CONTROL DESIGN FOR AN IRRIGATION CHANNEL FROM PHYSICAL DATA

Su Ki Ooi **E. Weyer**

CSSIP, Department of Electrical and Electronic Engineering
The University of Melbourne
Parkville VIC 3010 Australia
e-mail: {skoo, e.weyer}@ee.mu.oz.au

Keywords: PI controller, automatic controller tuning, physical modelling, system identification, irrigation channel.

Abstract

In this paper a tool for initial control designs for an irrigation channel is developed. The idea is that a physical model of the channel is obtained using the St. Venant equations, and a data set is generated by simulating this model. A first order nonlinear model is then estimated from the simulated data using system identification techniques, and a controller is designed based on the estimated model and the given design specifications. The controller is a PI controller augmented with a first order low pass filter in order not to amplify waves present in the channel. The developed routine for controller design is based on frequency response design, and configurations with and without feedforward from downstream gate are considered. The designed controllers have shown good performance, and they are able to track setpoint changes and the water levels recover from disturbances with small deviations from setpoints and without excessive oscillations.

1 Introduction

Water is becoming an increasingly scarce resource, and it is therefore important to manage the water resources well and minimize the losses. This applies particularly to networks of irrigation channels, where huge amounts of water are wasted due to poor management and control. These losses can be reduced by improving the control of the water levels in the channels.

Controllers that show good performance are obtained in [10]. Those controllers were tuned using frequency response techniques. Usually there are many gates, often more than 20 from the start till the end of a channel, and it can be a hard and time consuming task if control engineers have to tune each and every controller manually. In addition, the models used to design the controllers in [10] were obtained using system identification methods based on the operational data (see e.g. [9] and [7]). This requires measured data that are informative for identification purposes (see e.g. [4]). This kind of data is often not readily available. Even if there are measured data available, they are often irregularly or infrequently sampled, and hence not providing sufficient information about the relevant dynamics. Furthermore, in certain situations there is no operational data available, for example when implementing an automated control scheme in a channel for the first time.

Physical modelling only requires data like the length, height, cross section area, etc. In contrast to operational data, physical data are more widely available. Traditionally, the dynamics of an irrigation channel are modelled by the St. Venant equations, see e.g. [1]. Recent comparisons of the St. Venant equations against real data [6] have shown that the St. Venant equations are capable of capturing the relevant dynamics for control purposes of a real irrigation channel.

Our aim is to develop a tool to help with initial control design. Ideally this tool should be an automated routine for designing controllers for a channel based on physical data only, i.e. without any operational data. The tool developed is not a fully automatic one, but it will assist a control engineer in simplifying and speeding up the process of designing a large number of controllers. As this is an initial controller design, the main goal is to obtain controllers that stabilise the water levels without being overly sluggish. They can be fine tuned for better performance after operational closed loop data become available.

The idea is as follows. First, a data set is obtained by simulating the St. Venant equations based on physical data using the Preissmann scheme (see e.g. [1]). Then, a simplified model is estimated using system identification techniques from the simulated data. Given the obtained model and the controller design specifications: the phase margin, and the controller gain at the wave frequency, a controller is tuned by the developed routine.

This research is part of a collaborative research project between the Department of Electrical and Electronic Engineering and Rubicon Systems on modelling and control of irrigation channels.

In Section 2 a description of the irrigation channel is given. In the next section, the St. Venant equations are presented, followed by physical modelling and estimation of models using system identification techniques. Control design and results from simulation tests are presented in Section 5 and 6. Finally, conclusions are given in Section 7.

2 Channel Description

The channel considered is automated with overshot gates as shown in Figure 1. We refer to the stretch of the channel be-

tween two gates as a pool. We name the pool according to the number of the upstream gate, e.g. the pool in Figure 1 is pool i. y_i and y_j are the upstream water level of gate i and j respectively, and p_i and p_j are the position of gates. The amount of water above the gate is called the head over the gate, and denoted by h_i and h_j.

