# Supporting Alternative Designs of Reservoir Management Policies

Andrea E. Rizzoli[1], Rodolfo Soncini Sessa[2], Enrico Weber[2]
[1]IDSIA Lugano, Switzerland
[2]Politecnico di Milano, Italy
{andrea@idsia.ch, soncini@elet.polimi.it, weber@elet.polimi.it}

**Abstract.** TwoLe, a Decision Support System (DSS) for the efficient management of multi purpose reservoir networks is presented. TwoLe introduces the concept of "two level" decision support aimed at separating the role of the System Analyst (SA), who designs the management policies, and the Decision Maker (DM), who takes release decisions. TwoLe is designed around a software architecture aimed at the independence of data, models and optimisation algorithms. In this paper the use of TwoLe to design and evaluate alternative reservoir management policies is shown.

## 1. INTRODUCTION

Water supply for agriculture and for power generation is often provided by reservoir networks. In most cases, each reservoir collects water from a catchment, the power plants are usually located at the reservoir discharge, while the most important agricultural users are downstream the network output, so they can use the water collected from all the catchments covered by the network. In some cases, natural lakes are used as reservoirs. The major problem of the management of a reservoir network is due to the presence of conflicting objectives: to the above mentioned agricultural water supply and hydroelectric power generation, we must also add the protection from flood events and the satisfaction of environmental flows constraints.

Managing these systems is often a complex and demanding task. It has been shown that human experience alone is not sufficient to guarantee a n optimal performance. Moreover, the activity of policy design also requires substantial mathematical skills, which often are not in the background of the reservoir manager in charge of regulating the reservoir. For these reasons, TwoLe introduces the concept of two level decision support aimed at separating the role of the System Analyst (SA), who designs the management policies, and the Decision Maker (DM), who adopts those policies to take the daily water release decision (Soncini-Sessa et al. 1990).

We focus our attention on the first of the two levels, the Planning Level, which supports the SA in the policy design phase. The SA needs the possibility of testing various design alternatives before producing the policies that will be implemented at the second TwoLe level, the Management Level. The DSS architecture of TwoLe provides a set of components and tools (the domain and the model bases, the experiment base and a set of algorithms) which helps the SA in the policy design tasks.

Using TwoLe, the SA can build alternative models of the reservoir network (e.g. choosing different catchment models to describe the upstream water inflow, choosing among various water user models), and use them as constraints in the formulation of control problems. TwoLe allows to formulate multi-objective control problems, thus addressing the problem of conflicting objectives. TwoLe provides a set of solvers, based on different algorithms such as stochastic dynamic programming, neuro dynamic programming based on artificial neural networks, and Q-learning, which can be used to solve the control problem. The SA has therefore the freedom of testing alternatives, and, according to the chosen algorithm, the possibility of expressing his/her risk predilection or aversion, being thus able to better adapt the policy to the DM needs.

TwoLe is currently being employed in the study on the management of lake Maggiore, a trans-national lake,where the interest of the Swiss users (recreation and flood protection) are in conflict with those of the Italian users (irrigation and power generation) (Soncini-Sessa et al. 1999).

In the next section, the architecture of the TwoLe DSS is presented. In the last section, the use of TwoLe to design alternative reservoir management policies is demonstrated. Finally, some conclusions and suggestions for future work are drawn.

## 2. THE TWOLE ARCHITECTURE

The architecture of the TwoLe DSS is centred on the following modules:

- the domain base, which allows to create and edit the data describing the physical parameters of the water system;
- the model base, which contains the mathematical models representing the

behaviour of the water system components;

- the solver toolbox, which provides the TwoLe user with a set of calibration, simulation and optimisation algorithms, which can be employed to solve the various problems (parameter estimation, control policy synthesis, etc.), which arise in the management of water systems.

The user dialogue and interaction module provides a graphical user interface to present the services provided by these modules in a consistent framework.

## 2.1 The domain base

A component of a water system, such as a catchment, has a set of characteristic attributes. The attribute values are stored in the domain base. It is the first modelling layer: no assumptions are made on the kind of mathematical relationships, but only on the data and their representation. Being the data relative to a model of a natural resource such as a water system, we adopted a hierarchical approach to its classification, according to an object-oriented analysis paradigm.

The domain base contains basic domain objects (shortly BasicDObj) and compound domain objects (CompDObj). BasicDObj are used to keep together the data attributes of simple modelling components. Building a BasicDObj is the first modelling step: raw data is organised in data structures which hypothesise a modelling structure and purpose. Among basic domain objects we find: measurement stations, catchments, reservoirs, channel junctions and diversions, water channels and non consuming water users, consuming water users, alarm gauges.

