IFAC

Stable Controller Reconfiguration through Terminal Connections [*]

Klaus Trangbaek[*] Jakob Stoustrup[*] Jan Bendtsen[*]

[*] *Aalborg University, Denmark, e-mail: ktr@es.aau.dk*

Abstract: Often, when new sensor and/or actuator hardware becomes available for use in a control system, it is desirable to retain the existing controllers and apply the new control capabilities in a gradual, online fashion rather than decommissioning the entire existing system and replacing it with the new system. This paper presents a novel method of introducing new control components in a smooth manner, providing stability guarantees during the transition phase, and which retains the original control structure.

1. INTRODUCTION

All medium- to large-scale automation systems, such as power plants, refineries, factories, supermarkets or even large ships, invariably have control systems to handle the automated processes, such as production facilities, chemical batch processing, climate control or steam production. These control systems are often designed at the time of commissioning of the plant and tend to rely on PLCs or similar hardware to implement classically designed (and often conservatively tuned) control loops. However, as time goes by and new technology and knowledge becomes available, it may become desirable to introduce new sensor and/or actuator hardware.

There can be various reasons for this: wear and tear on the existing devices; new technology that can supplement with better or cheaper measurements or actuation becomes available; better knowledge about the process dynamics invites more precise control; etc.

On the other hand, there may be also be a strong argument for maintaining the existing control system, since it has a proven track record, and designing an entirely new control system from the bottom up is likely to be very costly both in terms of commissioning and operation stop. Furthermore, in addition to a linear control dynamic, the original controller may be part of a safety critical interlocking circuit as well.

The contribution of this paper is to provide a method of introducing new control components in a smooth manner, which provides stability guarantees during the transition phase, and which retains the original control structure intact.

In (Stoustrup et al. (1999)) a gain scheduling method was presented which solves this problem, but the implementation is of a rather high order, and the performance during transitions can be poor.

The method presented in this paper relies on the Youla-Kucera parametrization of all stabilizing controllers for a given plant. This methodology has the advantage of en-suring stability during the transition, and that the performance transfer function is affine in the design parameter, which means that the design problem has an open loop nature and that good performance can be expected during transition between controllers.

Section 2 provides the necessary background information on the Youla-Kucera parameterisation. Sections 3 and 4 present the novel method for modifying controller behaviour. In Section 5, a simulation example is presented, where the controller is modified after a new actuator is introduced.

2. CONTROLLER PARAMETERISATION

This section gives a short introduction to some basic concepts of coprime factorisation and the Youla-Kucera parameterisation of stabilising controllers. See (Youla et al. (1976); Kucera (1975); Anderson (1998); Niemann (2006)) for further details.
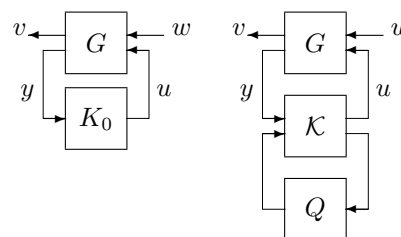

Fig. 1. *Left: The interconnection of the system G and the controller K_0. Right: Controller implemented as $K(Q) = \mathcal{K} \star Q$.*

Consider the control loop in the left part of Figure 1 and assume that the controller K_0 stabilises the system G. Factorise the lower right part of G as

$$G_{yu} = NM^{-1} = \tilde{M}^{-1}\tilde{N} \qquad (1)$$

with $N, M, \tilde{M}, \tilde{N} \in \mathcal{RH}_\infty$, and K_0 as

$$K_0 = UV^{-1} = \tilde{V}^{-1}\tilde{U} \qquad (2)$$

where $U, V, \tilde{U}, \tilde{V} \in \mathcal{RH}_\infty$, with the factors chosen to satisfy the double Bezout identity

$$\begin{bmatrix} \tilde{V} & -\tilde{U} \\ -\tilde{N} & \tilde{M} \end{bmatrix} \begin{bmatrix} M & U \\ N & V \end{bmatrix} = \begin{bmatrix} M & U \\ N & V \end{bmatrix} \begin{bmatrix} \tilde{V} & -\tilde{U} \\ -\tilde{N} & \tilde{M} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}.$$

