Machine Learning in Water Systems

Dragan Savić (editor)

AISB Convention 2013 • University of Exeter • 3rd–5th April, 2013

Foreword from the Convention Chairs

This volume forms the proceedings of one of eight co-located symposia held at the AISB Convention 2013 that took place 3rd-5th April 2013 at the University of Exeter, UK. The convention consisted of these symposia together in four parallel tracks with five plenary talks; all papers other than the plenaries were given as talks within the symposia. This symposium-based format, which has been the standard for AISB conventions for many years, encourages collaboration and discussion among a wide variety of disciplines. Although each symposium is self contained, the convention as a whole represents a diverse array of topics from philosophy, psychology, computer science and cognitive science under the common umbrella of artificial intelligence and the simulation of behaviour.

We would like to thank the symposium organisers and their programme committees for their hard work in publicising their symposium, attracting and reviewing submissions and compiling this volume. Without these interesting, high quality symposia the convention would not be possible.

Dr Ed Keedwell & Prof. Richard Everson
AISB 2013 Convention Chairs

Published by
The Society for the Study of Artificial Intelligence and the Simulation of Behaviour
http://www.aisb.org.uk

ISBN: 978-1-908187-33-8

CONTENTS

PREFACE

The emergence of water, alongside energy and food, as one of the three major, interlinked, global environmental security issues provides abundant challenges and opportunities for the application of Machine Learning to such problems as optimisation of water distribution and drainage networks' design and operation, modelling and prediction of fluvial, pluvial, urban and coastal flooding, sediment transport and water quality issues. Advances in GIS, remote sensing and weather forecasting techniques mean that environmental data is becoming increasingly abundant at the same time as demands for solutions and tools to work on these problems become more urgent.

Numerical models have been applied widely to improve the understanding and operational management of natural and manmade water systems. Traditionally, so-called "physically-based" models have been applied for such purposes. However, such models are often computationally demanding, and frequently require significant data to constrain model structures and parameters. Data-Driven Models (DDMs) based on Machine Learning techniques - which seek to provide a mapping between the inputs and outputs of a given system, with little prior process knowledge – have emerged as an attractive option for prediction and classification in water systems. The principal benefit of such DDMs is their fast execution time, which allows many more model evaluations for a fixed computational budget. Such models have been applied widely to address a variety of problems within water systems modelling, including: system simulation (e.g. rainfall-runoff modelling/rating curve prediction) when trained on measured data, and also when employed as metamodels and trained to emulate models with a stronger physical (or process) basis; to improve the speed of the optimisation procedure by acting as a surrogate model to the full fitness evaluation; to correct systematic errors in physically-based models during real-time forecasting; to provide uncertainty bound predictions during model forecasting when trained on uncertainty bounds derived from offline calibration; and in classification (for example of predicted severity of a hazard or exceedances of regulatory limits).

Despite their potential benefit, successful application of machine learning techniques is not straightforward. A variety of machine learning techniques, optimisation methods and evaluation procedures have been applied in the research literature. It is not always clear which methods will perform best in different settings, and how choices made will influence performance. As an example, different machine learning techniques might perform best depending on how their performance is evaluated within a given operational setting. Furthermore, although such methods are technically "black-box" models, system understanding may be required to choose the best input variables, and tailor the approach to the operational setting in question. With a view towards sharing the interdisciplinary knowledge required to make appropriate methodological decisions, papers are invited that explore issues of model design and application, and in particular, papers that compare different approaches for machine learning application.

A comparative study of artificial neural network architectures for time series prediction of water distribution system flow data

S. R. Mounce[1]

Abstract. Many water utility companies are beginning to amass large volumes of data by means of remote sensing of flow, pressure and other variables. For district meter area monitoring there has been increasing interest in using this sensor data for abnormality detection, such as the real-time detection of bursts. Research pilots have explored systems for generating 'smart alarms' and a key requirement is usually a prediction of future time series values. Artificial neural networks have been employed in this capacity, however built in temporal memory in the network architecture (tap delays, feedback etc.) has not been widely explored. In this comparative study, a number of artificial neural network architectures are evaluated for water distribution flow time series prediction, in particular by exploring using temporal memory. These models included multi-layer perceptron, mixture density network, time delay network and recurrent network. In addition the mean diurnal cycle (calculated from the data set) was utilised as a baseline prediction. Genetic algorithm optimisation was utilised in some cases to optimise the number of hidden processing elements and the learning rates parameters for the neural network. Two reference data sets are used as a case study originating from typical real world distribution systems and the performance assessed by means of mean absolute error. The results of the study show that of the static networks, the mixture density network is superior for repeatability and insensitivity to parameter settings. Similarly, the recurrent network is generally superior to the time delay network in this capacity. However, the use of either time delays or feedback results in approximately 50% less error than a static network for the best performing networks.

KEYWORDS: Time series prediction, water distribution systems, DMA flow, ANN, MLP, MDN, TDNN, Recurrent

1 INTRODUCTION

The worldwide water industry has been making increasing use of advances in sensor technology for monitoring parameters of water systems to identify performance shortfalls in order to improve asset management and hence provide better customer service, value and regulatory performance. For example, in water distribution systems (WDS) sensors for flow and pressure have become more widely used, especially on trunk mains and at District Meter Area (DMA) level, in order to facilitate zone-based asset management.

Continuous on-line monitors and sensors are increasingly being used to measure a wide range of potable water hydraulic and quality variables within WDS [1]. The proliferation and diminishing costs of automated data transfer, such as by SMS and GPRS systems, is allowing all types of recorded data to be transferred from many disparate points on the networks. Water utilities are struggling to archive or to transform the data effectively into knowledge with which to enable operational control. Data-driven modelling provides a mapping between the inputs and outputs of a given system, with the advantage of not requiring a detailed understanding of the physical, chemical and/or biological processes that affect a system – and it is emerging as an attractive option for prediction and classification in water systems. Data-driven models can complement and sometimes replace deterministic models [2] and Artificial Neural Networks (ANNs) are one such model. ANNs have been successfully applied to a range of water modelling problems and have displayed particular promise for forecasting applications. They can be evaluated based on physical model outcomes and experimental/field data can be further integrated in order to enhance their performance. Maier and Dandy [3] provide a comprehensive review of 43 papers dealing with the use of ANNs for the prediction and forecasting of water resources variables, as well as a useful protocol for developing such models. Their study found that in all but two papers reviewed, feedforward networks were used, and that most used the backpropagation training algorithm.

Sensor data (such as flow) obtained from a WDS is in the form of a time series—that is, a data stream consisting of one or more variables whose value is a function of time. Standard industry practice in the UK is to sample at a regular time interval of 15 minutes to produce a discrete series. Due to opportunities afforded in recent years near real time data acquisition is enabling new applications to be developed utilising this data. The challenge for data-driven approaches is to learn and predict the normal variability of a particular time series and then to use this in areas such as demand forecasting or abnormality detection.

Water demand forecasts are useful for estimating future water demands in different time scales and evaluating water demand management measures in urban areas. ANNs have been applied to both short-term demand forecasting for WDS, i.e., for 24 h demand forecasts [4, 5, 6] and for much longer term demand prediction, such as over a 10-year horizon [7]. These models often take in other factors, such as temperature, humidity etc. and operate at varying scales. Other techniques such as projection pursuit regression, multivariate adaptive regression splines, random forests and support vector regression [8] and genetic programming [9] have been considered.

[1] Dept. of Civil and Structural Engineering, Univ. of Sheffield, S1 4JF, UK. Email: s.r.mounce@sheffield.ac.uk.

In the second area of research, current work has explored the prediction of flow (and pressure) allowing for the DMA level detection of burst events or other abnormal demand. Generally, some sort of future time series prediction based solely on the observed data is produced and the resulting residuals (when comparing with the actual value) are examined in order to determine abnormality. A number of methodologies for the prediction of DMA monitoring readings (common in the UK) are described in the literature with the potential for real time processing of sensor data streams in mind. Mounce et al. [10, 11] used a Mixture Density Network (MDN) ANN, trained using a continually updated historic database that constructed a probability density model of the future flow profile. Romano et al. [12] used feed-forward multilayer perceptron (MLP) ANNs for short-term forecasting of future pressure/flow signal values. Adaptive Kalman filtering has been used to model normal water usage, so the residual of the filter represents the amount of abnormal water usage relating to bursts or other abnormal usage [13]. Support Vector Regression has also been explored for water time series prediction to enable novelty detection in WDS time series data [14]. In all these cases, some secondary process or methodology is required to analyse the residuals and make an event classification. It should be noted that the ANN approach employed in these applications has been to statically encode the temporal sequence on the input layer, and that built in temporal memory (tap delays, feedback etc.) has not been widely explored. It could be questioned whether this approach is sufficient to represent non-linear dynamic behaviour accurately.

This paper presents a comparison in performance of a number of ANN architectures for WDS flow time series data, in particular exploring using ANN temporal memory. Two reference data sets are used as a case study originating from typical real world WDS. The rest of the paper is organised as follows. In section 2, the case study data is outlined and the prediction task defined. Next, in section 3, the ANN architectures are described with a focus on their use for time series prediction and some implementation details. In Section 4, results are presented and discussed of applying the techniques described in section 3. Section 5 presents the conclusions.

2 CASE STUDY

2.1 Data sources

Two reference data sets were selected from available historic data. Each originated from a DMA inlet flow sensor in a UK water distribution system with a fifteen minute sample frequency. Both are characterised by a reasonably long time series, are fairly stationary and, since the goal was investigating parameters for the time series distribution prediction, contain few major events. Each data set time series was separated into a training set and a smaller test set.

