

Bechmarking control algorithms with current saturation aware references

Robot Programming and Control

Filippo Grotto VR460638

January 31, 2022

Contents

| | | |
|----------|-----------------------------------|-----------|
| 1 | Introduction | 1 |
| 2 | Noise Sensitivity Function | 2 |
| 3 | Control Architectures | 3 |
| 3.1 | Parameters Used | 4 |
| 3.2 | Position Control | 4 |
| 3.2.1 | Modulated reference | 5 |
| 3.3 | Velocity Control | 6 |
| 3.3.1 | Modulated reference | 7 |
| 3.4 | Basic Force Control | 8 |
| 3.4.1 | Modulated reference | 9 |
| 4 | Limitations | 9 |
| 5 | Other approaches | 10 |
| 6 | Conclusions | 10 |

1 Introduction

The aim of Forecast project [1] is to provide a testbed to benchmark control algorithms and get in this way comparable results especially in the context of force control algorithms. The procedure uses several performances indicators to assess the controller in use such as static error, dynamic error, overshoot and bandwidth. The latter is estimated using a sinusoidal function that increments frequency over time (e.g. 1 to 10Hz in 10 seconds) until we reach -3db over the desired reference. However this estimation suffers from the presence of saturation of the motor used to drive the experiments making the result about the controller not as general as we want. In the following section we will propose a method to reduce the amplitude of the reference signal before reaching the motor saturation in order to get a much more precise bandwidth estimation using general transfer function methods in case of fully linear systems.

2 Noise Sensitivity Function

If we consider the general feedback system in Fig.1. There are lots of transfer function that we are possible to define depending on our input-outputs pairs. In our cases we will consider $F(s) = 1$.

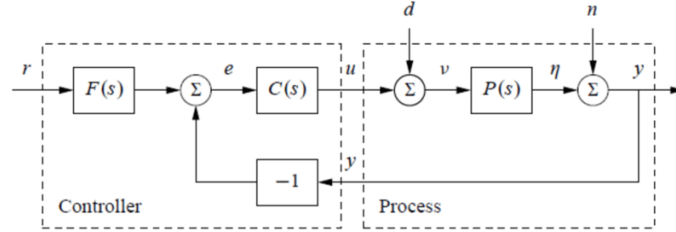


Figure 1: A general SISO feedback system taken from [2]

We are actually interested in having the relation between the input of the system r and the control input u in order to know when the current saturation is reached. We will assume our $P(s)$ plant known since we can identify all the parameters of our DC motors assuming it to be a linear system.

$$CS(s) = \frac{C(s)}{1 + C(s)P(s)} \quad (1)$$

The procedure proposed for the current saturations is derived considering the motor saturation u_{sat} usually provided in the datasheets:

$$\frac{u}{u_{sat}} < 1 \quad (2)$$

The equation (2) holds since we want u to not overcome the provided saturation. However a more strict condition can be imposed. We can then write:

$$\frac{u}{u_{sat}} = |CS(s)| \frac{r}{u_{sat}} \quad (3)$$

Considering the worst case scenario when $u/u_{sat} = 1$ we can derive:

$$r < \frac{u_{sat}}{|CS(s)|} \quad (4)$$

An equivalent condition for unitary input is:

$$\frac{|CS(s)|}{u_{sat}} < 1(0db) \quad (5)$$

Note that equation (4) and (5) use mixed time and laplace domain notations. In the following sections only the magnitude of the transfer function will be considered.

3 Control Architectures

Our aim is to replace the input of the different architectures with a correctly scaled sinusoidal function (sweep) which guarantees that motor saturation is not exceeded during the simulated experiment. To do so the signal is firstly processed by some matlab code which scaled it using the presented equations secondly the simulink model is considered using the generated signal and current/tau is considered.

