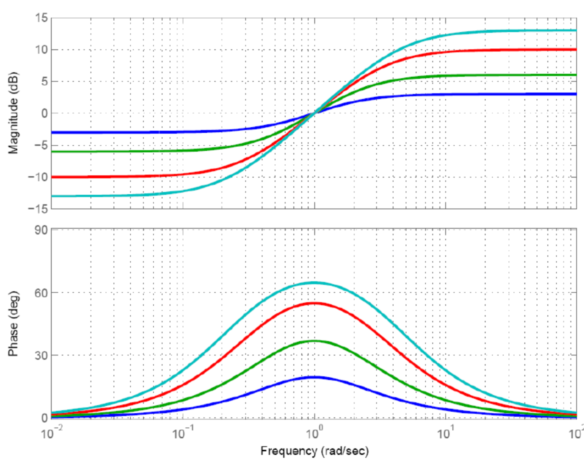
TEL AVIV UNIVERSITY
אוניברסיטת תל אביב

Faculty of Engineering

<u>Lab Brief – Velocity Control</u>



1. Lab Purpose:

- Understanding the principals involved with first-order dynamic systems.
- Experimenting with system-identification procedures.
- Knowing the Bode diagram and its components.
- Design a controller based on frequency-response requirements (lead/lag compensator).

2. System Description.

2.1. Quanser Qube USB Interface.

This is a smart box containing a motor, an encoder, amplifier and additional circuitry. The box receives power from a regular AC/DC adaptor, and is connected to a computer through a USB-B cable. The encoder, amplifier and microprocessor are stationed inside the box. The processor reads the motor angle and sends it to the computer, and receives from it the level of voltage it should supply to the motor. That voltage is transferred to the motor via the amplifier, which gets its power from the adaptor.

The motor shaft can be connected to various expansions using magnets, and in this experiment we shall use a red aluminum disc (for inertia).

Figure 1: Experiment system

3. Theoretical background.

3.1. Dead-Zone.

Permanent Magnet Brushed DC Motors actuate by using the Lorenz force. They run current within their coils which are stationed inside a magnetic field generated by the magnet. In order to move, one must apply a voltage difference to the motor. Every motor on earth has a certain voltage threshold required for it to move. Below that threshold, there will be no movement, and only above it the motor shaft will start spinning. That threshold is known as the dead zone. Adding or subtracting a constant amount of voltage to each spin direction allows us to move the characteristic lines of the dead zone, in such a way that we could narrow it down to zero, or even more! The larger the bias, the higher the lines move (in absolute value), yet they keep their incline, becoming closer to the shape of a linear function.

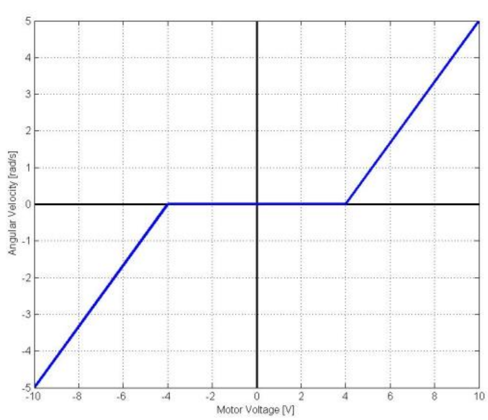


Figure 2: Dead-zone characteristic curve

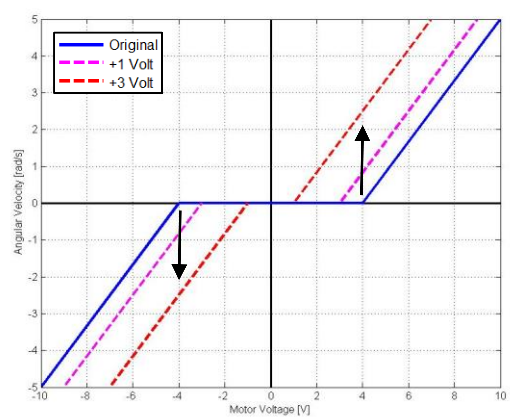


Figure 3: Biased dead-zone curves

3.2. System Identification.

There exist numerous ways of creating a mathematical model of a control system. Many times there are unknown parameters (such as viscous friction or motor torque constant), or parameters with uncertainties (like the moment of inertia of the motor shaft, or even the load disc – what is it made of, its density, exact shape, etc.).

