SWEDISH OPEN CHAMPIONSHIPS IN ROBOT CONTROL

Stig Moberg [1], Jonas Öhr [2]

[1] ABB Automation Technologies AB – Robotics - Motion Control Group
S-721 68 Västerås, Sweden
stig.moberg@se.abb.com

[2] ABB AB – Corporate Research – Mechatronics and Robot Automation Group
S-721 78 Västerås, Sweden
jonas.f.ohr@se.abb.com

Abstract: This paper describes a competition task, which was first presented at the Swedish control meeting (Reglermöte 2004), Chalmers University of Technology, Sweden. The competition concerns robot control where the process to be controlled is a nonlinear four-mass system. Only the so-called regulator problem will be treated and the task is, in brief, to design a discrete-time controller that optimizes performance for given robustness requirements. The deadline for handing in the contribution is 2004-11-30. Evaluation and results will be presented at a seminar held at ABB in Västerås 2005-01-24. All the participants are welcome to join this seminar. The winning solution is rewarded with 10000 SEK.

Keywords: Robust control, robot, competition, elasticity.

1. INTRODUCTION

It is a big challenge to control an industrial manipulator with high accuracy. A typical industrial manipulator has six links, which are controlled by electrical motors via gears. The manipulator can be described as a multivariable system where the six motor currents are the inputs and the six measurable motor angles are outputs. The goal is, however, to control the orientation and the position (six degrees of freedom) of the tool.

The dynamics of the system changes rapidly as the robot arms move fast within its working range and there are strong dynamic couplings between the arms. Moreover, the gears have nonlinearities in form of backlash, friction and nonlinear stiffness (elasticity). In addition to the elasticity of the gears, the arm structure also has significant elasticity. A typical industrial manipulator is shown in Figure 1.

The competition task described in this paper concerns only the so-called regulator problem and the controller should be designed such that the actual tool position is close to the desired reference, in the presence of motor torque disturbances and tool disturbances. Motor torque ripple and torque disturbances acting on the tool under material processing are examples of such disturbances.

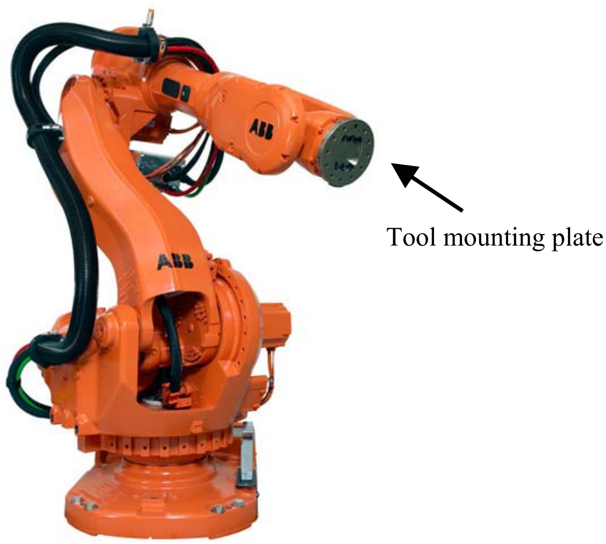

Fig. 1. *IRB6600 from ABB*

The problem has great practical significance since one direction of future manipulator development points toward increased elasticity. This will require the use of more advanced control methods than already used today.

2. THE SIMULATION MODEL

The simulation model used in this competition is a simplified, single variable model, which corresponds to one decoupled robot axis, which is linearized at a certain operating point. We will thereby ignore the strong couplings between the axes. Furthermore, the nonlinear friction present in the gear is also ignored.

The model to be used is a four-mass model having nonlinear elasticity in the gear and a time delay T_d in the measurement system of the motor position. The motor current- and torque control is assumed to be ideal so that the motor torque becomes the model input. The model is illustrated in Figure 2.

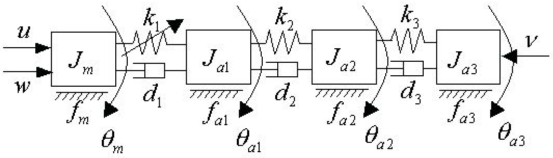

Fig. 2. *Nonlinear model of the robot arm.*

The damping and the nonlinear stiffness of the gear are represented by d_1 and k_1 respectively and the corresponding parameters for the arm are represented by d_2, k_2, d_3 and k_3. The nonlinear stiffness k_1 is illustrated in Figure 3. The moment of inertia of the arm is here split-up in the three components J_{a1}, J_{a2} and J_{a3}. The moment of inertia of the motor is J_m. The parameters f_m, f_{a1}, f_{a2} and f_{a3} represent viscous friction in the motor and in the arm structure respectively. θ_m is the motor (shaft) angle which is measured. θ_{a1}, θ_{a2} and θ_{a3} are arm angles of the three masses and together they define the position of the tool. The angles in this model are, however, expressed on the high-speed side of the gear so in order to get the real arm angles one must divide the model angles by the gear-ratio n (see the next section). The motor torque u, is the manipulated input of the system and w and v represent disturbances acting on the motor and the tool respectively.