The first dataset consists of a flow meter located at a PRV with the DMA having approximately 3000 properties. Figure 1 shows the raw data of the training set used in the example covering a period of 8 months. Missing data has been filled using an ARIMA approach (missing data segments appear dark). As can be seen, the minimum night flow is fairly constant at just below 10 l/s for this period. A consecutive period of six weeks was used as the test period (no filled data).

The second dataset consists of an inlet flow meter for a DMA fed from a ring main system with the DMA. Figure 2 shows the raw data of the training set used in the example covering a period of 3 months. No missing data was present. A consecutive period of five weeks was used as the test period (no filled data).

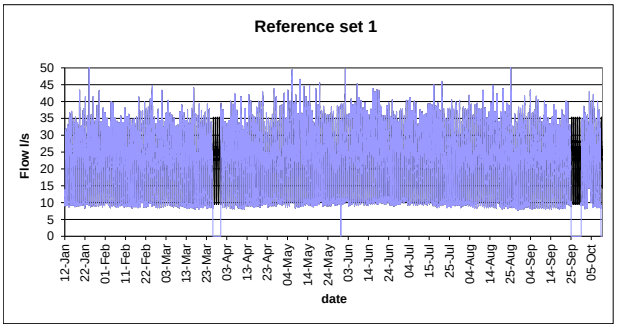

Figure 1. Reference Set 1

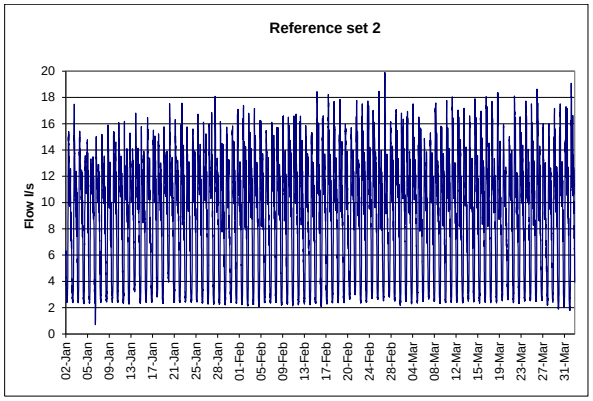

Figure 2. Reference Set 2

2.2 The prediction task

Forecasting future values of time series is useful in many domains. A set of temporal ordered observations can be used to predict future values using previous observations due to serial correlations along the series. The task is essentially one of function approximation, i.e., to approximate the underlying continuous valued function F producing the time series.

Time series data of real-world phenomena is inherently non-stationary. If the series is non-stationary, then all the typical results of the classical regression analysis are not valid and results will be spurious. Time series produced by sensors deployed in WDS are, in general, non-stationary and manifest significant noise (both observational and measurement) because of changing network characteristics (for example, a valve which is closed may result in a new flow profile) as well as consumption patterns altering over longer periods. However, hydraulic data can be described as ‘almost’ stationary in that for settled periods the data acts in a stationary fashion, albeit with a periodic component. A diurnal cycle generally manifested in hydraulic parameters is a reflection of the dominating residential consumption pattern. This is nearly always present in DMA level data although not necessarily as prominent in larger bulk water transfer data such as trunk main monitoring.

ANNs have shown to be a promising alternative to traditional techniques for non-linear temporal processing tasks [15]. ANNs are well suited to this task since they are, in theory, universal computing machines capable of arbitrary function approximation. The non-linear mapping performed can be thought of as a multi-dimensional mapping surface, which is molded to a desired response. Conventional ANN architectures and learning algorithms are mostly designed for detecting patterns that do not change in time (static patterns). By contrast, temporal pattern recognition involves the processing of patterns that evolve over time. The appropriate response at a particular point in time depends not only on the current input, but potentially on all past inputs. The output of the neural network y(t) is based on the input sequence x(t), x(t-1), x(t-2),.., x(0). There are two approaches to building time into an ANN: implicit and explicit. In the latter, time is given its own particular representation. Most general-purpose theory and architectures concentrate on an implicit representation. This can be achieved by embedding the temporal structure of the input signal within the spatial structure of the network (e.g. Time Delays) or by the use of feedback. Four categories can be defined [16]:

- Layer delay without feedback (time delay)
- Layer delay with feedback
- Unit delay without feedback
- Unit delay with feedback (self-recurrent loops)

Several mechanisms for implementing an implicit representation are described in section 3.

A training set of samples can be constructed using a particular dimensional delay vector **m** of the last *m* observations, where the sampling time is taken as uniform with τ the lag time i.e. the sampling rate and predicting **n** time steps into the future:

$$\underline{x}_t = \left[x(t), x(t-\tau),..., x(t-(m-1)\tau)\right] \text{ with target prediction}$$
$$y_t = x(t+\mathrm{n}) \qquad (1)$$

Previous work has largely concentrated on next step ahead prediction as the forecasting goal (i.e. n=1 in equation 1) - this is the usual approach for optimal accuracy of prediction a short period into the future. Here, a 24 hour ahead prediction is attempted as a harder problem. Note that this is not so called multi-step ahead time series prediction in which the model is applied step by step to predict future values. This type of approach uses predicted values from the past and a significant problem with this methodology is that errors from the past are propagated into future predictions. Rather, the goal is to predict the likely value 24 hours (or 96 time steps for 15 minute data) into the future based on **m.** Other work [6] has found the MAE for this type of prediction (not surprisingly) higher than for the one step ahead value so this is a more challenging task. Another rationale is that some water utilities use daily download, via SMS, so predicting further into the future could be useful in such a scheme. The lag size for static encoding was 96 values (sensitivity studies on lag size not reported here had revealed approximately 10% less MAE for static ANNs when using this lag compared to smaller lags).

Finally, it should be noted some previous work has explored using separate models for different days of the week (particularly weekend vs week days) since there is often a difference in demand profiles for different days (albeit more subtle than the diurnal profile). In this study the time series is kept as a continuous entity and not subdivided in this way in line with what ANN architectures using delays and feedback will expect.

3 METHDOLOGY

A number of ANN architectures were used to assess their efficacy for flow time series prediction in WDS. These included: MLP, MDN, Time Delay ANN (TDNN) and recurrent ANN. In addition the mean diurnal cycle (calculated from the data set) was utilised as a baseline prediction.

Mean diurnal cycle predictions and the MDN were developed in MATLAB[2] (the latter using the NETLAB toolbox [17]). Neurosolutions[3] was used for construction, training and testing of alternative architecture ANNs for time series prediction. MATLAB code was used for data pre-processing. To ensure a continuous data stream an ARIMA based filter fills in any periods of missing data. The input is then normalised by means of linear re-scaling with mean and standard deviation (Z-score) to a range of 0 to 1. Finally, the input stream is reformatted into a tapped delay line format (if required) in order to prepare for ANN presentation.

3.1 Mean Diurnal Cycle

The average diurnal cycle calculated from the training set was used as a 24-hour ahead prediction for the test sets.

3.2 MLP

The static network turns a temporal sequence into a spatial pattern encoded on the input layer. A MLP trained with backpropagation is the most popular method to do this [18]. The strategy for processing temporal information is to represent a sequence of input time series data 'simultaneously' on the input layer of the network. A traditional feedforward ANN can be used for time series forecasting by employing a sliding window over the input sequence. A set of N-tuples is used as input and a single output is the target value of the network. Figure 3 shows the basic architecture for lag size of three, and next step ahead prediction. No explicit reference to the dynamic nature of time is made. Full mathematical treatment of backpropagation can be readily found in the literature (e.g. [19]).

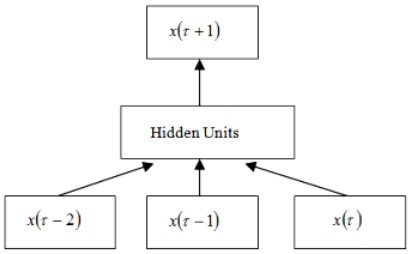

Figure 3. ANN for time series prediction using sliding window

The NeuroSolutions ANN simulator was used to build a static ANN for the 24-hour ahead prediction task. The training and test

[2] MATLAB version 7.10.0. Natick, Massachusetts: The MathWorks Inc., 2010.

[3] NeuroSolutions (Version 5.06; NeuroDimension, Inc.)

sets were assembled by statically encoding the temporal sequence on the input layer. A standard MLP was built in NeuroSolutions: a one hidden layer MLP trained with backpropagation with a momentum term. The transfer functions are tanh and the data was further normalized to between –0.9 and 0.9. For all the networks, a stopping criteria based on the Mean Squared Error (MSE) for a Cross Validation (CV) set consisting of 20% of the exemplars from the original training set was used. This is generally agreed to be an appropriate figure [20]. In most cases, if no improvement on the CV MSE was seen for 10-20 epochs, training was halted. Once CV MSE starts increasing, generalization is lost.

Training MLPs is often a trial and error procedure since there are a large number of degrees of freedom in terms of architecture and parameters (for example: number of hidden layers, size of hidden layers, step size for learning, momentum term, online or batch learning, transfer function type, bias levels and stopping criterion) and few rules or formal procedures for selecting these. Experimentation on network size and parameters was conducted based on experience of similar learning tasks. However, to speed selection of parameters, a Genetic Optimisation feature of NeuroSolutions was applied which implements a genetic algorithm (a general-purpose search algorithm based upon the principles of evolution observed in nature) to optimise parameters within the ANN. The Genetic Algorithm (GA) achieves this by combining selection, crossover, and mutation operators with the goal of finding the best solution to a problem. The solution to a problem is called a chromosome. A chromosome is made up of a collection of genes, which are simply the neural network parameters to be optimized. A GA creates an initial population (a collection of chromosomes) and then evaluates this population by training an ANN for each chromosome. It then evolves the population through multiple generations in the search for the best network parameters. In this case, the selected parameters were the number of hidden processing elements and the learning rates. The termination criteria used to evaluate the fitness of each potential solution is the lowest cost achieved during the training run.