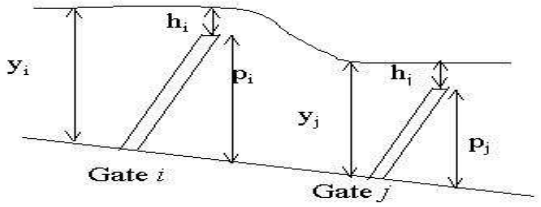

Figure 1: Schematic of channel with overshot gates

The water levels, in mAHD (meter Australia Height Datum), and the gate positions are the measured variables. The head over gate is computed from these variables. A fully shut gate has position of 0 meter and a positive value when the gate is open. The measured gate position is $\bar{p} = p_{max} - p$, where p_{max} is the position when the gate is fully shut. The head over the gate i and j is calculated as $h_i = y_i + \bar{p}_i - a_i$ and $h_j = y_j + \bar{p}_j - a_j$, where a_i and a_j are the gate adjustment constants necessary to convert from mAHD to meter.

3 St. Venant Equations

The St. Venant equations are derived from a mass and momentum balance, see e.g. [1] and given by

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0 \tag{1}$$
$$\frac{\partial Q}{\partial t} + \left(\frac{gA}{B} - \frac{Q^2}{A^2}\right)\frac{\partial A}{\partial x} + \frac{2Q}{A}\frac{\partial Q}{\partial x} + gA(S_f - \bar{S}) = 0$$

where A is the cross sectional area of the channel, B is the width of the water surface, $g = 9.81m/s^2$ is the gravity, $\bar{S}$ is the bottom slope, Q is the flow (discharge), and S_f is the friction slop. A commonly used relationship between the flow and the head over gate is $Q = ch^{3/2}$ (see e.g. [1]), where c is the gate constant. The gate constant of the upstream and downstream gate are labelled as c_{in} and c_{out}. From [2], for a sharp-edged rectangular channel, $c \approx 0.6\sqrt{g}b$ where b is the gate width.

According to the Manning equation, $S_f = \frac{n^2Q^2}{A^2R^{\frac{4}{3}}}$, where n is the Manning coefficient, which mainly depends on the surface roughness. Table values of n for different flow surfaces are available (see e.g. [8]). $R = \frac{A}{P}$ is the hydraulic radius, where the wetted perimeter, P, is defined as the length of line of intersection of the channel's wetted surface with a cross-sectional plane normal to the flow (see [1]). In this paper the Preissmann scheme with the weighting coefficient $\alpha = 0.6$ is used for simulation. See [6] for details.

The pools we study are pool 9 and 10 of the Haughton Main Channel (HMC). The physical data are given in [6]. Pool 9 is a short pool, 853 m long, and it has relatively fast dynamics. On the other hand, Pool 10 is 3129 m long and has slower dynamics than pool 9.

4 Modelling based on physical data

4.1 Input design and data simulation

In order to simulate the downstream water level using the St. Venant equations, input signals; which are head over upstream gate and downstream gate position are needed. In order to generate informative data, binary input signals are used and they are designed based on information obtained from a step test using the St. Venant equations. Step tests on pool 9 and 10 are performed by stepping the head over the upstream gate from 0.5 m to 0.7 m. The step response in pool 9 is plotted in Figure 2 (the result for pool 10 is not shown). From the

