Cooperative Control of Water Volumes of Parallel Ponds Attached to An Open Channel Based on Information Consensus with Minimum Diversion Water Loss*

Christophe Tricaud and YangQuan Chen
Center for Self-Organizing and Intelligent Systems (CSOIS)
Department of Electrical and Computer Engineering
4160 Old Main Hill, Utah State University, Logan, UT 84322-4160, USA
{ctricaud, yqchen}@cc.usu.edu

Abstract—Decision making through information sharing for a canal irrigation system is discussed in this paper. A consensus-based decision algorithm is used to manage water distribution into a parallel ponds network and achieve good quality of service and minimum water-loss. The algorithm is tested in a simulation software reproducing the major dynamics of the system for different scenarios. While not taking into account some of the non-linearities and focusing on feasibility, the algorithm shows interesting results under ideal flow control conditions especially regarding convergence. Robustness is estimated by Monte Carlo evaluation of the effects of delay uncertainty, and the strategy maintains convergence.

Index Terms—Consensus, Irrigation Control, Open Water Channel

I. INTRODUCTION

Because of the extensiveness of irrigation basins and the development of affordable remote sensing, crucial information about flows, water volumes and water levels is now easily available and can be centralized in a database and even connected to the internet. This enhancement of current technology gives more accuracy to the work of decision makers and gives the opportunity to develop automation. However, the non-linearities, delays and uncertainties inherent to open water channels make the conception of control algorithm delicate and require robust control strategies to handle them.

Information sharing is one of the fundamental concepts of cooperative control mostly used in vehicle formation. Cooperative control is used in domains where some type of repetition of identical or non-identical subsystems, which are interconnected together, occurs. In spite of its use is primarily control of multiple-agents, this theory has nevertheless been successfully applied to many different control problem such as streams temperature control ([3], [1]), furnace control ([2]), control of urban traffic network...

This paper attempts to apply such theory to the control of a irrigation basin described in the first part, a control algorithm is then developed to match certain specifications and finally simulations are performed to test the performances and robustness of the strategy.

II. SCIPIO CANAL IRRIGATION SYSTEM

The purpose of this section is to give the reader a clear idea of the considered problem ; as canal irrigation basins can have different architectures and hardware, each requiring specific considerations and, hence, specific control strategies. First, we will give an overview of the Scipio, Utah system and then will define the scheme and assumption made to create a model.

A. System Description

1) The Sevier River Water Management Cyber-Infrastructure: The Sevier River Basin in rural south-central Utah is one of the state’s major drainage. A closed river basin, it encompasses 12.5 percent of the state’s total area. From the headwaters, 250 miles south of Salt Lake City, the river flows north and then west 225 miles before reaching the Sevier Lake (figure 1).

At the early stages of the automation, the different monitoring and automation systems in the Sevier River Basin were giving significant data, but it was only available to a few water managers. Meanwhile, with the development of the Internet, many of the water managers were thinking about making all these data available on the Internet and how it would be a good way to distribute the data among the water users at low cost.

In 1997, StoneFly Technology contacted the users in the Upper Basin to propose connecting their database to the Internet. This proposal was accepted and a preliminary web site was tested. The web site was designed to serve a variety of users with different needs by proposing large panel of ways of presenting information.

There are several real-time data monitoring sensors in place across the entire basin. The current version of the website makes all of this data easily accessible and interpretable (figure 2). A user is able to click on the map indicating which of the six regions he or she would like to access information about. A map is then displayed showing all of the measurement stations in that region. Clicking on the specific station brings up a seven day graph of the data for that station. The window shows what the sensor is monitoring (i.e. Gate Height), the current reading on the sensor, and a color. The color index is the way of showing how recently the data being displayed was recorded (from green for recent to red for outdated).

In addition to the graphical representation of data, all the data can be displayed in tabular form. Graphs are dynamically generated to show the values for different periods of time. This can prove useful in recognizing patterns or seeing disturbances in the water flow or even use the data to create personalized plots.

Since 1999, automation equipment has been installed and is currently collecting data at stations throughout the entire basin, including the basin we are considering in this paper, Scipio. The site
has expanded to include sensors on reservoirs, canals, diversions, and the river itself.