The following code was used to generate the reference signal:

```
% Calculate time and frequencies vectors (they must have the same length)
t = start_T:dt:end_T;
freq = start_freq + ((end_freq-start_freq)/duration * t);

% Build original swept function
sweep = amplitude * chirp(t, start_freq, end_T, end_freq, 'linear', -90);

% Compute the magnitude vector /m
[magnitude_vector,~] = bode((CS*amplitude)/(u_sat), 2*pi*freq);

% Calculate deamplification factor only when the magnitude is larger than
% 0db (1 in linear scale).
deamplification = zeros(1,size(magnitude_vector, 3));
for i = 1:size(magnitude_vector,3)
    if magnitude_vector(:, :, i) >= 1
        deamplification(i) = amplitude/magnitude_vector(:, :, i);
    else
        deamplification(i) = amplitude;
    end
end

% Build the suggested sweep
suggested_sweep = deamplification .* chirp(t, start_freq,
                                           end_T, end_freq, 'linear', -90);
```

Briefly, the idea is to use the magnitude of the transfer function (6) to scale the reference only when we exceed 0db (1 in linear scale). That means we allow the user to define the reference and we scale only when we can't guarantee the saturation limits (forcing, in that scenario, the maximum current available).

$$\frac{|CS(s)|amplitude}{u_{sat}} \quad (6)$$

3.1 Parameters Used

The parameters used for the motor comes from direct identification of a maxon DCX22L, in particular $J_m = 0.0064 \text{ Kg m}^2$, $d_m = 0.0068 \text{ Nms}$, $K_t = 1.46$ and $u_{sat} = 2.26 \text{ A}$. The controller used for all the architectures is a PID/PD controller with different parameters.

3.2 Position Control

Let's consider a basic position control architecture for a linear motor model considering only the mechanical subsystem. The related equations of the plant P_l and the PD controller C are:

$$P_l = \frac{1}{J_m s^2 + d_m s} \quad C = P + Ds$$

The two transfer functions $T(S)$ (r to y) and $CS(s)$ (r to u) are presented in Fig 2.

For simplicity only few cases to fully understand the potential of this approach are reported. Other experiments can be performed to get familiar with how this approach works in real scenario. Moreover more sophisticated motor model with the electrical subsystem can be considered.

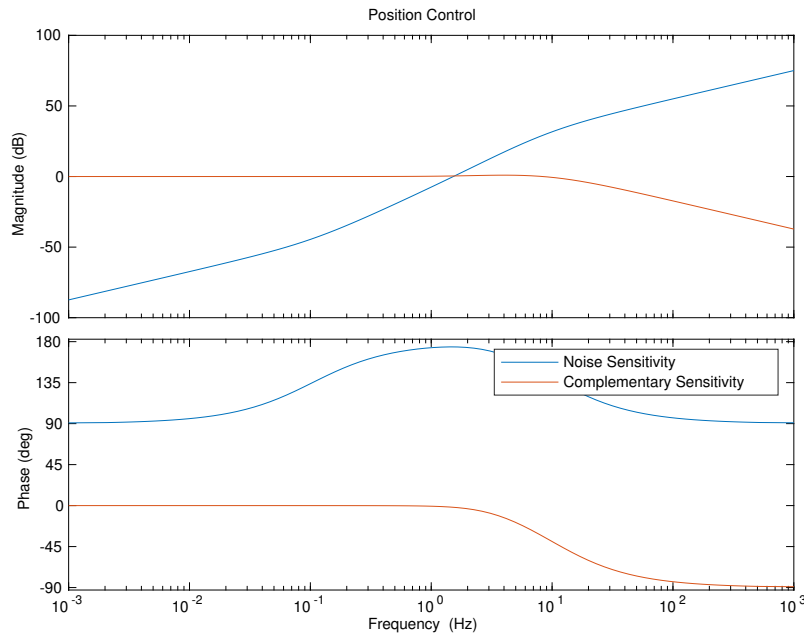


Figure 2: Position control bode plot of the complementary sensitivity function and noise sensitivity function

3.2.1 Modulated reference

As it is visible the reference requires high current in order to be performed, the code scaled the signal as soon as we reached the saturation in order to guarantee the saturation limits.

