Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005



# Reconfiguration schemes to mitigate faults in automated irrigation channels

Sumith Choy and Erik Weyer
Department of Electrical and Electronic Engineering
University of Melbourne, Parkville, VIC-3010, Australia
{schoy,ewey}@ee.unimelb.edu.au,

*Abstract*— **In this paper we develop reconfiguration schemes to mitigate water level sensor and gate (actuator) failures in automated irrigation channels. Analytical redundancy inherent in the system is used for control and signal reconfiguration. Sensor faults are dealt with by estimating the correct value of the signal measured by the faulty sensor and using the estimated signal as input to the controller. A generic rule for the design of estimators is given which facilitates rapid design of a large number of estimators as required in a large scale irrigation network. An actuator fault necessitates relaxation of the control objective, and it requires both signal and controller reconfiguration. It is shown in simulations that the reconfiguration schemes work satisfactorily throughout the entire operating regime of the irrigation channel.**

***Keywords: Controller reconfiguration, Sensor and actuator faults, Irrigation channel***

## I. INTRODUCTION

Water is precious, and it is rapidly becoming a scarce resource. Hence it is important to manage the water resources well. This has motivated studies on control of irrigation channels since agriculture uses 70% of the world's fresh water and a large amount of this water is lost due to poor operation of the irrigation channels [1]. The goal is to reduce the losses by applying improved control systems. Different control schemes were devised in e.g. [2], [3], [4], [5], [6], [7], [8], [9], [10] and successful implementations are described in e.g. [2], [3] and [8]. Irrigation channels are open water channels with flows and water levels regulated by gates located along the channel. An automated channel employs controllers which utilize water level measurements and give out control signals to position the gates. This requires a large number of sensors and actuators throughout the channel. Faults are not uncommon, and they lead to a reduced level of service to the farmers and loss of water. However, these losses can be contained and water can be made available to farmers by suitably reconfiguring the signals and controllers in the event of a fault. Control reconfiguration based on stored control laws tailored to anticipated faults is developed in this paper. The anticipated faults are sensor faults and actuator faults which are the most common faults in an automated irrigation channel.

Previous works on control reconfiguration have been mainly associated with the development of fault tolerant control systems. Different approaches to controller and signal reconfiguration have been discussed in [11], [12] and [13]. Methods developed for controller reconfiguration include pseudo-inverse method [14], eigen structure assignment [15], LQC [16], learning method [17] etc.

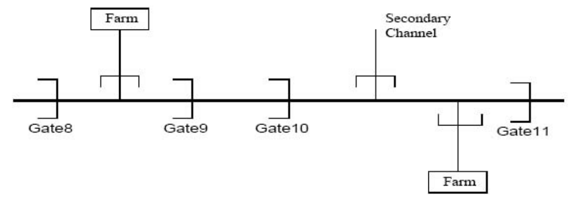


Fig. 1. Top view of an irrigation channel.

In this paper, we consider both controller and signal reconfiguration. The reconfiguration approaches utilize analytical redundancy inherent in the system to recover from faults. A sensor fault is taken care of by replacing the faulty signal from a sensor with an estimated value obtained from the other available measurements using an observer. An actuator fault necessitates a relaxation of the control objective and a reconfiguration of the control loop. The developed methods are illustrated in simulations.

The paper is organized as follows. Section II presents the models of the irrigation channel and the controllers used. Potential faults are described in section III, and in section IV we develop reconfiguration schemes to mitigate sensor faults. Reconfiguration schemes used to mitigate the actuator faults are considered in section V. Sections VI and VII give simulation results for sensor and actuator faults respectively. Conclusions are given in section IX.

## II. MODEL AND CONTROLLERS FOR IRRIGATION CHANNELS

Irrigation channels are open water channels with outflows to farms and secondary channels. A top view of such an irrigation channel is shown in Fig. 1. Mechanical gates are used to regulate the flow of water in the channel. Fig. 2 shows part of an irrigation channel with overshot gates. The stretch between two gates is called a pool and the pool between gate $i$ and gate $i+1$ is referred to as pool $i$. The measured variables at gate $i$ are the upstream water level $y_i$ and the gate position $p_i$. The height of water above the gate is called the head over the gate $h_i$ and it is given by $h_i = y_i - p_i$. In this paper, we focus our attention on the top three pools, pool $8, 9$ and 10, of the Haughton Main Channel (HMC) in Australia.