10.3182/20080706-5-KR-1001.2375

All stabilising controllers for G can now be parameterised according to the Youla-Kucera parameterisation

$$K(Q) = \mathcal{K} \star Q = K_0 + \tilde{V}^{-1}Q(I + V^{-1}NQ)^{-1}V^{-1},$$

with $Q \in \mathcal{RH}_\infty$, i.e., $G \star K(Q)$ is stable for any stable Q and for any stabilising controller K_i, a stable Q exists so that $K(Q) = K_i$. This linear fractional transformation setup is depicted in the right part of Figure 1, and, due to the Bezout identity, can also be implemented as in Figure 2.
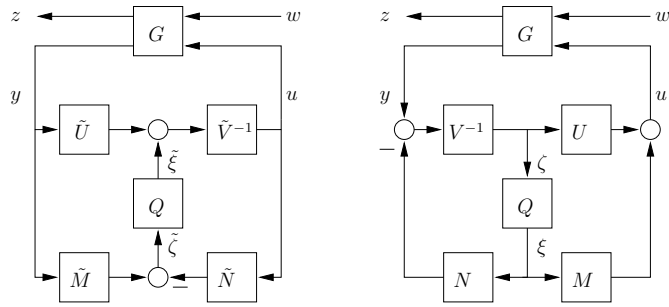

Fig. 2. *Left and right coprime factorisation-based Youla-Kucera parameterisation of all stabilising controllers.*

Thus, it is possible to implement a given controller as a function of a stable parameter system Q based on another stabilising controller, as depicted in the right part of Figure 1. As stated in (Niemann and Stoustrup (1999)) this implies that it is possible to change between two controllers online, say, from a nominal controller K_0 to another controller K_1, in a smooth fashion *without losing stability,* by scaling the Q parameter by a factor $\gamma \in [0; 1]$.
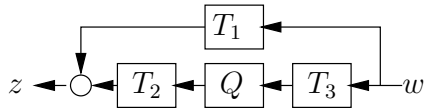

Fig. 3. *Classical model matching setup.*

One interesting feature of the parameterisation is that the performance transfer function from w to z is affine in Q, i.e.

$$T_{zw} = T_1 + T_2 Q T_3, \qquad (3)$$

also illustrated in Figure 3, where T_1, T_2, and T_3 are stable transfer functions. Thus, a control design can be carried out by finding a stable Q that minimises T_{zw} in some sense. This is known as a model matching problem (Francis (1987)).

Alternatively, if a desired transfer function for the a new stabilisng controller K_1 has been obtained, $K(Q) = K_1$ can be realised by factoring $K_1 = \tilde{V}_1^{-1}\tilde{U}_1$ with

$$\begin{bmatrix} \tilde{V}_1 & -\tilde{U}_1 \\ -\tilde{N} & \tilde{M} \end{bmatrix}\begin{bmatrix} M & U_1 \\ N & V_1 \end{bmatrix} = \begin{bmatrix} M & U_1 \\ N & V_1 \end{bmatrix}\begin{bmatrix} \tilde{V}_1 & -\tilde{U}_1 \\ -\tilde{N} & \tilde{M} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix},$$

and setting (Bendtsen et al. (2005))

$$Q = \tilde{U}_1 V - \tilde{V}_1 U = \tilde{V}_1(K_1 - K_0)V. \qquad (4)$$

Once a Q has been designed, the affine dependence also means that if Q is scaled by γ as mentioned above, then the performance will change in a predictable way for values of γ between 0 and 1. (In fact, stability will be preserved even for quickly time-varying γ (Hespanha and Morse (2002)), but that is not essential here.)

3. CONTROLLER MODIFICATION

We now turn our attention to a situation, where we wish to modify the controller behaviour but without removing the original controller. The reasons for the desired change can be numerous. The system may have changed due to equipment being added or replaced or simply due to wear and tear, or maybe a better understanding of the system has been obtained.