2) The Scipio River Basin: The Scipio system is a closed river basin situated within the Sevier River Basin. It covers around 10 miles of open-channels. The system is composed of a reservoir situated about 5 miles upstream of the first diversion, 4 diversion structures along the main canal diverting water into 4 parallel ponds. The water is pumped by the farmers from these ponds to perform their field irrigation.

A map of the Scipio River basin is shown in figure 3.

B. System Model

The description of the model used to develop the control strategy follows the scheme of figure 4.

Each canal is modeled using the ID model described in [8] with some modifications inherent to the peculiarities of the considered irrigation basin, namely there is no water storage before each diversion gate. The continuous time ID model for any considered canal is given by

$$ q_{\text{downstream}}(s) = e^{-Ds} q_{\text{upstream}}(s) $$

where $q_{\text{downstream}}$ is the flow at the end of the canal, $q_{\text{upstream}}$ is the flow at the origin of the canal, $D$ is the travel time of the considered canal.

Each diversion is modeled using a simple separation of the upstream flow into to separate flows: one going to a pond where the water will be stored, the other still in the main canal to provide water to the downstream ponds. The general model for a single diversion is given by

$$ q_{\text{upstream}}(t) = q_{\text{downstream}}(t) + q_{\text{pond}}(t) $$

where $q_{\text{upstream}}$ is the flow of water coming from the main canal, $q_{\text{pond}}$ is the flow going through the diversion gate and $q_{\text{downstream}}$ is the remaining main canal flow.

Each pond is modeled as the integration of the incoming flow from the diversion gate and the amount of water pumped by the surrounding irrigators. The model obtained for a single pond is

$$ \frac{dV}{dt}(t) = q_{\text{in}}(t) - q_{\text{out}}(t) $$

where $V$ represents the volume of water inside the pond, $q_{\text{in}}$ represents the incoming flow and $q_{\text{out}}$ is flow of water pumped out of the pond.

III. COOPERATIVE CONTROL

Cooperative control is the favorite framework in multiple-agents based problems. Most of the publications written about this topic are based either on mobile robots, UAVs or manipulators. The concept of agent is for the majority associated with these autonomous mobile devices. However, the definition of agent is not limited to this class of system and many of the results obtained can be applied to any group of similar actuators whose purpose is to achieve a common task. This framework fits well to canal irrigation control where water has to be distributed among different users whose needs and rights are different. In this section, general results and theorems developed for cooperative control of mobile robots will presented.

In most problems involving multiagent systems, groups of agents have to be in agreement with each other to be able to work together. Consequently, it is important to enunciate agreement problems in their general form for groups of agents with information flow.

From [4], [5], [6], if a system is modeled by

$$ \dot{\xi}_i = u_i $$

a consensus algorithm can be defined as

$$ u_i = -\sum_{j=1}^{n} g_{ij}k_{ij}(\xi_i - \xi_j) $$
where the weights \( k_{ij} > 0 \) are uniformly bounded, \( g_{ii} = 0 \); and \( g_{ij} = 1 \) if agent \( j \) sends information to agent \( i \) and 0 otherwise.

In this paper, we assume a full communication tree; that is, \( g_{ij} = 1 \), \( \forall i \neq j \). The graph Laplacian which represents the information flow is, according to the definition given in [4],

\[
L(s) = \begin{bmatrix}
-1/3 & -1/3 & -1/3 \\
-1/3 & 1 & -1/3 \\
-1/3 & -1/3 & 1 \\
\end{bmatrix}
\]  

(6)

Given \( L \), the state \( x(t) \) of the network of agents is evolving according to the following linear system:

\[
\dot{x}(t) = -Lx(t)
\]  

(7)

Considering that the solution of (7) with topology \( L \) is given by

\[
x(t) = \exp(-Lt)x(0)
\]  

(8)

by calculation, one can get the value of the state the group agreed on.

IV. CANAL OPERATIONS STRATEGY

In this section, we will first give the aim of the control strategy which will help us in the development of the control algorithm itself. Since the system is monitored and controlled remotely, and the measurements are obtained every hour, we discretize the problem with a sampling time of 1 hour. The system is controlled in discrete time and the sampling time for the update of the information is one hour. Following, we describe the main features of this algorithm.