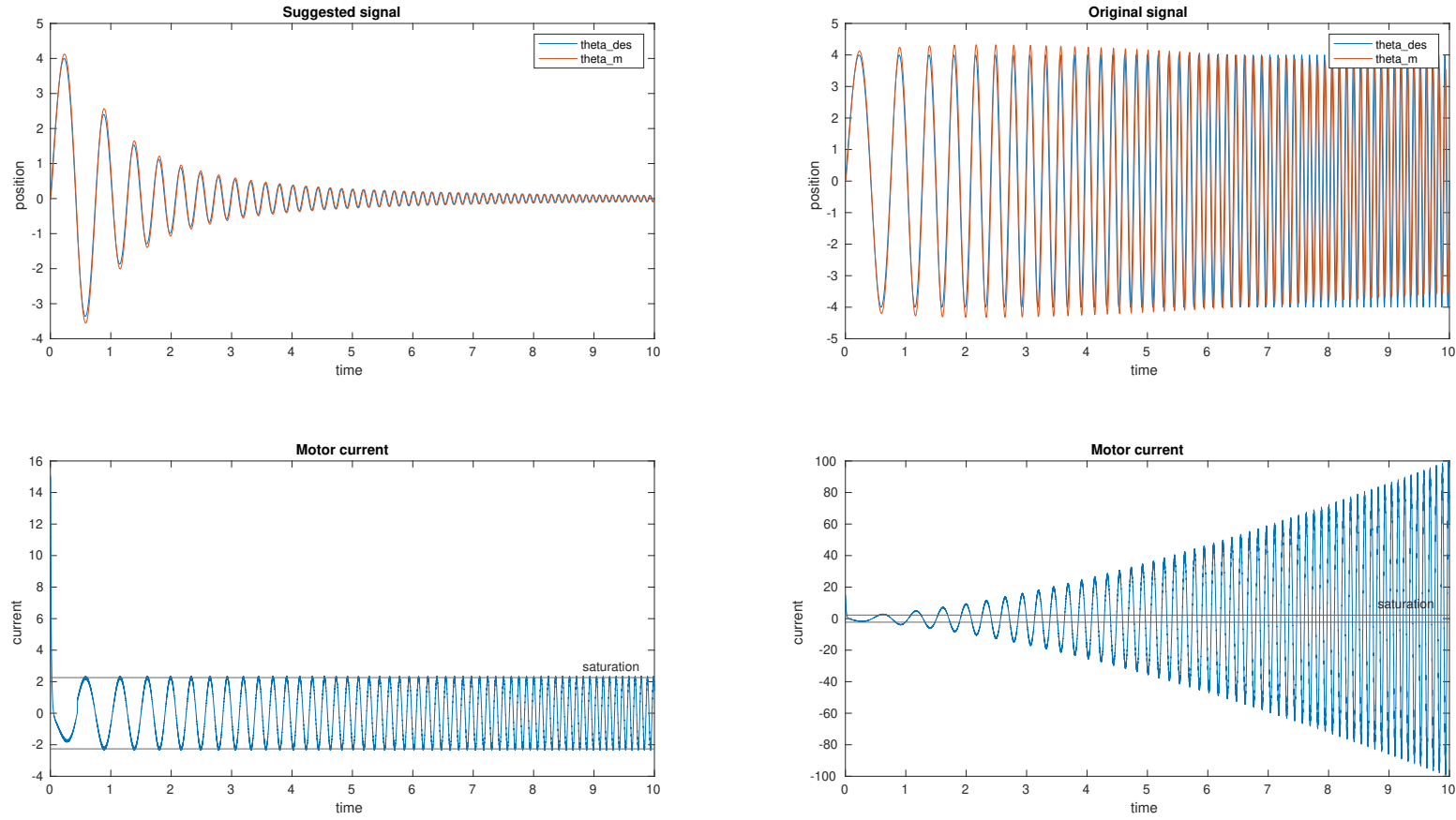


Figure 3: Comparison of the original position signal with the suggested saturation aware one

To obtain Fig 3 a PD with $P = 10$ and $D = 0.9$ was used. The current saturation is 2.26A, and the sweep signal goes from 1 to 10 Hz in 10 seconds. The motor used is the one described in the parameters section.