As mentioned in the introduction, the reasons for desiring to keep the original controller in the loop can also be numerous. It may for instance contain supervisory logic that we do not wish to replicate. Also, the operator will often be wary of removing a functioning controller with an entirely new replacement. Instead, adding a controller to the original one and slowly turning it on would be much more appealing.
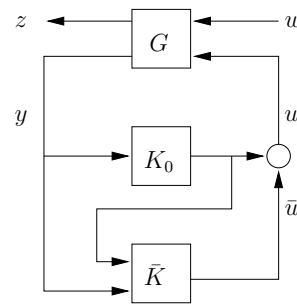

Fig. 4. *Modifying a controller through the control signal terminal.*

We assume that the original controller still stabilises the system, but we cannot access the inside of it, as Figure 2 would suggest is needed to use a Youla-Kucera parameterisation. Rather, the additional controller, $\bar{K}$, must be applied at the terminals of the existing controller as shown in Figure 4.

Thus, the task is to develop a method for designing an additional controller to be applied at the terminals of the original controller, which will improve the performance. It must be possible to perform the switch gradually while maintaining stability, so that the process can be monitored.
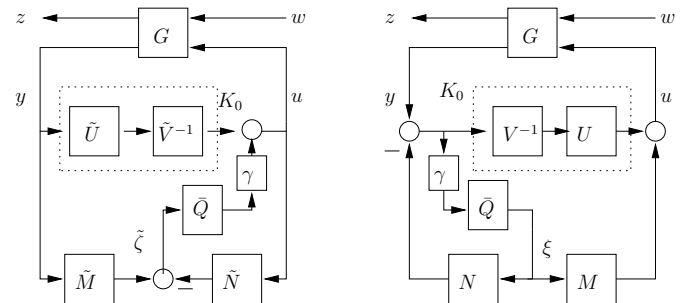

Fig. 5. *Controller parameterisation modified for connection to terminals of existing controller.*

By modifying the Youla-Kucera parameterisation in Figure 2, we arrive at the two possible setups in Figure 5. Here, the original controller, K_0 is kept in place and is only

accessed at the terminals. Stability of $\bar{Q}$ still implies stability of the closed loop, but not all stabilising controllers are parameterised by a stable $\bar{Q}$:

Theorem 1. Let $G_{yu} = \tilde{M}^{-1}\tilde{N}$ be a coprime factorisation of a system, and assume that $K_0 = \tilde{V}_0^{-1}\tilde{N}_0 = U_0 V_0^{-1}$, is a stabilizing controller, i.e. $G \star K_0 \in \mathcal{RH}_\infty$. Consider a second controller $K_1 = \tilde{V}_1^{-1}\tilde{N}_1 = U_1 V_1^{-1}$. Then

$$G \star K_1 \in \mathcal{RH}_\infty \wedge V_0^{-1}V_1 \in \mathcal{RH}_\infty \qquad (5)$$

$$\Updownarrow$$

$$\exists \bar{Q} \in \mathcal{RH}_\infty : K_1 = (I + \bar{Q}\tilde{N})^{-1}\begin{bmatrix} I & \bar{Q}\tilde{M} \end{bmatrix}\begin{bmatrix} K_0 \\ I \end{bmatrix} \qquad (6)$$

i.e., (6) is a parameterization of all stabilizing controllers that include the right half plane (RHP) pole structure of K_0.

Proof: First, assume that a controller K_1 satisfying (5) is given where, without loss of generality, we can assume that the parameterizations given satisfy the double Bezout identity. Define

$$\bar{Q} = U_1 - U_0 V_0^{-1} V_1 \in \mathcal{RH}_\infty$$

From (5) we infer $\bar{Q} \in \mathcal{RH}_\infty$. With this choice, we obtain:

$$(I + \bar{Q}\tilde{N})^{-1}\begin{bmatrix} I & \bar{Q}\tilde{M} \end{bmatrix}\begin{bmatrix} K_0 \\ I \end{bmatrix}$$

