REINFORCEMENT LEARNING IN THE OPERATIONAL MANAGEMENT OF A WATER SYSTEM

Andrea Castelletti [*] **Giorgio Corani** [*] **Andrea E. Rizzoli** [**]
Rodolfo Soncini Sessa [*] **Enrico Weber** [*]

[*] *Dipartimento Elettronica e Informazione, Politecnico di Milano, Milan, Italy*
[**] *Istituto Dalle Molle per l'Intelligenza Artificiale, Manno-Lugano, Switzerland*

Abstract: In this paper we present a variant of the classic Q-learning algorithm to design an operating policy for the management of a multi-purpose water reservoir. The model-free approach is applied only to a part of the model of the water system, the catchment, while the remainder, the reservoir, is described by mathematical models based on physical characteristics. This algorithm has been tested on the case of the regulation of Lake Como.

Keywords: Water Resources Management, Reinforcement Learning, Optimal Control

1. INTRODUCTION

To manage efficiently a regulated lake we need to know how much water to release and how to distribute it to the different users. The satisfaction of these users will be expressed in the primary objectives of management. Moreover we want to protect users that only indirectly are affected by the resource usage (e.g. the lakeside population). We can formulate this problem as an optimal control problem, so that its solution will be a management policy answering to our questions.

Unfortunately the control problem is not trivial at all: first, multiple management objectives can be present (satisfaction of water demand for irrigation and hydropower generation, upstream and downstream flood protection, protection of water for recreational use and lake navigation) and very often these objectives are conflicting. Second, the decisional problem is made more complex by the presence of stochasticity of the reservoir inflow process. Third, the complexity is increased by non-linearities in the objective functions and in the system's constraints (stage-discharge functions). Finally, the hydrological processes are usually cyclostationary and this also adds to the intricacy of the problem.

A methodology to design and synthetize reservoir management policies is Stochastic Dynamic Programming (SDP) (Bellman, 1957), which allows to obtain policies that take into account non-linearities and stochasticity, solving the optimal control problem for a dynamical systems with stochastic inputs (for a general introduction to the problem see the review papers by Yakowitz (1982) and Yeh (1985), for some case studies (Gilbert and Shane, 1982), (Read, 1989), (Hooper *et al.*, 1991)).

A limitation of SDP is the impossibility to take explicitly into account the exogenous information, such as average daily precipitation or mean temperatures in the catchment, unless either we write a mathematical model of its dynamic behavior or we describe it as a white process. This is a serious limitation when supplemental hydrological information is available and could be used. Note that this information is often known only qualitatively and it is difficult to describe it using mathematical

models. Even when these models are available, they are far too complex to be used in an optimal control problem that can be solved by SDP, mostly because they enlarge the dimension of the state space and therefore SDP incurs in *curse of dimensionality* (Bellman, 1957).

The object of this work is to propose a policy design algorithm overcoming these limitations. Algorithms based on Reinforcement Learning (see for instance Kaelbling *et al.* (1997) and Barto and Sutton (1998)), originally developed in the Robotics and Artificial Intelligence fields, have been shown to be able to generate control policies that make an effective use of qualitative information. In this study we used the Q-learning algorithm (Watkins, 1989) that finds the optimal policy by trial-and-error, testing alternative control actions and observing the resulting model state (model free approach). We propose a Q-learning variant, named in the following *Qlp* (Q-learning planning), which adopts a model free approach only for a subpart of the system: the catchment. Qlp is therefore able, as the original Q-learning algorithm, to use exogenous information without the need for a model. The rest of the problem is formalized as in the SDP case: the optimal policy is computed backwards in time, the dynamics of the reservoir and the cost functions are described by explicit quantitative models.

The paper is organized as follows: first we introduce the model of the system and the formulation of the control problem; then we describe the classical SDP algorithm and the Qlp algorithm. Finally, we describe the case study of Qlp for the management of Lake Como, in Italy.