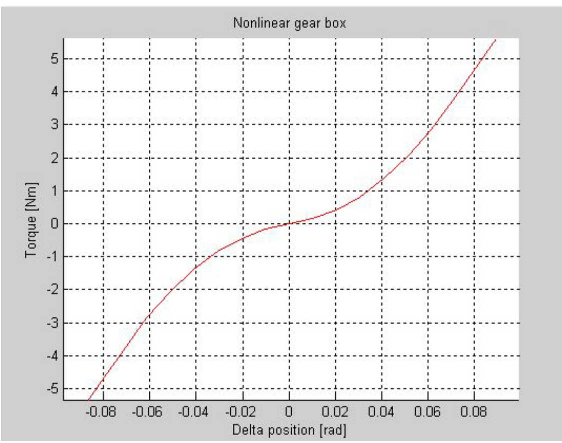

Fig. 3.*Nonlinear gear stiffness k_1.*

3. MATHEMATICAL DESCRIPTION OF THE LINEARIZED MODEL

The linearized model (k_1 = constant, $T_d = 0$) is given by the following equations:

$$J_m\ddot{\theta}_m + d_1(\dot{\theta}_m - \dot{\theta}_{a1}) + k_1(\theta_m - \theta_{a1}) + f_m\dot{\theta}_m = u + w$$
$$J_{a1}\ddot{\theta}_{a1} + d_1(\dot{\theta}_{a1} - \dot{\theta}_m) + k_1(\theta_{a1} - \theta_m) + d_2(\dot{\theta}_{a1} - \dot{\theta}_{a2}) + k_2(\theta_{a1} - \theta_{a2}) + f_{a1}\dot{\theta}_{a1} = 0$$
$$J_{a2}\ddot{\theta}_{a2} + d_2(\dot{\theta}_{a2} - \dot{\theta}_{a1}) + k_2(\theta_{a2} - \theta_{a1}) + d_3(\dot{\theta}_{a2} - \dot{\theta}_{a3}) + k_3(\theta_{a2} - \theta_{a3}) + f_{a2}\dot{\theta}_{a2} = 0$$
$$J_{a3}\ddot{\theta}_{a3} + d_3(\dot{\theta}_{a3} - \dot{\theta}_{a2}) + k_3(\theta_{a3} - \theta_{a2}) + f_{a3}\dot{\theta}_{a3} = v$$

The tool position z (which is the controlled variable) can for small variations around a given working point be calculated as

$$z = \frac{(l_1\theta_{a1} + l_2\theta_{a2} + l_3\theta_{a3})}{n},$$

where n is the gear-ratio and l_1, l_2, l_3 are distances between the (fictive) masses and the tool.

On state-space form the linearized system can now be described by

$$\dot{x}(t) = Ax(t) + Bu(t) + B_w(t)w(t) + B_v(t)v(t)$$
$$y(t) = Cx(t)$$
$$z(t) = Ex(t)$$

where y is the measured motor angle position and z the controlled variable. By selecting the states as

$$x = \begin{bmatrix}\theta_m & \theta_{a1} & \theta_{a2} & \theta_{a3} & \dot{\theta}_m & \dot{\theta}_{a1} & \dot{\theta}_{a2} & \dot{\theta}_{a3}\end{bmatrix}^T$$

we get

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -\frac{k_1}{J_m} & \frac{k_1}{J_m} & 0 & 0 & -\frac{d_1 + f_m}{J_m} & \frac{d_1}{J_m} & 0 & 0 \\ \frac{k_1}{J_{a1}} & -\frac{k_1 + k_2}{J_{a1}} & \frac{k_2}{J_{a1}} & 0 & \frac{d_1}{J_{a1}} & -\frac{d_1 + d_2 + f_{a1}}{J_{a1}} & \frac{d_2}{J_{a1}} & 0 \\ 0 & \frac{k_2}{J_{a2}} & -\frac{k_2 + k_3}{J_{a2}} & \frac{k_3}{J_{a2}} & 0 & \frac{d_2}{J_{a2}} & -\frac{d_2 + d_3 + f_{a2}}{J_{a2}} & \frac{d_2}{J_{a2}} \\ 0 & 0 & \frac{k_3}{J_{a3}} & -\frac{k_3}{J_{a3}} & 0 & 0 & \frac{d_3}{J_{a3}} & -\frac{d_3 + f_{a3}}{J_{a3}} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{J_m} \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad B_w = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{J_m} \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad B_v = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{J_{a3}} \end{bmatrix}$$

$$C = \begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}$$
$$E = \begin{bmatrix}0 & \frac{l_1}{n} & \frac{l_2}{n} & \frac{l_3}{n} & 0 & 0 & 0 & 0\end{bmatrix}$$

4. NOMINELL MODELL

The nominal model, which will be denote by M_{nom}, is defined by the following set of parameter values:

