Robust control of a laboratory servomechanism

Petr Urban, Zdeněk Hurák*+, *Michael Šebek*+

*Department of Control Engineering, Faculty of Electrical Engineering,
and

+Centre for Applied Cybernetics
Czech Technical University, Prague, Czech Republic

Abstract

This report documents the application of some recent linear matrix inequalities (LMIs) - based methods for design of a robust controller for a laboratory servomechanism. The report is application-oriented and therefore no theoretical backgroung is developed. Instead, references are made where references are due and matlab-like style is followed.

1 Introduction

This report documents the application for design of a robust controller for laboratory servomechanism with function *ptopdes.m*. The laboratory servomechanism is used for control education at the Department of Control Egineering CTU in Prague. Students identify and design feedback controllers. A set of new functions are being included to the new release 3.0 of the Polynomial Toolbox [1]. The new functions use optimization over linear matrix inequalities (LMIs) to solve various robust control problems [4].

2 System description

Classical DC servomotor with adjustable rotational friction and a set of removable disks bringing about an uncertainty in the description, namely uncertain rotational friction and uncertain moment of inertia.

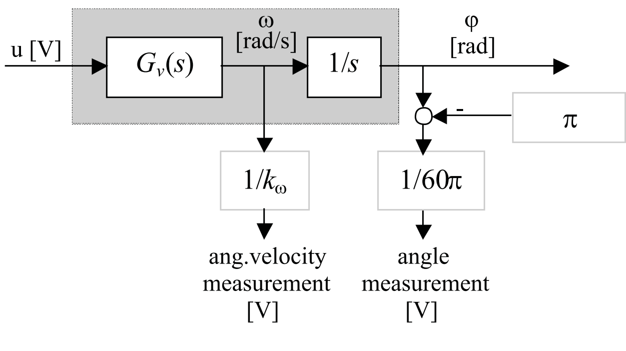

Figure 1: Block diagram

The plant input - voltage $u \in [-1, 1][V]$ applied to the armature of a DC motor (after amplification in the multifunction I/O card PCL-812).

The plant output - angular position $\varphi \in [0, 2\pi][rad]$ of the shaft and/or the rotational velocity $\omega[rad/s]$. The rotational velocity measured by the tachogenerator is scaled by $\frac{1}{k_\omega}$. Potentiometer measures the angle of the shaft with ratio $\frac{1}{60}$, such that $0[rad]$ coresponds to $-\frac{1}{60}[V]$, $\pi[rad]$ corresponds to $0[V]$ and $2\pi[rad]$ corresponds to $\frac{1}{60}[V]$.

The system parameters

```
L = 0.75e-3;         % armature inductance
R = 6.2;             % armature resistance
U0 = 10;             % nominal input (armature) voltage

J0 = 30e-6;          % nominal moment of inertia
J1 = 75.6e-6;        % additional inertia due to the disk #1
J2 = 330.5e-6;       % additional inertia due to the disk #1
J3 = 1.387e-3;       % additional inertia due to the disk #3

k = 0.032;           % motor constant
k_omega = 27.11;     % tachogenerator (scaling) constant

B0 = 3e-5;           % nominal (rotational) friction
B5 = 0.000042;       % additional friction for setting 5 mm
B10 = 0.000195;      % additional friction for setting 10 mm
B15 = 0.0004;        % additional friction for setting 15 mm
B(1) = B0; B(2) = B0 + B5; B(3) = B0 + B10; B(4) = B0 + B15;
```

2.1 Linear differential equations

The system is linear, classical DC servo system:

$$u(t) = Ri(t) + L\frac{di(t)}{dt} + k\frac{d\varphi(t)}{dt}$$
$$J\frac{d^2\varphi(t)}{dt^2} = ki(t) - \left(B_0 + B_{5\ (or\ 10,15)}\right)\frac{d\varphi(t)}{dt}$$

Neglecting the electrical circuit response (setting $L = 0$), we get

$$J\frac{d^2\varphi(t)}{dt^2} + \left(B_0 + B_{5\ (or\ 10,15)} + \frac{k^2}{R}\right)\frac{d\varphi(t)}{dt} = \frac{k}{R}u(t)$$

2.2 Transfer function - nominal model

From the armature voltage to the angle of the rotor shaft - assuming zero initial conditions, the transfer function (from $u(t)$ to $\varphi(t)$) for disk $\sharp 2$ and (rotational) friction indicator set to $5\,mm$:

```
J = J0+J2;              % system with disk #2
B = B0+B5;              % total rotational friction

num3 = k/(L*J)
num3 =
       1.1835e+005

den3 = s^3 +  (R/L + B/J)*s^2 + (k^2/(L*J)+R*B/(L*J))*s
den3 =
       5.4e+003s + 8.3e+003s^2 + s^3
```