2. THE PROBLEM

Ever since the first pioneering works by Maas (1962), the problem of the management of a regulated lake has been represented with a feedback control scheme with feedforward compensation (Figure 1). The control policy, which is the key element of the scheme, returns the volume u_t to be released from the reservoir, once the current storage value s_t is known. When feedforward compensation is present the policy also depends upon the vector I_t, that is, the state of the meteorological and catchment systems, which are affected by a stochastic disturbance ε_t. According to the chosen modelling representation, the components of the vector I_t can be, e.g., the piezometric head of groundwater or the reservoir inflow over the last 24 hours before the release decision; the stochastic disturbance ε_t can be, e.g., the atmospheric pressure, the solar radiation, the rainfall. The control policy can be determined as the solution of an optimal control problem.

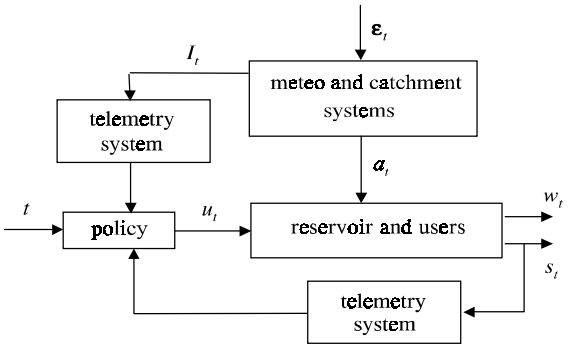

Fig. 1. Feedback control scheme for a water system.

The reservoir is represented by the mass conservation equation:

$$s_{t+1} = s_t + a_{t+1} - r_{t+1}(s_t, u_t, a_{t+1}) \qquad (1)$$

where $r_{t+1}(\cdot)$ is the actual release in the interval $[t, t+1)$.

The meteorological system and the catchment are the most difficult components to model, because of the complexity of the meteorological and hydrological processes. Very often only the catchment is considered and the reservoir inflow a_t is represented by simple stochastic autoregressive models of order q:

$$a_{t+1} = \chi_t(a_t, a_{t-1}, ..., a_{t-q-1}, \varepsilon_{t+1}) \qquad (2)$$

where ε_{t+1} is a white gaussian noise.

The model of the whole dynamic system can thus be represented in the compact vectorial equation:

$$x_{t+1} = f_t(x_t, u_t, \varepsilon_{t+1}) \qquad (3)$$

where the state vector x_t is composed by the couple (s_t, I_t). Because of the periodicity of the climate, function $f_t(\cdot,\cdot,\cdot)$ is periodic of period T equal to one year.

During the system evolution, the state transition from x_t to x_{t+1} produces an instantaneous cost g_t, given by the weighted sum of the elementary costs g_t^j associated with k management objectives:

$$g_t = \sum_{j=1}^{k} \lambda^j g_t^j \qquad (4)$$

where λ^j are weights. Also the step costs g_t^j are periodic of period T.

The decision u_t is assumed to be given by:

$$u_t = m_t(x_t) \qquad (5)$$

where $m.(x.)$ is the policy, which is a periodic function.

Given the discount rate α, the cost associated with a policy p and an initial state x_0 is defined as:

$$J(x_0, p) = \lim_{h\to\infty} \sum_{t=0}^{h} \mathop{\mathrm{E}}_{\varepsilon_1,\dots\varepsilon_{t+1}} \left[\alpha^t g_t(x_t, u_t, \varepsilon_{t+1})\right] \tag{6a}$$

$$x_{t+1} = f_t(x_t, u_t, \varepsilon_{t+1}) \tag{6b}$$

$$\varepsilon_{t+1} \sim \phi_t(\varepsilon_{t+1}|x_t, u_t) \tag{6c}$$

$$u_t \in U_t(x_t) \tag{6d}$$

$$u_t = m_t(x_t) \tag{6e}$$

$$p = [m_0(\cdot), m_1(\cdot), ..., m_{T-1}(\cdot)] \tag{6f}$$

The optimal policy p^o is therefore the solution of the following optimal control problem:

$$J(x_0) = \min_{p\in P} J(x_0, p) \tag{7}$$

3. THE SOLUTION BASED ON STOCHASTIC DYNAMIC PROGRAMMING

The solution of the optimal control problem (7), based on SDP, requires to evaluate, for each couple (t, x_t), the optimal cost-to-go $H_t^o(x_t)$ that is defined as the cost that the system would incur, from time t onward, when it is controlled by policy p^o and starts from state x_0. If the optimal cost-to-go would be known for every value of x_{t+1}, the optimal decision $m_t^o(x_t)$ at time t would be easily found minimizing the expected value of the present cost plus the discounted optimal cost-to go from time $t+1$:

$$m_t^o(x_t) = \arg\min_{u_t} \mathop{\mathrm{E}}_{\varepsilon_{t+1}} \left[g_t(x_t, u_t, \varepsilon_{t+1}) + H_t^o(x_{t+1})\right] \tag{8}$$

with $t = 0, 1, ...$

The optimal cost-to-go associated with the couple (t, x_t) is therefore given by the following equation:

$$H_t^o(x_t) = \min_{u_t} \mathop{\mathrm{E}}_{\varepsilon_{t+1}} \left[g_t(x_t, u_t, \varepsilon_{t+1}) + H_{t+1}^o(x_{t+1})\right] \tag{9}$$

$\forall x_t$ with $t = 0, 1, ...$

The optimal policy p^o is fully defined by the equation (8) for each time step t if the function $H_\cdot^o(\cdot)$ is known. This function, the *Bellman function*, can be obtained solving the recursive equation (9).

Under the previous hypotheses, the Bellman function turns out to be a periodic function of t, of period T, which can be obtained using the Successive Approximations Algorithm (SAA) (Bertsekas, 1995). This algorithm proceeds backwards in time, solvoing at each time step t equation (9) where the next state x_{t+1} is computed by means of the transition function (6b). Note that an explicit model for each one of the system's component must therefore be available.

4. THE SOLUTION BASED ON Q-LEARNING

In the field of Reinforcement Learning, Q-learning is the algorithm more similar to SDP, and it can be easily reconduced to its optimality principle (Bertsekas and Tsitsiklis, 1996). In Q-learning, the optimal costs-to-go are expressed not only as a function of state x_t but also of control u_t and they constitute the *Q-function*. The following relationship can be established between the Q-function and the Bellman function:

$$H_t^o(x_t) = \min_{u_t} Q_t^o(x_t, u_t) \tag{10}$$

and therefore the optimal policy p^o is given by:

$$m_t^o(x_t) = \arg\min_{u_t} Q_t^o(x_t, u_t) \tag{11}$$

Also the Q-function is periodical of period T, like the Bellman function. Equation (9) can thus be substituted by the following recursive expression:

$$Q_{t_{\mathrm{mod}\,T}}(x_t, u_t) \leftarrow (1-\gamma_k) Q_{t_{\mathrm{mod}\,T}}(x_t, u_t) + \gamma_k \cdot \cdot \left[g_t(x_t, u_t, a_{t+1}) + \alpha \min_{u_{t+1}} Q_{(t+1)_{\mathrm{mod}\,T}}(x_{t+1}, u_{t+1})\right] \tag{12}$$

with $t = 0, 1, ...$ and where γ_k represents the learning rate, which decreases with the number of updates k that have been already made of the value $Q_t(x_t, u_t)$.

Differently from SAA, which is off-line and works backwards, Q-learning works on-line and forward; the evaluation of next state by means of (6b) is thus substituted by the direct observation of the behaviour of the real world system. For this reason, the Q-learning algorithm is said to be *model free*. At every time step, a single control u_t is applied and only at time $t+1$, when the new system's state x_{t+1} is directly observable, the value $Q_{t_{\mathrm{mod}\,T}}(x_t, u_t)$ is updated using (12).

The Q-value is therefore updated by means of a weighted sum of its current value and of a term which keeps track of the recent system's past. Such a process (learning phase) continues until the learning rate γ_k is sufficiently close to 0, so that the contribution due to further exploration becomes negligible. When this happens, an empirical estimate of the optimal Q-function,

defined by (10), has been obtained. In practice, to reach a good approximation of the optimal policy by means of (11), it is necessary to explore extensively (in theory forever) the whole space of (t, x_t, u_t) (Watkins and Dayan, 1992).