3.3 Velocity Control

Let's consider a basic velocity control architecture for a linear motor model considering only the mechanical subsystem. The related equations of the plant P_l and the PI controller C are:

$$P_l = \frac{1}{J_m s + d_m} \quad C = P + \frac{I}{s}$$

The two transfer functions $T(S)$ (r to y) and $CS(s)$ (r to u) are presented in Fig 4.

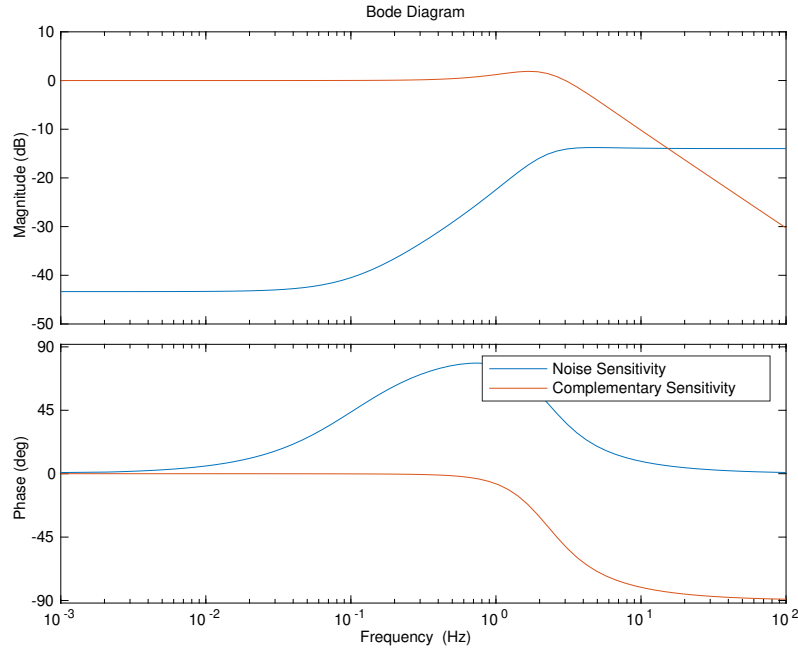


Figure 4: Velocity control bode plot of the complementary sensitivity function and noise sensitivity function

It is important to notice that the noise sensitivity function in Fig 4 doesn't reach the 0db level so it means we can't reach saturation with unitary references. However the amplitude of the reference (when > 1) will shift the transfer function and it will eventually reach 0db leading to current saturation.

3.3.1 Modulated reference

As it is visible the reference requires high current in order to be performed, the code scaled the signal as soon as we reached the saturation in order to guarantee the saturation limits.