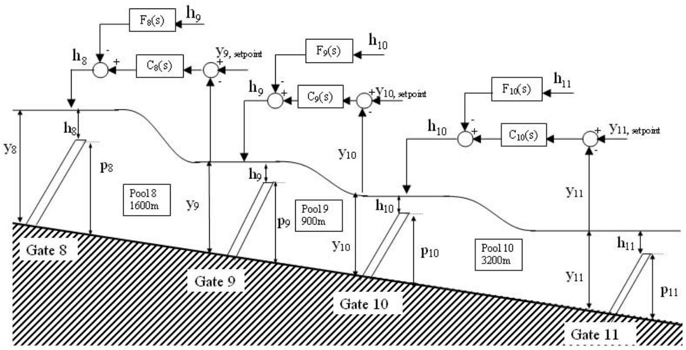


Fig. 2. Lateral view of irrigation channel with gates and decentralized distant downstream feedback controllers.

TABLE I

PARAMETERS FOR FIRST ORDER LINEAR MODEL, POOL LENGTH AND WAVE FREQUENCIES

| Pool | $c_{in}$ | $c_{out}$ | Length (m) | $\tau$ (min) | Wave freq |
|---|---|---|---|---|---|
| 8 | 0.014 | -0.017 | 1600 | 6 | 0.42rad/min |
| 9 | 0.046 | -0.042 | 900 | 3 | 0.72rad/min |
| 10 | 0.009 | -0.010 | 3200 | 16 | 0.20rad/min |

The water levels $y_i$ are measured in meters Australian Height Datum ($mAHD$), which is relative to sea level.

### A. System identification model

The models used here are developed based on a simple volume balance equation $\frac{dV(t)}{dt} = Q_{in}(t) - Q_{out}(t)$ where $V(t)$ is the volume of water in the pool and $Q_{in}(t)$ and $Q_{out}(t)$ are the flow over the upstream and downstream gates respectively. The flow over an overshot gate is approximated by $Q(t) = ch^{3/2}(t)$ [18] where $c$ is a proportionality constant and $h$ is the head over the gate. Assuming that the water level $y_{i+1}(t)$ is proportional to the volume of water in a pool, we get

$$\dot{y}_{i+1}(t) = \tilde{c}_{i,in} h_i^{3/2}(t-\tau) + \tilde{c}_{i,out} h_{i+1}^{3/2}(t) \quad (1)$$

where we have introduced a time delay $\tau$ accounting for the time it takes for the water flowing over gate $i$ to reach the point where $y_{i+1}$ is measured. For control purposes the model (1) was linearized to obtain

$$\dot{y}_{i+1}(t) = c_{i,in} h_i(t-\tau) + c_{i,out} h_{i+1}(t) \quad (2)$$

The parameters of this model have been found in [19] using system identification methods and the models have been successfully used for control design in [3] and [2].

The simulation models of the irrigation channel used in this paper are the third order nonlinear models from [19]. These models have been identified and validated through experiments in [19] and they capture the wave dynamics well. The parameters for the linear models, the pool lengths, time delays and dominant wave frequencies are given in Table I.

### B. Controller configuration

The offtakes to farms and secondary channels are gravity fed from the main channel, i.e. there is no pumping. Hence the main control objective is to maintain a given water level in a pool. This is essential in order to provide a certain standard of service to the farmers. A too low level will not enable us to deliver water and a too high level will cause flooding. The controller must also suppress the waves in the pool and reject the disturbances caused by offtakes of water.

Decentralized distant downstream controllers as shown in Fig. 2 are used at the HMC [2]. The controller takes the difference between the setpoint and $y_{i+1}$ ($e_{i+1} = y_{i+1,setpoint} - y_{i+1}$) as input and generates the head $h_i$ as its output (See Fig. 2). The gate position is then calculated from $p_i = y_i - h_i$. Taking the Laplace transform of (2) we get

$$y_{i+1}(s) = \frac{c_{i,in} e^{-\tau_i s}}{s} u_i(s) \quad (3)$$

where $u_i(s) = h_i(s) + \frac{c_{i,out}}{c_{i,in}} h_{i+1}(s) e^{\tau_i s}$. This relation depends on future signals, so in practise we use

$$u_i(s) = h_i(s) + \frac{c_{i,out}}{c_{i,in}} h_{i+1}(s) \quad (4)$$

Solving for $h_i$ we get

$$h_i(s) = u_i(s) - \frac{c_{i,out}}{c_{i,in}} h_{i+1}(s) \quad (5)$$

where $u_i(s) = C_i(s) e_{i+1}(s)$, where $C_i(s)$ is a PI controller augmented with a low pass filter to ensure a low gain at the dominant wave frequency in the pool, i.e.