A. Control Objectives

From [9], an irrigation system’s primary function is to provide water in an precise and accommodating way. Precision is achieved when the supply matches the demand on time, and flexibility is accomplished when the changing demand of the users is still met. This primary goal can be separated in two water level control problems.

First, the amount of water deducted from the reservoir should be able to meet the overall demand inside the irrigation basin. Water has to be released according to all the errors of the different ponds. This assures that the ponds will be provided with enough water to allow field irrigation operations.

\[
V(t) = \int_0^t q(u)du \geq \sum_{i=1}^{4} E_i(t + D_i)
\]  

(9)

where \( q \) is the outflow of the reservoir (in \( m^3/s \)), \( E_i \) represents the error (difference between the reference volume \( V_{ref} \) and the measured volume \( V_i \) of the \( i^{th} \) pond (in \( m^3 \)) and \( D_i \) represents the travel time of the water between the reservoir and the \( i^{th} \) pond (in \( s \)).

Second, the ponds should be controlled using the control structures (the diversion gates) situated upstream, at the diversion with the main canal. This certifies the demand is satisfied [9]; if the water level in one pond comes near its reference value, the diverted flow is higher than necessary, and the diverted flow should be decreased if not stopped.

In addition, in the Scipio system, no water rights apply. Therefore, to maintain equal rights between all the users, if the overall demand cannot be satisfied, the available water should be equally distributed among the different ponds. Doing so, an even amount of water is available to each user; maintaining a regular quality of service throughout the irrigation basin:

\[
V_{ref} - V_1 = V_{ref} - V_2 = V_{ref} - V_3 = V_{ref} - V_4
\]  

(10)

Finally, water loss should be minimized. To make sure all the water available in the system’s reservoir will be used for irrigation purpose that is to say diverted to the ponds, the control strategy should release an amount of water from the reservoir corresponding at least to the ponds need. And if the amount of water released is too large, the excessive water should be equally distributed among the ponds for future use by irrigators instead of flowing into the overflow of the main canal:

\[
\sum_{i=1}^{4} q_i(t + D_i) = q(t)
\]  

(11)

where \( q_i \) represent the inflow into the \( i^{th} \) pond, \( D_i \) represents the travel time of the water between the reservoir and the \( i^{th} \) pond (in \( s \)) and \( q \) is the outflow of the reservoir (in \( m^3/s \)).

B. Control Strategy

In this paper, we assume that the volume of each pond can be obtained and hence is known. It is reasonable to assume that we can estimate the volume of water inside each pond through the measurement of the surface water level and the knowledge of the geometry of the pond. In addition, we presume that we can control each diversion gate so that a given flow is punctured from the main canal and that each diversion gate is able to eventually divert all the flow of the main canal at once. In practice, such condition is very unlikely but we use it to illustrate the capability of the considered strategy to minimize the water loss. If such a condition is not met, the impact on the water loss would be the same for any strategy and would result as an offset independent of the considered strategy.

In order to apply consensus ideas to the system, we need to consider that the system has the dynamics described in Eqn. 4.

Therefore, the following simplifications are made:

- The delays between the different diversion gates are not taken into account
- The delays between the diversion gates and their corresponding pond are not taken into account

The control objectives mentioned in IV-A become:

\[
V(t) \geq \sum_{i=1}^{4} E_i(t + D_i)
\]  

(12)

\[
V_{ref} - V_1 = V_{ref} - V_2 = V_{ref} - V_3 = V_{ref} - V_4
\]  

(13)

\[
q(t) = \sum_{i=1}^{4} q_i(t + D_i)
\]  

(14)

These constraints demand an algorithm based on information sharing.

From the three conditions in (12), we derive the following control algorithm:

1) The volume error of each pond \( E_i \), which is the information shared by all the ponds is computed

\[
E_i(t) = V_{ref} - V_i(t)
\]  

(15)

2) The overall error for the irrigation system is calculated

\[
E(t) = \sum_{i=1}^{4} E_i(t)
\]  

(16)

3) The volume of water dedicated to each pond is computed.

\[
V_{ref}(t + 1) = \max(0, E(t) + \frac{(V(t) - D - E(t))}{4})
\]  

(17)

This equation displays two purposes: if the amount of water available is not large enough to fill all the ponds, the pond with the largest error will be delivered most of the water; and if the amount of water is sufficient to fill all ponds, water is equally distributed among ponds.
4) The total volume of water necessary is estimated

\[ V_c(t + 1) = \sum_{i=1}^{4} V_{ci}(t + 1) \quad (16) \]

