UNESCO-IHE
Institute for Water Education

R.K. Price and D.P. Solomatine

A Brief Guide to Hydroinformatics

Delft, 2009

1. WHY HYDROINFORMATICS?

The growing world water crisis is in part a failure of human society to be aware of the problem and its possible solutions. This crisis is perceived in different ways. For some it is the conflict between different interests or sectors for water resources, such as between environmentalists and agriculturalists. For others it is learning how to deal with the problems of climate change, whether leading to drought and desertification or more frequent and severe flooding in rivers and from coastal waters. For yet others it is dealing with the problem of pollution of existing resources, whether from point sources such as industry and urban wastewater or from diffuse (non-point) sources such as agriculture and varying land uses. The problems seem to be increasing and solutions in many cases are no longer simple to generate or implement. As engineers and scientists, we have become aware that water management involves an integrated view of a number of distinct systems that would previously have been dealt with in isolation, and consequently there is a need for collaboration with experts from a number of other disciplines. What is more, we need also to take into account the requirements of a range of stakeholders who have a direct interest in the performance of a given water-based system.

Many of the integrated systems that we now deal with have very complex interactions that are not immediately apparent. The normal way of trying to assimilate the complexity is to form a (single, integrated) model of the integrated systems. What our minds are unable to do because of the complexity and the calculations involved, we give to the computer. This is particularly important when we move from a steady state to an unsteady state analysis. Simulation modelling has therefore become an important tool in order to understand the behaviour of complex systems and to enable predictions to be made of that behaviour under changes to various boundary conditions or internal conditions, such as parameters or even functional representations of different identified phenomena. This approach to problem solving has pervaded a wide number of topic domains, not least the management of water-based systems.

Our ability to model and analyse complex water-based systems is due almost entirely to the development of digital technologies. These have revolutionised the way in which we can reproduce the behaviour of such systems, especially in using graphics to analyse and present data, to track the building of models and to visualise output in ways that replicate images of the real world. Such facilities are indicative of the way in which technology generally is pervading our lives and taking over the functions that we would normally have reserved for the activities of the human mind. There are some interesting consequences here for the way in which we relate to the world around us. It can be argued that our involvement with computers is a commitment to virtual reality. Such an experience taunts us with the prospect of being ‘in the picture’ ourselves. This is in contrast to our ‘traditional’, rational view where there is a rigorous separation between society and nature. We regarded nature as a pool of resources for us to master and dominate. Departmentalised knowledge became the ultimate power with which to reign over and to manage these resources. Now however, we increasingly see ourselves as part of the system. All knowledge is situated in a given context and is itself a resource. As such, it is subject to management itself. We talk about knowledge producers and knowledge consumers.

However, knowledge is not a scarce resource in that it can be ‘sold, exchanged and renewed indefinitely, without depleting the original store of knowledge’ (Abbott and Jonoski 2001). Access to knowledge is increasingly measured in terms of access to electronic networks. This demarcates three ‘worlds’: the first world having access both to the ‘content’ and possibility to influence it, the second consisting of those with the possibility of access but not of influence, and the third world consisting of those who neither have access or influence. ‘Knowledge circulation processes in the third world have positive potential for fighting poverty…. (through) new kinds of sociotechnical arrangements’ (Jonoski 2002). Hydroinformatics is the discipline that can implement this new paradigm.

2. WHAT IS HYDROINFORMATICS?

Hydroinformatics uses simulation modelling and information and communication technology to help in solving problems of hydraulics, hydrology and environmental engineering for better management of water-based systems. It provides the computer-based decision-support systems that now enter increasingly into the offices of engineers, water authorities and government agencies.

Defining hydroinformatics, Abbott (1991) points out that it is essentially a technology, in the sense that it concerns human action in which a revealing of ‘truth’ about something occurs, and which in turn influences social structures and relationships. Hydroinformatics can therefore be called a ‘sociotechnology’.

The emergence of hydroinformatics can be traced back to developments in computational hydraulics. Again, Abbott (1991) has identified several generations of modelling. The first generation was characterised by the use of (the first) computers as calculation devices of analytical expressions; in other words, as little more than superior slide rules. As users recognised the value of the sequential, repetitive and recursive modes of operation of their digital machines they turned to finite differences in order to represent and then solve differential equations numerically. This second generation of modelling resulted in one-off or customised models, which were imperfect approximations to (partial) differential equations that were regarded as ‘the ultimate and most perfect repositories of our belief systems’ (Abbott, 1993). From about 1970 onwards, developers recognised the possibility and value of producing software packages such that a given modelling package (or system) could be applied to the instantiation of models for a wide class of similar problems. This enabled resources to be invested on a system that could be used repetitively and refined in the light of experience. In addition, standards could be developed for input and output, preceding future links to databases, GIS and graphical display tools. Most importantly, the systems could be made available to a wide range of users. In turn, the effectiveness of these third generation systems became heavily dependent on the skills and experience of the users. It was at this stage that computational hydraulics became inevitably locked into the commercial side of the market.

The success of the third generation systems depended on ‘main frame’ computers. However, in the early 1980s personal computers appeared as serious professional tools, and it was natural for the modelling software packages of the third generation to be ported to them. In turn, the whole mode of operation of the systems aspect of the packages was rapidly improved. This resulted in the modelling systems being used by people who were not computational hydraulics specialists. They demanded high standards of robustness, consistency and ease of use from the software providers, who adopted production means from software engineering and the IT industry. It meant also that the focus of the developers was on the technology rather than scientific research. The resulting fourth generation modelling systems have subsequently been transformed through close integration with databases, GIS and sophisticated graphics display tools that were foreseen in the third generation.

The birth of hydroinformatics has been identified by Abbott (1996) to have occurred during the transition between the third and fourth generations of modelling. He points out that the revealing offered by the technology has been made to many thousands of users of fourth generation modelling systems, even if the users are still predominantly specialists in hydraulics, hydrology and water resources employed by different organisations. It is the range of these organisations: consultants, contractors, government ministries and agencies, contract laboratories, water companies, universities, investment and insurance companies, etc that indicates the social consequences of the modelling systems. Indeed the social and employment structures of these organisations are changing as a consequence of their using the systems.

At root, the fundamental key change that has distinguished hydroinformatics is a movement from representing knowledge in hydraulics and hydrology by symbols to signs. Here symbols function in the minds of water engineers as tokens by replacing features and processes in the real world. On the other hand signs point towards these features and processes. The generic knowledge encapsulated in the modelling systems of today is now available to a large number of people outside the comparatively small group of developers. This knowledge, operating on site-specific knowledge, enables new knowledge to ‘come to presence’ in the mind of the tool user, depending on their ability to interpret and assimilate that knowledge. The importance of this communication of knowledge has been emphasised by the increasing proportional investment in interfaces and supporting tools, as evidenced by the stress placed on modelling environments rather than modelling engines. Abbott (1992, 1993, 1994) highlights this view by referring to a model as a ‘collection of indicative signs that serves as an expressive sign’. Therefore, a model is not just a ‘simplified representation of reality’ or a ‘system for converting inputs into outputs’. Instead the emphasis is on ‘the choice, number and arrangement of the indicative signs in order to produce the expressive signs (that) determine the very expression that a certain model creates’. ‘Models serve as devices for communicating knowledge’.

Hydroinformatics therefore occupies the middle ground, not just between physical water sciences and ICT but including a third pillar or pole, namely ‘social’ (Fig. 2.1). Therefore, there is a real need to appreciate and understand the social context within which hydroinformatics operates. Jonoski (2002) states: ‘Because of its reliance on physical science, hydroinformatics has its strength, because of its employment of ICT it can become powerful, and because of its social awareness its applications provide value’.

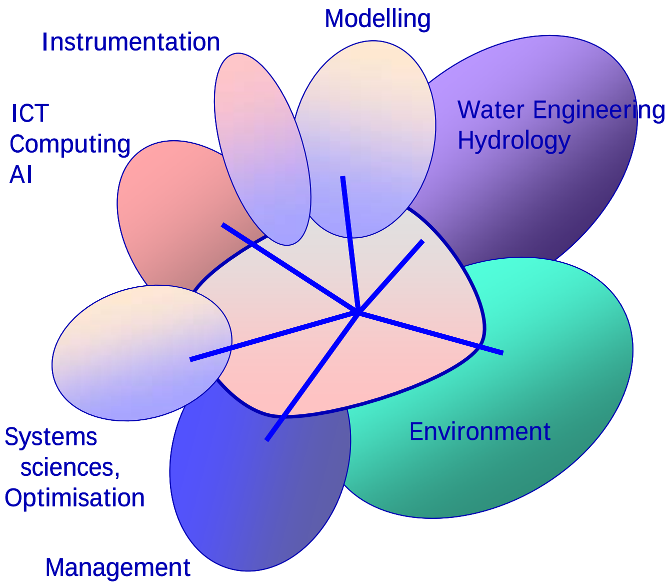

Fig. 2.1. Position of hydroinformatics

The fifth generation of modelling was foreseen by Abbott to come about through the introduction of artificial intelligence (AI) to support modelling and decision-making. Many attempts have been made to construct such fifth generation systems, but none has yet achieved the wide spread use of the existing fourth generation systems. This is partly because of the now recognised inadequacy of rule-based (expert) systems heralded in the mid-1980s as the way forward for computing. Since then there has been a wide diversification into alternative forms of AI, ranging from new forms of data driven modelling (artificial neural networks, genetic algorithms, chaos theory, model trees, fuzzy logic, etc) to intelligent agents, and emerging in different ways of providing decision support.