NOMINAL PARAMETERS

- J_m = **0.005** $kg \cdot m^2$
- J_{a1} = **0.002** $kg \cdot m^2$
- J_{a2} = **0.02** $kg \cdot m^2$
- J_{a3} = **0.02** $kg \cdot m^2$
- $k_{1,high}$ = **100** Nm/rad
- $k_{1,low}$ = **16.7** Nm/rad
- $k_{1,pos}$ = **0.064** rad
- k_2 = **110** Nm/rad
- k_3 = **80** Nm/rad
- d_1 = **0.08** $Nm \cdot s/rad$
- d_2 = **0.06** $Nm \cdot s/rad$
- d_3 = **0.08** $Nm \cdot s/rad$
- f_m = **0.006** $Nm \cdot s/rad$
- f_{a1} = **0.001** $Nm \cdot s/rad$
- f_{a2} = **0.001** $Nm \cdot s/rad$
- f_{a3} = **0.001** $Nm \cdot s/rad$
- n = **220**
- l_1 = **20** mm
- l_2 = **600** mm
- l_3 = **1530** mm
- T_d = **0.5E-3** s

The nonlinear gear stiffness illustrated in Figure 3 is approximated and described by five piece-vice linear segments in the attached simulation files. In the table above only the first segment, $k_{1,low}$, the last segment, $k_{1,high}$, and the position difference where the last segment begins, $k_{1,pos}$, are given.

5. PARAMETER VARIATIONS OCH MODEL SETS

Performance of the control systems will be evaluated for the nominal model M_{nom} and for two sets of models which will be denoted by M_1 and M_2, and which contain ten models, *m*, each. The set M_1 represents relatively small variations in the physical parameters and the set M_2 represents relatively large. For more details, see the file "RobotSystemSimulate.m".

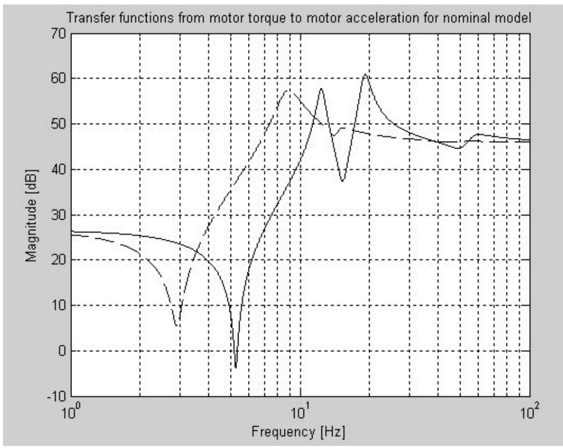

Fig. 4. *Amplitude spectra for (motor acceleration)/(motor torque) for* M_{nom}. *The solid line corresponds to the most stiff region of the gear (*$k_{1,high}$*) and the dashed line corresponds to the least stiff region (*$k_{1,low}$*).*

In Figure 4 the amplitude spectra for (motor acceleration)/(motor torque) $\left|\frac{\ddot{\theta}_m(\omega)}{U(\omega)}\right|$ for M_{nom} is shown. The solid line corresponds to the most stiff region of the gear ($k_{1,high}$) and the dashed line corresponds to the least stiff region ($k_{1,low}$). The Figures 5 and 6 show the amplitude spectra of the models $m \in \mathrm{M}_1$ and $m \in \mathrm{M}_2$ respectively, for the most stiff region of the gear ($k_{1,high}$). Note, however, that also these models are nonlinear in the simulation files.

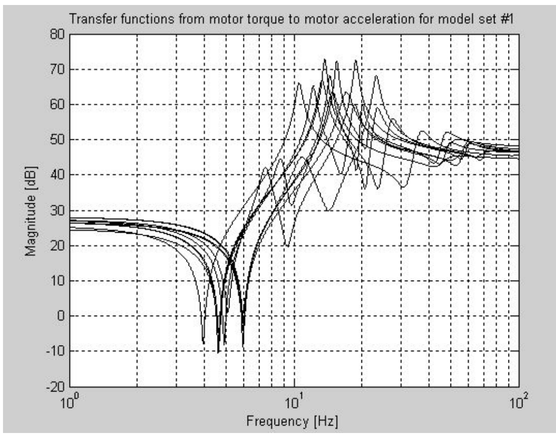

Fig. 5. *Amplitude spectra for(motor acceleration)/(motor torque)* $\forall m \in \mathrm{M}_1$.

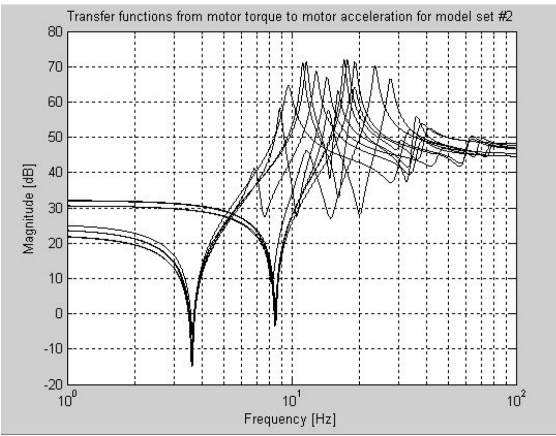

Fig. 6. *Amplitude spectra for (motor acceleration)/(motor torque)* $\forall m \in \mathrm{M}_2$.