$$C_i(s) = \frac{K_c(1+T_I s)}{T_I s} \frac{1}{(1+T_f s)} \quad (6)$$

where $K_c$ is the proportional gain, $T_I$ is the integral time constant and $T_f$ is the filter constant of the low pass filter.

Due to that we have ignored the time delay (5) is modified to

$$h_i(s) = u_i(s) - K_f B_i(s) \frac{c_{i,out}}{c_{i,in}} h_{i+1}(s) \quad (7)$$

where $B_i(s)$ is a second order lowpass Butterworth filter which ensures that the waves are filtered out and the gain is reduced by a factor $K_f = 0.75$ in order to avoid large overshoots due to the ignored time delay. The final controller structure is thus given by,

$$h_i(s) = C_i(s) e_{i+1}(s) - F_i(s) h_{i+1}(s) \quad (8)$$

where

$$F_i(s) = K_f B_i(s) \frac{c_{i,out}}{c_{i,in}}$$

The controller parameters are taken from [2] and is given in Table II.

TABLE II

CONTROLLER PARAMETERS

| Pool | $K_c$ | $T_I$ | $T_f$ |
|---|---|---|---|
| 8 | 2.5 | 142.86 | 16.67 |
| 9 | 2 | 50 | 10 |
| 10 | 1.6 | 142.86 | 20 |

## III. POSSIBLE FAULTS IN AUTOMATED IRRIGATION NETWORK

Faults can occur in the sensors measuring the water levels $y_i$ and the gate positions $p_i$. Moreover gate (actuator) failures can also occur. Sensor faults can vary from a drift in the sensor output to a complete failure. Incorrect information from a sensor will affect the controllers. Notice that a fault in $y_i$ or $p_i$ at a particular gate location $i$ will affect both the pool upstream and downstream of the location. This is because the controllers in the adjacent pools utilize the measurement for control purposes, see Fig. 2. The controller for the upstream pool uses the measurement $y_i$ directly for computing the setpoint error, while the controller for the downstream pool uses the measurement $y_i$ to calculate the position $p_i = y_i - h_i$ where $h_i$ is given by the controller for the downstream pool. In this paper we develop methods for mitigating the effect of faults in the sensor measuring $y_i$. Similar strategies can be used to mitigate faults in $p_i$.

A typical actuator fault is that a gate gets stuck in a fixed position. This happens when objects gets stuck in the gate or when the motor used to move the gate breaks down. The controller can then no longer achieve the control objective of maintaining the water level at setpoint in the downstream pool. In this case the control objective is changed and the controller is reconfigured to meet the new objective. Reconfiguration schemes to mitigate this type of faults are discussed in Section V.

## IV. RECONFIGURATION SCHEME TO MITIGATE SENSOR FAULTS

The effect of water level sensor faults are mitigated by using observers. They act as virtual sensors which supply an estimate of the signal measured by the faulty sensors. The nominal controllers are still applied except that the estimated signal is used as input signal. A block diagram of the irrigation channel with a reconfiguration scheme utilizing an observer is shown in Fig. 3. The observer makes use of the available measurements $h_i, p_{i+1}, h_{i+2}$ and $y_{i+2}$ to estimate the value of $y_{i+1}$ which is then used by the controllers.

### A. Linear observer based approach

Here we develop a Luenberger observer based on a combined state space model of the two consecutive pools affected by the fault. The model is constructed such that one of the states is the faulty signal. The inputs and outputs of the model are selected such that they are available for measurement.

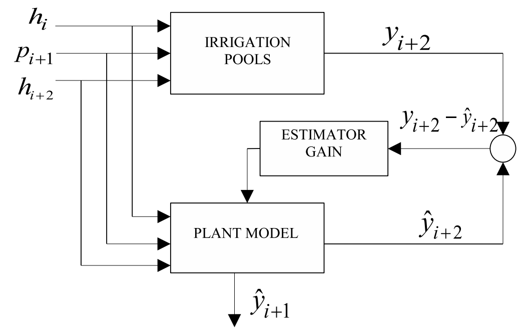


Fig. 3. Block diagram of the reconfiguration scheme to mitigate a sensor fault.