3. THE 'HOW' OF HYDROINFORMATICS

3.1. Modelling

A model is a simplified description of reality. Hydroinformatics typically deals with computer-based models, where such a model is defined as a computer program that attempts to simulate an abstract model of a particular physical process or system. It can be said that modelling is at the heart of hydroinformatics.

Modelling has two primary goals. The first is to improve understanding about the performance of the real world domain, such as a river catchment or reach, and aquifer, water distribution or drainage network, estuary, or coastal waters. This will involve reproducing past performances and understanding why they happened. The second goal is the ability to make predictions about the performance. There are two forms of prediction. One is to do with identifying the performance of the domain when it is physically altered through human intervention, such as building a dam in a river basin, rehabilitating an urban drainage network, or building embankments along a river to protect associated floodplains from inundation. The other form of prediction is to do with what happens to the future performance of the existing or modified domain. Such predictions may be used in real time to forecast flow variables for warning purposes or for action in operating structures such as barrages, gates, pumps or turbines.

There are three main modelling paradigms that are the focus of hydroinformatics:

1. Physically based (process based) modelling (also called numerical, or simulation modelling), which is based on a scientific understanding of the physics of the flow of water, the chemistry of the associated substances and the biology of the ecology in the aquatic environment. An example is the 1D continuity and momentum equations for open channel flow (Saint Venant equations):

$$\frac{\partial A}{\partial t}+\frac{\partial Q}{\partial x}=q_L$$

$$\frac{\partial Q}{\partial t}+\frac{\partial}{\partial x}\left(\frac{Q^2}{A}\right)+gA\frac{\partial h}{\partial x}-gAS_o+gAS_f=0$$

where Q=flow, h=depth, x=distance, t=time, A=cross-section, S_0=bottom slope, and S_f=energy grade line slope. These equations cannot be solved analytically except under very limited conditions. The alternative is to solve them numerically using a discretisation of the solution space and generating appropriate numerical equations based on a finite difference, finite element or finite volume approach to the original algorithmic equations. Particular algorithms are then used to solve the resulting numerical equations. The algorithms are coded in a particular software language and executed on a computer. The discretisation and solution of the mathematical equations describing the motion of water in the natural or urban environments is known as computational hydraulics.

2. Data-driven modelling (DDM), which is based on a direct the analysis of the data characterising the system under study. Such a model is defined on the connections between the system state variables (input, internal and output variables) with only a limited number of assumptions about the "physical" behaviour of the system. The contemporary methods go much further than those used in conventional empirical modelling in hydraulic engineering and hydrology. They allow for solving numerical prediction problems, reconstructing highly non-linear functions, performing classification, grouping data, and building rule-based systems. An example is a linear regression model, or a non-linear one (for example, an artificial neural network), linking past rainfall measurements and current river flow.

3. Agent-based modelling, where entities (agents) interact dynamically according to relatively simple rule-based computational codes. An example is the behaviour of a school of fish in a water stream where each individual fish is modeled separately. This approach is still in its infancy and will not be covered further in this paper.

Besides viewing a model as a description of the performance of a particular real world domain it is also important to be aware of the process by which a model is developed and instantiated. The process of modelling typically includes the following steps:

- State the problem (why do the modelling?)
- Specify the modelling methods and choose the tools
- Carry out the modelling:
    - Collect, prepare and survey the data
    - Choose variables that reflect the physical processes

  - Build the model
  - Calibrate the model parameters using the measured data
  - Evaluate the model uncertainty
  - Test (validate) the model using the “unseen” data sets
- Apply the model
- Evaluate results

The development of software packages for physically-based models follows a number of well-defined steps (Dee, 1993):

- Represent the (generic) physical laws in terms of mathematical algorithms
- Replicate the resulting (conservation) equations in terms of a digital representation of the algorithms
- Solve the resulting difference equations within particular boundary constraints and conditions
- Design, code and test the numerical procedures
- Design, produce and test the resulting software system that can be linked to other systems such as databases, GIS, CAD, 2D and 3D graphics, and so on.

The resulting software tools are then used with the data for a particular instance of a water-based system to generate (or instantiate) a computational model of that system.

The user would go on to identify and test scenarios, and to select the preferred scenario and carry out sensitivity tests as necessary.

These models are constructed by the process of conceptualising the real world system into structural and process objects and abstracting the collection of objects into a feasible system. A hydroinformatics model in Abbott’s sense has indicative signs that point to the structural and process objects and expressive signs that point to the output objects, such as graphics of the results. All these are designed by the developer who works from his/her own worldview, and particularly for the class of problems that the software package is supposed to address. It does not follow that the user has the same point of view as the developer. Consequently, there are big risks that the user will apply the software package outside the limits for which it was designed. Much is left up to the user concerning how to structure his/her model, what data to select, how to calibrate the model, how to interpret the results, etc. The decision maker is usually yet a third person, who is even more remote from the modelling process, but intimately concerned with what the model produces in terms of data that will assist in the decision making process.

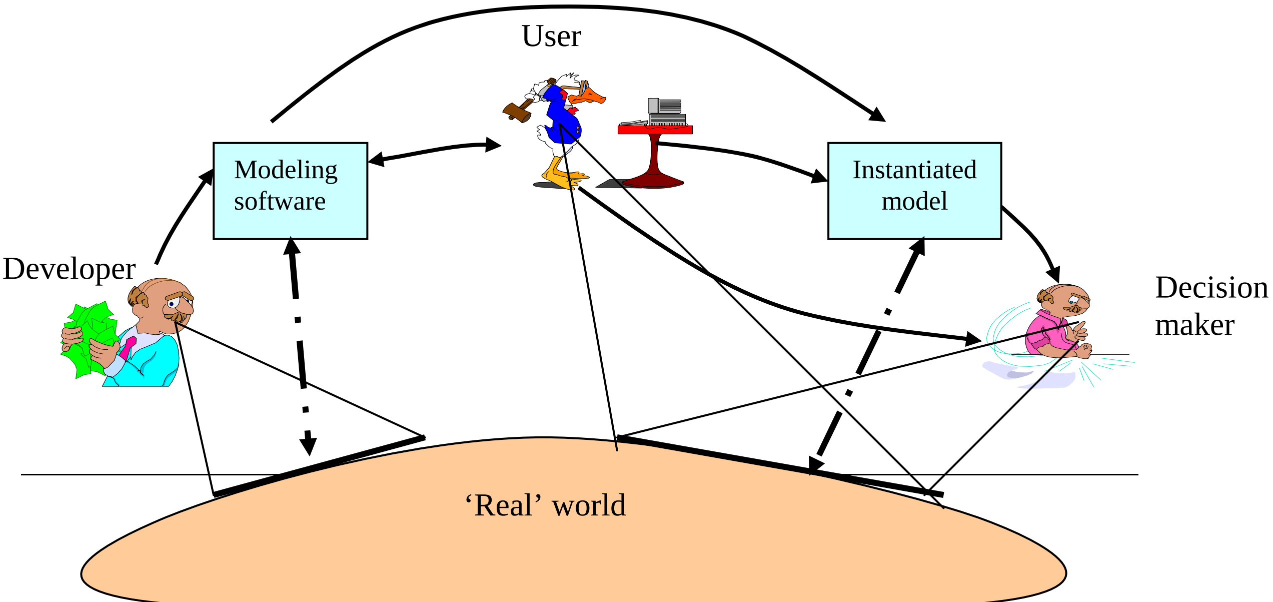

Fig 3.1 Relationship between developer, user and decision maker

3.2. Application of physically-based models

The application of modelling software tools has brought about radical improvements in our understanding of large-scale water-based systems, such as rivers, estuaries and coastal waters. The tools have been extended to include the advection and dispersion of pollutants in the flow, the transport of sediment in suspension and as bed load, the consequences of the flow environment for different biological species, the interaction of flow with structures, and so on. Better models for the closure problem in replicating turbulence, the possibility of feedback on flows from morphological changes whether in rivers or coastal waters, links between flow models and limited area atmospheric models, integration of detailed models of groundwater and surface flow, urban drainage, treatment and receiving waters, are changing the way in which modelling is being done. More emphasis is now being given to what is a safe and reliable software modelling system or an instantiated model, because it is on the basis of the results generated by instantiated models that important and far reaching decision are being made.