This data is usually provided, if at all, by the manufacturer, and they are not precise. Measuring them requires expensive industrial equipment. Adding calculations such as gear ratios and efficiencies makes the calculation very complex and probably not entirely correct.

For that reason, sometimes it is much better to perform measurements on the complete system and from them derive the components that define the transfer function. This method is good for modeling a simple, inherently stable system, like the one in out experiment (trying to do so for an inverted pendulum, for example, would be a bad idea). The method we would use, which is called a Bump-Test, requires only a transfer function form for which we try to make the approximation, and to apply a step input.

The form we would want to approximate to is: $\frac{y}{u} = \frac{\Omega_{(s)}}{V_{(s)}} = \frac{K_{sys}}{\tau \cdot s + 1}$.

The parameter K_{sys} is called the system gain (or DC gain), and τ is called the time constant. It is convenient to express the transfer function in this manner, because if we insert a step input: $V_{(s)} = \frac{R}{s}$, and then apply an Inverse-Laplace transform, the result will be (in the time domain): $\Omega_{(s)} = \frac{K_{sys} \cdot R}{s(\tau \cdot s + 1)} \rightarrow L^{-1}[\ ] \rightarrow \omega_{(t)} = K_{sys} R \left(1 - e^{-t/\tau}\right)$.

When, of course, <u>the value t is measured from the instance the step input begins.</u>

It is accustomed to apply a small step input, wait until the system reaches steady-state, and only then apply a bigger step response to make the measurements (see figure 4).

There exist a generalization of the equation, taking into account the time which the system was turned on, by adding a step time shift, but this is not necessary for the lab's understanding or application.

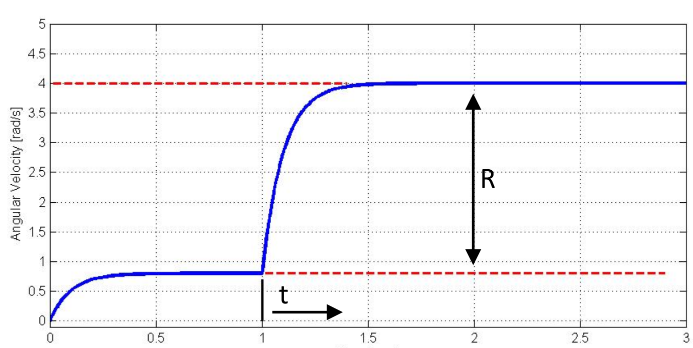


Figure 4: Characteristic system response for a classical bump-test

3.3. Bode Plots and Lead/Lag Compensators.

A Bode plot is the result of separating the Nyquist plot into two graphs – one represents the gain (the distance of a point from the origin in Nyquist) in decibels ($x[dB] = 20\log_{10}(x)$), and the second graph represents the angle between a line connecting that point with the origin and the positive side of the real axis in the complex plain. The meaning of the gain in terms of transfer functions is the ratio between the input and output amplitudes, and the meaning of the angle (phase) is a time delay between the input and output. Considering the fact that this delay is measured in degrees, <u>the value of the phase depends on the input frequency.</u>

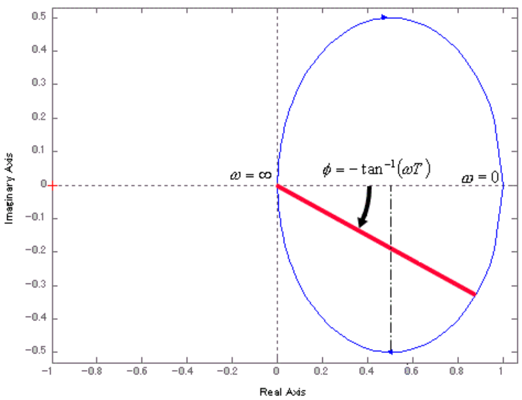


Figure 5: A general Nyquist plot.