3.3 MDN

The MDN is a mixture density model combined with an ANN [21]. They can provide a framework for modelling conditional probability density functions $p(t|x)$ for a time series prediction [22]. The distribution of the outputs t is described by a parametric model whose parameters are determined by the output of a feed-forward neural network. The ANN element of the MDN is implemented with a two-layer feedforward MLP having a single hidden layer of hyperbolic tangent (tanh) units and an output of linear units (the vector holding the parameters that define the Gaussian Mixture Model). An MDN can be used in a similar fashion to an MLP by employing a sliding window over the input sequence (an MDN is a more general model for non-linear regression than an MLP under this scheme). More detail on its application for water data can be found in [10] in particular on how a prediction confidence can be obtained.

The MDN algorithm is implemented in the NETLAB toolbox for MATLAB. This software library includes a number of data analysis techniques, including mixture model implementations, which are rarely, if ever, included in standard ANN simulation packages. The library contains an MDN module. A two-layer feedforward MLP is utilised in which the weights are drawn from a zero mean, unit variance isotropic Gaussian using the Matlab function randn. The mixture model is Gaussian with a single covariance parameter for each component. The mixture coefficients are computed from a group of softmax outputs, the centres are equal to a group of linear outputs, and the variances are obtained by applying the exponential function to a third group of outputs. The network is trained using a Scaled Conjugate Gradient (SCG) optimiser [23], which is a general-purpose optimisation routine.

A systematic sensitivity study was conducted to identify the optimal number of Gaussians, hidden units and number of training cycles for the reference data sets which for brevity is not reported in detail here. Generally, a wide range of parameters gave good results for the MDN architecture. Findings indicated 1 or 2 Gaussians and 10-15 hidden units was a good choice for 96 lag. The number of training cycles required to obtain a satisfactory error level was small. Good performance for prediction was achieved after approximately 100 cycles, when the majority of learning occurred. Additional cycles contributed to some fine-tuning. For training files covering several months around 1000 cycles was typical. The best performing nets were used for comparison with other architectures.

3.4 TDNN

An ANN can be provided with 'memory' to deal with the temporal dimension by introducing a time delay on connections. The input signal is delayed by one time unit. The network receives both the original and the delayed signals $x(t), x(t-1), \ldots, x(t-d)$. New information is placed at nodes at one end and the old information shifts down a series of nodes like a shift register controlled by a clock. This scheme effectively provides a static neural network with dynamic properties. Essentially, time delays are a way of building short-term memory (in contrast to the usual long-term memory a static ANN develops through supervised training), which is required to make the network dynamic. Time delays can be implemented in two general ways: at the synaptic level inside the network (*distributed*) or at the input level of the network (*focused*). A TDNN has been used previously for district meter area flow data, but only as a classification task for known events [24].

The NeuroSolutions ANN simulator was used to implement a TDNN for prediction. The NeuroSolutions implementation is a Time-Lagged Feedforward Network (TLFN). The TLFN is an MLP with memory components to store past values of the data in the network. The memory components allow the network to learn relationships over time. It is probably the most common temporal supervised neural network. Similar training settings were used as for the MLP, except the input file was not static.

3.5 Recurrent network

Recurrent networks allow recurrences through feedback connections. Temporal pattern recognition is achieved through recurrent connections into input, hidden or output neurons. The neuron receives two types of input, one is from the current incoming data and the other is from the state information at the preceding time, which is fed back in to the network. In this way

the neural network can integrate temporal information going back to the starting point. Feedback can be applied in two basic ways. Firstly, local feedback provides feedback at the level of a single neuron inside the network - this is relatively straightforward. More generally, global feedback involves feedback encompassing the whole network. An ANN of this form can be considered as a non-linear dynamical system.

Recurrent networks have two functional uses: associative memories (e.g. Hopfield network) and input-output mapping networks (e.g. simple recurrent network). The latter type, responding as they do temporally to an externally applied input signal, are sometimes referred to as dynamically driven recurrent networks. Feedback allows this form of network to acquire state representation.

Recurrent networks offer an attractive ability to recognise non-linear relationships between elements of time series. Another advantage of recurrent networks is the ability to recognise temporal patterns independent of their duration. However, the actual choice of feedbacks, delays and weights still depends largely upon empirical evaluations. It is important to point out that recurrent non-linear networks demand large training data sets and careful evaluation of their learning algorithm is necessary. They have a tendency to settle in a stable local minima. Also, if feedback is not applied in a correct manner, harmful effects can result.

The NeuroSolutions ANN simulator was used to implement a Time-Lagged Recurrent Network (TLRN). TLRNs are MLPs extended with short-term memory structures that have local recurrent connections. The NeuroSolutions architecture was built with a Gamma memory (cascade of leaky integrators). A low pass filter creates an output that is a weighted (average) value of some of its more recent past inputs. A focused architecture was selected, so that memory kernels are only connected to the input layer i.e. only the past is remembered.

The training algorithm is the backpropagation through time (BPTT) algorithm [25]. In contrast to MLPs, TLRNs have a smaller network size to learn temporal problems (since MLPs use extra inputs to represent the past samples). The recurrence of the TLRN provides the advantage of an adaptive memory depth (i.e., it finds the best duration to represent the input signal's past). From a system identification point of view, TLRNs implement nonlinear moving average (NMA) models.

4 RESULTS AND DISCUSSION

In each case, the reference training sets were used to train the ANN (or construct the average diurnal model) and performance compared by use of metrics. The precision of prediction models can be measured in term of the errors in the predicted values in relation to those observed. There are a variety of metrics that can be employed for assessing time series prediction, and no general agreement as to a definitive set of metrics as to some extent it will depend on the application and the issues that must be addressed such as repeatability or the precision in goodness of the prediction.

For this study, the relative prediction accuracy was assessed based on the Mean Absolute Error (MAE) of the validation data sets (equation 2) which is a commonly used metric in this field.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|o_i - y_i\right| \qquad (2)$$

where o is the observation and y the prediction.

A continuous error metric such as either MAE or MSE would be suitable, where the errors are summed over the validation set of inputs and outputs and then normalised to the size of the validation set. MSE penalizes distant errors more severely and therefore favours a network with few or no distant errors. MAE takes large errors into account, but does not weigh them more heavily. For classification tasks, the choice of MSE may be more appropriate since MSE penalises distant errors (i.e. clear misses on class targets) more severely and therefore favours a network with few or no distant errors.

The error term employed for training is dependent on particular architectures. For the MLP, TLFN and TLRN networks MSE was used to evaluate training epochs while for the MDN the negative log likelihood error function was used. For comparing training versus testing performance for a single ANN, a uniform metric should normally be preferred. But for evaluating the prediction accuracy here the most important consideration in the decision was to have a fair and uniform comparison across different ANN simulator environments, so MAE was chosen to facilitate this. The mean and standard deviation of this value for multiple ANN runs were also used as an indication of repeatability.

4.1 Mean Diurnal Cycle

Table 1 gives the MAE obtained by using the average diurnal cycle calculated from the normalised training set as a 24 hour ahead prediction for the test set. The processed error was calculated by first denormalising the prediction values before calculating residual averages in l/s.

Test Data Set	Average diurnal	
	Raw MAE	**Processed average error**
Reference set 1	0.0885	2.61 l/s
Reference set 2	0.0806	1.83 l/s

Table 1. Results for mean diurnal cycle

4.2 MLP

Two simulations were conducted using GA to optimise the selection of parameters (GA1 and GA2) on Reference Set 1. The first consisted of 100 epochs, 20 chromosomes and 2 generations. The second: 50 epochs, 20 chromosomes and 5 generations. Each simulation resulted in a best solution that was used to inform the choice of parameters for both reference sets. Thus, for MLP 4-6 the GA simulations provided the choice of step size and momentum. Generally, these values varied between 0.3-0.7 for the step size (η) and 0.6-0.85 (m) for the momentum (the GA optimizes these values for different layers). Table 2 documents the errors (to 4 d.p.) for the various networks. 'T' indicates the network was terminated, 'CV' indicates cross validation was used as the stopping criteria. The optimal architecture as indicated by the GA was between 2 and 4 processing units on the hidden layer. The best performing network based on MAE is highlighted. As we would expect, the ANN model shows a clear superiority for prediction error level compared to the average diurnal prediction.

ANN	Reference set 1			Reference set 2		
	Epochs	MSE (train)	MAE (test)	Epochs	MSE (train)	MAE (test)
MLP1 96,10,1 η 0.1, m 0.7	500 (T)	0.0203	0.0767	1900 (T)	0.0088	0.0439
MLP2 96,20,1 η 0.2, m 0.7	1420 (CV)	0.0111	0.0511	1200 (T)	0.0140	0.1141
MLP3 96,25,1 η 0.3, m 0.6	905 (CV)	0.0247	0.0795	-	-	-
GA1	100 Winner: 96,2,1	-	0.0413	100 Winner: 96,4,1	-	0.0405
MLP4 96,2,1 (GA)	1401 (CV)	0.0081	0.0405	985 (CV)	0.0086	**0.0398**
MLP5 96,3,1 (GA)	680 (CV)	0.0079	0.0403	2500 (T)	0.0091	0.0421
GA2	50 Winner: 96,4,1	-	0.0403	50 Winner: 96,3,1	-	0.0411
MLP6 96,4,1 (GA)	1240 (CV)	0.0072	**0.0385**	1546 (CV)	0.0105	0.0454

Table 2. Results for MLP

4.3 MDN

Table 3 documents the errors for a range of MDN networks. The conditional average is calculated from the conditional probability density and used as the prediction in the MAE calculations.