6. THE TASK

Your task is to design one or two sampled controllers for the systems described above. One of the controllers must be capable of controlling all the models $m \in M_{nom} \bigcup M_1 \bigcup M_2$ whereas the other controller should be able to control M_{nom} alone. The controllers can be linear or nonlinear.

In order to investigate how well a controller can perform once we have access to really good models the competitors are encouraged to design two different controllers where one is optimized for the control of M_{nom} only. This controller will in the sequel be denoted by R_1 and the other by R_2. Note that R_1 and R_2 can be identical, can have the same structure and differ only by different tuning or can have completely different structure and tuning parameters.

The control systems will be exposed to a sequence of torque disturbances acting on the motor and on the tool according to Figure 7. The disturbance sequence consists of steps, pulses and sweeping sinusoids (chirp) that are specified in the attached Matlab™ files. Figure 8 shows the tool position and Figure 9 shows the controller signal (motor torque) when the disturbance sequence in Figure 7 acts on the nominal system. A simple PID-type controller is used here.

As a tool for controller design, a cost (or criterion) function will be used. The smaller the cost function the better the design. Figures 8 and 9 show all the performance measures that are weighted together into this cost function. Peak-to-peak error of the tool position (e_1 - e_8), settling times ($Ts_{1,2,3,4}$), maximum torque T_{MAX}, adjusted rms torque T_{RMS} and torque "noise" (peak-to-peak) T_{NOISE}.

Note that T_{NOISE}, which can be measurement noise and/or chattering caused by a discontinuous controller, is measured by the simulation routines when the system is at rest but that a good controller would keep the chattering/noise on a decent level also when it operates actively.

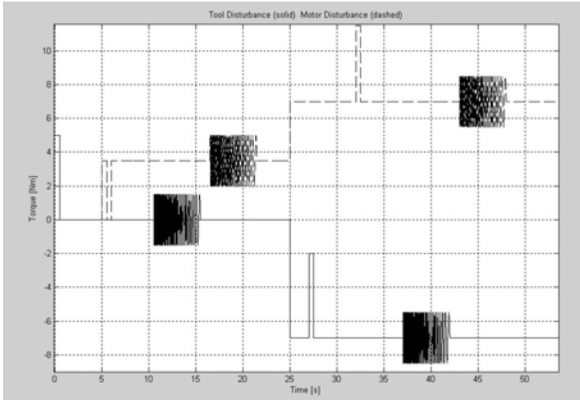

Fig. 7. *Torque disturbances on motor and tool*

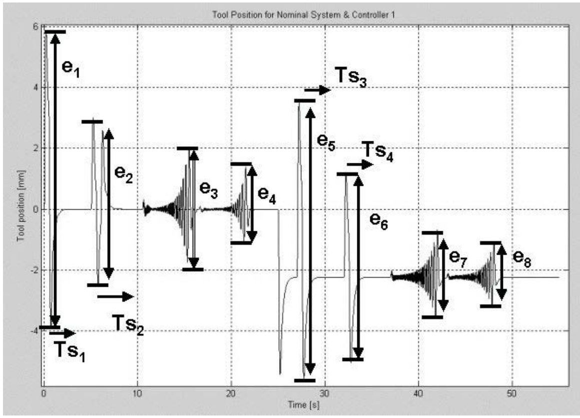

Fig. 8. *Tool position when using PID-control.*

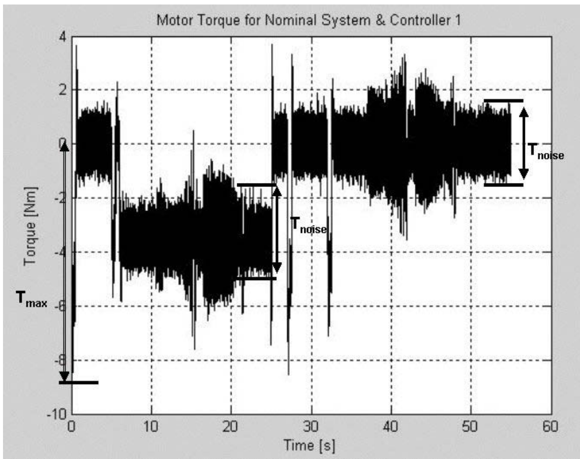

Fig. 9. *Motor torque when using PID-control.*

Magnified Figures 7, 8 and 9 can be found in Appendix 1.

For the nominal system the following cost function will be used:

COST FUNCTION FOR M_{nom} (R_1)

$$K_{nom} = \gamma \sum_{i=1}^{15} \alpha_i \cdot e_i$$

where e_i represents a generalized ”error” (position error, settling time or torque), α_i and γ are weights. It should be mentioned that K_{nom} is calculated only when R_1 is used for the control of M_{nom}.