As Jonkers (2001) points out, models of complex systems reduce complexity. Structures and processes are conceptualised, and simultaneously decomposed. The resulting concepts facilitate communication. They enable the structures and processes of the real world (society as well as nature) to be explored and mapped. The models themselves are valuable instruments for mining, refining and even formalising ‘tacit’ knowledge. In addition, they provide a rational framework for archiving and retrieving formalised knowledge and information. Finally, models are vital in supporting and governing the control of complex systems as well as in facilitating training and education.

A particular feature of computational hydraulic modelling is that it is a vigorous commercial activity. Organisations such as Danish Hydraulic Institute, Delft Hydraulics (Netherlands) and HR Wallingford (UK) have lead the way in making sophisticated modelling packages available. For example, MIKE11 from DHI is probably the leading commercial package for river modelling in terms of the number of copies released, with more than 2000 copies in use worldwide. There are however, many free, public domain products also available. The difficulty is that the latter products rarely have guaranteed ongoing support, unless it is from a major supplier in the US such as the US Army Corps of Engineers. There is growing need for better support in the *process* of modelling rather than just the modelling product. This speaks of the need for better training in using modelling products, especially as they become more and more sophisticated when the chance of misusing the products becomes greater.

Models are therefore a means to an end. At some stage, where models are used as tools in engineering design, decisions have to be made. Decision-makers need safe and reliable information on which to make their decisions. Yet there are inherent problems with the models themselves in terms of the uncertainties that are introduced through the development of the software tools and the art of collecting, selecting and implementing data from the real world with the modelling software to produce instantiated models. What is more, the decision-makers are no longer a select body of people: decision-making is increasingly being shared within our communities, with a number of different stakeholders involved. The more open the process of decision-making, the more transparent the modelling results have to be. It is with this in mind that Abbott and Jonoski (2001) have developed the dual concept of ‘fact’ and ‘judgement’ engines within the context of an Internet-based decision support environment, designed for a range of stakeholders. The emphasis therefore is turning more to the processes in which modelling systems are being used rather than the modelling systems themselves.

Another discernible trend is the movement towards greater reliance and use of data from the real world. Admittedly, many engineers have been sceptical about the value of modelling systems: they much prefer to deal with the real world at first hand. Even though the collection of data carries with it its own inherent problems, they have greater trust in collected data than the models that utilise the data. Having collected the data the engineer still has to analyse it in order to discover new information and knowledge about his or her system. They go through a process of ‘knowledge discovery’, looking for patterns and anomalies in the data. They may even attempt some form of statistical analysis on the data to deduce particular relationships. This approach to data modelling is now being extended using a range of data-driven modelling techniques. The idea is to look for connections between different categories or sets of data. Particular connectionist techniques include artificial neural networks, fuzzy logic generators, model trees, and so on.

As models are means to an end they are usually incorporated within engineering decision-making processes. A good example of the latest state in the development of hydroinformatics is its role in conceiving, designing and implementing the bridge and tunnel connection between Denmark and Sweden. The latest modelling tools were used to predict currents in conjunction with remote sensing and monitoring. A particular communication system was devised to take into account and keep informed a number of active stakeholders in the construction process. Eco-systems were preserved and costs were reduced. See Thorkilsen and Dynesen (2001). Similar developments are taking place in other areas such as urban water asset management such as for sewerage and water supply. An example of model development and application in river basin management is presented by Falconer et al., (2005) among others.

The safe instantiation of models is prejudiced by several factors. The first is errors or missing values in the data measurements, or insufficient data, for example when calibrating a physically based model or training a data driven model. Another factor is missing modelling objects in a physically based model, such as not including an overflow in an urban drainage network. Then there are missing processes in a model, such as not including evaporation in a data driven model for a catchment where evaporation is important. An unfortunate consequence of having missing or inadequate data is force fitting the model to the data. For example, this can result in model parameters, such as the boundary roughness in a physically based model of flow in a channel, calibrated to have a value outside the normal range. Again, due to the approximations and uncertainties in the input data, model structure and solution process, there is a need to be formally aware of the uncertainties in the model predictions. Finally we need to be assured that the model is 'fit for purpose'. In other words, it can be used with confidence for decision making.

Some "golden rules" of modelling can be formulated:

- Try to ensure data is good
- Follow appropriate modelling procedures
- Be prepared to use models of various types (for example, complement physically based models by the data-driven ones)
- Do not trust models blindly
- Learn what is inside a model

3.3. Data-driven modelling

Physically based modelling depends on a knowledge of the physics (or chemistry and biology) being encapsulated within the software in some way. The software then provides a direct link between the input to an instantiated model and the corresponding output. Usually such models are deterministic in that there is a unique output for a given input (provided the input data is complete). One of the advantages of such models is that following calibration and confirmation they can be applied with some degree of confidence within a range of input data covered by the calibration, and even for a limited degree of extrapolation due to the encapsulated physics. Precisely how far the model can be extrapolated with confidence depends on the quality of the structural data and possibly on the definition of some of the critical (conceptualised) processes (such as the definition of conveyance across a section in 1D river modelling). Physically based models can also generate a large amount of information on what happens away from the boundaries (where data is prescribed as input or output). In addition, modifications can be made to the structural objects in order to assess the performance of the system for different scenarios that involve structural change.

Data-driven modelling is very different to physically based modelling, despite its similar purpose of connecting one set of data (the output) with another corresponding set (the input). The basic idea is to work with data only on the 'boundaries' of the domain where data is given, and to find a form of relationship(s) that best connects the specific data sets. The relationship can take a form that has little to do with the physical principles that might be used in, say, a physically based model.

The main feature of data driven modelling is, in fact, *learning* from available data, which incorporates the so far unknown mappings (or dependencies) between a system's inputs and outputs (Mitchell 1997). By data we understand the known samples that are combinations of inputs and corresponding outputs. As such, a dependency (viz. mapping, or 'model') is discovered (induced), which can then be used to predict (or effectively *deduce*) the future system's outputs from known input values.

By data we usually understand it to be a set K of examples (or instances) represented by the duple $<\mathbf{x}_k, \mathbf{y}_k>$, where $k = 1,\ldots, K$, vector $\mathbf{x}_k = \{x_1,\ldots,x_n\}_k$, vector $\mathbf{y}_k = \{y_1,\ldots,y_m\}_k$, n = number of inputs, m = number of outputs. The process of building a function (or 'mapping', or 'model') $\mathbf{y} = f(\mathbf{x})$ is called *training*. Often only one output is considered, so $m = 1$.

In the context of water modelling the inputs and outputs are typically real numbers ($\mathbf{x}_k$, $\boldsymbol{y}_k \in \Re^n$), so the main learning problem to be solved is numerical prediction (regression). Sometimes problems of clustering and classification are also solved.

Consider an example where an attempt is made to build a data-driven model linking the input variable X and output Y (Fig. 2.1). A set of observations (x_i, y_i) is given (denoted by points). A data-driven model representing this data set could be a linear regression model $Y = a_0 + a_1 X$ (Model 1). Other, non-linear models can also be built: Models 2 and 3. We now ask: What is the best model? And how do we define "best"?

If we look purely at the model error then the Model 3 would be the best – its plot goes through all the points so it has zero error on the training set. However in real life the data may be noisy so it should not be seen as a very accurate representation of the modelled system. So a model that is very accurate on the training data set, may have captured not only the general trend in data, but also the noise. It is said that such model "overfits" the data. Hence, if the purpose of modelling is to capture a general trend in the data, then Model 2 would be a better representative (approximator) of the data set (whereas linear Model 1 is too simple and inaccurate). Indeed it is said Model 2 has a higher *generalisation*, since in a general case of encountering new unknown data it has a better chance of making a better prediction of the output. It is up to a modeller to judge the quality of the data and to decide what type of model is most adequate in a particular situation.

There are many machine learning techniques that can be used to build non-linear data-driven models. One of the most popular is an artificial neural network.

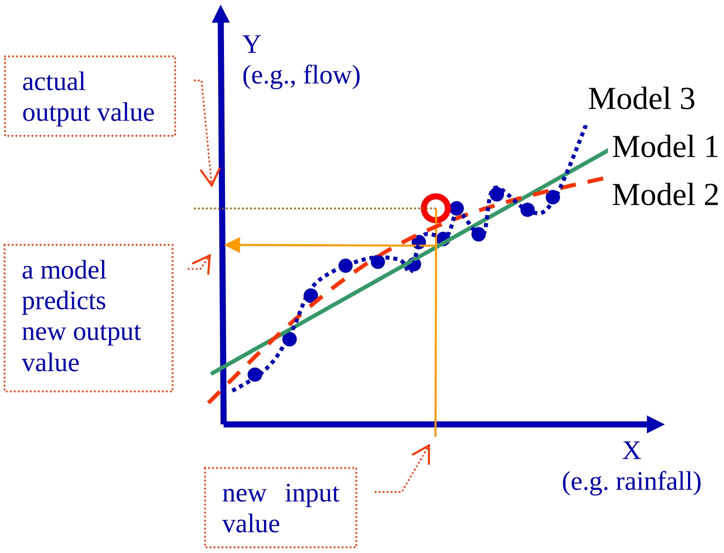

Figure 2.1. Examples of linear and non-linear data-driven models. What is the "best" model?