Using (2), the first order linear models for pool $i$ and pool $i+1$ are given by

$$\dot{y}_{i+1}(t) = c_{i,in}h_i(t-\tau_i) + c_{i,out}(y_{i+1}(t) - p_{i+1}(t)) \quad (9)$$

$$\dot{y}_{i+2}(t) = c_{i+1,in}(y_{i+1}(t-\tau_{i+1}) - p_{i+1}(t-\tau_{i+1})) + c_{i+1,out}h_{i+2}(t) \quad (10)$$

The following measured variables are taken as the input to the system; $u_1(t) = h_i(t-\tau_i)$, $u_2(t) = p_{i+1}(t)$ and $u_3(t) = h_{i+2}(t)$. The output is $y_{i+2}(t)$, which is also measured. The system is completely observable and controllable, and it is marginally stable since it contains an integrator. Notice that the time delay in pool $i$ is incorporated in the input $u_1$. A Pade approximation is used to approximate the time delay in pool $i+1$ in equation (10). The observer thus designed is as shown below.

$$\begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \\ \dot{\hat{x}}_3 \end{bmatrix} = \begin{bmatrix} \frac{-2}{\tau_{i+1}} & \frac{4}{\tau_{i+1}} & 0 \\ 0 & c_{i,out} & 0 \\ c_{i+1,in} & -c_{i+1,in} & 0 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \end{bmatrix} + \begin{bmatrix} 0 & -\frac{4}{\tau_{i+1}} & 0 \\ c_{i,in} & -c_{i,out} & 0 \\ 0 & c_{i+1,in} & c_{i+1,out} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} K_{e,1} \\ K_{e,2} \\ K_{e,3} \end{bmatrix} \epsilon_{i+2} \quad (11)$$

$$\hat{y}_{i+2} = \hat{x}_3 \quad (12)$$

where $\epsilon_{i+2} = y_{i+2} - \hat{x}_3$, $\hat{x}_1$ is a state variable used in the Pade approximation for the delay in pool $i+1$, $\hat{x}_2$ is an estimate of the water level $y_{i+1}$ in pool $i$ [which is the state of interest], $\hat{x}_3$ is the estimate of the water level $y_{i+2}$ in pool $i+1$. The values of the parameters in (11) are given in Table I. The real value of $y_{i+2}$ is available for measurement, and the error between the estimated $\hat{y}_{i+2}$ and the real $y_{i+2}$ is fed back to the observer through a gain vector $K_e$. The state $\hat{x}_2$ gives the estimate of the faulty signal.

### B. Generic rule for observer pole placement

A rule of thumb for observer pole placement is given here. The poles of the state space model of the two consecutive pools $i$ and $i+1$ are located at

$$\begin{bmatrix} 0 & -\frac{2}{\tau_{i+1}} & c_{i,out} \end{bmatrix}^T \quad (13)$$

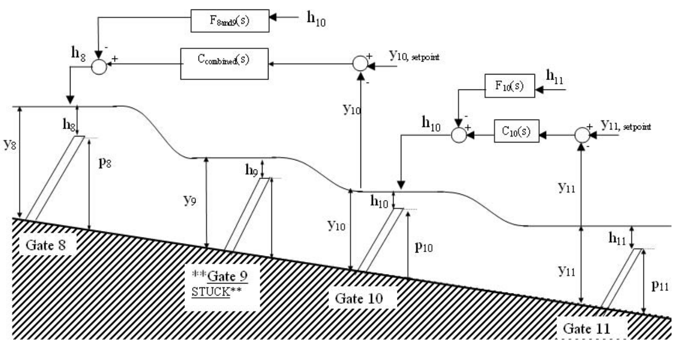


Fig. 4. Control scheme with gate $i+1$ (gate 9) completely stuck.

The observer poles should be placed to the left of the system poles, i.e. the observer dynamics should be faster than the system dynamics. Experience has shown that the following pole locations

$$\begin{bmatrix} -c_{i,in} \times 10 & -\frac{4}{\tau_{i+1}} & c_{i,out} \times 10 \end{bmatrix}^T \tag{14}$$

normally yield a well conditioned observer. This can be used as a guideline to place the poles initially. The design can be further fine tuned later on if required. Note that the pole locations in (14) are in direct relationship with the dynamics of the combined pools.