$$= (I + (U_1 - U_0 V_0^{-1}V_1)\tilde{N})^{-1}\begin{bmatrix} I & (U_1 - U_0 V_0^{-1}V_1)\tilde{M} \end{bmatrix}\begin{bmatrix} K_0 \\ I \end{bmatrix}$$

$$= (M\tilde{V}_1 - U_0 V_0^{-1}V_1\tilde{N})^{-1}\begin{bmatrix} U_0 V_0^{-1} - U_0 V_0^{-1}V_1\tilde{M} + U_1\tilde{M} \end{bmatrix}$$

$$= (M\tilde{V}_1 - U_0 V_0^{-1}N\tilde{V}_1)^{-1}\begin{bmatrix} -U_0 V_0^{-1}N\tilde{U}_1 + M\tilde{U}_1 \end{bmatrix}$$

$$= \tilde{V}_1^{-1}(M - U_0 V_0^{-1}N)^{-1}\begin{bmatrix} -U_0 V_0^{-1}N + M \end{bmatrix}\tilde{U}_1$$

$$= \tilde{V}_1^{-1}\tilde{U}_1 = K_1$$

Conversely, assume that K_1 is given by:

$$K_1 = (I + \bar{Q}\tilde{N})^{-1}\begin{bmatrix} I & \bar{Q}\tilde{M} \end{bmatrix}\begin{bmatrix} K_0 \\ I \end{bmatrix} \qquad (7)$$

We rewrite (7) as

$$K_1 = (I + \bar{Q}\tilde{N})^{-1}\begin{bmatrix} V_0^{-1}U_0 + \bar{Q}\tilde{M} \end{bmatrix}$$
$$= (V_0 + V_0\bar{Q}\tilde{N})^{-1}\begin{bmatrix} U_0 + V_0\bar{Q}\tilde{M} \end{bmatrix}$$
$$= (V_0 + Q\tilde{N})^{-1}\begin{bmatrix} U_0 + Q\tilde{M} \end{bmatrix}$$

with $Q = V_0\bar{Q} \in \mathcal{RH}_\infty$, and we see that K_1 is a stabilizing controller due to the Youla-Kucera theorem.

In order to prove that V_1 contains the RHP zero structure of V_0, we rearrange (7) into

$$(I + \bar{Q})U_1 V_1^{-1} = U_0 V_0^{-1} + \bar{Q}\tilde{M}$$

and further into

$$(I + \bar{Q})U_1 - \bar{Q}\tilde{M}V_1 = U_0 V_0^{-1}V_1 \qquad (8)$$

Since the left hand side of (8) is stable, so is the right hand side. Due to coprimeness of U_0 and V_0 there occur no RHP cancellations in forming the product $U_0 V_0^{-1}$, and since V_1 is stable, the product $V_0^{-1}V_1$ itself must be stable. $\triangleleft$

Thus, the setup in the left part of Figure 5 corresponding to (6) parametrises all stabilising controllers containing

the same unstable poles as K_0, i.e. we cannot move these unstable poles, but we can introduce new ones.

As with the Youla-Kucera parametrisation, the performance transfer function is affine in $\bar{Q}$, and the controller can still be designed by a model matching method, where the T_1, T_2, and T_3 transfer functions are the same as in Figure 3, but V or $\tilde{V}$ are introduced as shown in Figure 6.
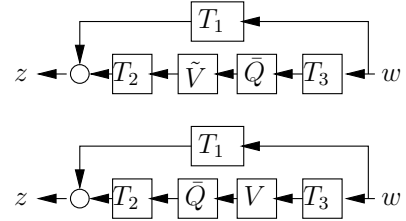

Fig. 6. *Modified model matching setup.*

In particular cases, T_2 and T_3 will be invertible and $\bar{Q}$ can be designed from

$$\bar{Q} \approx -\tilde{V}^{-1}T_2^{-1}T_1 T_3^{-1}. \qquad (9)$$

If exact equality could be achieved, this would imply $T_{zw} = 0$, but of course the inverses will usually have to be approximated to obtain a stable $\bar{Q}$.