3.3.1. Artificial Neural Networks

An increasingly popular data driven modelling method is the artificial neural network (ANN). This paradigm is inspired by the way, in which the human brain processes information. This is done through a biological neural system that solves a specific task that it has been trained to do by handling a number of input signals, processing them, and outputting the result(s). The brain is composed of a very large number of neurons that are massively interconnected. Each neuron is a specialised cell that can propagate an electrochemical signal. The neuron has a branching input structure (the dendrites), a cell body and a branching output structure (the axon). The axons of one cell connect to the dendrites of another cell via a synapse. When a neuron is activated, it fires an electrochemical signal along the axon. This signal crosses the synapses to other neurons, which may in turn fire themselves. A neuron only fires if the total signal received by the cell body exceeds a certain level called the firing threshold. The strength of the signal received by a neuron depends critically on the nature of the synapse. Each synapse consists of a gap across which neurotransmitter chemicals are poised to transmit the signal. Learning consists essentially of altering the strength of the synaptic connections. From a system consisting of a large number of very simple processing units, the brain appears able to carry out extremely complex tasks. An ANN is a greatly simplified model of this perception of the human brain.

So an ANN consists of a large number of processing elements called neurons. Each neuron has an internal state called its activation or activity level. This is a function of all the inputs it has received. It then sends one signal at a time depending on its activation result to several other neurons. Typically an ANN developed for modelling the connectivity between a time series input and a corresponding time series output will consist of three layers of neurons: an input layer with a number of specific inputs, a hidden layer containing again a (different) number of neurons, and an output layer with one or more neurons.

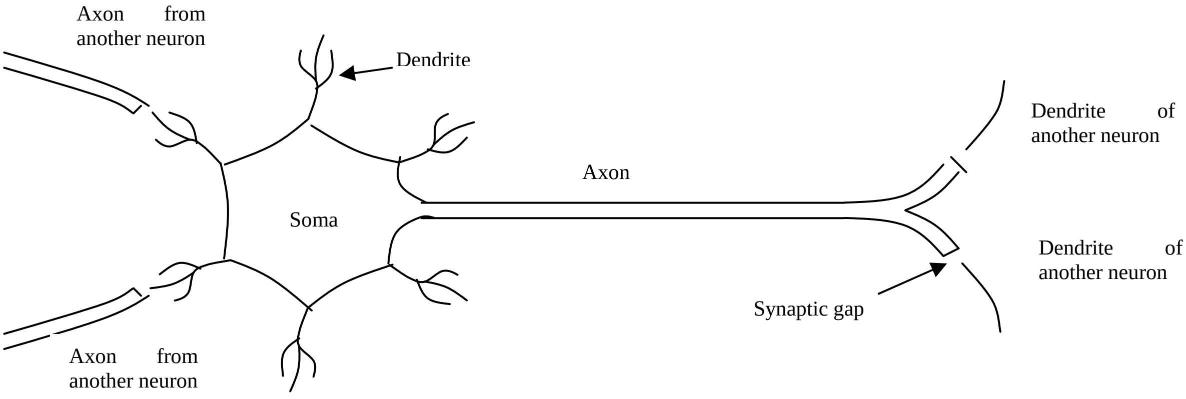

Fig 2.2 The basic features of a biological neuron

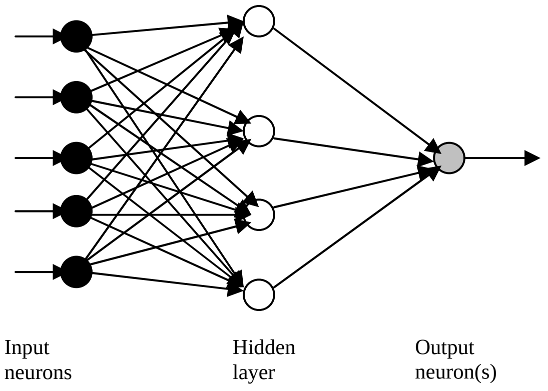

Fig 2.3 Structure of a simple artificial neural network

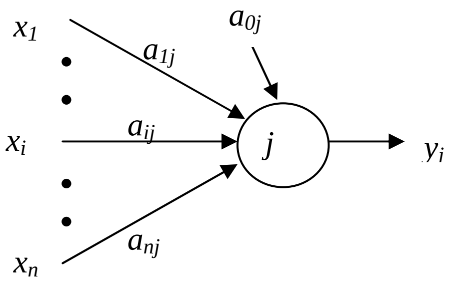

Fig 2.4 Schematic diagram of node *j*

The inputs form an input vector $X=(x_1,\ldots, x_i,\ldots, x_n)$, and the corresponding weights leading to the node form a weight vector $A=(a_1,\ldots, a_i,\ldots, a_n)$. The output of node *j* is obtained by computing the value of the function *f*:

$$y_j = f\left(\sum_{i=1}^{n} a_{ij}x_i + a_{0j}\right) \tag{3.1}$$

where the activation function *f* can be the sigmoid function:

$$f(u)=\frac{1}{1+\exp(-\lambda u)} \tag{3.2}$$

which is well behaved between 0 and 1. Note that $-a_{oj}$ is the threshold value such that the function *f* is zero for values less than zero. Other activation functions can also be chosen such as the threshold function (a) and the linear or saturation function (b). (c) is the sigmoid function.

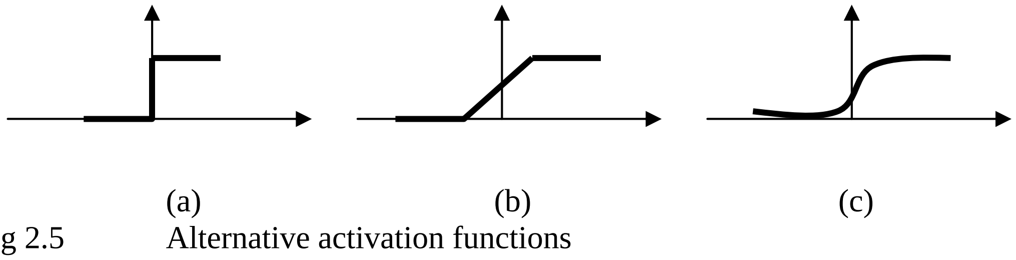

Fig 2.5 Alternative activation functions

The combination of the outputs from each of the *m* nodes in the hidden layer feeding the output node gives

$$z = f\left(\sum_{j=1}^{m} b_j y_j + b_0\right) \tag{3.3}$$

where *B=(b₁,…., bⱼ,…., bₘ)* is the weight vector on the outputs from the neurons in the hidden layer.

The ANN generates an output vector Z=$(z_1,\ldots, z_k,\ldots., z_K)$ (for *K* output data values) that is as close as possible to the target vector $T=(t_1,\ldots, t_k,\ldots., t_K)$ of observed values. This is the training process, also called the learning process, during which the weights a_{ij} and b_j are optimised. Normally this is done through minimising a predetermined error function of the form:

$$E = \frac{\sum_{k=1}^{K} (z_k - t_k)^2}{2K} \tag{3.4}$$

A range of methods have been developed whereby the weights can be determined. Such methods are usually variants of a gradient-based techniques (like Levenberg-Marquardt).

This type of ANN is called a multi-layer perceptron. It is a feed forward network in that information is fed through from the input to the output layer. It learns through back propagation from the errors of the prescribed output data. This is called supervised learning.

Another popular form of ANN is the radial basis function (RBF) network (Haykin 1999), which consists of

$$y_j = f\left(\sqrt{\sum_{i=1}^{n} (x_i - a_{ij})^2}\right) \tag{3.5}$$

where

$$f(u) = \exp(-\lambda u^2) \tag{3.6}$$

The output is given by

$$z = \frac{\sum_{j=1}^{m} b_{jk} y_j}{\sum_{j=1}^{m} y_j} \tag{3.7}$$

This form of ANN has a faster learning process than back propagation networks, though the accuracy of the solution is highly dependent on the data range and quality (Dibike, 1997).

A successful implementation of an ANN depends on a number of unknowns. For example, what input data should be used for a given output? How many hidden layers should there be? How many neurons (nodes) should be used in a hidden layer? Can the number of nodes be reduced to limit the time taken in training the network? How can ‘overfitting’ be avoided?

Generally, one hidden layer is sufficient to reproduce any non-linear function. Similarly, the number of nodes in the hidden layer is typically selected to be not more than twice the number of input nodes. Overfitting is a particular problem that can be identified by dividing the data set into three parts: training set, validation set and test set. The validation set is used during the training to check whether the error on this set starts to increase even when the error on the training set is decreasing.