The domain base also contains a set of objects that represent the system structure: compound domain objects. An example of CompDObj is the water system that can be seen as composed by a number of catchments, reservoirs, channels, junctions, diversions and water users. A part of it is another CompDObj, the distribution network, itself composed by water users, channels, junctions and diversions.

We called this repository the domain base, in accordance with the proposal by Del Furia et al. (1995) since it contains the structural knowledge regarding the objects appearing in the modelled domain. The knowledge contained in the domain base must be completed with the interactions and relationships among the modelled objects (the model equations): this happens in the model base.

## 2.2 The model base

Domain objects play the important role of logically dividing the physical world from the mathematical one: different models can be associated with the same domain object. This is the key to model re-use and to the easy prototyping of alternative water system models (Rizzoli et al. 1998). This feature is a keystone in the design of alternative management policies as it will be shown in Section 3.

Models can be classified according to their purpose. A first classification based on the model use leads to the following model classes: descriptive models, control problems, regulators.

Descriptive models are mathematical models of the objects contained in the modelling domain, for instance:reservoirs, catchments, channels, water users, and so on. These models are defined basic models since they cannot be decomposed further. They can be assembled to construct compound models: a water system is described by a compound model made of, at least, catchments, reservoirs, channels, and water users.

The definition of the problem to compute a release policy or a distribution policy is a control problem. For this reason, we divide them into release control and distribution control problems. Release control problems are in turn divided into single and multi objective problems. Single objective problems are further specialised into a priori and a posteriori problems, according to the use they make of available information. A posteriori control problems compute on-line policies to take advantage of the updated information provided by a telemetering network (Nardini et al., 1994). Multi objective control problems are made by a collection of single objective problems and can be used to explore the Pareto frontier of the problem (see Section 3). Solving a control problem produces a control policy, which is implemented in a regulator. In some cases, the regulator also makes use of a state reconstructor, if the state vector includes the catchments states.

## 2.3 The solver toolbox

The third module in the TwoLe architecture is the solver toolbox. The toolbox contains:

- calibration algorithms that can be applied to descriptive models to calibrate their parameters;
- optimisation algorithms to be applied to control problems to generate regulators;
- validation and simulation algorithms to be applied to regulated models (descriptive models where decisions are taken by regulators);

This set of solvers has been designed in order to be compliant with the specifications of the interfaces

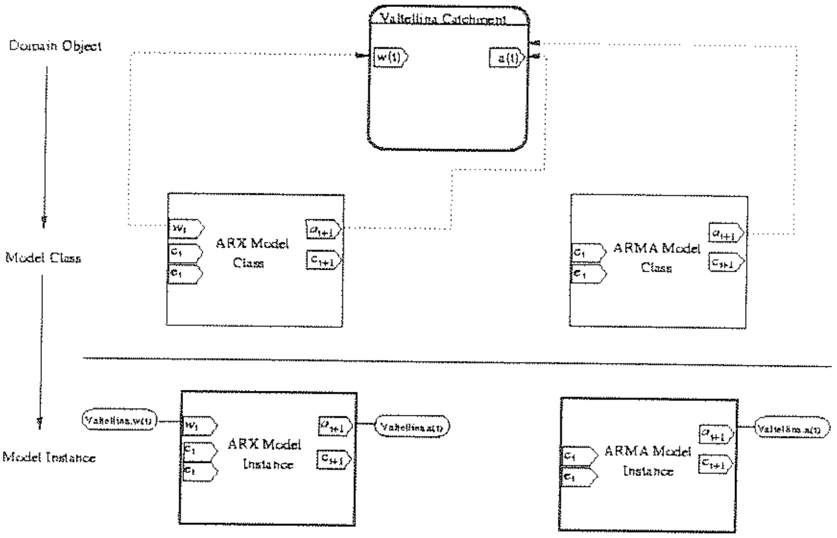


Figure 1. Alternative modelling options for a catchment domain object.

of the models stored in the model base. This allows the independence of the models from the solvers: adding a new solver to the toolbox is as easy as adding a new model in the model base as long as the interface specifications are respected.

In Twole the following problems can be solved: (a) water system optimisation; (b) distribution network optimisation; (c) catchment calibration and validation; (d) catchment outflow forecasting; (e) design of flood alarm policies; (f) water system simulation.