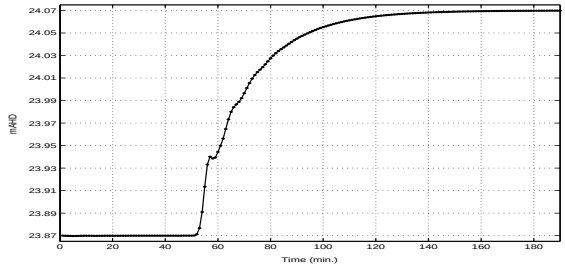

Figure 2: Pool 9 step response

step tests, we find that the time constant of pool 9 and 10 are around 19 min and 66 min. According to Section 13.3 in [4], the clock frequency of the input signal should be around 2.5 times the bandwidth of the system. Hence, the clock period of h_9 is set to 15 min., i.e. h_9 was constant for a multiple of 15 min, and the clock period of $\bar{p}_{10}$ is set to 60 min. For pool 10, h_{10} and $\bar{p}_{11}$ have clock periods of 60 min and 25 min respectively. The clock periods are chosen based on characteristics of the pool downstream from the gate the signal is associated with. This way we obtained signals similar to those we expect to encounter in practise. Our choices of clock periods are shorter than the commonly suggested values. However this is a reasonable choice taking into account that the time constant is obtained from linear consideration about a nonlinear system, and the time constant will in fact decrease with higher flows. Furthermore, the bandwidth of the closed loop system will be larger than the open loop bandwidth, and we also want to capture the wave, which has a higher frequency than the bandwidth.

Simulations are performed using the St. Venant equations with the designed input signals. The simulated data for pool 9 are shown in Figure 3. The data to the left of the vertical line are used for estimation purposes and those to the right for validation purposes. From the simulated water level, it is clear that there are waves present. The same wave effect is also found in the step test (Figure 2), and the wave periods are around 9 min

and 29 min for pool 9 and 10.

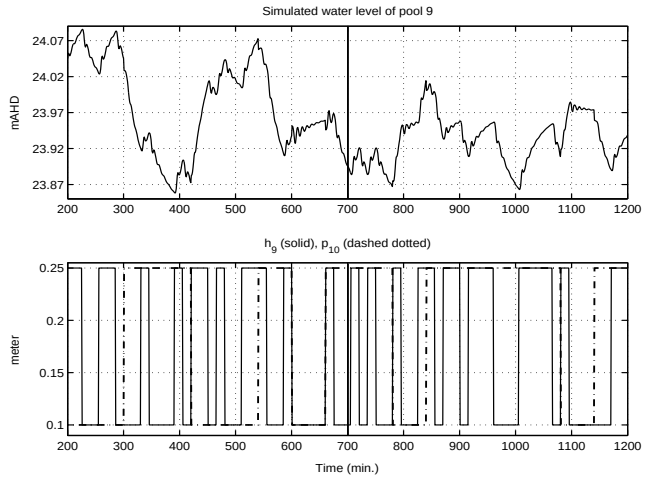

Figure 3: Pool 9 water level (top), and head over gate and gate position (bottom)

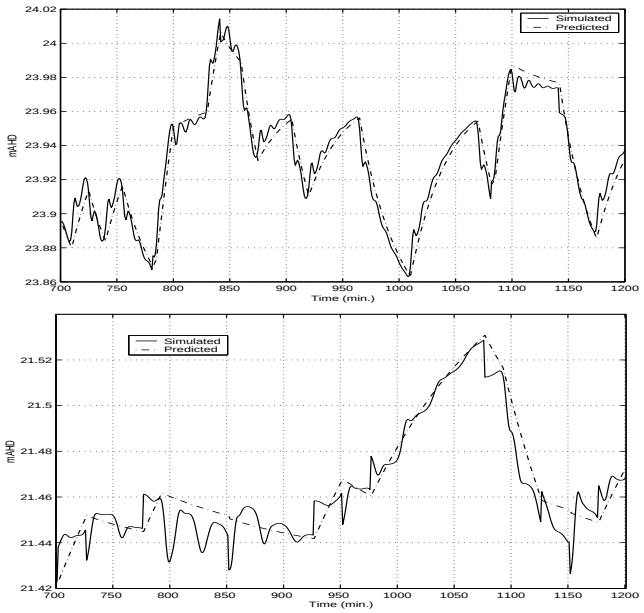

Figure 4: Pool 9 (top) and 10 (bottom) simulated and predicted water levels

4.2 Model structure and parameter estimation

The model structure considered is a discrete time first order nonlinear model, which is derived from a simple mass balance, see [9]. The models are identified using a prediction error method with quadratic criteria, see e.g. [9] or [7]. The predictor associated with the model for pool $i-1$ ($i = 10, 11$) is

$$\begin{aligned}\hat{y}_i(t+1,\theta) &= \hat{y}_i(t,\theta) + c_{i-1}h_{i-1}^{3/2}(t-\tau) - c_i(\hat{y}_i(t,\theta) \\ &\quad +\bar{p}_i(t) - a_i)^{3/2} \end{aligned} \tag{2}$$