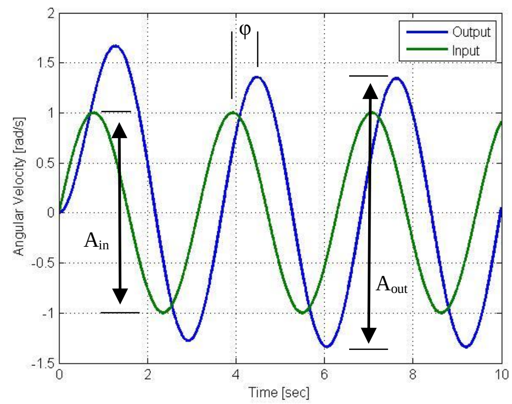


Figure 6: Input and output amplitude and phase

In control theory, we use components called Lead / Lag Compensators in order to modify our system's response according to our needs. There is of course a tradeoff between the response characteristic, which we would have to take into consideration.

One way of expressing a lead compensator is: $\frac{1+\alpha T\cdot s}{1+T\cdot s}$, while α>1. In that form, one should choose the frequency in which the phase is to be lowered, that would be ω_m. Then, α is calculated by the amount of phase needed to be lowered, in the following way: $\varphi = \arcsin\left(\frac{\alpha-1}{\alpha+1}\right)$.

Finally, T is calculated by: $T = \frac{1}{\omega_m\sqrt{\alpha}}$, which hands us the compensator.

Note that using this compensator results in amplification of the gain in higher frequencies and suppression of the gain in lower frequencies, which changes the cut-off frequency (likely moving to the right). To fix this we have a few options: adding a constant gain to the compensator (raising or lowering the entire gain graph as it is), or adding more phase then needed to begin with.

Using this method, the gain change at ω_m is $\sqrt{\alpha}$ [dB].

4. Pre-Lab Questions.

a) Draw the graph depicted in figure 3, should we add a bias of 4[v], and of 6[v].

- Is it desired to have a definite zero dead zone in a real system? Explain.
- What behavior problems are expected from a system with negative dead zone?
- Investigate – why do we even have dead zone in motors?

b) Look at section 3.2, assume step size of R=1, and answer:

- How can we calculate the system gain from its response? Show this by using the transfer function (frequency domain).
- How can we know what the time constant is? Show by using the time domain.

c) Assuming we do not manipulate the input signal, why do we first insert a small step, wait for it to settle, and only then insert the real step? What are we trying to avoid?

d) Assuming we do manipulate the input signal, what other method can we use to avoid that same problem, so we can insert the real step right from the beginning?

e) A transfer function is given in the following form: $P = \frac{a}{s+b}$. Find an expression for the final value of the output, assuming R=1 (Note, in an open-loop there is no meaning for the term “error”).

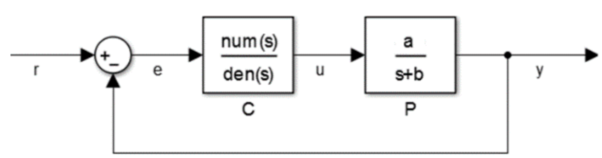
f) A sinusoidal input with a frequency of ω_0 [rad/s] is inserted to that same system, and between the peaks of the input and output waves a time delay of t_0 was measured. Derive a formula to calculate the system phase (in degrees), depending on ω_0 , t_0 .

g) <u>For this question alone</u>, substitute in that same system with: a=7, b=3.