ANN	Reference set 1			Reference set 2		
	Epochs	Error (train)	MAE (test)	Epochs	Error (train)	MAE (test)
MDN1 96,3,1 1 GMM	900	-36210	0.0375	800	-14698	0.0432
MDN2 96,10,1 1 GMM	1000	-36736	0.0370	1100	-15175	0.0427
MDN3 96,15,1 1 GMM	1200	-37310	0.0372	1000	-15291	0.0432
MDN4 96,3,1 2 GMM	900	-36276	0.0381	1200	-15077	0.0426
MDN5 96,10,1 2 GMM	900	-37355	**0.0362**	1100	-14621	0.0424
MDN6 96,15,1 2 GMM	1100	-38046	0.0363	1000	-15490	**0.0417**
MDN7 96,10,1 3 GMM	1000	-37932	**0.0362**	1200	-15796	0.0421

Table 3. Results for MDN

Training times (epochs) were similar to the MLP. But one important consideration was that the MLP was more sensitive to parameter choice than the MDN – selection of architecture and parameters was much more critical to success compared to the MDN. Otherwise best MAE levels were similar, which was expected since both networks use static encoding.

4.4 TDNN

Table 4 gives the results for the TLFN networks used (single hidden layer networks trained using backpropagation with momentum). A smaller learning parameter value (0.01) was used in later epochs. The number in parentheses after the model name indicates the number of taps. A GA was also used for optimising the parameters with 50 population and 4 generations. Experiments were also conducted on both a high tap number (up to 48) and on using a tap delay larger than 1. This delay represents a number of samples between successive taps (e.g. tap 4, delay 24 covers 24hrs in 4 values for 15 min data). Both of these factors were found not to contribute any improvement for the prediction task. Some experimentation was carried out using iterative prediction. For a multi-step prediction problem one approach is to use current and previous data to predict N-steps ahead. Alternatively, iterative prediction just predicts one sample ahead. It then uses the output of this prediction as an input to a prediction of the next sample. This cycle is continued until a prediction N-steps ahead is made. However, it was found that iterative prediction was not suitable for this application because the prediction into the future was too distant (larger prediction lengths lead to instability when using iterative teaching).

ANN	Reference set 1			Reference set 2		
	Epochs	MSE (train)	MAE (test)	Epochs	MSE (train)	MAE (test)
TD1 (3) 96,10,1 η 0.02, m 0.6	1190 (CV)	0.0076	0.0481	145 (CV)	0.0104	0.0195
TD2 (5) 96,10,1 η 0.02, m 0.6	337 (CV)	0.0075	0.0751	180 (CV)	0.0108	0.0204
TD3 (3) 96,10,1 η 0.1, m 0.7	539 (CV)	0.0075	0.0211	110	0.0107	0.0300
TD4 (2) 96,10,1 η 0.1, m 0.7	350 (CV)	0.0072	**0.0160**	110	0.0099	0.0210
TD5 (1) 96,10,1 η 0.1, m 0.7	240 (CV)	0.0071	0.0721	265 (CV)	0.0102	0.0240
GA	50 Winner: 96,5,1 1 tap	-	0.0580	50 Winner: 96,10,1 1 tap	-	**0.0192**
TD6 GA Win	450 (CV)	0.0074	0.0580	230 (CV)	0.01069	0.0222
TD7 (1) 96,5,1 η 0.1, m 0.7	360 (CV)	0.00757	0.0674	125 (CV)	0.0104	0.0223

Table 4. Results for TDNN

Training observations included:

- Training time was very short compared to the MLP. A small value of MSE for the training set was obtained after only a few epochs. Without a CV set training rapidly results

in over specialization to the training set. Most subsequent training is aimed at reducing the MSE of the CV set.

- By using much smaller values of learning parameters a better convergence of CV MSE could be obtained, at very little expense of training time. Further, problems with local minima are less likely.
- The good results obtained for some neural nets were not guaranteed (particularly for reference set 1). Runs conducted with the same set of parameters gave a wide range of results. Initial starting conditions (weight values) played a significant part, as well as encountering local minima. Stopping criteria proved more vital as large oscillations were observed for MSE compared to the MLP. However, the best performing networks were superior to the MDN/MLP as should be expected for a topology that incorporates the temporal dimension.

4.5 Recurrent network

Table 5 gives the results for the TLRN networks.

ANN	Reference set 1			Reference set 2		
	Epochs	MSE (train)	MAE (test)	Epochs	MSE (train)	MAE (test)
R1 (6) 96,50,1 η 0.1, m 0.7	400	0.0216	0.0220	500	0.0230	0.0244
R2 (6) 96,200,1 η 0.1, m 0.7	400	0.0202	0.018	500	0.0222	**0.0219**
R3 (6) 96,500,1 η 0.1, m 0.7	400	0.0196	**0.0168**	500	0.0220	0.0219
R4 (6) 96,600,1 η 0.1, m 0.7	400	0.01980	0.0172	133 (CV)	0.0353	0.0426
R5 (3) 96,500,1 η 0.1, m 0.7	400	0.0201	0.0186	500	0.0225	0.0228
R6 (12) 96,500,1 η 0.1, m 0.7	400	0.0190	0.0186	500	0.0216	0.0244

Table 5. Results for TLRN

Training observations included:

- Training was rapid, with most of the reduction in MSE observed over 20-50 epochs. The graph of MSE was observed to be smooth, in particular compared to MLP/TLFN.
- A broad range of parameters gave good MAE results for the test sets. Hence performance was more predictable compared to the TLFN.

4.6 Summary

Figure 4 provides a graphical summary of best MAE levels for the various techniques. Table 6 gives a comparison of actual test set prediction errors (mean and standard deviations over total runs) for the various ANN architectures. Clearly, MDN has a similar best performance level to the MLP. Similarly, the TLFN and the TLRN networks are comparable and with an approximate 50% reduction in MAE level compared to the two static architectures. This finding is not very surprising as TLFN/TLRN networks have memory components to store past values of the data in the network thus allowing networks to learn relationships over time. We should expect the extra memory structure in the time-lagged architectures to result in a performance gain for time series prediction. In contrast, static networks must turn a temporal sequence into a spatial pattern encoded on the input layer. However, an important caveat is revealed after inspecting table 6. The MDN is superior to the MLP in that it has less sensitivity to initial condition and parameter values (the standard deviations for the MAE error levels reveal that the MDN shows an order of magnitude less variation for different network set-ups). Therefore the average MAE for the MDN is significantly better than the MLP. The same applies to the TLFN when compared to the TLRN – indeed the average MAE of the TLFN was worse than that of the MDN for reference set 1. So while superior MAE error levels are achievable with the two time-lagged networks, the MDN can be relied on to give similar MAE results for a broad range of the architecture parameters used. This is a useful property when time or human resources are not available to find an optimal solution of parameter values for different data sets. Of course, if time series prediction error residuals are the only consideration, a time-lagged architecture should be selected.

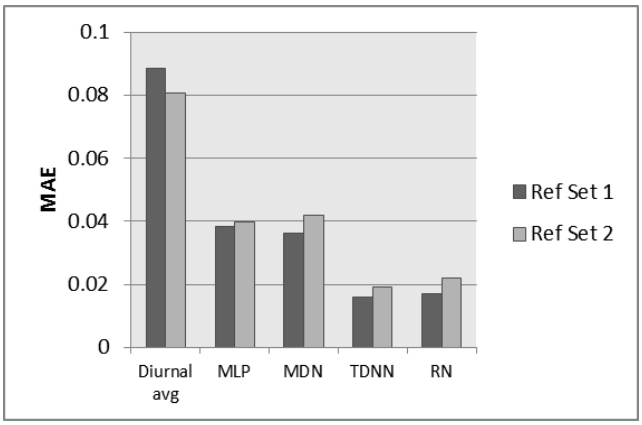

Figure 4. Comparison of best MAE levels for prediction techniques

Test Data Set	MDN prediction	MLP prediction	TLFN prediction	TLRN prediction
	Mean/s.d. MAE	Mean/s.d. MAE	Mean/s.d. MAE	Mean/s.d. MAE
Ref Set 1	0.0369/ **0.0006**	0.0510/ 0.0171	0.0520/ 0.0223	**0.0260**/ 0.0210
Ref Set 2	0.0425/ **0.0009**	0.0524/ 0.0273	**0.0223**/ 0.0034	0.0258/ 0.0075

Table 6. Comparison of prediction MAE for ANN architectures across all networks

5 CONCLUSIONS

This paper has investigated how a number of ANN architectures perform when predicting future values of WDS DMA flow. The

approach of the research has been to use time series produced by sensors to directly construct an empirical model by use of an ANN. A systematic study based upon two reference sets was performed comparing the performance achievable from several techniques: mean diurnal cycle, MLP, MDN, TDNN and recurrent ANN. The key results and findings were as follows:

- The results of the study demonstrate how ANNs can provide superior predictions of future flow values in WDS. All ANN models had at least 50% less MAE than a prediction based on the average diurnal cycle directly calculated from the data.
- Of the static networks, the MDN is superior for repeatability and insensitivity to parameter settings. Similarly, the recurrent network is generally superior to the TDNN in this capacity. However, the use of either time delays or feedback results in approximately 50% less error than a static network for the best performing networks.
- The advantages of the ANN approach for this problem are that it is a data driven paradigm, with robustness to noisy or incomplete data, with knowledge that can be relatively easily updated (retraining), generalisation on unseen data and, once trained, fast execution speed.