This is a very brief introduction to ANNs for time series analysis on input and corresponding output data sets. ANNs can also be used very effectively for classification. These facilities can be very important in complementing the time series analysis facility in modelling.

3.3.2. Choice of relevant variables

The choice of variables is an important subject, and some studies suffer from the lack of relevant analysis. Apart from the expert judgement and visual inspection, there are formal methods that help in justifying this choice, and the reader is directed to the paper by Bowden et al. (2005) for an overview of these methods. Note that the input data may require pre-processing (e.g. filtering to remove noise), and this may increase the total number of possible inputs (and their combinations) to consider (see e.g., Solomatine and Xue 2004). In case of a high number of inputs, methods such as principal component analysis (PCA) may help.

Several main approaches to inputs selection can be distinguished. Of course, the initial set of candidate inputs is selected on the basis of expert judgement and *a priori* knowledge of the system being modelled. Further, stepwise selection of inputs can be employed. In forward selection we begin by finding the best single input, and in each subsequent step we add the input that improves the model performance most. Backward elimination starts with a set of all inputs, and sequentially removes the input that reduces performance the least. An optimal way would be to train many models on various sets of inputs and selecting the model with the lowest error; we may go for an exhaustive automated optimisation search across all possible combinations, or use a limited set of combinations. These methods need model runs for input selection. The so-called model-free approach is based either on statistical methods like cross-correlation, or information-theory based methods.

The information-theory based approach is in determining the information content between, say, the time series input data (eg rainfall) and the corresponding output time series (eg discharge). Our own experience using the Average Mutual Information (Abebe and Price, 2003; Solomatine and Dulal, 2003) shows that this simple and reliable method can help in selection of relevant input variables. The AMI is the measure of information in bits that can be learned about one data set in comparison with another known data set. It is based on Shannon's theory of entropy (Shannon, 1948). The AMI between two measurements, a_i and b_j, drawn from sets A and B respectively can be written as

$$I_{AB} = \sum_{a_i b_j} P_{AB}(a_i, b_j) \log_2 \left[\frac{P_{AB}(a_i, b_j)}{P_A(a_i) P_B(b_j)} \right] \tag{3.8}$$

where $P_{AB}(a_i, b_j)$ is the joint probability density for measurements A and B resulting in values a_i and b_j, $P_A(a_i)$ and $P_B(b_j)$ are the individual probability densities for the measurements of *A* and *B*. If the measurement of a value from *A* resulting in a_i is completely independent of the measurement of a value from *B* resulting in b_i, then the mutual information I_{AB} is zero. Compared to techniques such as linear correlation, the advantage of AMI is that it can be used to detect non-linear relationships as well as linear ones since it employs set theoretic principles and is not bound to any specific function. However, for discrete measurements actual values depend on technicalities such as the number of class intervals used to calculate the probability densities. The value of AMI in the case of rainfall-runoff modelling say, is that it clearly determines the lag time between the rainfall and the runoff, thus enabling a proper choice of input data for the input layer of the ANN.

3.3.3. Other machine learning modelling techniques

ANNs are one technique that can be used for data-driven modelling. There is a range of other techniques that have become popular in recent years, including:

Nearest neighbour	Based on the assumption that nearby points are more likely to be given the same classification than distant ones. The learning set $\{v,k\}$ is taken as a collection of known cases $[v,k]$ and a search is made for a given pattern v to be recognised for the best match among the precedents v_j. The class label k of the nearest neighbour $v_{nearest}$ is forwarded as a result of the classification
Fuzzy rule based systems	Consists of input-output membership functions, fuzzy rules and an inference engine. Crisp inputs are fuzzified, the fuzzy rules are applied and the inference engine is used to recover a crisp output; see Bardossy and Duckstein (1995)
Genetic programming	A functional form is allowed to evolve according to prescribed evolutionary rules such that the resulting function most closely generates the output set given the input set.
Decision/Model trees	Instances are classified by sorting them up the 'tree' from the 'root' to some 'leaf' node that provides a classification of the instance. Each node in a tree specifies a test of some attribute of the instance, and each branch descending from a node corresponds to on of the possible values for this attribute. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the branch corresponding to the value of the attribute. This process is repeated for the sub-tree root based at the new node (Witten an Frank 2000).
Support vector machines	The approximating function is chosen on how well it fits the verification set (minimising the structural risk) as well as the training set (minimising the empirical risk) using statistical learning theory; see Vapnik (1998)

3.3.4. Chaos

Besides the connectionist techniques above there are other techniques that consider the underlying structure of a time series. The idea is that an apparently random time series may have an underlying deterministic structure that can be determined in an appropriate state or phase space. Data vectors are constructed from the time series $\{s(n)\}$ such as

$$y(n) = [s(n), s(n+T), s(n+2T),...., s(n+T(k-1))] \tag{3.9}$$

where $T\tau_s$ is the time lag k is the dimension of the space. T is determined from the first minimum of the AMI for the series $\{s(n)\}$ and $\{s(n+T)\}$.

d is the lowest (integer) dimension which unfolds the possible attractor (underlying structure) for the time series. This is done using nearest neighbour techniques and by progressively increasing the dimension d. Finally the stability of the system is determined by the Lyapunov exponents. These are determined by studying the separation of two points, a_0 and b_0, on two trajectories after some number n iterations. The global Lyapunov exponent is formulated as

$$\lambda_a = \lim_{n\to\infty}\frac{1}{n}\lim_{|a_0-b_0|\to\infty}\log_2\left|\frac{a_0-b_0}{a_n-b_n}\right| \tag{3.10}$$

If one or more of the Lyapunov exponents is positive, which indicates that the neighbouring points are diverging, then this implies the existence of chaos. A zero exponent means that the neighbouring points will remain at the same distance from each other and the system can be modelled by a set of differential equations. Negative Lyapunov exponents indicate that the neighbouring points are converging and the system is dissipative. The largest Lyapunov exponents can also be used to establish a window of predicatability, T$_p$, of a system in time:

$$T_p = \frac{\tau_s}{\lambda} \tag{5.11}$$

Chaos is used extensively now to forecast ahead, given the time series alone. For a good introduction to chaos theory; see Ararbanel (1996). See also Solomatine et al (2000) for an example of the application of chaos theory to the prediction of surge water level in the North Sea close to Hoek van Holland.

3.3.5. Applications of data driven modelling

An obvious use of ANNs is in modelling the rainfall-runoff process, or in routing flows from one point to another along a river. Other uses include prediction of currents in the sea from meteorological conditions, the interpretation of cone penetration tests, the estimation of sedimentation in dredged channels, forecasts of demand in water distribution networks, prediction of intermittent overflows in drainage networks, and so on. Such modelling can be done without the support of any physically based modelling. However, there is increasing use of ANNs (and other data driven modelling techniques) to complement physically based models. This is done by arranging for an ANN to model the error of a physically based model, that is, the difference or the ratio between the observed and predicted values of the output. This is a particularly simple form of data assimilation.

For example applications of various data modelling methods see the following:

- modelling rainfall-runoff processes using ANNs: Hsu et al. (1995); Minns and Hall (1996); Dawson and Wilby (1998); Dibike et al. (1999); Abrahart and See (2000); a collection of papers edited by Govindaraju and Ramachandra Rao (2000); Hu et al. (2007); Abrahart and See (2007).
- modelling river stage-discharge relationships with ANNs (Sudheer and Jain, 2003; Bhattacharya and Solomatine, 2005);

- Moradkhani et al. (2004) used RBF ANNs for predicting hourly streamflow hydrograph for the daily flow for a river in USA as a case study, and demonstrated their accuracy if compared to other numerical prediction models. In this study RBF was combined with the self organising feature maps used to identify the clusters of data;
- using a fuzzy rule-based system for the prediction of precipitation events (Abebe, Solomatine and Venneker, 1999);
- using fuzzy logic in the analysis of groundwater model uncertainty (Abebe, Guinot and Solomatine, 2000);
- using ANNs and fuzzy rule-based system to build an intelligent controller for water management in polder areas (Lobbrecht and Solomatine, 1999)
- modeling a channel network using ANN (Price et al. 1998);
- surge water level prediction in the problem of ship guidance using ANN and chaos theory (Solomatine et al., 2000);
- using M5 model trees to predict discharge in a river (Solomatine and Dulal, 2003);
- using support vector machines (SVM) in prediction of water flows for flood management (Dibike, Velickov, Solomatine & Abbott, 2001).

One of the applications of data-driven models is to replicate physically-based models. A number of such studies have been reported:

- replicating the behaviour of hydrodynamic and hydrological models of the Apure river basin (Venezuela), where ANNs are used in model-based optimal control of a reservoir (Solomatine and Torres, 1996);
- building an assisting surrogate model in calibration of a rainfall-runoff model (Khu et al., 2004);
- emulating by an MLP network and replacing the hydrologic simulation component of multiobjective decision support model for watershed management (Muleta and Nicklow, 2004). In this study an alternative to the back propagation training was used – a direct search method (evolutionary algorithm) that reportedly allowed for avoiding local minima during training.