- Use your formula from the previous question to find the phase for an sine input with a frequency of 10 [rad/s].
- Design a lead compensator H that would bring the system to a phase-margin (PM) of 150° without changing the cut-off frequency.

h) To the function from question e we add a PI controller of the form: $C = k_p + \frac{k_i}{s}$,

closing the loop in the following manner:



- Find an expression for the transfer function between the input r and the output y, depending on k_p , k_i , a, b.
- Find an expression for the transfer function between the input r and the error e, depending on k_p , k_i , a, b.

5. Lab Process.

First thing's first – inside Matlab, make a copy of the folder "Velocity Control" and change the name of the copy to your names. Enter that folder.

5.1. Fixing the Dead-Zone.

Open the Simulink model: "q_find_deadzone.slx":

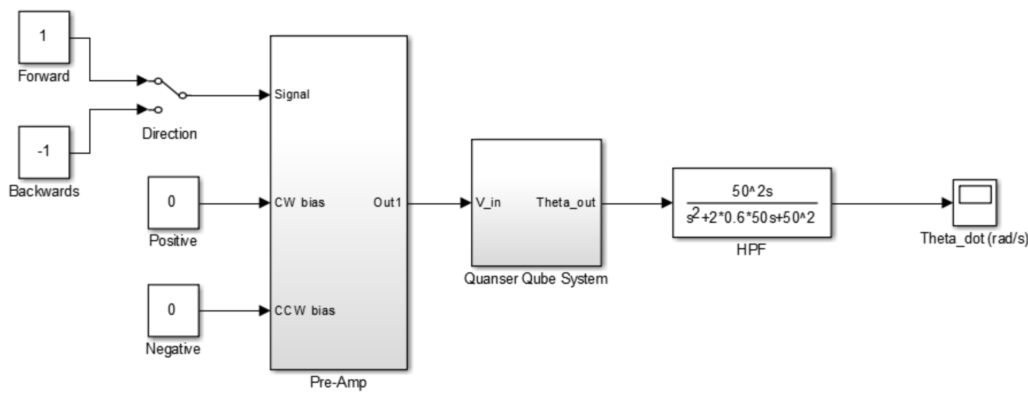


Figure 7: System model in an open loop

The model represents an open loop of the system. Before we model it, we would like to bring the dead zone effects to the minimum. We will try to do so by adding a constant bias to the motor voltage in each spinning direction via a "Pre-Amplifier".

- Make sure that the "Direction" switch points to "Forward".
- At the top menu, click: Quarc → Build. The model will compile.
- At the top bar click "Connect to target" (??) and then "Play" (??).
- Change the value inside "Positive" block by increments of 0.01 [V] until the motor shaft starts spinning. When this happens, subtract 0.01 from that value and write it down. Return the value to 0.
- Move the switch so that it now points to "Backwards".
- Repeat the previous stages, this time with the "Negative" block.
- Keep the 2 values you found (BEFORE subtraction) for the post-lab report.
- Choose the smaller value of the two and save it as a variable in the Matlab workspace under the name "pre_amp" (case-sensitive).

5.2. System Identification.

At this stage we would want to perform a bump test in order to identify the system characteristics. In order to accomplish that, we will insert a step input signal and measure parameters to define a transfer function of the form: $\frac{K_{sys}}{\tau \cdot s+1}$.

Open the Simulink model: “q_velocity_control.slx”.