## V. Reconfiguration scheme to mitigate an actuator fault

Here we consider the situation when gate $i+1$ gets completely stuck. Once gate $i+1$ gets stuck, we change the control strategy, and we give up on controlling the water level $y_{i+1}$ just upstream of the gate. Instead we consider the two pools on either side of the fault location to be a single pool, and we control the water level $y_{i+2}$ as shown in Fig. 4. The control loop needs to be reconfigured, and the new controller takes the difference between $y_{i+2}$ and $y_{i+2,setpoint}$ as input and gives $h_i$ as output, which is then used to position gate $i$. The model used to design the controller is derived from (9) and (10). Taking the Laplace transform of (9) we get

$$y_{i+1}(s) = \frac{e^{-\tau_i s} c_{i,in}}{s - c_{i,out}} h_i(s) \tag{15}$$

since $p_{i+1}$ is fixed (gate $i+1$ is stuck). Taking the Laplace transform of (10), we get

$$y_{i+2}(s) = \frac{e^{-\tau_{i+1} s} c_{i+1,in}}{s} y_{i+1}(s) + \frac{c_{i+1,out}}{s} h_{i+2}(s) \tag{16}$$

Substituting (15) into (16) we get

$$y_{i+2}(s) = \frac{e^{-\tau_{i+1} s} e^{-\tau_i s} c_{i,in} c_{i+1,in}}{s(s - c_{i,out})} h_i(s) + \frac{c_{i+1,out}}{s} h_{i+2}(s) \tag{17}$$

The model derived here is of a higher order than the integrator model (3) used previously. An additional lead element is now required in the controller to get sufficient phase lead. The new controller is given by

$$h_i(s) = \bar{C}_i(s) L_i(s) (y_{i+2,setpoint} - y_{i+2}) - \bar{F}_i(s) h_{i+2}(s) \tag{18}$$

TABLE III

Controller parameters and lead compensator used in the event of an actuator fault

| Model | $K_c$ | $T_i$ | $T_f$ | Lead compensator $L_i$ |
|---|---|---|---|---|
| Second order (17) | 1.5577 | 105 | 4 | $\alpha = 10$, $T_l = 4$ |

where $L_i(s) = \frac{1+\alpha T_l s}{1+T_l s}$ and $\bar{C}_i(s)$ has the same structure as in (6) and $\bar{F}_i(s) = K_f B_i(s) \frac{c_{i+1,out}}{c_{i,in} c_{i+1,in}}$. The parameters of the new controller are given in Table III.

## VI. Simulation-Sensor faults

The simulation models used are the third order nonlinear models from [19] discussed in Section II-A. In the beginning the water levels in pool $8, 9$ and $10$ were at the set points $26.55mAHD$, $23.85mAHD$ and $21.15mAHD$ respectively. The head over the last gate $h_{11}$ in the channel was kept constant at $0.5m$. All the pools were assumed to be running normally with their respective decentralized controllers. A setpoint change of $-0.05m$ from $26.55mAHD$ to $26.50mAHD$ for $y_9$ took place at $t = 860min$. The setpoint was brought back to $26.55mAHD$ at $t = 1220min$. An offtake of water was simulated by adding a constant outflow to pool 9. The offtake began at $t = 2500min$ and lasted till $t = 2700min$.

### A. No fault scenario

The linear observer was run in parallel to the simulation model to estimate the water level $y_9$ in pool 8, see Fig. 3. Fig. 5 shows the estimated water level and the real water level in pool 8 under a no fault scenario. It can be observed that the linear observer tracks the real water level satisfactorily. The observer poles were initially placed using the rule developed in section IV-B at $P = [-0.14 - 1.333 - 0.17]$. The corresponding observer gain was $K_e = [-4.2185, 0.7878, 0.9596]$. There was a small offset of $-0.0047m$ from the real steady state level using this gain. These pole locations were then fine tuned to $P_{ft} = [-0.07 - 1.6 - 0.27]$ to get the gain $K_e = [-6.0069, 0.6750, 1.2563]$. This gain yields a near zero offset (around $10^{-4}m$).