5) Each volume is weighted by the amount of water available since equation (5) is non-linear

\[ V_{ci}(t + 1) = V_{ci}(t + 1)(V(t - D) / V_c(t + 1)) \quad (17) \]

6) Each volume is converted into a flow to be diverted

\[ V(t + 1) = E(t) - \sum_{i=0}^{D} V(t - i) \quad (18) \]

This algorithm ensures the respect of the specification regarding the achievement of the water supply and quality of service. However, a large amount of water is wasted during the operation (see Section V).

One explanation for this loss of water is the inability of the strategy to manage the delays between the diversion gates. Consequently, the diversion gates are actuated at the same instant whereas the considered available flow of water is only present at the first gate. Therefore, when the gate command at instant \( t \) is smaller than the one at instant \( t - 1 \), the amount of water at the gate is larger than the expected amount. This difference is hence discarded into the end of the canal where it is not available for irrigation.

To compensate this flaw in the strategy, the diversion commands are delayed according to the travel times between the considered gate and the first diversion gate. Such approach actuates the gates when the flow of water is actually there and not ahead of time. Practically, this can be achieved thanks to the hardware that allows the emission of commands at anytime through radiocommunication.

Once tested (see Section V), this upgrade demonstrates very good results regarding water loss with only residual loss. However, the overshoot of the level is too high and since the only control action we possess is positive (no water can flow upstream back to the canal!), this creates an overflow in the ponds.

These overflows can be explained by the fact that the delays between the gates and the ponds are not considered and hence compensated by the strategy. The water volume measurements used by the algorithm are taken while the control action is not completed. The measured error is larger than predicted and the strategy releases more water from the reservoir than necessary, engendering these overshoots.

To counterbalance this phenomenon, a module that weights the measured volume according to the gate-pond delays and the previous control value is created.

\[ V_i'(t) = V_i(t) + \frac{D}{T_s} V_{ci}(t) \quad (19) \]

where \( V_i'(t) \) is the new value of the measured volume of pond \( i \) and \( T_s \) is the sampling time (1h).

The results of this addition to the algorithm can be observed in the next section, as well as the results of the previously described strategies.

The three strategies will further be referred as “1st strategy” for the initial algorithm, “2nd strategy” for the strategy with the delayed commands and “3rd strategy” for the algorithm with both compensations.

V. SIMULATION AND RESULTS

A. Simulation Platform

To simulate the system, we create a Simulink model based on the dynamics described in II-B.

The values chosen for the travel time in each canal and the volumes of the ponds were approximated from figure 3 and from the knowledge of the water manager in the Scipio irrigation basin. The values are summarized in tables I and II.

<table>
<thead>
<tr>
<th>Canal</th>
<th>Time (in h)</th>
<th>Canal</th>
<th>Time (in h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reservoir</td>
<td>Div.1 4</td>
<td>Div.1 → Pond 1</td>
<td>0.1</td>
</tr>
<tr>
<td>Div.1 → Div.2</td>
<td>1.3</td>
<td>Div.2 → Pond 2</td>
<td>0.1</td>
</tr>
<tr>
<td>Div.2 → Div.3</td>
<td>0.2</td>
<td>Div.3 → Pond 3</td>
<td>0.05</td>
</tr>
<tr>
<td>Div.3 → Div.4</td>
<td>0</td>
<td>Div.4 → Pond 4</td>
<td>0.6</td>
</tr>
</tbody>
</table>

TABLE II

<table>
<thead>
<tr>
<th>Maximum Volume of Ponds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pond 1</td>
</tr>
<tr>
<td>Pond 2</td>
</tr>
<tr>
<td>Pond 3</td>
</tr>
<tr>
<td>Pond 4</td>
</tr>
</tbody>
</table>

The simulation scheme (figure 5) used to test the developed control algorithm embeds: a controllable reservoir outflow, 4 controllable diversions, 4 pond water level sensors and a monitoring of the amount of water lost along the operations. These features allows the proper work of the control algorithm as well as the possibility to simulate field irrigation operations via pumps on each pond.