There is little doubt that the variety of data driven modelling techniques offers considerable scope for analysing time series data and deducing appropriate ‘black-box’ models. Such models are normally considerably easier to set up than physically based models. They are particularly powerful in situations where it is difficult to determine the physical processes, or when accurate forecasts are needed based on what is known of the system up to time now. There are however, obvious limitations in that these modelling techniques rely on there being no change in the structural (physical) domain that can change the assumed functional relationship between the input and output data sets. These could include modifications to a catchment land use, river training works, or alterations to control structures. Furthermore, ANNs, for example, are well known to have difficulty in extrapolating outside the range of the training data. There are ways of reducing this difficulty, but users should be aware of the problem.

On the latest trends and applications of DDM see, for example, Solomatine and Ostfeld (2008).

3.4. Uncertainty in modelling

A huge issue in modelling is the confidence that the decision-maker can put in the results from an (instantiated) model. Every model is, by definition, an approximation to reality. The decision-maker needs to know therefore how safe and reliable the results from a model are in affecting the decision made. Bearing in mind that measurements, particularly of flows, can have an uncertainty of 20% or more, then the models that are built (calibrated, confirmed) using such data will have uncertainties of at least a similar order. As seen above, considerable effort is now put into trying to reduce the model error by complementing a physically based model with a data-driven model of the 'errors' of the physically based model. However, the fact is that decision-makers still have to live with uncertainty. This means that they have to consider such uncertainties when making decisions. The whole area therefore of decision-making in civil engineering (risk analysis etc) needs ongoing attention; see Maskey (2001).

An excellent paper by Pappenberger and Beven (2006) presents some reasons why uncertainty estimation is still rarely used. They state: "a significant part of the community is still reluctant to embrace the estimation of uncertainty in hydrological and hydraulic modelling. … we summarize and explore seven common arguments: uncertainty analysis is not necessary given physically realistic models; uncertainty analysis cannot be used in hydrological and hydraulic hypothesis testing; uncertainty (probability) distributions cannot be understood by policy makers and the public; uncertainty analysis cannot be incorporated into the decision-making process; uncertainty analysis is too subjective; uncertainty analysis is too difficult to perform; uncertainty does not really matter in making the final decision".

There are two main types of uncertainty:

- epistemic uncertainty that is due to imperfection of our knowledge – it can be reduced by more research or acquiring more data;
- variability uncertainty which is due to inherent variability (randomness) in behaviour of natural or human systems.

The main sources of uncertainty in modelling have three main sources:

- input data uncertainty
- model uncertainty consisting of structural uncertainty, and parametric uncertainty;
- output data uncertainty, which influences the calibration procedure and leads to uncertainty in the model output.

In relation to water-related issues, there were many studies done where the model uncertainty was estimated. Several main approaches can be identified.

The *first* approach is to forecast the model outputs probabilistically and it is often used in hydrological modeling (like Bayesian Forecasting System of Krzysztofowicz, 2000). The *second* approach is to estimate uncertainty by analyzing the statistical properties of the model errors that occurred in reproducing the observed historical data. This approach has been widely used in statistical (Wonnacott and Wonnacott, 1996) and machine learning communities (Nix and Weigend, 1994). For time series forecasting; uncertainty is estimated in terms of confidence interval or prediction interval. A method that can be also attributed to this group (meta-Gaussian model) was developed by Montanari and Brath (2004). The *third* approach is to use sampling based techniques, generally referred to as a Monte Carlo method. This method is used, for example, in a *generalized likelihood uncertainty estimator*, GLUE (Beven and Binely, 1992) that is popular in hydrologic modelling. Monte Carlo methods are typically used to estimate the models' output uncertainty due to the uncertainty of model parameters (parametric uncertainty). The *fourth* approach is based on fuzzy theory based method (Maskey et al., 2004). This provides a non-probabilistic approach for modelling the kind of uncertainty associated with vagueness and imprecision.

The first and the third approaches mentioned above require the prior distributions of the uncertain input parameters or data to be propagated through the model to the outputs. In contrast, the second approach requires certain assumptions about the data and the errors, and obviously the relevancy and accuracy of such approach depends on the validity of these assumptions. The last approach requires knowledge of the membership function of the quantity subject to the uncertainty.

Recently Shrestha and Solomatine (2006) presented an approach termed UNcertainty Estimation based on local model Errors (UNEEC). It is based on an idea to build local data-driven models predicting the properties of the error distribution for particular (hydrometeorological) situations. Further, it uses a scheme based on fuzzy clustering to aggregate the outputs of these models and to train an overall uncertainty prediction model. This is a distribution free, non parametric method to model the propagation of integral uncertainty through the models, and it was tested in forecasting river flows in a flood context.

This discussion leads us to reflect on decision support environments.

3.5. Integrated water modelling

However, why stop at integration of models in the physical sphere alone? In fact, a hierarchy of modelling is done in most water engineering organisations. For example, an urban water supply or wastewater disposal organisation has to model not only its water networks but also to model (in comparatively simple terms) the economics of making choices between one scenario or option and another. This points to a hierarchy of models in a similar manner to the integrated modelling of a river basin referred to above.

One way of exploring the concept of such a hierarchy is to adopt Jonker's (2001) classification of the physical, biological and social spheres. He postulates a three-dimensional knowledge space of systems (geo-, bio- and socio-spheres), processes (plan, design, construct, manage) and tools (data sets, technologies, models, courseware, people, etc). This is intended to be all encompassing. As such it runs the danger that it becomes unwieldy due to its complexity such that we cannot grasp its implications. Nevertheless, the hierarchy of models needs to work with this knowledge space in a way that we can retain some form of control on its complexity.

One way of doing this is to address the hierarchy of the models in terms of model complexity. For example, if we were considering a river basin then at the top level the (systems) model would integrate the inter-related knowledge domains on the systems axis, as well as specific processes and associated tools. Such a model would follow the parsimony principle, and at the same time be capable of answering the key questions/objectives that are raised; that is, the model should be ‘fit for purpose’. It would probably be based on cybernetics principles. In order to make the model tractable it would also consist of a number of sub-models. This is way of saying that at the next level down there will be models defined separately for each system domain. They will necessarily introduce an order of magnitude increase in complexity, but they should still be tractable. There will in addition be models below these intermediate models that address yet finer details of the problem. In the river basin, for example, the second layer would include models of the economics of particular branches of industry or city development, integrated water resources, or recreation development.

Examples of these sorts of models exist at IHE for ‘role play’ in water resources management. At the bottom layer (and we can of course envisage other intermediate layers) there will be detailed models such as MIKE-SHE (from DHI) for integrated surface and groundwater modelling, and InfoWorks (from Wallingford Software) for integrated water supply/distribution, wastewater and storm water collection, wastewater treatment, and pollution impact on receiving streams. The models at each layer would be visualised in and make use of a suitable GIS. What is going to be extremely important is that there is consistency between the models at each layer, for example, between the MIKE-SHE and InfoWorks models and the integrated water resources model above them. There exists the possibility of training a cruder, higher-level model on a more detailed, lower level model to achieve consistency. This points to a ‘bottom-up’ model development that works from greater to lesser complexity. This is not the usual way in which computational hydraulics has developed. In general there has been a striving for the models to become more detailed with the assumption that as detail is achieved so the latest model will include everything that a less detailed model will encompass (and more besides). Whereas this is undoubtedly the case, in going to higher levels there is much less interest in the details: attention is focussed more on global or boundary conditions. This is where ‘conceptual’ or data-driven models trained on (to replicate) the more detailed models come into their own.

Apart from conceptual integration, the models need to be integrated in terms of software and hardware, and the development of tools that make such integration easier and more effective is very much needed. Recently various research and development groups are reporting interesting approaches to such integration using flexible straightforward protocols like file exchange and XML descriptors used in Delft-FEWS system (Werner, 2008), web services (Donchyts, 2007; Horak et al., 2007) and object-oriented framework OpenMI , which allows a tight connection between two computational processes (Fortune, 2008). The latter approach is a result of a joint effort between several major competing suppliers of hydraulic and hydrologic modelling software (DHI, Delft Hydraulics and Wallingford Software), see www.OpenMI.org.

3.6. Optimization

Optimization can be defined is a process of finding such values of the variables characterizing some system that would bring a particular function to a minimum (or maximum). This would mean that the system is in a certain sense “optimal”. The variables are called the decision variables, and the function – an objective function. Examples of water-related issues that require solving an optimization problem follow:

1) Find such release of the reservoir(s) with a hydropower dam that would lead to the maximum yearly production of the electrical power, and satisfy the water consumers downstream (Labadie, 2002).