### B. Fault scenario

A fault (drift) was simulated in the sensor measuring $y_9$ at time $t = 100min$. The fault went undetected till $t = 200min$. It can be observed from Fig. 6 that the system deviated from its normal behavior in this period. Once the fault was detected, the faulty sensor was removed from the loop and replaced by the estimated signal. It can be observed that the system recovered smoothly and tracked the setpoint changes and recovered from offtakes. The shift in the water level using the observer designed using the generic rule is due to the offset discussed in section VI-A. When the signal from the observer is fed to the controller it looks like the water level is higher than the set point and the controller brings the water level down.

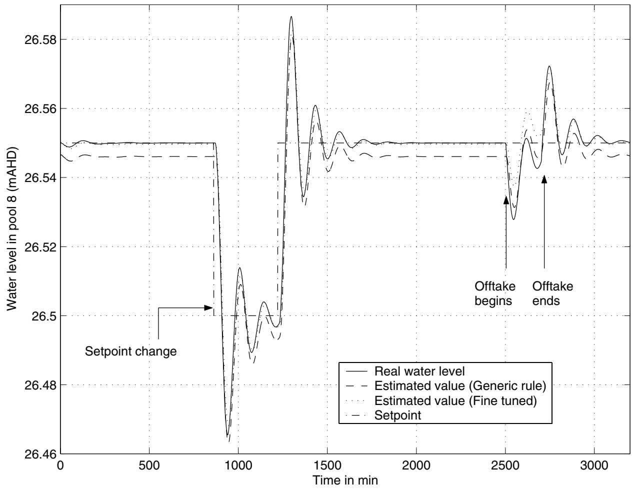


Fig. 5. Estimated water level using the observer approach and real water level in pool 8 when there are no faults.

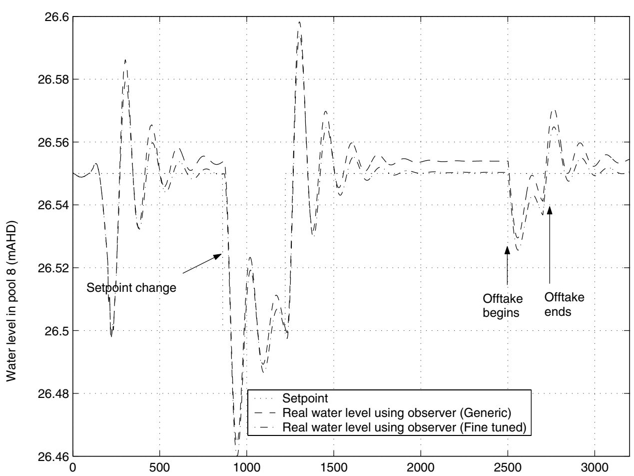


Fig. 6. Water levels in pool 8 when there is a sensor fault.

### C. Performance under different flow conditions

To check the performance of the reconfiguration scheme over the entire flow range under no fault, we forced the observer to operate under different flow conditions. This was achieved by varying the head over the last gate $h_{11}$. A large head over the gate (eg. $1m$) corresponds to a high flow and a small head (eg. $0.2m$) corresponds to a low flow. Fig. 7 shows the estimated water level and real water level across the entire operating range. The linear observer satisfactorily estimates the water level throughout the entire flow range. The observer was designed to give a near zero steady state offset for $h_{11} = 0.5m$. Maximum offsets of $0.0224m$ and $-0.0218m$ were observed under high and low flows respectively. These offsets are small since deviations of $\pm 0.05m$ from the setpoint are usually acceptable. Hence the performance of the linear observer is good throughout the operating regime.

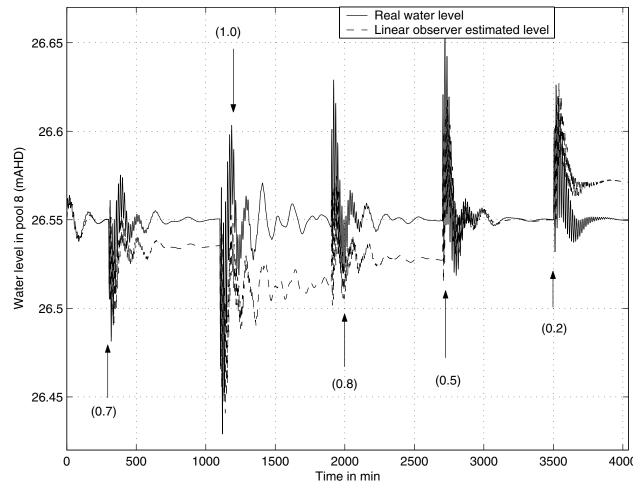


Fig. 7. Estimated water level and real water level with flow varying across the entire operating range (The numbers in brackets are the head over gate 11 in $m$).