Two different scenarios are used to test the strategies: one with each pond with a different low initial volume and no disturbance to see the nominal behavior of each strategy, one with evaporation (small permanent loss) and field irrigation operations (large withholds). Both simulations lasted 100 hours.

B. Results

The results of the simulations are displayed here. Figure 6 shows the response of the system with the 1st strategy, under nominal condition (No disturbance). In figure 7, the conditions are operational, that is, water is pumped out of the pond during certain periods of time. Figures 8 and 9 correspond to the 2nd strategy. Finally, figures 10 and 11 show the results for the 3rd strategy.

Even though all strategies achieve the control of the four ponds, a good way to compare their performance is to observe the total
The algorithm that achieved the best overall performance is the strategy including compensations for both delays between gates and between gate and pond.

VI. MONTE-CARLO SIMULATION

Since the assumption of constant travel times along open-water channels is inaccurate, the robustness of the 3rd strategy is tested. 200 simulations are run and each delay is randomly chosen within the ranges described in table IV. These range are chosen as the nominal value of the delay ±20%. The average amount of water released is 185 m³ and the average water loss observed is around...
15 m$^3$ making an overall waste of 8%. An interesting observation in the results is the curve representing the waste of water for each value of travel time from the reservoir to the first pond (figure 12). In figure 12, one can observe that the distribution is parabolic which means that the error on the estimation of the delay should be limited if the water waste is to be minimized.

A second sequence of simulations is performed to test the robustness of the strategy when the travel time from the reservoir to the first diversion gate is known. This time, the average amount of water released is 174 m$^3$ and the average water loss observed is around 4 m$^3$ making an overall waste of 2%. Once again, the display of the water waste function of the largest delay proves interesting (figure 13). Contrary to figure 12, the distribution is linear and therefore, maintaining the estimation of the delay within a reasonable range is not as consequential.

### VII. Conclusion

A control algorithm using information throughout an irrigation basin was developed. Starting with a consensus algorithm with simplistic assumptions and simplifications, we managed to obtain good results by taking into account all the dynamics of the system. However, the lack of unsteady-flow simulation software did not allow to test the strategy further. Nevertheless, we illustrated the fact that cooperative control in canal irrigation can give good results regarding water economy, quality of service and performance. In addition, the strategy has been proved robust towards quality of service and convergence when large uncertainties were considered.

### Table III

<table>
<thead>
<tr>
<th>Nominal</th>
<th>Case</th>
<th>Operational</th>
<th>Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume (m$^3$)</td>
<td>Loss (%)</td>
<td>Volume (m$^3$)</td>
<td>Loss (%)</td>
</tr>
<tr>
<td>1st Strategy</td>
<td>147.5</td>
<td>8.8</td>
<td>215</td>
</tr>
<tr>
<td>2nd Strategy</td>
<td>148.7</td>
<td>0.05</td>
<td>179</td>
</tr>
<tr>
<td>3rd Strategy</td>
<td>153.2</td>
<td>0.10</td>
<td>170</td>
</tr>
</tbody>
</table>

### Table IV

<table>
<thead>
<tr>
<th>Canal</th>
<th>Max. Time (in h)</th>
<th>Min. Time (in h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reservoir → Div.1</td>
<td>4.8</td>
<td>3.2</td>
</tr>
<tr>
<td>Div.1 → Div.2</td>
<td>1.55</td>
<td>1.05</td>
</tr>
<tr>
<td>Div.2 → Div.3</td>
<td>0.24</td>
<td>0.16</td>
</tr>
<tr>
<td>Div.3 → Div.4</td>
<td>0.1</td>
<td>0</td>
</tr>
<tr>
<td>Div.1 → Pond 1</td>
<td>0.12</td>
<td>0.08</td>
</tr>
<tr>
<td>Div.2 → Pond 2</td>
<td>0.12</td>
<td>0.08</td>
</tr>
<tr>
<td>Div.3 → Pond 3</td>
<td>0.06</td>
<td>0.04</td>
</tr>
<tr>
<td>Div.4 → Pond 4</td>
<td>0.7</td>
<td>0.5</td>
</tr>
</tbody>
</table>

### REFERENCES


---

Fig. 12. Water Waste (%) vs. Reservoir Delay Value (h)

Fig. 13. Water Waste (%) vs. 1st→2nd Pond Delay Value (h)

---

3288