## 3. DESIGNING RESERVOIR MANAGEMENT POLICIES

The TwoLe architecture has been designed with the objective of making a DSS in which the DM could have confidence. The management of complex water systems is a task that has been traditionally delegated to skilled operators with a long on-site experience. Their proprietary knowledge allows them to integrate the operating rule (usually a lower and higher limit for the release) with the on-line information made available by weather forecasts, the reservoir state, the expected water demands. Most DMs have operated satisfactorily in the past and it is very difficult to convince them that the computer-generated policies can sensibly improve real-world management (Guariso et al., 1990). For this reason the DSS must gain the DM trust. To achieve this objective, the TwoLe architecture, as presented in Section 2, must be put into action to serve the DSS cause. In the remainder of this section we show how TwoLe supports: (a) model prototyping; (b) innovative optimisation algorithms; (c) analysis of the decision space; (d) policy validation and assessment.

### 3.1 Model prototyping

There is no such thing such as the definitive water system model. A class of widely used optimisation algorithms are based on Dynamic Programming, which is an exhaustive search method which explores all the solution space. Such algorithms are computationally intensive and even the use of parallel algorithms and computers can only reduce but not eliminate the dimensionality problem (Piccardi and Soncini-Sessa, 1991). It is therefore important to balance the accuracy of the model with the complexity (which is often related to the dimension of the model state vector). Trials and errors, together with a feeling for the problem, are required to achieve this balance. The dichotomy between the domain base and the model base is aimed at this objective: the user can test alternative model formulations always relating them to the domain objects which contain the essential characteristics of the physical system.

An example of the use of this approach is given by the reservoir model, The same physical reservoir may be described by many alternative models which differ only by the storage discretisation. Another example is reported in Figure 1. The catchment domain object Valtellina catchment defines a compact interface, made of the exogenous input w(t) and the catchment outflow a(t). In this example two alternative model classes can be constructed on the domain object, the first one is an ARX model that makes use of the exogenous inflow, the second one is an ARMA model. These model classes add their own interface variables (in this case the disturbance $\varepsilon_t$ and the inner state $c_t$) which are relative to the adopted mathematical description. The model classes retrieve data (e.g. the input and output time series) from the domain object. With these data, catchment models can be calibrated using the calibration and validation algorithms provided by the solver toolbox.

### 3.2 Innovative algorithms

Classical optimisation algorithms have several shortcomings: (a) they propose a unique value for the control action, the optimal value. The user can only choose to adopt this proposed control or leave it. When the suggested control is against the DM s intuition, it is often rejected. The consequence is that the remainder of the policy is compromised; (b) classical algorithms based on stochastic dynamic programming (SDP) perform an exhaustive search in the solution space. These algorithms are therefore impractical to use in water systems with more than three state variables (this is a ballpark figure, actually it depends on the reservoirs characteristics and on the discretisation adopted. Each reservoir needs a state variable and also catchments need state variables in their models); (c) most of the information available on catchments is often disregarded in the formulation of the catchment models for the need to keep them simple, to avoid the enlargement of the state vector, for the same reasons as above.

The algorithms designed and implemented in TwoLe cannot solve all these problems, but they are an interesting mix of traditional and innovative methods, thanks to the flexible software architecture of TwoLe.

TwoLe implements traditional algorithms based on SDP, but it also enhances these algorithms with the concept of set-valued policies (see Aufiero & Soncini-Sessa 1995 a,b,c for details). The DM is proposed a set of alternative controls, instead of a single point. This allows the DM to select the control that best matches his/her own feeling and experience, without losing the optimality of the solution. These algorithms are implemented in two variants, according to the formulation of the cost function in the control problem: efficiency and risk-aversion. While the efficiency criterion is shown to produce the best performance, the risk-aversion criterion is liked by the DM since it protects against extreme events (i.e. floods, droughts) since it aims at minimising the worst situation. These algorithms are now well assessed and have been applied to different case-studies.

The problem of dimensionality in unfortunately not yet solved, but we have implemented in the TwoLe solver toolbox an algorithm based on Neuro-Dynamic Programming (Bertsekas and Tsitsiklis, 1996) to overcome this problem. The basic idea is to combine the function approximation power of Artificial Neural Networks with the traditional Dynamic Programming approach based on Bellman's equation. Thus, to compute a Bellman function it is enough to calibrate a Neural Network approximating it, saving us from exploring the whole solution space. Applied research is on-going on the application of this methodology in the TwoLe DSS and first preliminary results showed its practicability.