## VII. Simulation-Actuator faults

Here we simulated the reconfiguration scheme when gate 9 got completely stuck. All the sensors were working correctly. The water levels in pool $8, 9, 10$ were at the set points of $26.55mAHD$, $23.85mAHD$ and $21.15mAHD$ respectively. The head over the last gate $h_{11}$ was $0.5m$. An offtake was simulated by adding a constant outflow in pool 9 from $t = 500min$ to $t = 700min$ and from $t = 2000min$ to $t = 2200min$. An actuator fault was simulated by making gate 9 stay constant at $p_9 = 26.22mAHD$ in the middle of the first offtake at $t = 550min$. As a result, we gave up on controlling the water level $y_9$ but tried to maintain water level $y_{10}$ at setpoint. As discussed in section V, the control loop was reconfigured as shown in Fig. 4. When the fault occurred, the new controller was initialized by running it offline (in non-real time) with the setpoint error for the last 10hrs before the fault occurred as input. This was done in order to ensure a bumpless transfer. The response is shown in Fig. 8. The fault was detected and the recovery action began at $t = 600min$. It can be seen from Fig. 8 that the new controller recovered and settled smoothly at the setpoint and performed well in the presence of the offtake at $t = 2000min$.

## VIII. Discussion

Observers were also designed and tested using nonlinear approaches e.g. gain scheduling and output injection. All the approaches worked well. Bumpless transfer schemes as proposed in [24] and [25] were also designed and tested. Bumpless strategies did not have any significant effect on sensor fault reconfiguration but actuator fault reconfiguration benefited by having smaller transients. Multiple actuator and sensor faults were also addressed and the reconfiguration schemes were able to mitigate the faults effectively.

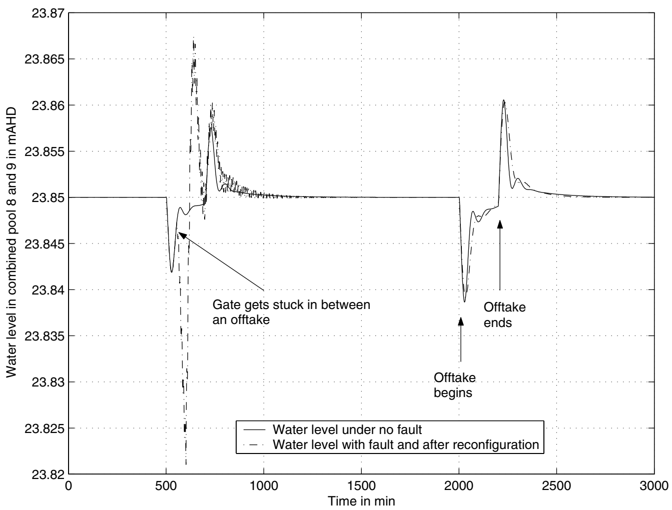


Fig. 8. Water level in pool 9 after reconfiguration in the event of an actuator fault.

## IX. Conclusion

Control and signal reconfiguration schemes to mitigate the effects of sensor and actuator faults were developed in this paper. A generic rule for where to place observer poles which facilitates rapid design of large number of observers as required in a large scale irrigation network was also formulated. The reconfiguration schemes were able to mitigate the effects of sensor and actuator faults and the irrigation channel was brought back to its normal operation smoothly. The reconfiguration scheme based on a first order linear model worked satisfactorily throughout the entire flow range.

## Acknowledgment

This research is part of a collaborative research project between the University of Melbourne and Rubicon Systems on modelling and control on irrigation networks. This research was supported in part by the Centre for Sensor Signal and Information Processing (CSSIP) under the Cooperative Research Centre Scheme of The Commonwealth Government of Australia and in part by the Australian Research Council's Linkage Grant Scheme

## References

[1] Mareels I, E. Weyer, S. K. Ooi, "Irrigation Networks: A systems engineering approach", *Fourth International Conference on Control and Automation, ICCA*, 2003.