For the sets M_1 och M_2 (using controller R_2) the maximum error from the simulations is used and the cost functions are given by the following expressions:

COST FUNCTION FOR M_1 (R_2)

$$K_1 = \gamma_1 \sum_{i=1}^{15} \alpha_i \cdot \max_{m \in M_1}(e_i)\,,$$

COST FUNCTION FOR M_2 (R_2)

$$K_2 = \gamma_2 \sum_{i=1}^{15} \alpha_i \cdot \max_{m \in M_2}(e_i)\,.$$

The total cost function K_{total} is given by

$$K_{total} = \beta_{nom} \cdot K_{nom} + \beta_1 \cdot K_1 + \beta_2 \cdot K_2\,.$$

We also put the following requirements on the systems:

REQUIREMENTS

- Settling times for M_{nom} (R_1) and M_1 (R_2): $Ts_{1,2,3,4}$ < 3 s, error band ± 0.1 mm
- Settling times for M_2 (R_2): $Ts_{1,2,3,4}$ < 4 s, error band ± 0.3 mm
- Torque noise T_{NOISE} < 5 Nm for all systems.
- Stability for all systems.
- Stability for M_{nom} when increasing the loop gain by a factor 2.5 for R_1 and R_2.
- Stability for M_{nom} when increasing the delay T_d from 0.5 ms to 2 ms for R_1 and R_2.

By ”stability” we also mean that limit cycles larger than 10 μm peak-to-peak are not allowed.

The control requirements we put on the systems M_{nom} and M_1 above can be motivated by e.g. differences between robot individuals, non modeled nonlinearities, model reduction, uncertainties due to varying applications and incorrect payload definition. The use of the model set M_2 is motivated by the fact that a real control system must be stable even for relatively large deviations between the model and the real manipulator dynamics. It might then seem strange to weight performance for the set M_2 into the total cost function but the reason is that we want to reward the robustness of the proposed controller.

7. THE CONTROL SYSTEM

The control system is shown in form of a Simulink[TM]-model in Figure 10. The following conditions rules:

- Sampling time 0.5 ms
- Time delay T_d 0.5 ms.
- Torque saturation limits ± 20 Nm (the saturation function in the controller block should not be removed!).
- Relatively strong measurement noise (only the motor shaft angle is measured).
- Only the blocks ”Controller1” and ”Controller2” and the files ”Controller_1.m” and ”Controller_2.m are allowed to be changed in the attached the simulation system. The system comes with a simple PID-controller used as an example.
- No continuous-time blocks can be added.
- No knowledge of deterministic nature about the noise and disturbances can be used in the controller (of course).
- The controller R_2 must have the same initial states and parameter values for all the simulations of $m \in M_{nom} \bigcup M_1 \bigcup M_2$.

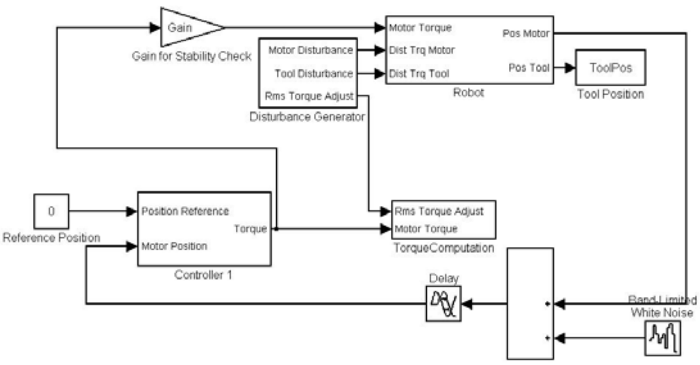

Fig. 10. *The control system.*

8. EVALUATION OF THE CONTRIBUTIONS

An expert panel from ABB will judge the competing contributions. The value of the cost function K_{total} and fulfillment of the requirements will have the strongest impact on the judgment but also other properties such as implementation- and tuning complexity will be taken into account.

Adaptive solutions are of course useful but one must consider the fact that the robot must have optimal performance even the first time it makes a movement after e.g. change of payload which could change the dynamics significantly. Furthermore a typical robot movement does not excite the dynamics as strongly as the disturbances used here do.

Adaptive solutions, learning control, auto-tuning etc. for the control of certain robot individual or a special application are powerful methods and they have their place in robot control. In this competition we are, however, more interested in robust control solutions.

9. TIME PLAN

The solution (theory and tuning) shall be described briefly in a document and the controllers should be able to run in Matlab™/Simulink™. Only the Simulink™-block ”Controller 1” / ”Controller 2” and the Matlab™-files ”controller_1.m” / ”controller_2.m” need to be handed in. Let us know what integration method you used in Simulink™!

The deadline is 2004-11-30!

If you plan to participate in the competition please send us a note so we can keep you updated with the latest versions of the material.

All questions, notes, corrections, you registration and finally the contributions can be sent to one of the authors of this paper.

In case the contribution is developed e.g. as a part of a master degree work, the final thesis do not have to be finished before the deadline.

Algorithms shall be open and visible (no ”black box” or dll-file will be accepted). No matter what tool is used for the design, Matlab™ / Simulink™ / Control Systems Toolbox™ shall be enough for running and testing the solution.