A necessary requirement for the applicability of Q-learning is that the time constant of the physical system must be sufficiently small to obtain a satisfactory control policy in a reasonable time. A second requirement is that the system can be trained by a trial-and-error procedure. In the case of a regulated reservoir, both requirements are not satisfied. The decisional time step is daily and in many cases the dam is operated at an even lower frequency (i.e. weekly). As a consequence, the training horizon would be in the order of decades. Moreover, it is not possible to perform trial-and-error experiments, since they would cause costs which could not be socially acceptable (for instance, the direct and indirect damages caused by a single flood day in lake Como, Italy, can be estimated as close to 500.000 Euros).

For this reason, we have devised a variant of Q-learning, which we have named Qlp (Q-learning planning). It allows to overcome the limitations of SDP and standard Q-learning, integrating the off-line approach, typical of SDP, and the model-free characteristics of Q-learning.

Qlp exploits a structural characteristic of regulated reservoirs: the model is composed of subsystems in cascade: the meteorological and catchment systems and the reservoir itself. Moreover, the dynamics of the first two systems is free, in other words, it is not affected by the control applied to the reservoir. These two characteristics allow to replace the models of the meteorological and catchment systems with the historical trajectories of the reservoir inflow a_t and of the information I_t. As in the SDP case, the state transition of the reservoir is simulated at each time step, for all the values of the state s_t, and for all the feasible controls, thanks to (1), but, unlike in SDP, this happens in correspondence of the historical value of the inflow $\bar{a}_t$ and of the information $\bar{I}_t$ and not for values synthetically generated by mathematical models. It is therefore easier to quickly and effectively explore the whole (t, x_t, u_t) space.

Equation (12) is therefore substituted by:

$$Q_{t_{\mathrm{mod}\,T}}(s_t, \bar{I}_t, u_t) \leftarrow (1-\gamma_k) Q_{t_{\mathrm{mod}\,T}}(s_t, \bar{I}_t, u_t) + \\ + \gamma_k \left[g_t(s_t, u_t, \bar{a}_{t+1}) + \right. \\ \left. + \alpha \min_{u_{t+1}} Q_{(t+1)_{\mathrm{mod}\,T}}(s_{t+1}, \bar{I}_{t+1}, u_{t+1}) \right] \quad (13)$$

which must be computed at time t for all couples (s_t, u_t) all over the N years that are available to train the policy.

It is therefore no more necessary to identify the probability distribution of the disturbance, since stochasticity is now implicitly represented in the natural sequence of inflow and meteorological information. The constraints (6b-6e) become:

$$s_{t+1} = s_t + \bar{a}_{t+1} - r_{t+1}(s_t, u_t, \bar{a}_{t+1}) \quad (14)$$

$$u_t \in U_t(s_t, \bar{I}_t) \quad (15)$$

$$u_t = m_t(s_t, \bar{I}_t) \quad (16)$$

5. CASE STUDY

The proposed methodology has been applied to the design of control policies for Lake Como. The performances obtained by simulating these policies over a 15-years horizon have been compared with those achieved applying SDP and those actually realized by the DM over the same period.

Lake Como, situated in Northern Italy, has an active storage of 260 $10^6 m^3$, and receives water from a catchment of about 4500 km^2 generating an yearly mean inflow of 4730 $10^6 m^3$. It was transformed into a reservoir in 1946 by constructing a regulation dam at Olginate at the outflow of the lake. Since then the lake has played an important role in the development of the Padana plain economy.

The water released from the lake supplies six agricultural districts, for a total irrigated surface of 144.000 hectares, and seven run-of-river hydroelectric power plants with a total installed capacity of about 92 Mw for an annual mean production of 473 GWh.

The primary objectives of the regulation are to supply water to downstream agricultural users and to provide flood protection on the lake shores. The hydropower supply has not been considered as one of the system's primary objectives, since a preliminary study (Guariso *et al.*, 1986) has shown how the mean yearly electricity production is not substantially lowered by the improvement of the other objectives.

Two different informative systems have been compared: the former (named IS1) assumes that, at time t, when the decision u_t must be taken, only the water storage s_t is known (i.e. I_t is empty); the second (IS2) that the inflow a_t in the last 24 hours is also known (i.e. $I_t = a_t$). Using the SDP algorithm, the inflow is described as a white noise when the first informative system is considered and as an autoregressive filter of first order (AR(1), see (2) with $q = 1$) in the other case.