where $\theta = [c_{i-1}, c_i]$, and y_i, h_{i-1} and $\bar{p}_i$ are as defined in Section 2, and $a_{10} = 23.97$ and $a_{11} = 21.43$. The time delay, τ is obtained from the step test. The parameter values together with the squared prediction errors on the validation set, which is computed as $V_{i-1}(\hat{\theta}) = \frac{1}{501}\sum_{t=700}^{1200}(y_i(t) - \hat{y}_i(t,\hat{\theta}))^2$ are shown in Table 1. The water level predicted by (2) on the validation set is shown together with the simulated water level using the St. Venant equations in Figure 4. The models (2) are able to track the main trends in the water levels very well but they cannot capture the waves.

Pool $(i-1)$	c_{i-1}	c_i	τ	$V_{i-1}(\hat{\theta})$
9	0.0447	0.0448	3min.	5.557×10^{-5}
10	0.0112	0.0111	11min.	7.127×10^{-5}

Table 1: Estimates, averaged squared prediction errors and time delays for pool 9 and 10

Remark. This part of the routine is not fully automated, as one needs to obtain the time constant, time delay and wave frequency from the step test and simulated data. However, it is sound practise always to look at the data before using them for estimation purposes. Furthermore, after a few investigations, one will have a rough idea of the relationship between the length of the pool and its time constant, time delay and wave period. In this case, the time constant and wave period increase with about 2 min and 1 min respectively for every 100 m in length, and the time delay increases with about 1 min for every 300 m in length. Moreover, it is not difficult to implement an automatic procedure for calculation of the time constant and time delay from the step test. Hence, for a channel with large number of gates, one will quickly be able to obtain rough estimates of time constants and wave frequencies.

5 Control design

In this section, an automatic routine for designing a continuous time controller based on the first order nonlinear model is developed. By automatic, we understand that given the model and some user specified criteria, like the phase margin, the routine generates controller parameters which satisfy the design specifications. Many automatic controller tuning methods have been proposed, e.g. the Ziegler-Nichols tuning rules, see e.g. [5], and the relay feedback method, see e.g. [3] and the references therein. However, these auto-tuning methods cannot be applied in this case, because the experiment needed produces undesired behavior in the channel, e.g. oscillations. Furthermore, most, if not all of the standard methods deal with the P, PI or PID controllers. Here we use a PI controller augmented with a low pass filter (see below).

The controller must be able to reject load disturbances. This is because the offtake of water from the channel is equivalent to a load disturbance. In addition, the controller must be able to track water level setpoint changes. However, from a practical point of view, the ability to reject load disturbances is much more important than tracking setpoint changes. This is

because offtake of water occurs much more frequently than setpoint changes. A distant downstream controller configuration is used, where the downstream water level is controlled by the upstream head over gate. The controller we consider is a PI controller augmented with a low pass filter and we refer to this combination as a robust PI controller. The integral action is needed in order to reject load disturbances and the low pass filter in order to suppress waves present in the channel. The transfer function of the robust PI controller is (for pool $i-1$)

$$C_{i-1}(s) = \frac{K_c(1+T_c s)}{T_c s} \cdot \frac{1}{(1+T_f s)} \tag{3}$$

We base the design on an integrator with delay model. The discrete time model (2) is derived from a continuous time model by an Euler approximation with sampling interval one minute (see [9]), hence when converting to s-domain $y_i(t+1) - y_i(t)$ is substituted by $sy_i(s)$, and c_{i-1} and c_i remain unchanged. We therefore obtain the following integrator with delay model.