The easier it is to collect and analyse large data sets the more water utilities will collect and, in a decade, tens or even hundreds of Petabytes of data may be routinely available. There is a growing need for sophisticated data analysis using the resultant data and ANNs are one type of data driven model that can provide predictive capabilities. Future work in this area may involve developing hybrid temporal architectures for addressing specific types of water resources data sets collected.

ACKNOWLEDGEMENTS

This work was part supported by the Pipe Dreams project funded by the UK Science and Engineering Research Council, grant EP/G029946/1. The author wishes to also acknowledge the support given for this research through data provision by Yorkshire Water Services Ltd., UK.

REFERENCES

[1] Z. Y. Wu, M. Farley, D. Turtle, Z. Kapelan, J. B. Boxall, S. R. Mounce, S. Dahasahasra, M. Mulay and Y. Kleiner, *Water Loss Reduction*, Bentley Systems, ed. Zheng Wu, (2011).

[2] D. Solomatine, Data-driven modelling: paradigm, methods, experiences. In: *Proc. 5th international conference on hydroinformatics*. p. 1-5, (2002).

[3] H. R. Maier and G. C. Dandy, Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. *Environ. Modell. Software*, 15: 101–124, (2000).

[4] A. Jain and L. E. Ormsbee, Short-term water demand forecasting modelling techniques-conventional versus AI. *Journal AWWA*, 94, 64–72, (2002).

[5] G. Greenaway, R. Guanlao, N. Bayda and Q. Zhang, Water distribution systems demand forecasting with pattern recognition. *Proc., 8th Water Distribution System Analysis Symp*., USEPA/Univ. of Cincinnati, Cincinnati, (2006).

[6] F. K. Odan and L. F. R. Reis. Hybrid Water Demand Forecasting Model Associating Artificial Neural Network with Fourier Series. *Proc., 12th Water Distribution System Analysis Symp*., ASCE, Tucson, Arizona, United States, (2010).

[7] N. Chang and A. Makkeasorn, Water demand analysis in urban region by neural network models. *Proc., 8th Water Distribution System Analysis Symp*., USEPA/Univ. of Cincinnati, Cincinnati, (2006).

[8] M. Herrera, L. Torgo, J. Izquierdo and R. Perez-Garcia. Predictive models for forecasting hourly urban water demand. *Journal of Hydrology*, 387: 141–150, (2010).

[9] Z. Y. Wu and Z. Yan. Applying genetic programming approaches to Short-term water demand forecast for district water system. *Proc., 12th Water Distribution System Analysis Symp.*, ASCE, Tucson, Arizona, United States, (2010).

[10] S. R. Mounce, A. Khan, A. S. Wood, A. J. Day, P. D. Widdop and J. Machell, Sensor-fusion of hydraulic data for burst detection and location in a treated water distribution system. *International Journal of Information Fusion*, 4(3): 217–229, (2003).

[11] S. R. Mounce, J. B. Boxall and J. Machell, Development and Verification of an Online Artificial Intelligence System for Burst Detection in Water Distribution Systems. *ASCE Water Resources Planning and Management*. 136(3): 309–318, (2010).

[12] M. Romano, Z. Kapelan, and D. A. Savić, Evolutionary Algorithm and Expectation Maximisation Strategies for Improved Detection of Pipe Bursts and Other Events in Water Distribution Systems, *Journal of HydroInformatics*, In Press, (2013).

[13] G. Ye and R. Fenner, Kalman Filtering of Hydraulic Measurements for Burst Detection in Water Distribution Systems. *ASCE Journal of Pipeline Systems Engineering and Practice*, 2(1): 14–22, (2010).

[14] S. R. Mounce, R. B. Mounce and J. B. Boxall, Novelty detection for time series data analysis in water distribution systems using Support Vector Machines. *Journal of HydroInformatics*, 13(4): 672-686, (2011).

[15] M. C. Mozer, Neural net architecture for temporal sequence processing. *In A. S. Weigend and N. A. Gershenfeld (Eds.), Time Series Prediction: Predicting the Future and Understanding the Past*, pp. 243-264. Addison-Wesley, Redwood City, CA, (1994).

[16] C. Ulbricht, *State Formation in Neural Networks for Handling Temporal Information*. Dissertation, Institut fuer Med. Kybernetik u. AI, University of Vienna, (1995).

[17] NETLAB Toolbox, based on techniques of R. Neuneier, F. Hergert, W. Finnoff and D. Ormoneit, Estimation of conditional densities: A comparison of neural network approaches, in: M. Marinaro, P.Morasso (Eds.), *Proceedings of ICANN 94*, Springer, Berlin, 689–692 (Department of Computer Science and Applied Mathematics, Aston University, Birmingham, UK), (1994).

[18] D. E. Rumelhart, G. E. Hinton and R. J. Williams, Learning representations of back-propagation errors. *Nature* (London), vol. 323, pp. 533-536, (1986).

[19] S. Haykin, *Neural Networks – a comprehensive foundation*. 2nd Ed. Prentice Hall, New Jersey, (1999)

[20] M. Kearns, A bound on the error of cross validation using the approximation and estimation rates, with consequences for the training-test split. *Advances in Neural Information Processing Systems*, 8: 183-189, Cambridge, MA:MIT Press, (1996).

[21] C.M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, NY, (1995).

[22] D. Husmeier and J.G. Taylor, Neural networks for predicting conditional probability densities: Improved training scheme combining EM and RVFL, *Neural Networks*, 11 (1): 89–116, (1998).

[23] M. Moller, A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 6(4):, 525-533, (1993).

[24] S. R. Mounce and J. Machell, Burst detection using hydraulic data from water distribution systems with artificial neural networks. *Urban Water Journal*, 3(1): 21-31, (2006).

[25] P. J. Werbos, Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE*, 78: 1550-1560, (1990).

Rare Events and Extreme Flood Predictions: An Application of Monte Carlo based Statistical Blockade

Renji Remesan[1], Jimson Mathew[2] and Ian Holman[1]

Abstract. Extreme hydrological events have gained significant interest in hydrology as such events have the potential to cause human and economic losses. Meanwhile predictive uncertainty and imprecise peak (extreme flow) estimations are continuing major perplexing concerns in the field of hydrology. A conjunctive application of machine learning and extreme value theory can provide useful solutions to address the extreme values of hydrological series and thus to enhance modelling of values which fall in the 'Tail End' of hydrological distributions. This study introduces a novel Monte Carlo (MC) technique named Statistical Blockade (SB) which focuses on significantly rare values in the tail distributions of data space. The capability of Statistical Blockade is compared with well-trained Artificial Neural Networks (ANN) and Support Vector Machines (SVM) to assess the accuracy of Statistical Blockade. The required optimum input space and training data length for the aforementioned models were identified using Gamma Test. The study was performed on the Beas river catchment in the Himalayan region of India. The SB method has proved its capability to offer better predictive accuracy to find the peaks over a given threshold, which has significance in such catchments with high variability in discharge volume.

1 INTRODUCTION

Reliable and well-timed information on the discharge of Himalayan rivers is of great importance, as most rivers originating from the Himalayas and flowing through India are heavily exploited for hydro-power generation, irrigation development and domestic supply, and manageed with flood control structures. The response of Himalayan rivers is largely unpredictable as it depends on the extent of snowcover and volume of snowpack in their respective catchments. Many studies have shown the incapability of traditional precipitation/snowmelt-discharge models in predicting peak values in glacier-dominated catchments [1]. Meanwhile, the continuing development of digital computers has brought new possibilities in hydrologic modelling with the help of mathematical and data based approaches like Neural Networks, Fuzzy Logic and Support Vector Machines. This paper explores a new realm in hydrological modelling combining Statistical Blockade (which combines extreme value distributions with machine learning to classify extreme data [2]) with vector support classifiers. This is a new application of such methods in hydrology, though it has been successful in circuit and memory design [2].

[1] Cranfield Water Science Institute, Cranfield University, MK43 0AL, UK. Email: {r.remesan, i.holman}@cranfield.ac.uk

[2] Dept. of Computer Science, Univ. of Bristol, BS2 8BB, UK. Email: jimson.mathew@bristol.ac.uk

This study is focusing on the Beas River and its dominantly flash flood prone catchment in Himachal Pradesh, one of the Himalayan states of India. The accuracy of Statistical Blockade is also compared with well-trained SVMs and ANNs to check its credibility in predicting number of peaks above a threshold value.

2 METHODOLOGY

The methodology adopted in this study is shown in Figure 1. In this study, we have:

1. Applied the Gamma Test on the daily hydrological and meteorological data of the Beas Catchment to identify effective input series and optimum training data length for modelling

2. Applied the Statistical Blockade (SB) for identification of peak flood over a given threshold value

3. Built an optimum Support Vector machine (SVM) model for the study region and sorted the results to check the capability of SB.

4. Constructed an ANN model using the data sets from the Beas catchment; then compared the results with that of SVM and to check the clustering capability of SB.