As in (4, the design can also be done by designing a desired K_1 and finding Q solving

$$\bar{Q}V = \tilde{U}_1 V - \tilde{V}_1 U = \tilde{V}_1(K_1 - K_0)V, \qquad (10)$$

or

$$\tilde{V}\bar{Q} = \tilde{U}_1 V - \tilde{V}_1 U = \tilde{V}_1(K_1 - K_0)V, \qquad (11)$$

but since V and $\tilde{V}$ usually are not inversely stable, $\bar{Q}$, must be chosen as a stable approximation.

Note that the implementation in Figure 5 only requires the factorised plant model, although the model of the original controller is of course needed for the design of $\bar{Q}$.
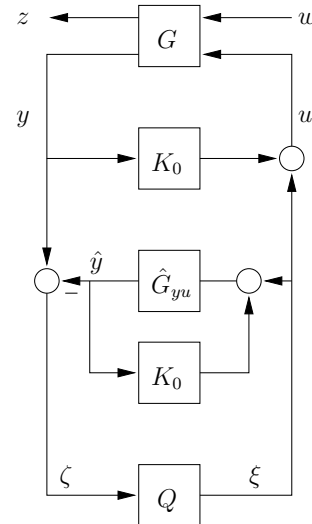

Fig. 7. *Pre-stabilised internal model control.*

As an aside, we note that it is possible to fully parameterise all stabilising controllers without doing any factorisation as shown in Figure 7 (Rotkowitz (2006)), still only accessing the terminals. Here, a stable Q implies a stable closed-loop, and vice versa. It does however require copying the

controller and plant models, and the resulting implementation could be of a very high order. On the other hand, this parameterisation makes it possible to deal with nonlinear plants and controllers, which will be the topic of further research.

4. ADDING SENSORS AND ACTUATORS

The main purpose of this work is to arrive at methods for automatic reconfiguration when new sensors and actuators are plugged in. The above method works for more general changes to the system, but in case of an additional sensor or actuator, we simply append the system model G with the new part, and add zero columns or rows to the model of the original controller before doing the factorisation.

Given a state space factorisation, (Niemann (2006)) provides extensions to the factors which preserve the original parts when adding sensors and actuators. However, for now we are not concerned with the particular structure of the factors.

5. SIMULATION EXAMPLE

In the buffer tank example shown in Figure 8, the fluid level M is controlled by a pump and a valve in series. The tank is disturbed by an unmeasured load flow $\dot{m}_L$. The only measurement is the fluid level.
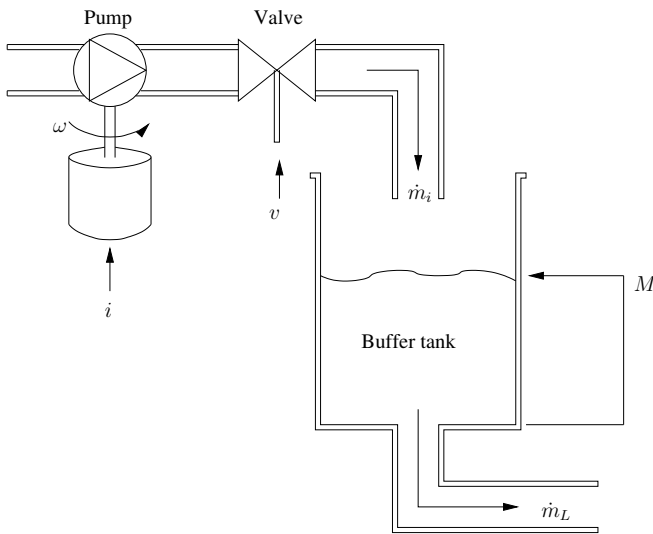

Fig. 8. *Example system.*

In (Trangbæk et al. (2006)) a first principles model was linearised to obtain the model shown in Figure 9.
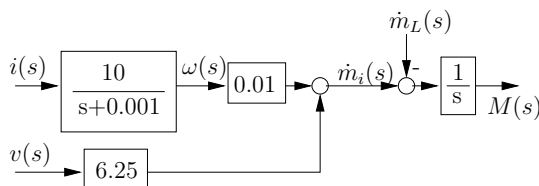