$$y_i(s) = \frac{c_{i-1}e^{-\tau s}}{s} u_{i-1}(s) \tag{4}$$

where $u_{i-1}(t) = h_{i-1}^{3/2}(t) + \frac{c_i}{c_{i-1}} h_i^{3/2}(t+\tau)$ ($i = 10$ and $i = 11$ for pool 9 and 10). $u_{i-1}(t)$ depends on future signals, so in practise we will use $u_{i-1}(t) = h_{i-1}^{3/2}(t) + \frac{c_i}{c_{i-1}} h_i^{3/2}(t)$. Hence, the total controller with the feedforward is

$$\begin{aligned} u_{i-1}(s) &= C_{i-1}(s)\left(y_{i,setpoint}(s) - y_i(s)\right) \\ h_{i-1}^{3/2}(s) &= u_{i-1}(s) - K_{ff,i-1}F_{i-1}(s)\frac{c_i}{c_{i-1}} h_i^{3/2}(s) \end{aligned} \tag{5}$$

where $K_{ff,i-1}$ is the feedforward gain, $F_{i-1}(s)$ is a low pass filter, and $h_{i-1}^{3/2}(s)$ and $h_i^{3/2}(s)$ are the Laplace transform of $h_{i-1}^{3/2}(t)$ and $h_i^{3/2}(t)$. As in [10], our choice of $F_{i-1}(s)$ is a second order Butterworth filter with cut off frequency around half the wave frequency and $K_{ff,i-1} = 0.75$. Figure 5 shows the side view of the irrigation channel with the controllers.

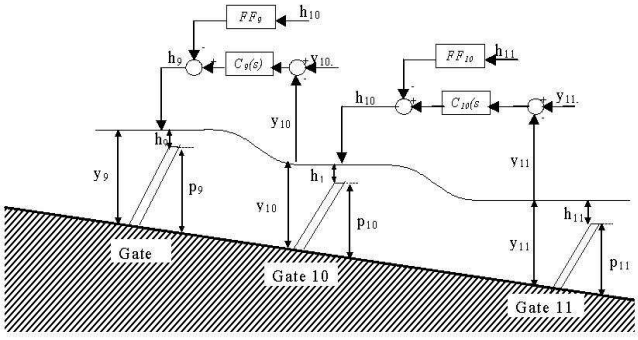

Figure 5: Side view of irrigation channel with distant downstream controllers with feedforward

When no feedforward is used, equation (5) becomes

$$h_{i-1}^{3/2}(s) = C_{i-1}(s)\left(y_{i,setpoint}(s) - y_i(s)\right) \tag{6}$$

5.1 Automatic controller tuning routine

The automated routine for controller tuning is based on the frequency response design for the lead compensator. $C_{i-1}(s)$ in (3) can be rearrange as $C_{i-1}(s) = \left(\frac{K_c}{T_c s}\right)\left(\frac{1+T_c s}{1+T_f s}\right)$, and the second term is a lead compensator. For the moment we treat $G_i(s) \cdot (\frac{K_c}{T_c s})$ as the uncompensated system, where $G_i(s)$ is the open loop transfer function (4). From equation (3) and (4), we observe that there are integrators both in the controller and the model, and hence the phase is -180° initially. Therefore, a phase lead is needed. The amount of phase lead is determined by the ratio between T_f and T_c, and given the required phase margin, ϕ_m, this ratio

$$\beta = \frac{T_f}{T_c} = \frac{1-\sin(\phi_m + \Delta\phi)}{1+\sin(\phi_m + \Delta\phi)} \tag{7}$$

can be computed. Note that in equation (7), on top of ϕ_m, an additional phase of $\Delta\phi = 10°$ is added in order to compensate for the phase drop that will occur due to the so-called gain amplification effect (see e.g. [5]) and also due to the time delay. The maximum phase lead is at the geometric mean frequency, $\omega_m = \frac{1}{T_c\sqrt{\beta}}$, and there is a gain amplification of $A = \frac{1}{\sqrt{\beta}}$ at this frequency. In order for the lead compensator to produce its maximum phase lead at the new gain crossover frequency, we set ω_m to be the frequency where the gain of the uncompensated system is $\frac{1}{A}$, i.e. $|G_i(j\omega).(\frac{K_c}{T_c j\omega})|_{\omega=\omega_m} = \frac{c_{i-1}K_c}{\omega_m^2 T_c} = 1/A$, and by substituting $\omega_m = \frac{1}{T_c\sqrt{\beta}}$, we have

$$K_c = \frac{1}{c_{i-1}T_c\sqrt{\beta}} \tag{8}$$

We also require the gain of the controller at the wave frequency ω_w to be a certain value, M_{wave}, i.e.