[2] Weyer E, "Decentralised PI control of an open water channel", *Proceedings of the 15th IFAC World Congress*, 2002.

[3] Weyer E, "LQ control of an Irrigation channel", *Proceedings of 42nd IEEE conference on Decision and Control*, vol. 1, pp. 750-755, 2003

[4] Weyer E, "Controller configurations for irrigation channels derived from sequential minimum variance control", *Proceedings of 2003 IEEE Conference on Control Applications-CCA*, vol.2, pp. 1209-1214, 2003

[5] Ooi S. K and E. Weyer, "Control design of Irrigation channel from physical data", *Proceedings of European Control Conference*, 2003.

[6] Schuurmanns J, A. Hoff, S. Dijkstra, O. H. Bosgra and R. Brouwer, "Simple water level controller for irrigation and drainage canals", *Journal of irrigation and drainage engineering*, no. 4, vol. 125, pp. 189-195, 1999.

[7] de Halleux, C. Prieur, J. M. Coron, B. d'Andrea Novel and G. Bastin, "Boundary feedback control in networks of open channel", *Automatica*, no. 8, vol. 39, pp. 1365-1376, 2003.

[8] Litrico X, V.Fromion, J. P. Baume and M. Rijo, "Modelling and PI control of an irrigation channel", *Proceedings of European control conference*, 2003.

[9] Malaterre P. O, "PILOTE: Linear Quadratic optimal controller for irrigation canals", *Journal of irrigation and drainage engineering*, no. 4, vol. 124, pp. 187-194, 1998.

[10] Elfawal H. M, D. Georges and N. Ohnishi, "Optimal control of an open channel irrigation system based on nonlinear models", *Proceedings of TENCON 2000*, vol. 3, pp. 308-313, 2000.

[11] Rauch H. E, "Intelligent Fault Diagnosis and Control Reconfiguration", *Plenary speech from IEEE International Symposium on Intelligent Control*. 1994.

[12] Frank P. M, "Fault diagnosis in dynamic systems using analytical and knowledge based redundancy-A survery and some new results", *Automatica*, vol. 26, No. 3, pp. 459-474, 1990.

[13] Blanke M, M. Kinnaert, J. Lunze and M. Staroswiecki, "Diagnosios and fault tolerant control", Springer, 2003.

[14] Gao Z and Antsaklis. P. J, "Stability of pseudo-inverse method for reconfigurable control systems", *Int. J. Control*, 53(3), pp. 717-719, 1991.

[15] Zhang Y and Jin. Jiang, "Design of proportional-integral reconfigurable control systems via eigen structure assignment", *Proceedings of the American Control Conference, Chicago, Illinios*, June 2000.

[16] Sauter D, F. Hamelin, H. Naura, "Fault tolerant control in dynamic system using convex optimization", *Proceedings of the IEEE conference on Decision and Control*, pp. 187-192, 1998.

[17] Polycarpou M, A. Vemuri, "Learning methodology for fault detection and accomodation", *IEEE control systems magazine*, vol. 15, no. 3, pp. 16-24, 1995.

[18] Bos M. G, *Discharge measurement structures*, International Institute for Land Reclamation and Improvement/ILRI, The Netherlands: Waageningen, 1978.

[19] Weyer E, "System identification of an open water channel", *Control Engineering Practice*, vol. 9, pp. 1289-1299, 2001.

[20] Ooi S. K and E. Weyer, "Closed loop identification of an irrigation channel", *Proceedings of the 40th IEEE Conference on Decision and Control*, vol.5, pp.4388-4343, 2001.

[21] Luenberger D. G, "An Introduction to observers", *IEEE Transactions on Automatic Control*, vol-16, no 6, pp. 596-602, December 1971.

[22] Khalil H. K, *Nonlinear systems*,Third Edition, Prentice Hall, 2002.

[23] Ciccarella G, M. Dalla. Mora and A. Germani, "Luenberger like observer for nonlinear system", *Int. J. Control*, vol. 57, no. 3, pp. 537-556, 1993.

[24] Zaccarian L, A. R. Teel, "Antiwindup bumpless transfer and reliable designs: a model based approach", *Proceedings of American Control Conference*, 2001.

[25] Zaccarian L, Y. Li, E. Weyer, M. Cantoni, A. Teel "Antiwindup for marginally stable plants and its application to open water channel control systems", *Submitted for publication*, 2004.