The corresponding simplified model (neglecting the electrical response, $L \to 0$) is

```
num = k/(R*J), den = s^2 + (B+k^2/R)/J * s
num =
      14.3170
```

```
den =
      0.66s + s^2
```

Comparing both transient (step, impulse) and frequency (bode) response we conlude that second order model given by

```
G = rdf(num,den)        %right denominator fraction
G =
    14     /      0.66s + s^2
```

is sufficient in the practical frequency range ($< 10^2\, rad.s^{-1}$). Moreover, with the sampling period $T = 5ms$, the higher frequencies cannot even be measured:

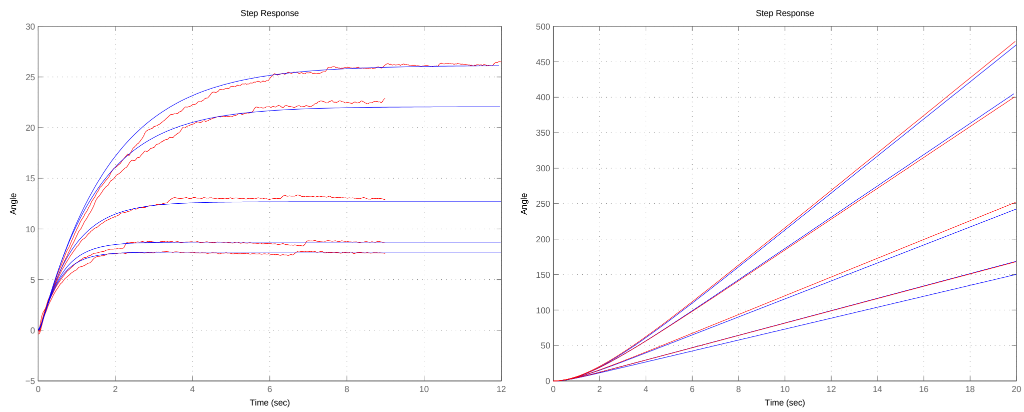

Figure 2: Step and impuls response model and real servomechanism

From the armature voltage to the angular velocity of the rotor shaft - assuming zero initial conditions, the first-order transfer function (from $u(t)$ to $\omega(t)$) for disk $\sharp 2$ and (rotational) friction indicator set to $5\, mm$:

```
numw = num,
denw = ldiv(den,s),     %left polynomial matrix division
numw =
       14.3170
denw =
       0.66 + s

Gw = rdf(numw,denw)
Gw =
     14     /      0.66 + s
```

The simplified mathematical model corresponds to the complete mathematical model and real identified model. That show the Fig. 3 on the next page.

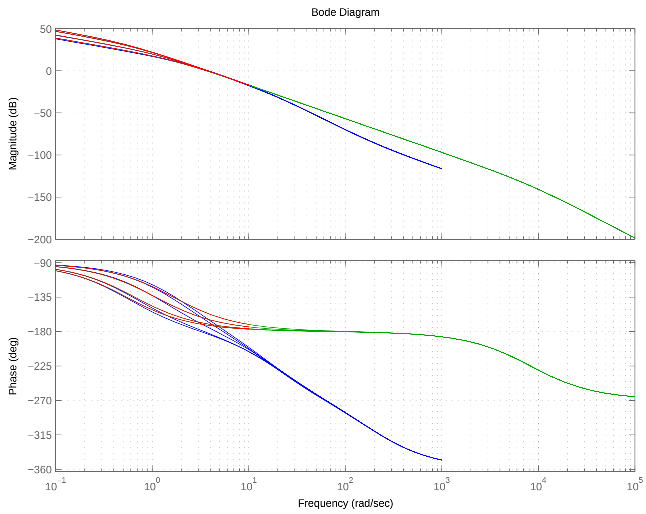

Figure 3: Comparison of the mathematical model and simplified model and real servomechanism

3 Robust control with Polynomial toolbox with LMI-based methods by Didier Henrion

LMI optimization software packages like SeDuMi [2] in combination with recent achievements in robust control theory [3] give a control engineer a new powerful tool. One useful instance of this is Matlab function *ptopdes.m* written by Didier Henrion. The function requires the Polynomial Toolbox [1]. In this section, we ilustrate the usefulness of this function for design of robust P and PI controller for (angular) position control.