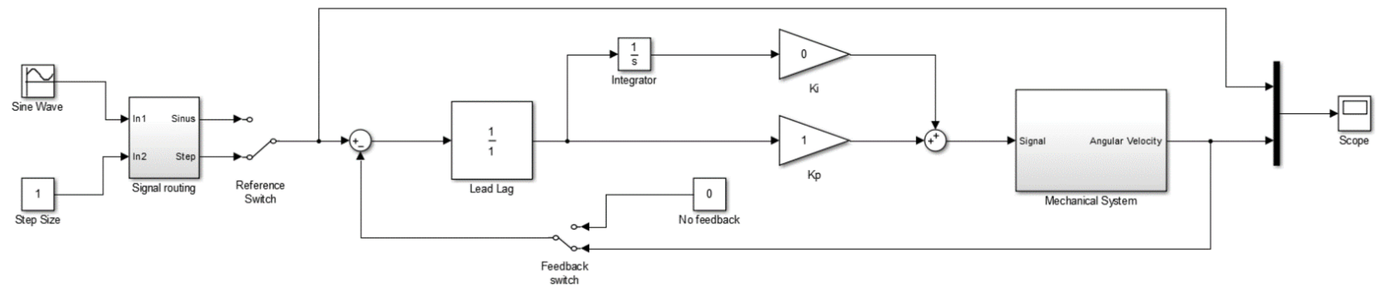


Figure 8: System model for the lab process

- Make sure that “Reference Switch” is pointing downward to “Step”, and that the step size is 1.
- Make sure that the “Feedback Switch” is pointing at 0 (meaning that the feedback is nulled, which means this is an open loop diagram).
- Compile the model by clicking: Quarc → Build.
- Connect to target, Play the model and wait a few seconds.
- When you believe that the system has reached steady state, click Stop.
- In Matlab, run the script “Plot_velocity.m” in order to plot the system response.
- Identify the input curve and the output curve and edit the graph legend so that it will correctly point out the curves (include units!).
- Use the display marker to pin all the relevant data points for calculating the desired unknown parameters (to pin more than 1 point, press and hold left Alt).
- Save the plot for the post-lab report.
- Calculate the system’s transfer function.

5.3. Analyzing the System's Behavior.

In the next stage, we will use the previously found system in order to compare theory to practice. Using the same Simulink model as in the previous section, use your calculations from the pre-lab questions and perform the following tasks:

a. Make sure you still work in an open loop. Run the system a number of times using different inputs and fill in the following table. In that table, Step input is the size of the step, Expected output is the actual value of the steady-state response (the measurement is noisy, pick a general point), Expected output is the steady-state value that the transfer function would reach using the finite-value theorem, and finally, the time it takes for the output to reach the given percentage of the ss value.

Step input [rad/s]	Measured output [rad/s]	Expected output [rad/s]	Time to 63.2% [sec]
1			
2			
3			

You may remove the input curve, if it exists. Plot 1 figure with 3 graphs, marking points of interest.

b. Close the loop by changing the feedback switch so that it points at the feedback signal, instead of 0. Run the system a number of times with a constant step size of 1, a Ki value of 0, and varying proportional gains, Kp. First, you need to reverse-calculate the gains by using the expressions found in the pre-lab.

K_p	Measured SS error [rad/s]	Expected SS error [rad/s]
		0.5
		1/3
		0.1

Save a plot of the last run you made, with points of interest.

c. Make sure the step size is 2, and that Kp=1. Read the section all the way through.
- Use the expressions from the pre-lab in order to analytically calculate the Ki value that is supposed to bring the identified system to critical dampening.
- Run the system after inserting the said Ki value into the model. There is no need to save a plot.

- Run the system a few more times in order to find the actual critical Ki, for which the response moves from being overdamped to underdamped.
- Run the system again 4 additional times, twice with a lower value than that found in the previous part and twice with a higher value. **Keep Ki between the interval [0.3..5].**
- Eventually, you need to finish this section with 1 figure containing 5 graphs: one of the response with the calculated critical Ki, and the four higher/lower values. Add a legend to each curve, and save the values of the calculated and empirical critical Ki's for the post-lab report.

5.4. Bode Plot.

Before the beginning of this section, make sure that in the Simulink model it is defined:
Step=2, Lead/Lag=1/1, Ki=0, Kp=1, and that the loop is OPEN.
Move the input switch so that it points towards a sine input.
Open the "Sine wave" block and make sure that:
Amplitude=0.5 [rad/s] , Bias=2, Frequency=5, Phase= -2*Frequency.
The phase needs to be changed every time you change the frequency, in order to synchronize between the input and output. <u>Attention, there is no connection between the said phase and the bode-plot phase, these are two completely different things.</u>

- Compile the system using Quarc → Build.
- Run the system and plot the graph using the "Plot_velocity.m" script.
- Inside that plot, mark up 1 local minimum and 1 local maximum on each curve (hint: not the first ones…). From this data, calculate the gain (in dB) and phase lag (in degrees) for the given frequency.
- Save the plotted figure and repeat the section again for 0.5, 1 [rad/s]. **Don't forget to change both the frequency and the phase in the block.**
- In Matlab, create a bode plot of the identified system. Find the gain and phase values for a frequency of 0.5 [rad/s]. Do these numbers match the calculations made in the previous part?
  If not, check your calculations, if they do, move on.