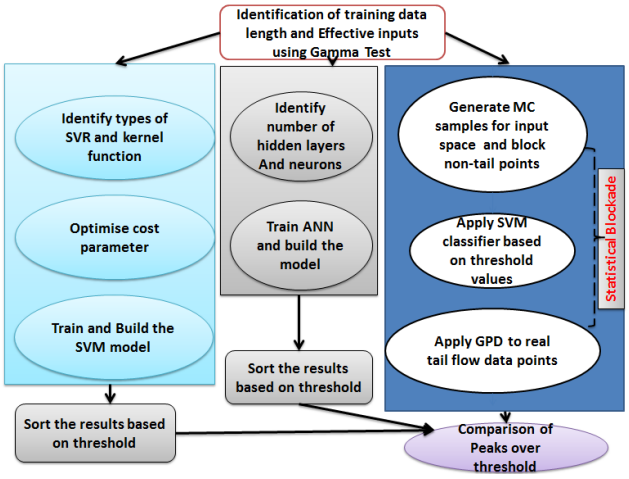

Figure 1. The Methodology Adopted in the Study

3 STUDY AREA AND DATA SETS

The study was performed with discharge data from the Beas River which originates in the Himalayas and flows for approximately 470 km before joining the Sutlej River. The Beas River has a basin area of around 12,561 km^2. There are two major dams along the river, Pong dam and Pandoh dam, which are predominantly used for irrigation, hydro-electric and flood control purposes. Pandoh dam is a diversion dam which diverts nearly 4,716 million cubic metres of Beas waters into the Sutlej River. The daily meteorological data used for the study are from National Centers for Environmental Prediction (NCEP) Climate Forecast System Reanalysis (CFSR) gridded data. The study area is shown in the Figure 2.

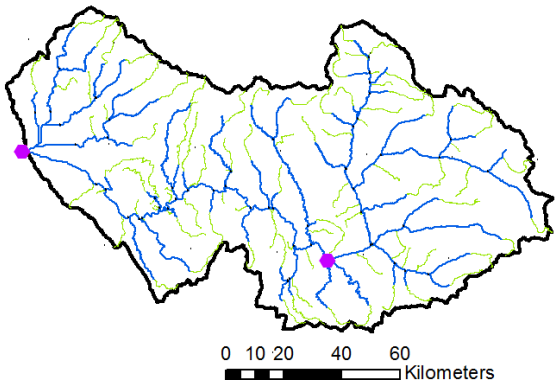

Figure 2. The Study Area of the Beas catchment in Himachal Pradesh, India (Positions of Two Dams are Marked in the Figure)

The available data from the above mentioned data sets are daily precipitation (mm/day) [Precip], daily maximum temperature (^{0}C) [Tmax], daily minimum temperature (^{0}C) [Tmin], daily average solder radiation (MJ/m^2/day) [Solar], daily average wind velocity (m/s) [Wind] and relative humidity (%) [RH]. The study has used river discharge time series (daily inflow data to the Pong dam) obtained from Bhakra Beas Management Board (BBMB). The study has used 5 years data for the analysis from 1st January 1998 to 31st December 2002. The statistics of the daily data sets used for this study are shown in Table 1.

	Precp (mm /day)	RH (%)	Solar (MJ/ m^2 /day)	Tmax (^{0}C)	Tmin (^{0}C)	Wind (m/s)	Flow (m3 /s)
Mean	1.68	0.46	19.85	19.42	6.53	3.59	285.34
Max	97.08	0.95	32.38	31.61	17.99	7.03	5475.67
Min	0.00	0.14	0.89	1.94	-9.27	1.55	16.57
Skews	9.53	0.68	-0.24	-0.37	-0.29	0.18	4.32
Kurt-osis	125.93	-0.09	-0.39	-0.93	-1.21	0.55	28.50
SD	5.38	0.15	6.29	6.83	6.78	0.82	478.92

Table 1. Summary Statistics of Input and Output Data Space used for the Study (Dam Positions Shown)

The high value of standard deviation (SD) and maximum value of the discharge time series in the study period indicates the intensity of possible flood threats in the catchment and it also gives an idea of 'how far' the event could be from its expected average value.

4 INPUTS AND DATA LENGTH SELECTION

In mathematical modelling, it is important to identify the representative inputs and reasonable training data length for modelling. This helps the modeller to avoid over training of the model and to avoid redundant variables from the input space. This study has employed Gamma Test [3, 4, 5] to identify the best possible inputs among our available data sets. Gamma test estimates the minimum mean square error which is achievable in any continuous non- linear model with unseen data.

Only a brief introduction on the Gamma Test is given here as further details can be find in 3, 4, 5. The basic idea is quite distinct from the earlier attempts with nonlinear analysis. Suppose we have a set of data observations of the form

$$\left\{\left(\mathbf{x}_i, y_i\right), 1 \le i \le M\right\} \quad (1)$$

Where the input vectors $\mathbf{x}_i \in R^m$ are vectors confined to some closed bounded set $C \in R^m$ and, without loss of generality, the corresponding outputs $y_i \in R$ are scalars. The vectors $\mathbf{x}$ contain predicatively useful factors influencing the output y. The only assumption made is that the underlying relationship of the system is of the following form

$$y = f\left(\mathbf{x}_1 \ldots \mathbf{x}_m\right) + r \quad (2)$$

Where f is a smooth function and r is a random variable that represents noise. Without loss of generality it can be assumed that the mean of r's distribution is zero (since any constant bias can be subsumed into the unknown function f) and that the variance of the noise Var(r) is bounded. The domain of a possible model is now restricted to the class of smooth functions which have bounded first partial derivatives. The Gamma statistic Γ is an estimate of the model's output variance that cannot be accounted for by a smooth data model.

The Gamma Test is based on $N[i,k]$, which are the kth $\left(1 \le k \le p\right)$nearest neighbours $\mathbf{x}_{N[i,k]}\left(1 \le k \le p\right)$ for each vector $\mathbf{x}_i\left(1 \le i \le M\right)$. Specifically, the Gamma Test is derived from the Delta function of the input vectors:

$$\delta_M(k) = \frac{1}{M}\sum_{i=1}^{M}\left|\mathbf{x}_{N(i,k)} - \mathbf{x}_i\right|^2 \quad \left(1 \le k \le p\right) \quad (3)$$

Where $|\ldots|$ denotes Euclidean distance, and the corresponding Gamma function of the output values:

$$\gamma_M(k) = \frac{1}{2M}\sum_{i=1}^{M}\left|y_{N(i,k)} - y_i\right|^2 \quad \left(1 \le k \le p\right) \quad (4)$$

Where $y_{N(i,k)}$ is the corresponding y-value for the k th nearest neighbour of $\mathbf{x}_i$ in Equation 3. In order to compute Γ a

least squares regression line is constructed for the p points $(\delta_M(k), \gamma_M(k))$

$$\gamma = A\delta + \Gamma \quad (5)$$

Calculating the regression line gradient can also provide helpful information on the complexity of the system under investigation [6]. The graphical output of this regression line (Equation 5) provides very useful information. First, it is remarkable that the vertical intercept Γ of the y (or Gamma) axis offers an estimate of the best MSE achievable utilising a modelling technique for unknown smooth functions of continuous variables [6]. Second, the gradient offers an indication of the model's complexity (a steeper gradient indicates a model of greater complexity).

The Gamma test is a non-parametric method and the results apply regardless of the particular techniques used to subsequently build a model of f. We can standardise the result by considering another term V_{ratio}, which returns a scale invariant noise estimate between zero and one. The V_{ratio} is be defined as

$$V_{ratio} = \frac{\Gamma}{\sigma^2(y)} \quad (6)$$

Where, $\sigma^2(y)$ is the variance of output y, which allows a judgement to be formed independent of the output range as to how well the output can be modelled by a smooth function. A V_{ratio} close to zero indicates that there is a high degree of predictability of the given output y.

We can also determine the reliability of the Γ statistic by running a series of Gamma tests for increasing *M*, to establish the size of data set required to produce a stable asymptote. This is known as the M-test. The M-test result helps to avoid the wasteful attempts of fitting the model beyond the stage where the MSE on the training data is smaller than Var(r), which may lead to *overfitting*. The M-test also helps to decide how much data is required to build a model with a mean squared error which approximates the estimated noise variance. The study has used WinGamma™ implementation for our analysis.

We have tabulated the Gamma Static values corresponding to all data sets in our input space and selected the first four inputs with minimum Gamma Static value. The embedding 000001 model (i.e.: wind speed data as input and flow data as output) was identified as the best structure in comparison to other models with single inputs for daily discharge modelling in the Beas catchment. Also the humidity data is considered as second most effective input data series because of following reasons viz. its low noise level (Γ value), the rapid fall off of the M-test error graph, relatively low V-ratio value (indicating the existence of a reasonably accurate and smooth model), the regression line fit with slope A = 1.3882 (low enough as a simple non-linear model with less complexity). The relative importance of six input data sets in modelling are in the form of Wind speed > Humidity > Minimum Temperature > Precipitation > Maximum Temperature > Solar Radiation. The variations of Gamma Static value for different data sets are shown in the Figure 3 along with corresponding coefficient of correlation values.