2) Identify and present to a decision maker several rehabilitation plans for a drainage (or combined sewer) system that would a) lead to smaller (ideally, minimum) flood damages in case of heavy rainfalls, and b) be within budget constraints (the lower the costs the better) (Barreto et al., 2010). This is an example of a multi-objective optimization problem.

3) Find an optimal groundwater remediation strategy leading to a smallest possible concentration of a pollutant (or concentration below a certain limit) in a given time (smaller the better) (see Maskey et al., 2002).

3) For a hydrologic model, find the values of the parameters (which cannot be measured) that would lead to the smallest possible error of this model (Solomatine et al., 1999).

4) Find a combination of models, knowledge sources and human experts that would solve a particular water management problem in an optimal way.

Optimization techniques and tools complement the arsenal of the modelling tools, and play an important role in Hydroinformatics.

3.7. Decision support environments for local and distributed decision making

Hydroinformatics incorporates computational hydraulics, but as has been stressed by many authors following Abbott it is more than simply modelling. This is because a modelling software product is primarily a tool. Like all tools it is created to be ‘fit for purpose’ within specific contexts. There are safe and reliable ways of applying a software tool just as there are unsafe and unreliable ways. The (engineering) user is therefore a critical component in the application of the software. Too often the interaction of the engineer with the modelling product is viewed as being ‘outside the picture’, and therefore not part of the application. This view can no longer be sustained. He (or she) has to make many decisions involving personal judgement based on experience. In other words, what the user does in implementing the modelling software product is as important as the final instantiated model.

And the user does not work alone. He is dependent on the situation in which he is working, and on his relationship to clients, stakeholders and personnel with different contributing functions within the organisation in which he works. The flow of the right information at the right time and in the right place becomes important for the success of the project involving the software. This is illustrated in the case of sewerage rehabilitation projects. There are, for example, many thousands of sewerage systems in Europe that at some stage will need rehabilitation for one reason or another. It follows that there can be considerable cost savings by transferring and disseminating as widely as possible knowledge on best practices in sewerage rehabilitation. The traditional way of doing this in engineering terms is to detail specific procedures that highlight the ‘lessons learned’ at each phase. Usually there is a hierarchy of phases or tasks that are implemented in particular sequences. Each task has its own attributes, including explicit information or knowledge on the execution of the task acquired from experts. An example of such a procedure or best management practice is the Sewerage Rehabilitation Manual (WRc).

It is recognised that such engineering procedures are most effective where there are many repetitions of a similar process. This is more likely to be the case with urban water-based systems, which have very well defined characteristics (even if there can be significant cultural differences between national practices). The application to natural systems, such as rivers, or coastal waters is much more dependent on the peculiarities of each situation, and therefore it is more difficult to define a suitable procedure for each type of goal or objective. Nevertheless, there is considerable scope for compiling 'lessons learned' on an encyclopaedic basis in support of applications. This has yet to be done effectively, although some expert systems have been developed for a few specific cases. In general, these expert systems have proved to be too simplistic or unwieldy, and a more open, unstructured access to information and knowledge is preferred.

3.8. Development issues in hydroinformatics

Hydroinformatics is a comparatively young subject that has attracted vigorous attention from a number of researchers world-wide. For example, new departments of hydroinformatics are being set up in different universities (such as Technical University of Delft, Newcastle, Bristol, Singapore, Iowa, among others). The diversity of the research specialisations of these departments, even within hydroinformatics, is growing. This is partly adding to a certain initial confusion surrounding the subject, which is still defined in alternative ways by different people, and this offers interesting possibilities to enterprising developers.

There are a number of key development areas in hydroinformatics. For example, there is still the need to develop more efficient and accurate difference schemes and solvers for computational hydraulics, to improve the data mining and knowledge discovery techniques, and to explore new and more versatile data modelling methods. There is a clear need to develop procedures making it possible to integrate physically-based and data-driven methods, thus building hybrid models.

Experience in developing and applying computational hydraulic models is now very extensive. Nevertheless, researchers and practising engineers have gained such experience by focussing on specific types of application. So, for example, drainage and sewerage engineers have become proficient in modelling wastewater and storm drainage collection systems. Other public health engineers have a corresponding but very different experience of modelling processes in treatment works. Yet other, river engineers have developed techniques and experience for modelling pollutant impact on the water quality of receiving streams. The modelling experience in each case is similar but sufficiently different that when it comes to integrating the different models there are problems of interfacing. This has been tackled successfully in this particular area, and others are working to bring together water, structure and groundwater modelling. The benefits for dealing holistically with complex situations, such as high-speed rail tunnels, off-shore structures, and other infrastructure projects are considerable.

Hydroinformatics integrates various data sources (remote sensing, gorund measurements etc.), various types of models, and management and decision support processes, and therefore serves the various stakeholders. Further development of software tools making this whole process effective and efficient is of great importance.

4. REFERENCES

Abebe, A J (2003) Forecasting the accuracy of numerical surge forecasts along the Dutch coast. Final report, Nautilus 10.10 project, for RIKZ, Rikswaterstaat, Den Haag, The Netherlands

Abebe, A J and Price, R K (2004) Information theory and neural networks for managing model uncertainty in flood routing. *ASCE J. of Computing in Civil Engineering*, 18(4), 373-380.

Abebe, A J and Price, R K (2005) Decision support system for flood warning in urban areas. J of Hydroinformatics, 7(1), 3-15.

Abebe, A J Guinot, V and Solomatine, D P (2000) Fuzzy alpha-cut vs. Monte Carlo techniques in assessing uncertainty in model parameters, Proc. 4th International Conference on Hydroinformatics, Iowa City, USA, July 2000.

Abebe, A J, Solomatine, D P and Venneker, R (2000) Application of adaptive fuzzy rule-based models for reconstruction of missing precipitation events, *Hydrological Sciences J.*, 45(3), 425-436.

Abbott, M B (1991) Hydroinformatics: Information technology and the aquatic environment. Avebury Technical, Aldershot, UK, 145 p.

Abbott, M B (1992) The theory of the hydrological model, or: The struggle for the soul of hydrology, in Advances in Theoretical Hydrology: a tribute to Jim Dooge, ed, O’Kane P, pp237-254, Elsevier, Amsterdam

Abbott, M A (1993) The electronic encapsulation of knowledge in hydraulics, hydrology and water resources, Adv. Wat. Resources, 16, 22-39.

Abbott, M B (1994) Hydroinformatics: a Copernican revolution in hydraulics, J of Hydr. Res., 32, Extra Issue, pp3-14

Abbott, M B (1996) The social dimensions of hydroinformatics, Hydroinformatics ’96, Muller(ed), Balkema, Rotterdam

Abbott, M A and Jonoski, A (2001) The democratisation of decision-making processes in the water sector II, J of Hydroinformatics, 3(1), 35-48

Abarbanel, H D I (1996) Analysis of observed chaotic data, Springer-Verlag New York Inc., New York

Abrahart, R J and See, L. (2000) Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecast in two contrasting catchments. Hydrological processes, 14, 2157-2172.

Abrahart, R.J., and See, L. M. (2007). Neural network modelling of non-linear hydrological relationships. Hydrol. Earth Syst. Sci., 11, 1563–1579.

Bardossy, A. and Duckstein, L. (1995) Fuzzy Rule-Based Modeling with Applications to Geophysical, Biological and Engineering Systems. CRC Press Inc, Boca Raton, Florida, USA

W. Barreto, Z. Vojinovic, R.K. Price, D.P. Solomatine. A Multi-objective Evolutionary Approach to Rehabilitation of Urban Drainage Systems. *ASCE Journal of Water Resources Planning and Management*, 2010 (in review).

Beven, K. J. and Binely, J. (1992). The future of distributed models: model calibration and uncertainty prediction. Hydrological Processes, 6, 279-298.

Minns, A.W. and Hall, M.J., (1996). Artificial Neural Network as Rainfall-Runoff Model. Hydrological science Journal, 41(3), 399-417.

Bhattacharya, B and Solomatine, D P (2000) Application of artificial neural network in stage-discharge relationship, Proc. 4th International Conference on Hydroinformatics, Iowa City, USA, July 2000.

Dee, D P (1993) A Framework for the validation of generic computational models. Tech Report X109, Delft Hydraulics, The Netherlands.

Dibike, Y, Solomatine, D P and Abbott, M B (1991) On the encapsulation of numerical-hydraulic models in artificial neural network, Journal of Hydraulic Research, No. 2, pp147-161

Dibike, Y B, Solomatine, D P, Velickov, S, and Abbott, M B (2001) Model induction with support vector machines: Introduction and applications. ASCE J of Computing in Civil Engineering, July 2001, Vol 15, No 3, p 208-216

Donchyts, G., Treebushny, D., Primachenko, A., Shlyahtun, N., and Zheleznyak, M. (2007). The architecture and prototype implementation of the Model Environment system. *Hydrol. Earth Syst. Sci. Discuss*., 4, 75–89.