Fig. 9. *Linearised model.*

At the original design, the valve was manually operated, meaning that the control system could only use the pump

to control the fluid level. Since it was desired to suppress ramp disturbances in the load, a controller, K_0, with a double integrator was designed:

$$u(s) = \begin{bmatrix} i(s) \\ v(s) \end{bmatrix} = K_0(s)M(s) \tag{12}$$

with

$$K_0(s) = \begin{bmatrix} \dfrac{-0.1(s+0.01)(s+0.001)^2}{s^2(s+0.1)} \\ 0 \end{bmatrix}. \tag{13}$$

After some time, it is found that the performance is not satisfactory and that the strain on the pump is too high. Therefore the manual valve is replaced with an electronically controlled one. However, we still wish to keep the original K_0 in the loop for several reasons. First of all, the controller contains some safety critical logical circuitry in addition to the linear controller. Secondly, in periods with very small disturbances, we may want to be able to fully open the valve and only use the pump for control in order to save energy. Furthermore, the plant operator will be most happy, if the new controller can be tuned in slowly, so that the effects can be monitored.

Thus, the new controller should be implemented as in Figure 4, and it should be possible to scale the influence of it while preserving stability and a satisfactory performance. We therefore choose to implement the additional controller as in Figure 5. The disturbance w is the load flow, and the performance output is chosen as the fluid level and the pump current deviations, i.e. we want to maintain a stable level without using the pump a lot. With this choice, both T_2 and T_3 in Figure 6 are invertible, so $\bar{Q}$ can be designed using (9). $\tilde{V}^{-1}T_2^{-1}T_1T_3^{-1}$ has poles in $s = 0$, and is approximated by moving these slightly to the left in s-plane.

Figure 10 shows the effects of the additional controller. The top row shows the response with the original controller to a step in the load flow. The fluid level drops, resulting in an increased pump speed. Due to the slowness of the pump, it takes hundreds of seconds before the level is returned to normal. The bottom row shows the results of a similar load flow step but with the additional controller applied. Now the valve immediately reacts to a fluid level drop and almost completely removes the effects while maintaining the same pump speed. The middle rows show the results for different scalings of the additional controller. An important point is that a good performance is insured for these intermediate steps, making it possible to perform a gradual change from one controller to the other.

It is also worth noting that although step disturbances give no steady state error, the additional controller does not contain integrators in itself, but borrows these from the original controller, transferring the action from the pump to the valve.