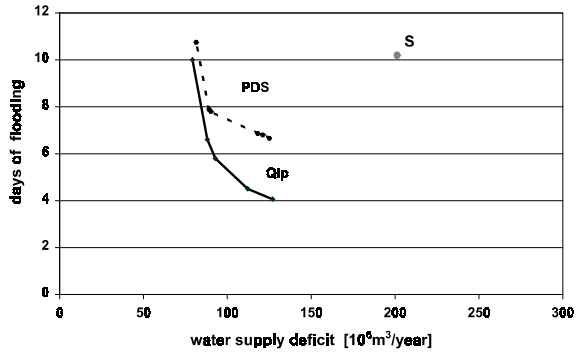

Fig. 2. Performances in calibration with IS1.

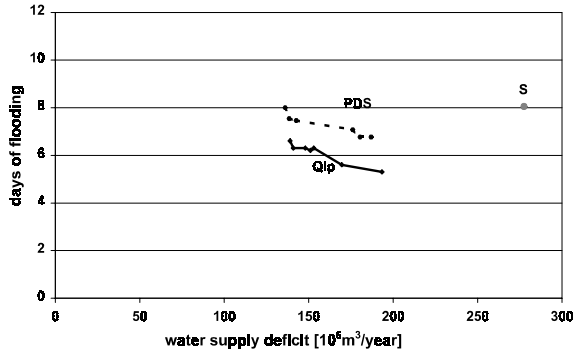

Fig. 3. Performances in validation with IS1.

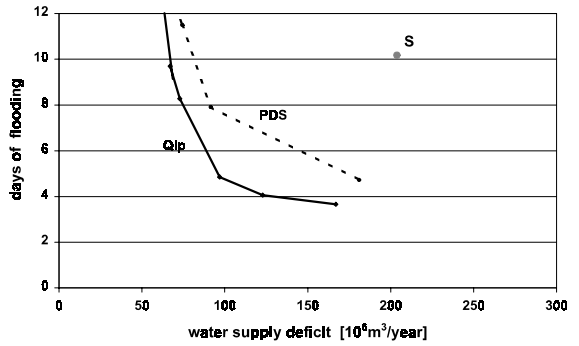

Fig. 4. Performances in calibration with IS2.

Given a value for the weight λ in (4), two optimal policies have been calibrated for each informative system: namely one with SDP and one with Qlp. The years (1965-1979) have been used for that scope. By varying the weight λ the Pareto boundary is obtained for each of the four cases. To evaluate the results, in both cases the performance of the controlled system has been computed via simulation on the years (1980-1997). The results obtained with IS1 are presented in Figures 2-3. Points S correspond to the performances attained by the historical management. Notice that the policies designed by SDP (dotted line) are outperformed by the policies obtained by Qlp (continuous line). The improvement has been obtained in correspondence of both objectives and therefore Qlp solutions dominate in Paretian sense SDP solutions.

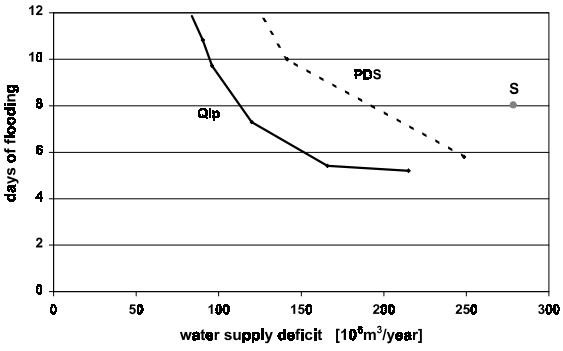

Fig. 5. Performances in validation with IS2.

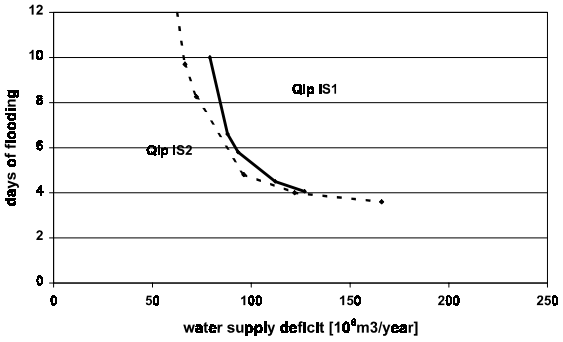

Fig. 6. Qlp performances with the two informative systems.