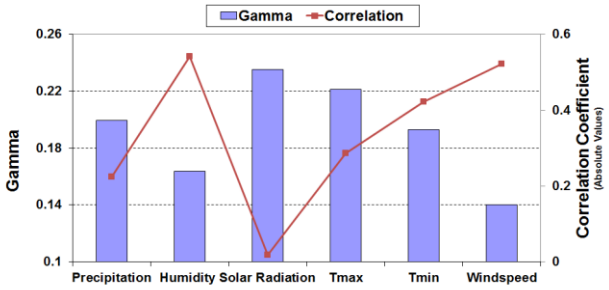

Figure 3. The Variation of Gamma and Coefficient of Correlation Values for Different Time Series

The cross correlation method is the traditional method for identifying the effective inputs suitable for mathematical/statistical modelling. To check the authenticity of above results obtained from the Gamma Test, we performed a cross-correlation analysis between the target discharge data set and different input time series. The outcome from the cross correlation analysis is given in the Figure 3 as a secondary axis. These cross correlation results are matching the results obtained from the Gamma test with a slight disparity between Maximum Temperature and precipitation. Cross Correlation suggests that Maximum daily temperature has more information than that of precipitation data in the Beas catchment to predict discharge in contrast to the results suggested by the Gamma Test. However, it should be pointed out that there is one caveat with this cross correlation procedure viz. cross correlation is suited to linear systems as it is a linear procedure. However in this study we have used four inputs for modelling as suggested by the Gamma Test and those inputs are daily values of wind speed, humidity, minimum temperature and precipitation.

Overfitting or overtraining is a statistical phenomenon associated with nonlinear data-based models when a model is generally complex with too many degrees of freedom, in relation to the amount of data available. So it is important to identify the suitable length of data for training the model. The quantity of the available input data (four input time series in this case) to predict the desirable output was analysed using the M-test (a repeat execution of the Gamma Test with different number of input data lengths). The results obtained from the M-Test with four inputs (110011 Model) are shown in the Figure 4.

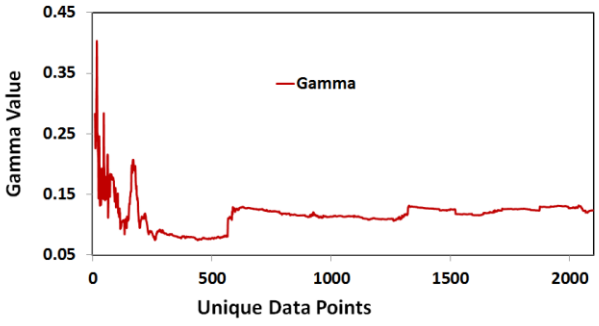

Figure 4. Variation of Gamma Static (Γ) with Unique Data Points Corresponding to Four Input (110011 Model)

The M-test produced an asymptotic convergence of the Gamma statistic to a value of 0.0805 at around 566 data points, then increases. The analysis also shows that the Statistical error (SE) corresponding to M = 566 is very small (0.0019) and that the complexity slightly increases after this point [i.e. Gradient (A) is increasing]. These values altogether can give a clear indication that it is quite adequate to construct a nonlinear predictive model using around 566 data points with reasonable accuracy. So in this study the training data length is selected as 566 data points.

5 SUPPORT VECTOR MACHINES

The SVM approach is a machine learning strategy; which was introduced by Vapnik [7] as an implementation of the structural risk minimization principle. This approach is widely used for analysing data, recognition of patterns, classification and regression analysis. The ν-SV regression structure from LIBSVM [8] was used for this study with four different Kernels (Linear, Polynomial, Radial and Sigmoid).

6 SVM MODELLING RESULTS

The conventional SVM model has used the first 566 data points of the available data points as the training data set using four input data time series as per the recommendations of the Gamma Test. The scaling of the input lists are important in SVM modelling as the difference between extreme values is reduced, which makes it easier and fast to run the SVM algorithm. So, we have normalized the whole data sets in a zero to one range. The proper identification of the kernel function out of the four functions is important in SVM based modelling as kernels are the components which simplify the learning process. Trial and error modelling was adopted to identify the suitable kernel functions and the corresponding results are given in Figure 6 in terms of mean square error (MSE) and correlation of determination (R^2). It was found that the ν-SVR with polynomial kernel function is the best model for the discharge modelling in the Beas basin for the selected input space. The cost parameter (C) of error assigns a penalty for the number of vectors falling between the two hyperplanes in the SVM hypothesis. Estimation of the optimum number of cost is very important as it has an influence on the quality of the data used for the modelling. To ascertain the optimum cost value, the support vector machine made from the best model ν-SVR regression algorithm with polynomial kernel was run several times, with differing values of C between C = 1 to C = 20 (Figure 7).

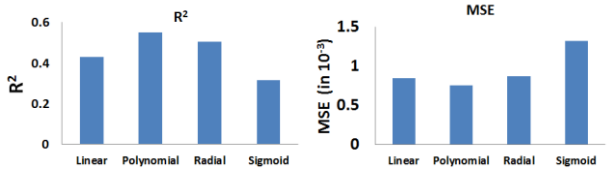

Figure 6. The Modelling Performance of ν-SVR with Different Kernels in Beas Discharge Modelling using Scaled Data Sets [6(a). Variation of R_2 and 6(b). Variation of MSE]

The performance of the models was compared by calculating the mean square error (MSE) of the daily discharge outputs given by the SVM model with that of the actual observed Beas flow into the Pong Dam. The Figure has shown that the mean least error is the lowest when the C parameter is 2. So we set the value of C to 2 for reliable results.

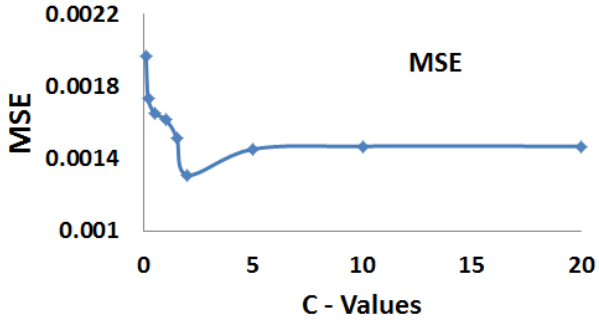

Figure 7. Variation of MSE in ν-SVR with Polynomial Kernel Function Corresponding to Different Cost Parameters in Daily Beas Discharge Modelling.

Now, we have applied reasonably optimised ν-SVR with polynomial kernel to the training and testing data. The modelling result during the training phase is shown in the Figure 8 and corresponding results during Testing or validation phase is shown in the Figure 9.

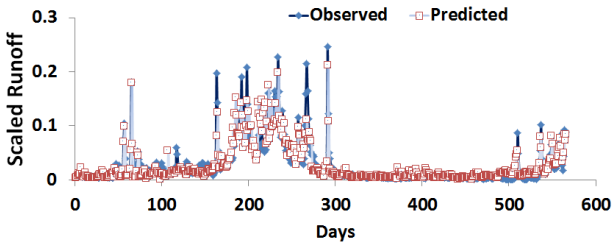

Figure 8. Beas Daily River Flow Modelling Results on Scaled Data Sets during Training Phase using ν-SVR

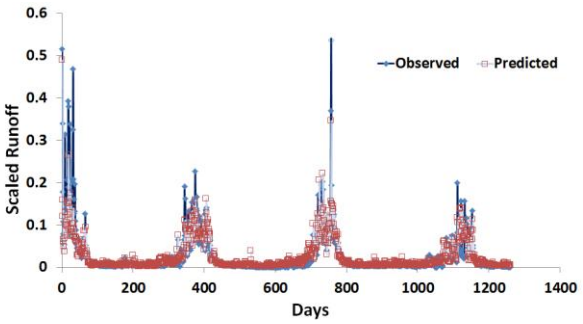

Figure 9. Beas Daily River Flow Modelling Results on Scaled Data Sets during Testing Phase using ν-SVR

One can note from the pictorial results that the model has failed to simulate sudden variations of flood flow during monsoon season but has better agreement during the non-monsoon months. The performance of the model was compared using major statistical indices like coefficient of correlation (CC), Nash and Sutcliffe Efficiency (NS) and percentage bias error. The Nash coefficient of efficiency of the conventional SVM model, which determines capability of the model in predicting discharge values, was calculated as 0.55 with a CC value of 0.76 and a bias value of 57.1% during training phase. The equivalent values during the validation phase are given in Table 2 along with the corresponding values of an ANN model. Considering the peculiar nature (highly skewed data with very high extreme values) of the hydrological data in the Beas region the conventional SVR gave a good fit with the data in the central area around the mean and the mode, but not in the tails. In other words, the whole data consists of some higher values which show a tendency to form clusters (i.e., for storm events) and the conventional model has failed to simulate those clusters of extreme values.

Training Data			
	CC	**NS Efficiency**	**Bias%**
SVM	0.76	0.55	57.10
ANN	0.88	0.76	37.98
Testing Data			
SVM	0.72	0.46	61.46
ANN	0.87	0.75	37.82

Table 2. Comparison of Some basic Performance Indices of SVM and ANN Models in Daily Discharge Modelling

7 ARTIFICIAL NEURAL NETWORKS

The story of ANNs started in early 1940's when McCulloch and Pitts developed the first computational representation of a neuron [9]. Later Rosenblatt proposed the idea of perceptrons [10]; single layer feed forward networks of McCulloch-Pitts neurons, and focussed on computational tasks with the help of weights and training algorithm. The applications of ANNs are based on their ability to mimic the human mental and neural structure to construct a good approximation of functional relationships between past and future values of a time series. The supervised one is the most commonly used ANNs, in which the input is presented to the network along with the desired output, and the weights are adjusted so that the network attempts to produce the desired output. There are different learning algorithms and a popular algorithm is the back propagation algorithm that employs gradient descent and gradient descent with momentum that are often too slow for practical problems because they require small learning rates for stable learning. Algorithms like Conjugate gradient, quasi-Newton, Levenberg–Marquardt (LM) etc. are considered as some of the faster algorithms, which all make use of standard numerical optimization techniques. Architecture of the model including number of hidden layers is also a very important factor.