5.5. Lead Compensator.

- Use the pre-lab answers in order to design a lead compensator, so that it pulls the phase lag of the system at 0.5 [rad/s] to **zero** degrees.
- Plot a Bode of the system along with your compensator and make sure the phase is in fact 0. Find the numerical gain value at the same frequency.
  Inverse that number, and multiply the result with your compensator in order to bring the Bode gain to 0 [dB].
- Plot 1 figure containing 3 Bode plots: one of the unchanged system, one with the system after adding the compensator, and one with the compensator and gain change. Add a relevant legend and mark points of interest.
- <u>After getting the instructor's approval,</u> insert the gain-inverse to the Kp block and your compensator to the Lead/Lag block. Make sure you still work on a frequency of 0.5 [rad/s].
- Compile the model.
- Run the model and watch what's happening. Save a plot of the input and output together, and calculate the peak-to-peak gain and phase lag.
- Have we accomplished our goals? If not, keep adjusting the controlled until you manage to do that.
-

6. Post-Lab Questions.

A. Draw (by hand or using Matlab) a graph of the system's dead-zone, before and after the correction. Why were you asked to take 1 step backwards?

B. Present the plot taken at section 5.2, along with a detailed derivation of the transfer function that had been identified.

C. Present the table filled in the beginning of section 5.3.

   - Within the tested limits, would you say that the modeling made in the lab is reasonably accurate?
   - How would you explain the change in ss value vs the change in rising time between each run?

D. Present the table filled in the middle of section 5.3.
    - What is the smallest and largest possible ss error for the system in an open loop? And in a closed loop?
    - Assuming that the motor must not receive more than 10 [V] in any given moment, what would be the maximal Kp gain allowed for a 1 [V] step input? What would be the ss error in this case?

E. Present the plot saved in the end of section 5.3.
    - What can you tell about the settling speed with reference to the Ki value (at the lower values)?
    - What response characteristic did you get when entering Ki values larger than the critical one? How is that possible if we work with a first-order transfer function?
    - For the 3 highest Ki values chosen, and Kp=1, calculate the system's closed-loop natural frequency (ω_n) and dampening coefficient (ζ) approximately. From them, calculate the time for reaching first peak, and compare with the graphs taken. From what does the difference come from?

F. Present all the plots and calculations made at section 5.4.
    - What's the conclusion? What would we get if we were to repeat this section an infinite amount of times, for an infinite amount of frequencies, and put all the results on a graph?
    - What are the units of the output's amplitude? What are the units of the input's frequency? Is there any connection between them?
    - What would happen at the motor shaft if we wouldn't have entered a bias and feed the sine waves around 0? Why did we want to avoid that? (hint: through which non-linear element would we cross?).

G. Present the Bode plots taken at section 5.5 and the calculations made to reach the desired compensators.

- Explain verbally what you did during the section and why.
- Present the plot of the last system run, together with the compensator.
- After adding the compensator, supposedly, the system gain is 1 and the phase lag is 0. Watch the response and Bode plot and try to guess, why aren't the tow curves sitting one on top of each other?
- Try to guess, why were you asked not to mark the first peak?
- It is now given that the system input signal was modified in order to conduct this experiment, by adding the following transfer function before the input reaches the motor: $\frac{0.0109s+0.0837}{s+1}$. Find the true transfer function of the motor, express it in the form: $\frac{K}{\tau \cdot s+1}$.