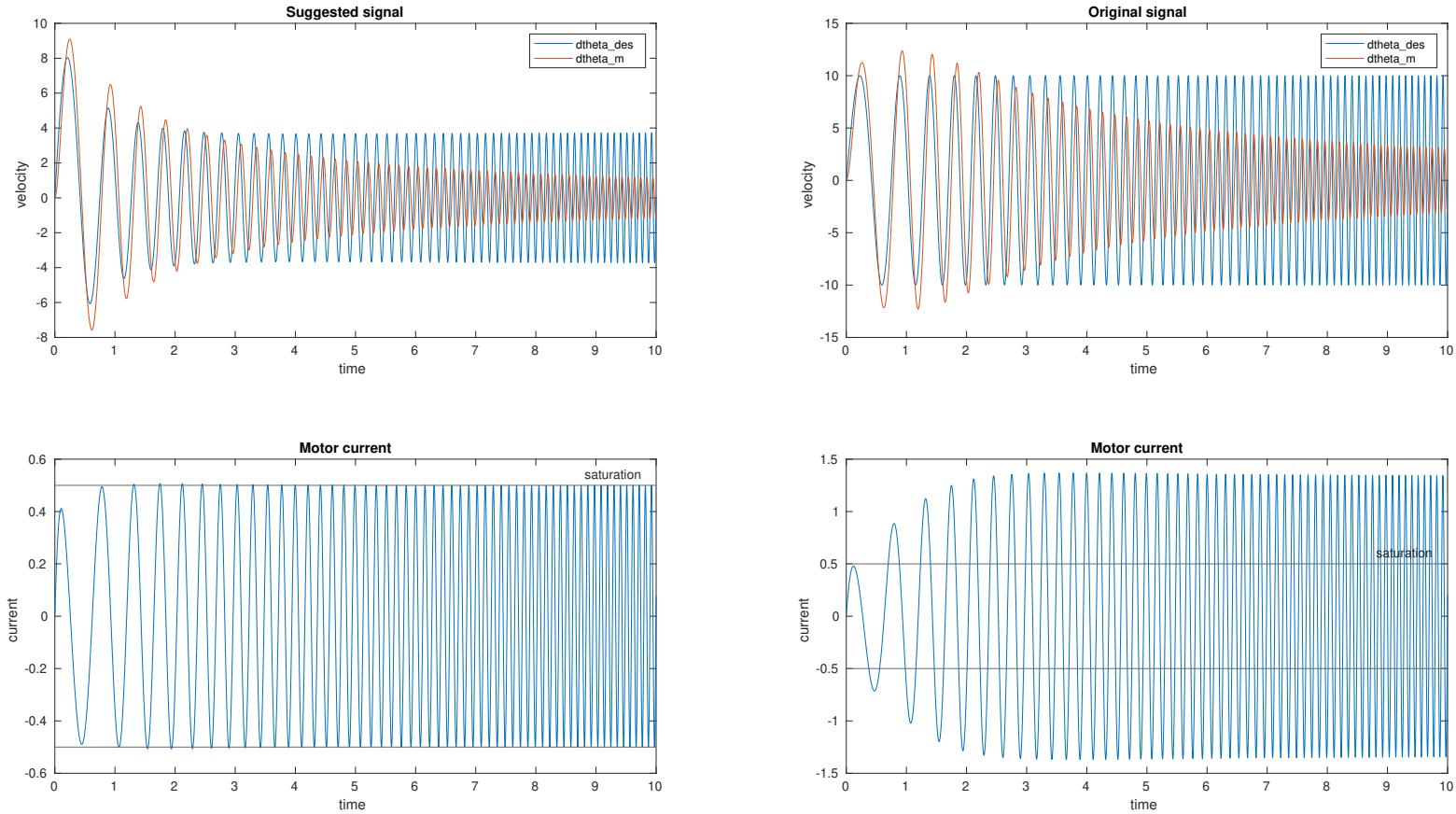


Figure 5: Comparison of the original velocity signal with the suggested saturation aware one

To obtain Fig 5 a PD with $P = 0.2$ and $I = 2$ was used. The current saturation is 0.5A, and the sweep signal goes from 1 to 10 Hz in 10 seconds. The motor used is the one described in the parameters section.