$$|C_{i-1}(j\omega)|_{\omega=\omega_w} = \frac{K_c\sqrt{1+T_c^2\omega_w^2}}{T_c\omega_w\sqrt{1+\beta^2 T_c^2\omega_w^2}} = M_{wave} \tag{9}$$

This specification is used instead of the standard gain margin because we do not want to amplify the wave, and whenever this specification is satisfied, a large gain margin is also guaranteed. Substituting K_c (equation (8)) into equation (9), we obtain a sixth order polynomial in T_c: $\alpha\beta^2\omega_w^2 T_c^6 + \alpha T_c^4 - \omega_w^2 T_c^2 - 1 = 0$, where $\alpha = (M_{wave}c_{i-1}\sqrt{\beta}\omega_w)^2$. Hence, given the system identification model, and the design specifications: the phase margin, and the controller gain at the wave frequency, β is computed (equation (7)) and the sixth order polynomial is then solved numerically for T_c. After that T_f is computed as βT_c, and finally K_c is computed using equation (8). The routine is programmed to check if the value of T_c is reasonable in the sense that T_c must be larger than $\frac{1}{c_i}$ in order to increase the bandwidth.

A disturbance rejection test is also performed before the controller is put into action to make sure that the overshoot is small and that the maximum deviation from setpoint is acceptable. The closed loop transfer function from the disturbance, d_i to y_i is $G_{d_i}(s) = \frac{y_i(s)}{d_i(s)} = \frac{G_i(s)}{1+C_{i-1}(s)G_i(s)}$. The test is carried out

by applying a step disturbance of size -0.1. Presently, these checks are done manually, but an automatic routine could be implemented to check the disturbance rejection criterion.

5.2 Pool 9

In this section a robust PI controller is designed for pool 9 using the routine developed. From Section 4.1, we have that the wave period is around 9 min. Based on operational experience, we specified ϕ_m to be $30°$, and $M_{wave} = -10$ dB. With the model in Table 1, the parameters $K_c = 1.493$, $T_c = 32.141$, and $T_f = 6.989$ were obtained. We can see that T_c is larger than $\frac{1}{c_{10}} \approx 22.32$. The Bode plots of the robust PI controller, model, and model with robust PI controller, and the disturbance rejection test result are shown in Figure 6. Note that the time delay is approximated using a first order Páde approximation. The

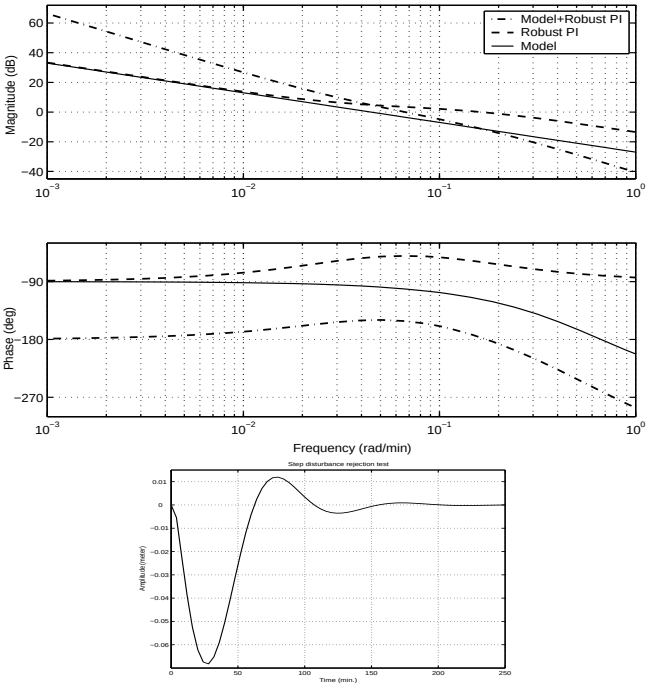

Figure 6: Pool 9: Bode plots (top) and step disturbance rejection test (bottom)