A - Design of a proporcional controller The nominal proporcional controller is $C = Y/X = 1$.

```
x0 = 1; y0 = 1;
```

There are two vertices of the polytopic family of plants:

```
den_p = cell(2,1); num_p = cell(2,1);
```

the model with minimum value of rotational friction $B = B0$

```
num_p{1} = num; den_p{1} = den(1);
```

and the model with maximum value of the rotational friction $B = B0 + B15$

```
num_p{2} = num; den_p{2} = den(4);
```

The first task is to choose the so-called central polynomial. We stabilize the first vertex with a proportional controller and the closed-loop polynomial becomes

```
charpol = den_p{1}*x0+num_p{1}*y0
charpol =
     14 + 0.54s + s^2
```

Now, we call the function *ptopdes.m* with given parameters

```
[x,y] = ptopdes(den_p,num_p,charpol,[0 1;1 0],'P')
x =
     1
y =
     0.7343
```

Simulation results:

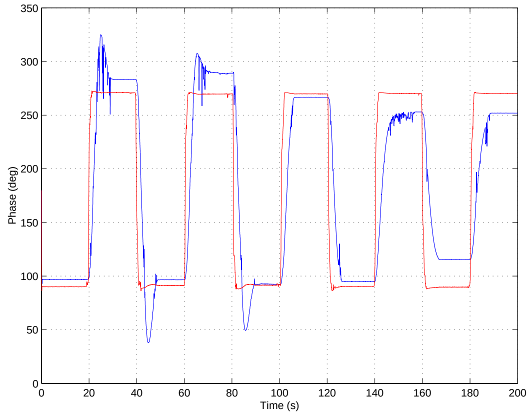

Figure 4: Control with P conroller

The friction B was changed every 40s from B0 to B0+B15 x0 = 1; y0 = 1;

B - Design of a PI controller The nominal PI controller C(s) = Y(s)/X(s) is given by

`x0 = s; y0 = 10*(2.5*s+1)*(0.85*s+1);`

Again, the uncertain family of plants is described as a polytope with two vertices. The central polynomial si found using the PI controller stabilizing the nominal plant:

```
charpol = den_p{1}*x0+num_p{1}*y0
charpol =
     1.4e+002 + 4.8e+002s + 3e+002s^2 + s^3
[x,y] = ptopdes(den_p,num_p,charpol,[0 1;1 0],'PI')
x =
     s
y =
     5.8e-008 + 5.7s
```

Simulation results:

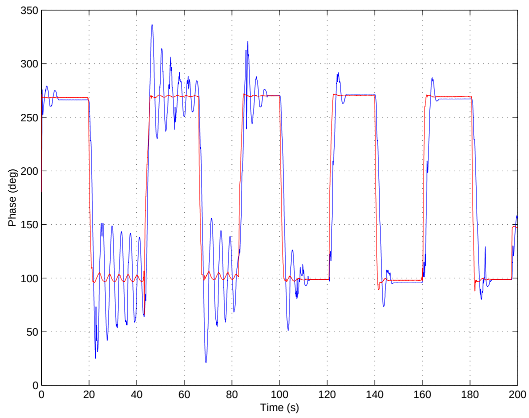

Figure 5: Control with PI conroller

The friction B was changed every 40s from B0 to B0+B15

C - Design of a PI controller with guaranteed stability margin Assessing the results of simulation with a robust PI controller, is might be reasonable to pose stricter requirements on the damping of the closed loop system. With *ptopdes.m* function it can be done via specification of some other stability domain. The simplest case is that of a shifted stability region D.
The shifted left half-plane guarantees some fixed stability margin, for example

$$\mathbf{D} = \{s \in \mathbf{C} : \mathrm{Re}\, s \ < -0.15\}$$

In the syntax of *ptopdes.m* function, this region can be fully described using the following matrix

```
S=[0.3 1; 1 0];
```

Now, call the function *ptopdes.m* with the given parameters and options:

```
[x,y] = ptopdes(den_p,num_p,(s+1)^3,[0.3 1;1 0],'PI')
x =
     s
y =
     0.011 + 0.081s
```

However, the designed controller performs poorly in simulations, because the control signal is small and the effect of the nonlinearity (dead zone) is not negligible.

4 Acknowledgements

This work was supported by the Ministry of Education of the Czech Republic under Project LN00B096.

5 Conclusion

We used the simplified mathematical model for all designed controllers. The simplified mathematical model corresponds with the complete mathematical model and real identified model. That is shown in Fig. 3. With the function *ptopdes.m* we can design P, PI and PID controllers. It is not (always) necessary to stabilize a vertex plant to obtain a central polynomial, very often the simple choice $(s+1)^k$ with k a suitable power (fixing the order of the controller) is a good candidate. It is described in [4] how to design other controllers with next functions that were created by Didier Henrion.

References

[1] PolyX, Ltd. *Polynomial Toolbox 3.0, (prerelease)*, <http://www.polyx.com>.

[2] Sturm, J., *SeDuMi 1.05*, <http://fewcal.kub.nl/sturm/software/sedumi.html>.

[3] Henrion, D., Arzelier, D., Peaucelle, D., *Positive Polynomial Matrices and Improved LMI Robustness Conditions. Proceedings of the IFAC World Congress*, Barcelona, Spain, July 2002.

[4] Henrion, D., Šebek, M., *New robust control functions for the Polynomial Toolbox 3.0*, May 17, 2002.