The competition is open for every one!

The winning contribution will be rewarded by 10000 SEK (Swedish Crowns).

Evaluation and results will be presented at a seminar held at ABB in Västerås 2005-01-24. All the participants are welcome to join this seminar.

10. ABOUT THE ATTACHED FILES

Matlab™ och Simulink™ files attached in Socrcsw.zip. A short description of a file ”xx.m” can be obtained by typing ”help xx” in the command window, see the example below.

```
>> help RobotSystemSimulate

  RobotSystemSimulate - Main File

  1. Simulates Nominal Model & Controller 1
     Compute Performance Criterion K_nom. Stability must be manually inspected.
  2. Check if Gain Margin and Delay Margin are fullfilled for Nominal Model and Controller 1.
     Result must be manually inspected
  3. Check if Gain Margin and Delay Margin are fullfilled for Nominal Model and Controller 2.
     Result must be manually inspected
  4. Simulates Model Set 1 & Controller 2
     Compute Performance Criterion K_1. Stability must be manually inspected.
  5. Simulates Model Set 2 & Controller 2
     Compute Performance Criterion K_2. Stability must be manually inspected.
  6. Outputs Total Performance Criterion if all requirements are fullfilled

  N.B.1. Stability has to be manually inspected
  N.B.2. The criterion computation is probably not fail-safe. Manual inspection is recommended.
  N.B.3. Tool Position, Torque and Amplitude Frequency Response of linearized robot system
         are plotted for each simulation.
  N.B.4. Simulation of model set 1 and 2 can be disabled.
  N.B.5. Of course it is allowed to have Controller 1 = Controller 2, i.e.
         Controller_1.m == Controller2.m & RobotSystem_1.mdl == RobotSystem_2.mdl
  N.B.6. Controller 1 & 2 may have the same structure but different tuning parameters
         (Controller_1.m != Controller2.m & RobotSystem_1.mdl == RobotSystem_2.mdl)
         or have completely different structures
         (Controller_1.m != Controller2.m & RobotSystem_1.mdl != RobotSystem_2.mdl)

>>
```

There is no guarantee that the attached file for automatic evaluation of performance always works so it is therefore wise to check the manually that the results are reasonable.

Stability must be investigated manually. The tool position may for instance seem stable while the system is internally unstable.

Does all this seem to be complicated ?

Unzip the file ”Socrcsw.zip”, start Matlab and type RobotSystemSimulate and you are in business!

You will soon se the following on the screen:

```
>> RobotSystemSimulate
PERFORMANCE NOMINAL MODEL:
[e1 e2 e3 e4 e5 e6 e7 e8] = [9.75 5.52 3.80 2.47 9.09 6.27 2.87 2.09]
[Ts1 Ts2 Ts3 Ts4] = [1.22 1.11 1.20 1.17] (Ts < 3.0 for tolerance 0.1 mm)
Torque noise = 3.06 (< 5.0), Max Torque = 8.80, RMS Torque = 1.19
Nominal Performance Criterion = 85.9
ROBUST PERFORMANCE FOR MODEL SET 1:
[e1 e2 e3 e4 e5 e6 e7 e8] = [12.79 6.48 4.64 2.71 11.92 7.57 3.71 2.39]
[Ts1 Ts2 Ts3 Ts4] = [1.38 1.35 1.23 1.20] (Ts < 3.0 for tolerance 0.1 mm)
Torque noise = 2.79 (< 5.0), Max Torque = 9.75, RMS Torque = 1.36
Robust Performance Criterion for Model Set 1 = 100.0
ROBUST PERFORMANCE FOR MODEL SET 2:
[e1 e2 e3 e4 e5 e6 e7 e8] = [14.64 6.91 4.82 2.51 14.34 8.33 4.06 2.42]
[Ts1 Ts2 Ts3 Ts4] = [1.07 1.07 0.97 0.96] (Ts < 4.0 for tolerance 0.3 mm)
Torque noise = 2.80 (< 5.0), Max Torque = 10.08, RMS Torque = 1.39
Robust Performance Criterion for Model Set 2 = 102.5
Weighted Robust Performance Criterion Nominal Model,
Model Set 1 and Model Set 2 = 182.3
Stability must be manually checked by inspection of the signals
```

The simulations may take some and a good idea might be to start with the nominal case.

11. FUTURE WORK

A complete robot controller must be multivariable and gain scheduled. A second step could therfore be to look at the multivariable case and/or using additional measured variables like e.g. arm position or tool position.