controller gain at the wave frequency (0.698 rad/min) is -10.46 dB, and the gain margin is 12.14 dB at 0.174 rad/min and the phase margin is $28.6°$ at 0.067 rad/min (see Figure 6). Hence, other than the slightly smaller phase margin, the controller satisfies our specifications. From the disturbance rejection test, there is an overshoot of less than 1.2 cm and the maximum deviation from the setpoint is about 6.8 cm, hence the designed controller provides acceptable disturbance rejection.

5.3 Pool 10

The same procedure was repeated for pool 10. We obtained $T_f = 25.110$, $K_c = 1.658$, and $T_c = 115.477$, and again $T_c > \frac{1}{c_{11}} \approx 90.09$. The overshoot in the load disturbance test

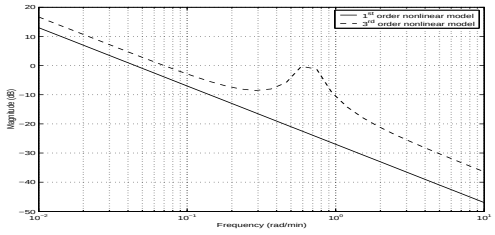

Figure 7: Pool 9 bode plots of the first and third order models

is around 1.1 cm and the maximum deviation from setpoint is about 6 cm but with a much slower response than in pool 9. The controller gain at the wave frequency (0.217 rad/min) is -10.46 dB, the gain margin is 11.97 dB at 0.048rad/min and the phase margin is $28.34°$ at 0.019 rad/min. Again, other than the slightly smaller phase margin, the designed controller satisfies the specifications and gives acceptable disturbance rejection.

6 Control performance

In this section, the controllers are put into action. A very accurate third order non-linear model is used to simulate the true system (see [9]). Obviously there is a mismatch between the simulation model and the model used for control design. This can be observed from the bode plots in Figure 7 of the first and third order nonlinear models, i.e. the transfer functions from $h_{i-1}^{3/2}$ to y_i. There is a 4 dB mismatch in the low frequency region for pool 9. The corresponding mismatch is 2.5 dB for pool 10.

6.1 Control of pool 9 and 10

In this section we investigate the performance of the controllers. We consider configurations with and without feedforward from the downstream heads (see equations (5) and (6)). During the test, gate 11 is kept at a given gate position, and gate 9 and 10 are controlled by the robust PI controllers.

The test is as follows. At time 0 minute both water levels are in steady state at the setpoints of 23.97 mAHD and 21.43 mAHD. The setpoint of pool 10 is kept constant throughout the whole test. At time 100 min. the position of gate 11 is changed from 0.22 m to 0.42 m (lowered). This can be viewed as an offtake in pool 10, i.e. a disturbance. Then, at time 600 the setpoint in pool 9 is increased from 23.97 to 24.00 mAHD, and at time 1100 the position of gate 11 is reduced back to 0.22 m. The water level responses in pool 9 and 10 are shown in Figure 8.

6.2 Discussion

From Figure 8, we can see the effect of the offtake in pool 10 propagating upstream to pool 9. For pool 9, we observe that the controller without feedforward is able to track the setpoint change and it recovers from the disturbances caused by the offtakes in pool 10 without excessive oscillations, but with