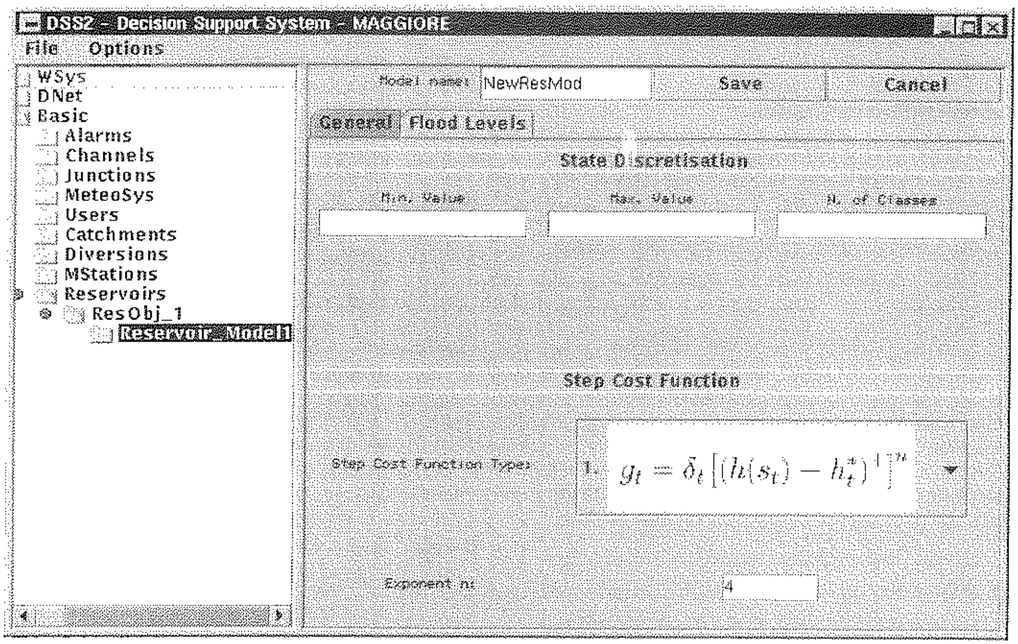
In a more advanced stage of development is our work on a Q-Learning variant of the DP algorithm (Corani and Castelletti, 1999), which was designed to make a better use of the historical rainfall scenarios that are available for most catchments. In the proposed approach, we used Q-learning, which is a technique of Reinforcement Learning, to design an algorithm able to learn the catchment dynamics from historical scenarios and the associated outflows. The approach was not completely model-free since the water system model still includes the equations for the reservoir and the downstream users. The algorithm has been applied to the case of lake Como, in Italy.

### 3.3 Analysis of the decision space

The management of reservoir networks often implies the presence of many conflicting objectives (Tauxe et al. 1979). Even small alpine reservoirs must nowadays comply with environmental flows constraints, thus adding this extra objective to the original one of hydro-power supply generation. TwoLe supports the decisional process allowing the DM to explore the decision space by defining and solving a set of single-objective control problems.

Each single-objective control problem has a unique performance indicator to be optimised thanks to a weighting (or reference, in the risk-aversion case) method. The single-objective control problem is defined by selecting a water system model. Each basic model in the water system may have an associated cost function: typically power plants have a cost function measuring the production of power as a function of the water supply, while the cost functions of the agricultural users express the price of water deficits. The different objectives are then weighted to produce a unique value of the performance indicator.

The DM can build alternative control problems for the same water system model selecting alternative formulations of the cost functions. This is another level of user control in the design of the management policies: from the structural definition



Model name: NewResMod

Step Cost Function Type: 1. $g_t = \delta_t \left[ (h(s_t) - h_t^*)^+ \right]^n$

Exponent n: 4

Figure 2. A snapshot of the graphical user interface.

of the system (the domain base) to the mathematical representation (the model base), and finally, in the control problem the DM selects how to evaluate the information produced by the models building the performance indicator

The solution of multiple single-objective control problems for different values of the weights can be plotted in the objectives' space. The set of efficient solutions determines the Pareto frontier of the multi-objective control problem. The Pareto frontier can then be a valuable tool in the negotiation phase which leads to the selection of a release policy to be implemented in a water system.

### 3.4 Policy validation

Finally, TwoLe completes its range of services providing the DM with simulation and validation tools, which can be used to test and assess the impact of the proposed policies. Currently, three simulation methods are implemented in TwoLe: (a) Markov simulation for the water systems that can be described as a Markov chain. They are all the water systems solved by pure SDP methodologies. Markov simulation allows to compute the probability that the reservoir storage will be in a given discrete state over time. (b) scenario based simulation can be used for all the models, since it only needs pre-recorded time-series to be input in the water system. Typically, the catchment inflow sequences. (c) Monte Carlo simulation is a case of scenario based simulation, where the input time series are corrupted by a stochastic noise. Different simulation runs are performed and statistics are collected.