APPENDIX 1:

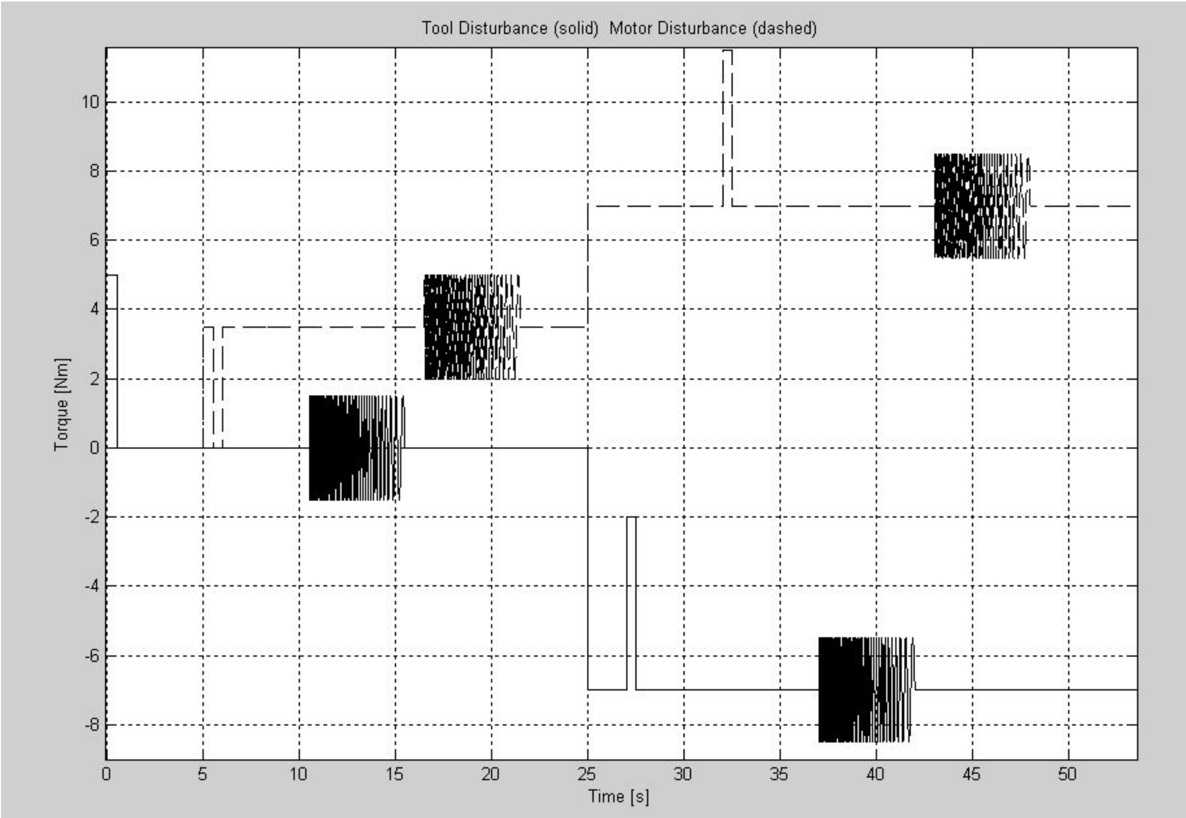

Fig. 11. *Torque disturbances on motor and tool*

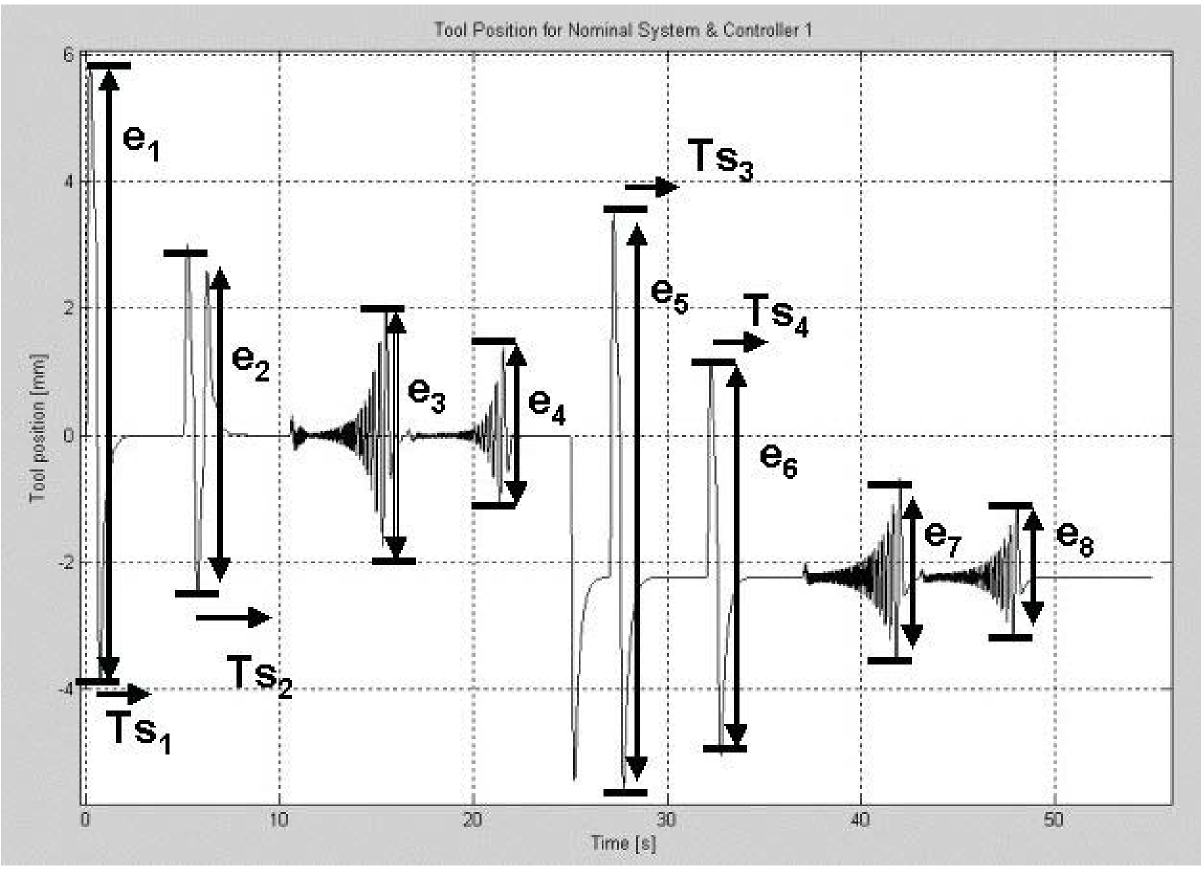