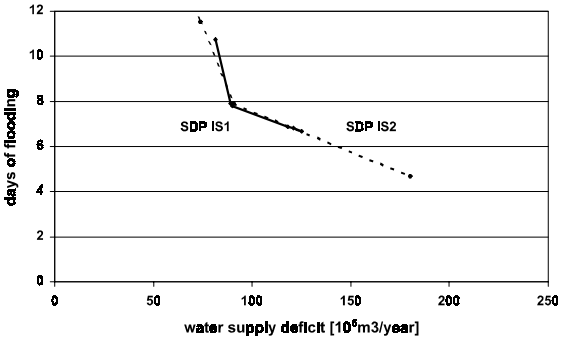

Fig. 7. SDP performances with the two informative systems.

Figures 6 and 7 show that the performances attained with the system IS2 dominate the performances of system IS1 when Qlp is adopted. The reason is that when system IS2 is used Qlp is able to exploit all the orders of correlation that exist in the inflow process, while SDP only the first order.

6. CONCLUSIONS

A new algorithm, named Qlp, has been proposed. It is a variant of the Q-learning algorithm and it can be used to solve reservoir management problems. Such systems are composed by a non-controllable part (the meteo and catchment systems) that is hard to model and a controllable part (the reservoir) that is easy to model. Therefore it

is of definitive interest to avoid the modelling of the first part. The Qlp algorithm, proposed in this work, allows to reach this goal. The case study of Lake Como shows how policies generated with Qlp are more efficient (in Paretian sense) than the ones produced by SDP.

7. REFERENCES

Barto, A. and R. Sutton (1998). *Reinforcement learning: an introduction*. MIT Press. Cambridge, MA.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press. Princeton, NJ.

Bertsekas, D.P. (1995). *Dynamic Programming and Optimal Control*. Athena Scientific. Boston, MA.

Bertsekas, D.P. and J.N. Tsitsiklis (1996). *Neuro-Dynamic Programming*. Athena Scientific. Boston, MA.

Gilbert, K.C. and R.M. Shane (1982). Tva hydroscheduling model: theoretical aspects. *J. Water Res. Plann. Manage. Div. Am. Soc. Civ. Eng.* **108**(1), 21–36.

Guariso, G., S. Rinaldi and R. Soncini-Sessa (1986). The management of lake como: multiobjective analysis. *Water Resour. Res.* **22**(2), 109–120.

Hooper, E. R., A.P. Georgakakos and D.P. Lettenmaier (1991). Optimal stochastic operation of salt river project, arizona. *J. Water Res. Plann. Manage. Div. Am. Soc. Civ. Eng.* **117**(5), 556–587.

Kaelbling, L. P., M.L. Littman and A.W. Moore (1997). Reinforcement learning: a survey. *J. of Artif. Int. Research* **4**, 237–285.

Maas, A., M.M. Hufschmidt, R. Dorfam, H.A. Thomas, S.A. Marglin and G.M. Fair (1962). *Design of Water Resource Systems*. Harvard Univ. Press.

Read, E.G. (1989). A dual approach to stochastic dynamic programming for reservoir release scheduling. In: *Dynamic Programming for Optimal Water Resources Systems Analysis* (A.O. Esogbue, Ed.). pp. 361–372. Prentice-Hall. Englewood Cliffs, NJ.

Watkins, C.J.C.H (1989). Learning from Delayed Rewards. PhD thesis. Cambridge University. Cambridge, U.K.

Watkins, C.J.C.H. and P. Dayan (1992). Q-learning. *Machine Learning* **8**, 279–292.

Yakowitz, S. (1982). Dynamic programming applications in water resources. *Water Resour. Res.* **18**(4), 673–696.

Yeh, W. (1985). Reservoir management and operations models: a state of the art review. *Water Resour. Res.* **21**(12), 1797–1818.