### 3.5 The user interface

The user interface reflects the hierachical structure of the domain and the model bases. The domain objects are represented as folders in a dedicate window.

In the folder structure each top level item in the folders box at the left can be expanded, to reveal the meta-domain objects belonging to that class (such as WSys for water systems and DNet for distribution networks). This structure is recursive, each meta level object opens on the real domain objects (such as the Maggiore Domain Object Catchment) and, in turn, each domain object will be expanded to reveal its models (Reservoir Model 1 in our example). Compound domain objects and models can be built from existing basic domain objects/models.

Solvers can be applied to models. According to the model type, the user is presented the possible tool to apply, for instance, a catchment can be calibrated, or used as a forecaster, the user can select the corresponding tool.

The "toolbox and folders" metaphor has the clear advantage of giving a logical structure to the user desktop: possible actions always assume that the user has picked an item from the folders and a tool from the toolbox. The folder structure is represented by an example snapshot in Figure 2.

The TwoLe user interface also reflects the object-oriented approach of its design. Objects on the screen are the active elements that can be accessed and operated either by mouse clicks or by menu item selections.

### 3.6 Implementation notes

Most of the algorithms have been implemented in C++ code which has been ported and tested on various UNIX platforms (Solaris 5.1, Digital DG-UX, HP/UX, Linux). The domain and the model bases have been implemented in PostgreSQL, an Open Source relational database (http://www.postgresql.org). The user interface has been implemented in Java (http://java.sun.com) and it is therefore platform independent.

## 4. CONCLUSIONS

The Planning module of TwoLe, a two-level DSS for water management, was presented. An advanced model management system allows the testing of alternative modelling options, while the data base management system enables the decision makers to generate scenarios corresponding to many possible external influences. This makes TwoLe a powerful environment in which alternative management policies can be built and tested, thus building the DM s confidence in them. Applications of this DSS scheme are currently under development in various sites.

## 5. REFERENCES

Aufiero, A. and R. Soncini-Sessa, Set-valued control laws in minimax control problems. *Tech. Report no.95/07*, DEI-Politecnico di Milano, Milano, Italy. 1995a.

Aufiero, A. and R. Soncini-Sessa, Set-valued control laws in TEV-DC control problems. *Tech. Report no.95/30*, DEI-Politecnico di Milano, Milano, Italy. 1995b.

Aufiero, A. and R. Soncini-Sessa, Discounted minimax control problems and set-valued control laws. *Tech. Report no. 95/31*, DEI-Politecnico di Milano, Milano, Italy. 1995c.

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

Castelletti, A. and G. Corani, Q-Learning nella Gestione dei Sistemi Idrici. *Master Thesis*, Politecnico di Milano, Milan, Italy (in Italian). 1999.

Del Furia, L., A: Rizzoli and R. Arditi, Lakemaker: a general object-oriented software tool for modelingthe eutrophication process in lakes. *Environmental Software*, 10(1), 43-64. 1995.

Guariso, G., S. Rinaldi, and R. Soncini-Sessa, A decision support system for the management of Como lake: structure, implementation and performance, *CISM-IIASA School on Methodology, Implementation and Applications of DSS*, Udine, Italy. 1990.

Nardini, A., C. Piccardi, and R. Soncini-Sessa, A decomposition approach to suboptimal control of discrete-time systems. *Optimal Control and Methods*, 15, 1-12. 1994.

Piccardi, C. and R. Soncini-Sessa, Stochastic dynamic programming for reservoir optimal control: dense discretization and inflow correlation assumption made possible by parallel computing. *Wat. Resources Res.* 27(2), 729-741. 1991.

Rizzoli, A.E., J.R. Davis, and D.J. Abel, A model management system for model integration and re-use. *Decision Support Systems*, Vol 24, pp. 127-144. 1998.

Soncini-Sessa, R., A. Nardini, Gandolfi, C. and A. Kraszewski, Computer aided water reservoirm anagement: a prototype two level DSS. Invited at: *NATO ARW on Computer Aided Support Systems in Water Resources Research and Management*, September. 23-28, Ericeira, Portogallo. 1990.

Soncini-Sessa, R., D. Canuti, A. Colorni, E. Laniado, F.B. Losa, A.E. Rizzoli, L. Villa, B. Vitali, and E. Weber, Planning and management of a transnational multipurpose water system, the case of Lake Maggiore, Italy-Switzerland. *Submitted to Water International*. 1999.

Tauxe, G.W., R.R: Inman, and D.M. Mades, Multiobjective dynamic programming with application to a reservoir. *Wat. Resources Res.* 15(6), 1403-1408. 1979.