Fig. 12. *Tool position when using PID-control.*

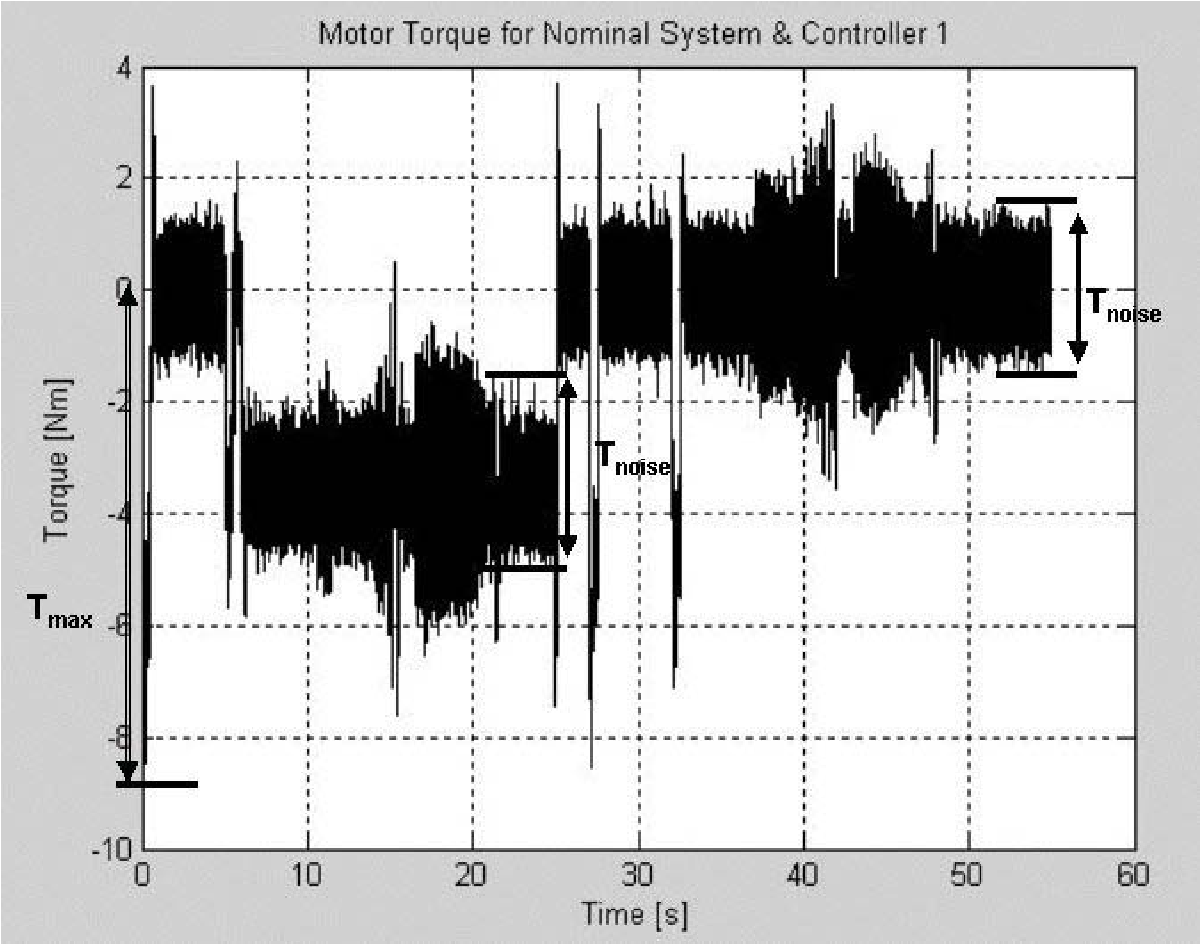

Fig. 13. *Motor torque when using PID-control.*