Falconer R., Lin B., and Harpin R. (2005). Environmental modelling in river basin management. *J. of River Basin Management*, 3(3), 169-184.

Fortune, D., Gijsbers, P., Gregersen, J., Moore, R. (2008). OpenMI – Real Progress Towards Integrated Modelling. Werner, M. (2008). Open model integration in flood forecasting. In: *Hydroinformatics in practice: computational intelligence and technological developments in water applications* (Abrahart, B., See L.M., Solomatine, D.P., eds.). Springer (in press).

Govindaraju, R S and Ramachandra Rao, A (eds.). (2001) Artificial neural networks in hydrology. Kluwer, Dordrecht

Hassibi, B. & Stork, D. G. (1993), Second order derivatives for network pruning: Optimal Brain Surgeon, in C. L. Giles, S. J. Hanson & J. D. Cowan, eds, Advances in Neural Information Processing Systems, Vol. 5, Morgan Kaufmann, San Mateo, CA

Haykin, S. (1999). Neural Networks: a comprehensive foundation. Prentice Hall, Upper saddle River, New Jersey.

J. Horak, A. Orlik, and J. Stromsky (2007). Web services for distributed and interoperable hydro-information systems, *Hydrol. Earth Syst. Sci. Discuss*., 4, 1879–1891.

Hsu, K L, Gupta, H V and Sorooshian, S (1995) Artificial neural network modeling of the rainfall-runoff process. Wat. Res. Res., Vol 31(10), pp2517-2530

Hu, T., Wu, F. and Zhang, X. (2007). Rainfall–runoff modeling using principal component analysis and neural network. Nordic Hydrology, 38(3), 235-248.

Jonkers, L (2001) Personal communication.

Jonoski, A and Abbott, M B (1998) Network distributed support systems as multi-agent constructs, Proc. In. Conf on Hydroinformatics 1998, Balkema, Rotterdam

Jonoski, A., (2000), AquaVoice: A prototype for Internet distributed decision support system, Proc. Int. Conference on Hydroinformatics 2000, Cedar Rapids, IA, USA.

Jonoski, A (2002) Hydroinformatics as Sociotechnology: Promoting individual stakeholder participation by using network distributed decision support systems. PhD thesis, Swets & Zeitlinger BV, Lisse

Khu, S.-T., Savic D., Liu, Y., and Madsen, H. (2004). A fast evolutionary-based meta-modelling approach for the calibration of a rainfall-runoff model. Trans.2nd

Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs: Manno, Switzerland.

Krzysztofowicz, R. (2000). The case for probabilistic forecasting in hydrology. Journal of Hydrology, 249, 2-9.

J.W. Labadie (2004). Optimal Operation of Multireservoir Systems: State-of-the-Art Review. ASCE Journal of Water Resources Planning and Management, 130(2), 93-111.

Lobbrecht, A and Solomatine, D P (1999) Control of water levels in polder areas using neural networks and fuzzy adaptive systems. Water Industry Systems: Modelling and optimisation applications: Vol 1, eds D Savic and G Walters, Research Studies Press Ltd, Baldock, UK pp509-518

Maskey, S (2001) Uncertainty analysis in flood forecasting and flood warning systems using expert judgement and fuzzy set theory. Safety and Reliability, Ed E Zio, M Demichela, M Piccinini, Proceedings of the ESREL 2001 Conference, Turin, Italy, Sept 2001, p 1787-1794

Maskey, S., Jonoski, A., Solomatine, D.P. (2002). Groundwater remediation strategy using global optimization algorithms. ASCE Journal of Water Resources Planning and Management, 128 (6), 431-440.

Maskey, S., Guinot, V., and Price, R.K. (2004). Treatment of precipitation uncertainty in rainfall-runoff modelling, Advances in Water Resources, 27, 889-898.

Minns, A W and Hall, M J (1996). Artificial Neural Network as Rainfall-Runoff Model. Hydrological Science Journal, 41(3), 399-417.

Montanari, A., and A. Brath (2004). A stochastic approach for assessing the uncertainty of rainfall-runoff simulations. *Water Resources Research*, 40, doi:10.1029/2003WR002540.

Moradkhani, H., Hsu, K. L., Gupta, H. V., Sorooshian, S. (2004). Improved streamflow forecasting using self-organizing radial basis function artificial neural networks. J. of Hydrology, 295 (1), 246-262.

Nix, D. and Weigend, A. (1994). Estimating the mean and variance of the target probability distribution. Proc. Intern. Joint Conf.on Neural Networks, IEEE, 55-60.

Pappenberger, F. and Beven, K. J. (2006). Ignorance is bliss: or seven reasons not to use uncertainty analysis. Water Resources Research, 42(5), W05302, doi:10.1029/2005WR004820.

Price, R K, Ahmad, K and Holz, P (1996) Hydroinformatics Concepts. In Hydroinformatics Tools for planning, design and operation and rehabilitation of sewer systems, NATO ASI Series, 2. Environment - Vol 44.

Price, R K (2000) Future perspectives of ICT in urban water management. Paper presented to the Nordic Conference on Urban Water Management, Lillehammer 22-24 November 2000

Price, R K, (2000) Hydroinformatics for river flood management J.Marsalek et al (eds), Flood issues in contemporary water management, pp237-250, Kluwer, Amsterdam

Price, R K, Samedov, J and Solomatine, D P (1998) Network modeling using artificial neural networks, Proc. International Conference on Hydroinformatics, Balkema, Rotterdam.

Shannon, C E (1948) A mathematical theory of communication. Bell System Technical Journal vol. 27, 379-423 and 623-656, July and October.

Shrestha, D.L. and Solomatine, D.P. Machine learning approaches for estimation of prediction interval for the model output. *Neural Networks*, 2006, 19(2), 225-235.

Solomatine, D P and Torres, L A (1996) Neural network approximation of a hydrodynamic model in optimizing reservoir operation. Proc. 2nd Intern. Conference on Hydroinformatics, Zurich, 201-206 September 9-13

Solomatine, D.P., Rojas, C., Velickov, S. and Wust, H. (2000). Chaos theory in predicting surge water levels in the North Sea. Proc. 4th Int. Conference on Hydroinformatics, Cedar-Rapids.

Solomatine, D.P., Dibike Y., Kukuric N. Automatic calibration of groundwater models using global optimization techniques. *Hydrological Sciences Journal*, 44(6), 1999, 879-894.

Solomatine, D.P. and Dulal, K.N. (2003). Model tree as an alternative to neural network in rainfall-runoff modelling. *Hydrological Sciences J*.: **48** (3), 399-411.

Solomatine, D. P. and Xue, Y. (2004). M5 model trees and neural networks: application to flood forecasting in the upper reach of the Huai River in China. *ASCE J. Hydrologic Engineering,* 9 (6), 491-501.

Solomatine, D.P. and Ostfeld, A. (2008). Data-driven modelling: some past experiences and new approaches. J. Hydroinformatics, 10(1).

Thorkilsen, M and Dynesen, C (2001) An owner's view of hydroinformatics: Its role in realising the bridge and tunnel connection between Denmark and Sweden. J. of Hydroinformatics, IWA Publishing, 3.2

Vapnik, V N (1998) Statistical Learning Theory, Wiley, New York

Werner, M. (2008). Open model integration in flood forecasting. In: *Hydroinformatics in practice: computational intelligence and technological developments in water applications* (Abrahart, B., See L.M., Solomatine, D.P., eds.). Springer (in press).

Witten, I.H. and Frank, E. (2000). *Data mining*. Morgan Kaufmann: San Francisco.

Wonnacott, T. H., & Wonnacott, R. J. (1996). *Introductory statistics*. New York: Wiley.

5. SOME USEFUL WEB SITES

Additional material can be found at the following web site

Universal locator of knowledge

www.google.com (typically finds everything you are searching for)

Commercial software suppliers

www.wldelft.nl
www.dhisoftware.com
www.wallingfordsoftware.co.uk
www.haestad.com
www.bossintl.com

Free hydraulic and hydrologic software

www.hec.usace.army.mil (HEC-RAS, HEC-HMS, etc.)
www.epa.gov/nrmrl/wswrd/dw/epanet.html (EPANET)
www.epa.gov/ednnrmrl/models/swmm/index.htm (SWMM)

Journals

www.iwaponline.com/jh/toc.htm (Journal of Hydroinformatics)
www.hydrology-and-earth-system-sciences.net (Hydrology and Earth Systems Sciences)
www.hydrologicalprocesses.com (Hydrological Processes)
www.elsevier.com/locate/jhydrol (Journal of Hydrology)
www.journalhydraulicresearch.com (Journal of Hydraulic Research)
www.agu.org/journals/wr (Water Resources Research)

Other sites

http://www.sahra.arizona.edu/software/index_main.html (SAHRA – Software archive, University of Arizona)
www.datamining.ihe.nl (Introduction to the use of data-driven models in civil engineering, UNESCO-IHE)
www.kdnuggets.com (Guide to data mining and machine learning software)