In practice, it may be difficult to implement a $\bar{Q}$ designed from (9), since it tends have high order and gain. In this particular example it would probably give more moderate gains if the valve action was included in the performance output z. Then, T_2 would no longer be invertible, and a more traditional model matching method should be used.
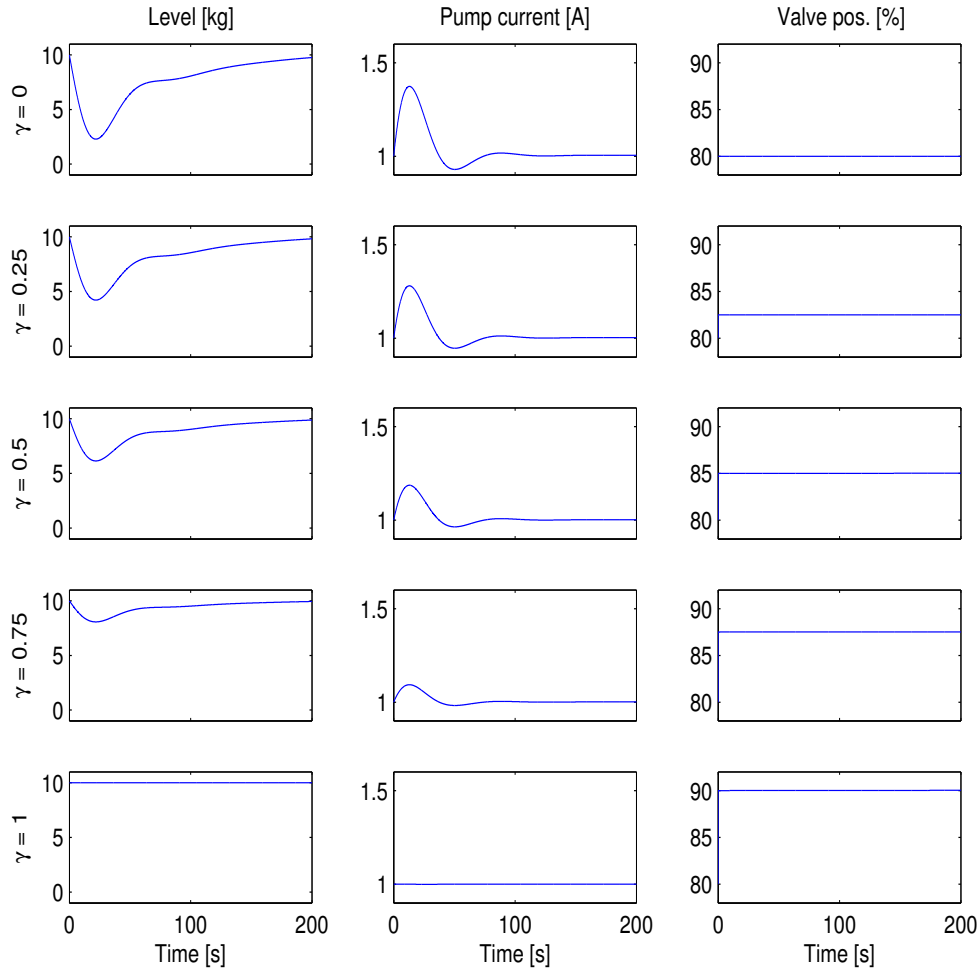
Fig. 10. *Simulation of a step in the load flow. Top row: Original controller. Bottom row: Modified controller.*

6. CONCLUSIONS

In this paper we have presented a novel method for modifying controller behaviour while keeping the original controller in place. By using a parametisation of stabilisng controllers, stability and performance are ensured even during transitions.

Future work may include methods that more explicitly address sensor and actuator addition and an extension to nonlinear systems.

REFERENCES

B. D. O. Anderson. From Youla-Kucera to identification, adaptive and nonlinear control. *Automatica*, 34:1485–1506, 1998.

Jan Dimon Bendtsen, Jakob Stoustrup, and Klaus Trangbæk. Bumpless transfer between observer-based gain scheduled controllers. *Int. J. of Control*, 78(7):491–504, 2005.

Bruce Francis. *A Course in $\mathcal{H}_\infty$ Control Theory.* Springer-Verlag, 1987. ISBN 3-540-17069-3.

J. P. Hespanha and S. A. Morse. Switching between stabilizing controllers. *Automatica*, 38(11):1905–1917, 2002.

V. Kucera. Stability of discrete linear feedback systems. *Proc. IFAC World Congress*, pages 44.1/1–6, 1975.

H. Niemann. Parameterisation of extended systems. *IEE Proceedings on Control Theory and Applications*, 153(2), 2006.

H. Niemann and J. Stoustrup. An architecture for implementation of multivariable controllers. In *Proc. American Control Conf.*, 1999.

Michael Rotkowitz. Information structures preserved under nonlinear time-varying feedback. *Proceedings of the American Control Conference*, pages 4207–4212, 2006.

J Stoustrup, P Andersen, T Pedersen, and M Hangstrup. Gain-scheduling procedures for unstable controllers: a behavioral approach. *American Control Conference, 1999*, pages 4004–4008, 1999.

Klaus Trangbæk, Jakob Stoustrup, and Jan Dimon Bendtsen. Gain scheduling of observer-based controllers with integral action. *Proc. IEEE Conf. Decision and Control*, 2006.

D. C. Youla, J. J. Bongiorno, and H. A. Jabr. Modern Wiener-Hopf design of optimal controllers. II. the multivariable case. *IEEE Transactions on Automatic Control*, 21(3):319–38, 1976.