Carinthia University of Applied Sciences

Degree Program: Systems Design

EXTRA QUESTIONS

"Robot Control"

Student: ID:	Patrick Hofer 1410528006
Lecturer: Submitted: Submitted: Grade:	Dr. Andreja Rojko February 21, 2016

2. CT

It is stated 'A Computed torque (CT) controller is a nonlinear controller, which linearizes the error system and applies then a linear control law for the stabilization of the closed loop system' and little later 'This is the linearising control law, which nullifies the nonlinear parts of the robot dynamics and reduces the system to a linear double integrator system'. Can you explain how CT linearizes error system and which conditions need to be fulfilled that this actually happens. What if CT doesn't manage to completely linearise the system; does the control law then still work and what happens with the stability of the system?
CT controllers linearise the system by inverting the system dynamics. Therefore, the model of the system needs to be completely known. Otherwise, the system is not linearised and is still influenced by the nonlinearities. When this happens, the controller becomes a CT-like controller. Depending on the gains of the linear control law, this can still provide good results. The controlled system is still stable but asymptotic stability cannot be guaranteed anymore.

2.1 PD controller for outer loop

How does the choice of natural frequency influence the error? If the system dynamics is not completely linearised, can we still use the same procedure (choosing damping 1) to set the parameters of feedback controller?
When the system is completely linearised, the natural frequency influences the time it takes for the error to converge towards zero. This design procedure can still be used if the nonlinearities are not completely compensated (as it was done in all CT-like controllers, such as PD+gravity compensation), but then the choice of the natural frequency does not only influence the transient behaviour but also the steady state. The higher the natural frequency is chosen, the smaller is the steady state error of the closed loop system. This is caused by an increase of the controller gains $k_{p,i}$ and $k_{d,i}$. Therefore, a high natural frequency is preferable. On the other hand, increasing the natural frequency also results in a more aggressive controller, which means that the applied torque is increased. This limits the value of the natural frequency, as an actuator is not capable of applying infinite torque.

Normally we don't have a model of the friction available, so we don't use it in CT controller as well. What do you think what is efficiency of CT-PD when applied to the real system with friction? Note, that our model of the robot also doesn't includes the model of the friction.
When applying a CT PD controller to a real system with friction, the stability of the closed loop system would most certainly change from asymptotic stability to stability. This means, the error would not converge to zero anymore and a steady state error would remain. Therefore, the controller could not be used for exact positioning anymore. In this case, a PID outerloop controller would be a better choice.

3.1 PD controller with gravity compensation

In which situation (which work tasks respectively for which control requirements) it would be wise to use PD+gravity compensation?
A CT-like controller with PD and gravity compensation results in no error, if a fixed set point is used, which means zero velocity. Furthermore, if some disturbance occurs, again a steady state error happens. Therefore, the controller works best for positioning in a disturbance free environment. On the other hand, when following a trajectory, an bounded error will occur.

6. Sliding mode controller

How the amplitude of switching part should be chosen that the tracking error goes almost to zero?
The needed amplitude of the sliding part depends on the accuracy of the used model. The closer the parameters of the model are to the parameters of the real system, the less the amplitude needs to be. This can be seen in the time-derivative of the Lyapunov function, which is used in design process:

$$\dot{V}(\mathbf{s}) = \mathbf{s}^T((\mathbf{M} - \hat{\mathbf{M}})\ddot{\mathbf{q}}_s + (\mathbf{V}_m - \hat{\mathbf{V}}_m)\dot{\mathbf{q}}_s + (\mathbf{g} - \hat{\mathbf{g}})) - \sum_{i=1}^{n} k_i|s_i|$$

In order to be negative definite, the value of k_i needs to be larger, than the amplitude of the difference between the real parameters and the modelled parameters.

What it happens to stability/error convergence with the introduction of boundary layer?
When using a boundary layer, the controlled system is only stable. This means, that the error is bounded but it does not converge to zero. By increasing the amplitude of the boundary layer, the error can still be reduced to very small values.

7. Adaptive control

If we would like to introduce friction model in the controller and would like to have an adaptive viscous friction coefficient, what would be the procedure to derive suitable adaptation and control law (just general description, no derivation is necessary)?
When introducing a friction model, it needs to be added to the robot model first. Therefore, a new vector, denoted by $\mathbf{f}(\dot{\mathbf{q}})$, needs to be included. The coefficients of the viscous friction, which have to be adapted, are then gathered in the parameter vector $\hat{\varphi}$. As a next step, a Lyapunov function, containing the error of the parameters, has to be chosen. The time derivative of this Lyapunov function needs to be negative definite. This can be obtained by choosing the control law and an adaptation rule.