3.4 Basic Force Control

Let's consider a basic force control architecture for a linear motor model considering only the mechanical subsystem and a PD controller. I will assume only some stiffness h following the analysis in [4]. The related equations of the plant P_l and the PD controller C are:

$$G = P_l = \frac{1}{\frac{J_m}{h}s^2 + \frac{d_m}{h}s + 1} \quad C = P + Ds;$$

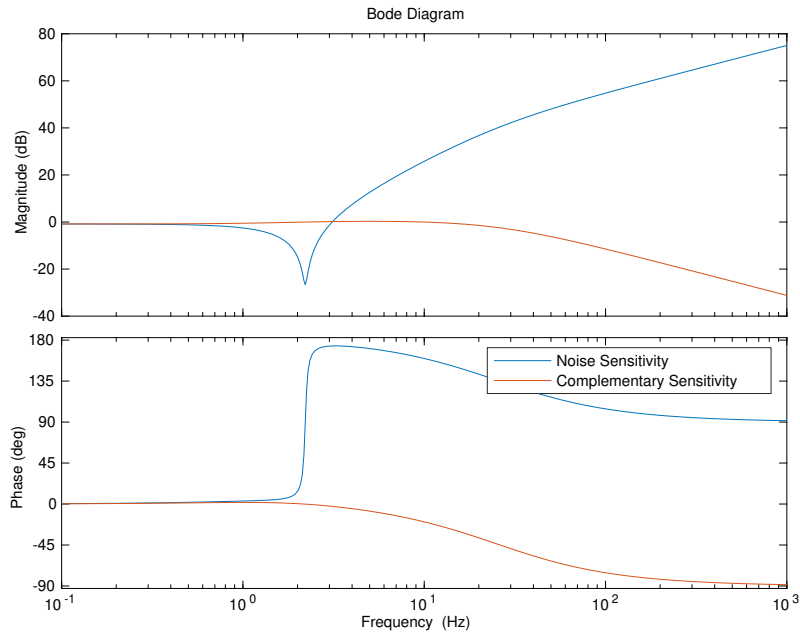


Figure 6: Force control bode plot of the complementary sensitivity function and noise sensitivity function

It is important to notice that the peak in the noise sensitivity function in Fig 6 depends on the stiffness of the environment and it can't be eliminated. However our method consider the amplification only when the transfer function exceeds 0db so the peak is always neglected.

3.4.1 Modulated reference

As it is visible the reference requires high current in order to be performed, the code scaled the signal as soon as we reached the saturation in order to guarantee the saturation limits.