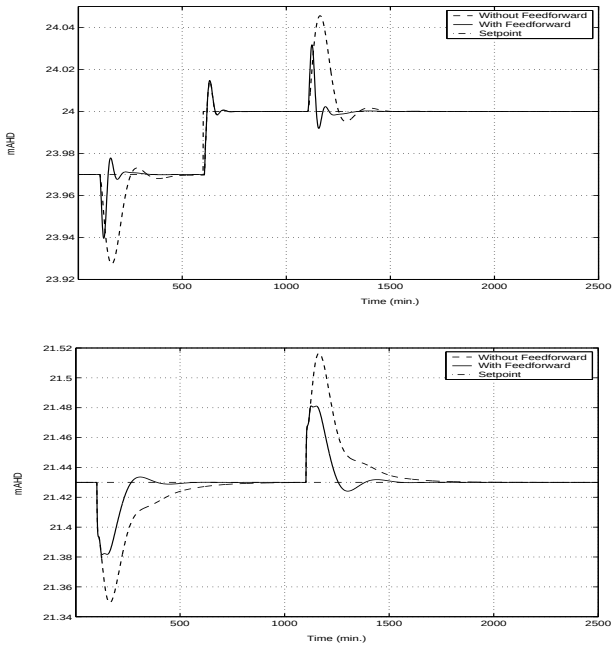

Figure 8: Control performance of Pool 9 (top) and 10 (bottom)

a slow response time. With feedforward, the response is much faster and the maximum deviation from the setpoint is only 3 cm, which is quite small. The response in pool 10 is similar but as expected the response is much slower, since pool 10 is more than three times longer than pool 9. These results show that the designed controllers give good performance and fulfill the required objectives, namely rejecting load disturbances and tracking setpoint changes. Feedforward from the downstream head improves the response significantly, i.e. faster response and smaller deviation from setpoint, hence controller configuration with feedforward is highly favorable. The designed controllers in this paper are comparable to those designed in [10]. Note that first order linear models were used in [10] for control design.

7 Conclusion

In this paper a tool is developed for speeding up the process of designing a large number of decentralised controllers when no operational data is available. The water levels of the channel are simulated using the St. Venant equations based on the physical data. Using the simulated data, a discrete time first order nonlinear model is obtained using system identification methods. An automated routine for designing a robust PI controller, which is a standard PI controller augmented with a low pass filter, based on the estimated model is developed. This routine is only used to provide the user with an initial control design, hence it is enough for the routine to provide controllers that stabilise the water levels without being overly sluggish. The designed controllers more than fulfill the required objectives and show good performance. They are able to track the step change in the setpoint quickly with small overshoot and reject load disturbances with small deviations from the setpoints and without excessive oscillations. The controller configuration with feedforward from head over downstream gate improves the closed loop response significantly with faster response and smaller deviation from setpoint. The results show that the tool developed is very useful, and it helps a control engineer in simplifying and speeding up the process of designing a large number of controllers given only physical data.

There are some room for improvement in the controller performance, and when data from the closed loop systems become available, the controllers can be fine tuned to give better performance. Hence, performance monitoring is a natural extension of this work, and it is a topic for future research.

Patent: A patent has been applied for to cover the developments that are described in this paper.

References

[1] M. Hanif Chaudhry. *Open-Channel Flow*. Prentice Hall, 1993.

[2] J. D. Fenton. *421-423: River Hydraulics (lecture note)*. The University of Melbourne, 2001.

[3] W.K. Ho, Y. Hong, A. Hansson, H. Hjalmarsson, and J.W. Deng. Relay auto-tuning of PID controllers using iterative feedback tuning. *Automatica*, 39:149–157, January 2003.

[4] L. Ljung. *System Identification: Theory For The User*. Prentice Hall, 2nd edition, 1999.

[5] K. Ogata. *Modern Control Engineering*. Prentice Hall, 3rd edition, 1997.

[6] Su Ki Ooi, M.P.M. Krutzen, and E. Weyer. On physcial and data driven modelling of irrigation channels. *To be presented at the 13th IFAC Symposium of System Identification*, 2003.

[7] Su Ki Ooi and E. Weyer. Closed loop identification of an irrigation channel. *Procedings of the 40th IEEE CDC, Orlando, USA*, pages 4338–4343, 2001.

[8] URL. *http://www.lmnoeng.com/manningn.htm*. LMNO Engineering, Research, and Software, Ltd., Athens, Ohio, USA, 2000.

[9] E. Weyer. System identification of an open water channel. *Control Engineering Practise*, Vol. 9:pp. 1289–1299, 2001.

[10] E. Weyer. Decentralised PI controller of an open water channel. *Procedings of the 15th IFAC World Congress, Barcelona, Spain*, 2002.