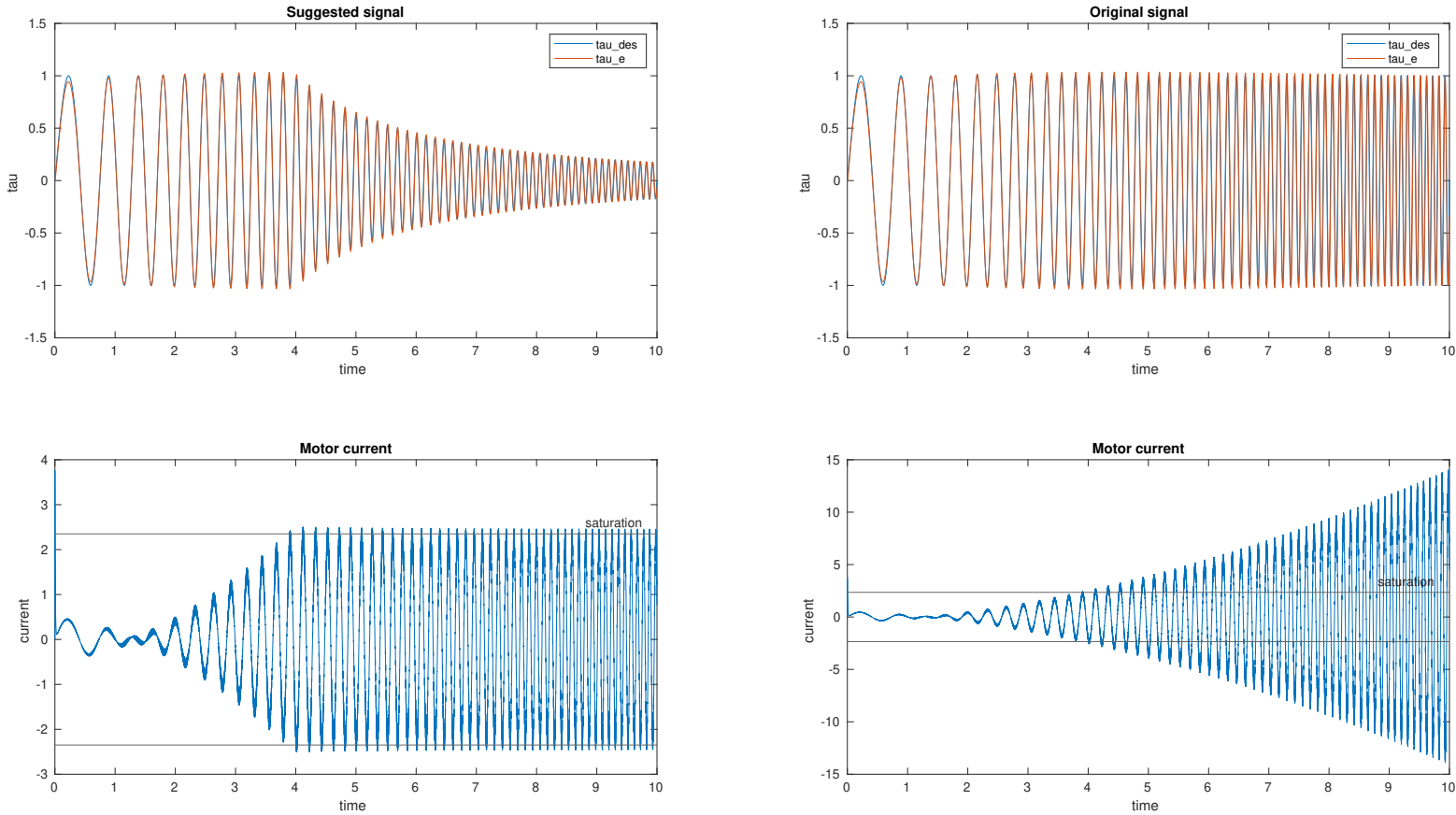


Figure 7: Comparison of the original force signal with the suggested saturation aware one

To obtain Fig 7 a PD with $P = 10$ and $D = 0.9$ was used. The current saturation is 2.26A, and the sweep signal goes from 1 to 10 Hz in 10 seconds. The motor used is the one described in the parameters section. The stiffness of the environment is $2Nm/rad$.

4 Limitations

The current approach based on the use of the transfer function $CS(s)$ has lots of limitations:

- The approach assumes full knowledge of the considered system, everything has to be identified correctly and this is even a bigger limitation for force control because we don't have informations about the environment which makes the use of the transfer functions not reliable at all.

- We assume a linear system which is a big limitation since it doesn't consider non-linear behaviours of the DC motors. For this cases having at least one real experiment to validate the data is a must.

5 Other approaches

Simulating the experiments in simulink is in general a good way to verify the presence of saturation but it is affected by the same problems related to prior knowledge and linear behaviours (at least without implementing a non-linear system). A good way to generate a current saturation aware reference signal is adapting the reference in real time by reading directly the current provided to the motor. If this exceeds a certain threshold just drop a certain percentage. The bandwidth estimation can be performed using the Matlab toolbox over that data.

6 Conclusions

The report is an attempt to provide a way to generate current saturation aware signals using noise sensitivity transfer function. The reported approach is pretty flexible and can be extended to guarantee also voltage or velocity limits as in [3]. The basic idea was provided as well as some simulation results. The main limitations and other possibilities were discussed and analysed.

References

- [1] Forecast Project <https://eurobench2020.eu/developing-the-framework/force-control-algorithms-testbench-forecast/>
- [2] Karl Johan Astrom, Richard M. Murray Feedback Systems An Introduction for Scientists and Engineers
- [3] Chan Lee, Sehoon Oh Performance Analysis of Series Elastic Actuator based on Maximum Torque Transmissibility
- [4] Eppinger, Seering Introduction to Dynamic Models for Robot Force Control