This study has used a three-layer feed forward neural network (one input layer, one hidden layer and one output layer) which is the most commonly used topology in hydrology. This topology has proved its ability in modelling many real-world functional problems. The selection of hidden neurons is the tricky part in ANN modelling as it relates to the complexity of the system being modelled. In this study we have used 15 hidden neurons which was identified through a trial and error method. The performance of the developed ANN model was compared with SVM models using three global statistics (correlation efficiency, efficiency and bias error) as shown in Table 2. Figures 10 and 11 show resulting line plots of ANN computed and observed daily discharge values in the Beas basin during the training and validation periods. The estimated discharge values using the ANN model for 566 data points resulted in the higher NS efficiency value of 0.76, compare to observed daily discharge and the percentage bias error observed as 37.98%. The corresponding values during testing phase are 0.75 and 37.82% which are much better than that of the SVM predicted results.

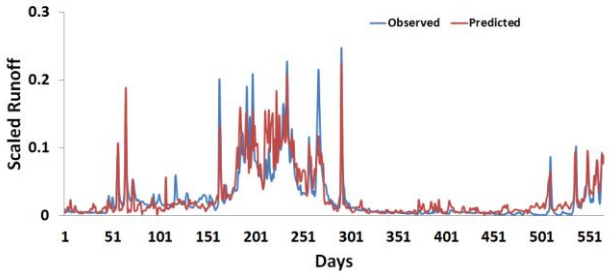

Figure 10. Beas Daily River Flow Modelling Results on Scaled Data Sets during Training Phase using ANN Model

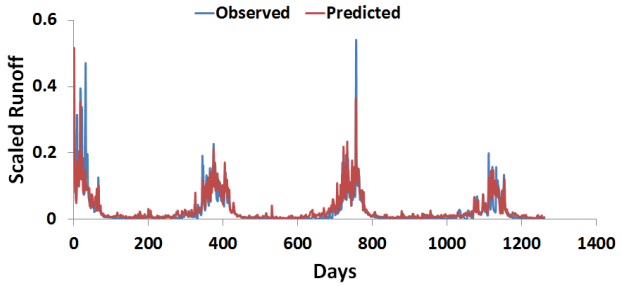

Figure 11. Beas Daily River Flow Modelling Results on Scaled Data Sets during Testing Phase using ANN model

8 STATISTICAL BLOCKADE

Statistical Blockade is an extension of extreme value distribution and its related theorem, in which we 'block' the values unlikely to fall in the low probability tails (hydrological peaks). Usually in rare event analysis approaches, , we may need to generate

enormous number of samples to obtain both samples and statistics for rare events. We normally use Monte Carlo (MC) simulation for efficiently simulating synthetic values based on the distribution. But the MC method is inefficient when we consider only the peak flood values (rare events) as MC still follows the complete distribution in generating synthetic samples. In contrast, Statistical Blockade uses a classifier (Support Vector classifier in our case) to filter out candidate MC points that will not generate values of our interest in the tail. In short, a SV classifier works in the tail portion of our distribution to pick the peak values (based on our threshold) from the MC generated values. In another way, we can say that this approach is partially similar to peaks over threshold (POT) method [generalized Pareto distribution (GPD) to the exceedances over some threshold]. In POT method, the used data are from historical record and not synthetically generated; whereas in statistical blockade 'synthetically' generated data are used [11]. A schematic representation of Statistical Blockade is given in the Figure 12 assuming that threshold value flood time series is at 80%.

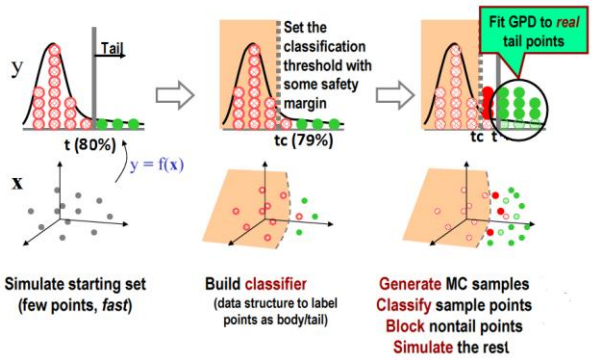

Figure 12. The Representation of Statistical Blockade (Modified from [12])

In our case study, the SB method was used to identify the occurrence of flood discharge events above a threshold value. The SB method produces MC simulated synthetic value input space which are likely to be fall in the tail region. The SVM based classifier 'blocks' the simulation in such a way that it won't generate input space below the user defined threshold value. It is accomplished by building the classification boundary at a classification threshold t_c that is less than the tail threshold t.

The four input space (wind speed, Humidity, Minimum Temperature and Precipitation) along with the observed Beas discharge values were used to train the Statistical Blockade with the SVM classifier which works by means of a radial basis function (RBF) kernel. As an example, the classified input space obtained from the Statistical Blockade at a threshold value of 90% is shown in the Figure 13. The blue dots are corresponding to the 'non-tail region' data and brown dots are corresponding to 'tail region' of the discharge data for a threshold value of 90%. The modelling capabilities of SB has been assessed by comparing with that of the ANN and SVM models in predicting the number of flood flows that could exceed a range of threshold values (Table 3).

Threshold limits	ANN	SB	SVM
90%	5%	4%	9%
80%	6%	7%	10%
70%	12%	9%	13%

Table 3. Percentage Changes in Number Flood Events Predicted by Different Models

The results show that the performance of Statistical Blockade is comparable to that of ANN model in predicting number of flood events falling above a threshold value at 90% and 80% limits. The SB has outperformed the ANN towards lower limits of the threshold. The high error values of SVM model can be related to the underestimation of the model during monsoon seasons; this highlights the facts that the SVM model requires much better tuning of its parameters for getting better performance.

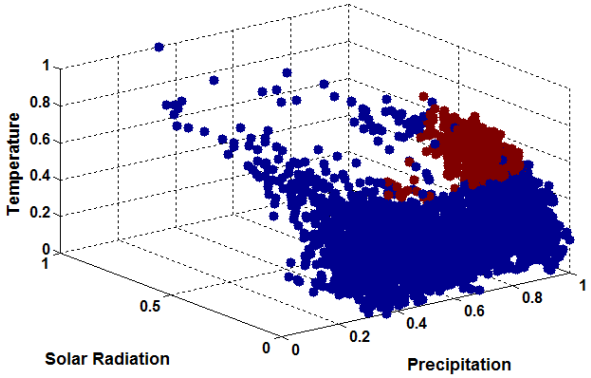

Figure 13. Classification of Monte Carlo Generated Samples at 90% threshold value [the brown colour shown flood values that falls in tail region]

An important observation that can be used is that the conditional distribution of the events in the tail region tend toward a generalized Pareto distribution (GPD). The generalized Pareto distribution CDF generated in our case study for different peak values is shown in the Figure 14 . This graph could generate meaningful information about the tail region of the discharge in the study area.

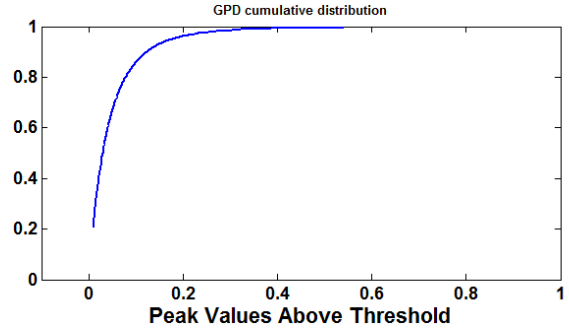

Figure 14. The CDF of generalized Pareto distribution (GPD) generated by SB on Scaled Data Sets .

9 CONCLUSIONS

Beas River daily discharge modelling is highly relevant in the region for hydroelectricity generation/planning, irrigation and flood prevention. This paper has introduced a rare event analysis method called Statistical Blockade to the field of hydrology and applied this approach to the Himalayan Beas river. The study has also explored the capabilities of the Gamma Test utilizing the daily data from the Beas catchment in conjunction with mathematical models like SVM, SB and ANNs and identified suitable inputs and training data length from the study region. An inter-comparison between ANN and SVM has shown that ANN is a superior model in the Beas basin in predicting peak floods especially in the monsoon season. Although this study made an attempt to optimize the SVM model, it requires further refinements to enable it to tackle monsoon flows. A comparison of SB approach with ANNs and SVMs have highlighted its capabilities in identifying the number of flood events above a given threshold value, utilizing extreme value theorem. This work has introduced the SB method in a single case study, but it has wider applications in the field of hydrology and we urge more hydrology studies to focus on it.

10 ACKNOWLEDGEMENTS

This work was partly funded by the UK-NERC (Project NE/I022337/1- Mitigating Climate Change Impacts on India Agriculture through Improved Irrigation Water Management, MICCI), as part of the Changing Water Cycle (South Asia) thematic programme. We would like to acknowledge the help extended Dr Sanjay Jain, NIH Roorkee.

REFERENCES

[1] S.K. Jain, J. Tyagi and V. Singh. Simulation of Discharge and Sediment Yield for a Himalayan Watershed Using SWAT Model. *J. Water Resource and Protection 2:* 267-281 (2010)

[2] A. Singhee and R. A. Rutenbar. Statistical Blockade: Very Fast Statistical Simulation and Modeling of Rare Circuit Events and Its Application to Memory Design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 28(8): 1176: 1189. (2011).

[3] S. Agalbjorn, N. KonHar and A.J. Jones. A note on the gamma test. *Neural Comput. Appl*. 5 (3), 131–133. (1997)

[4] P.J.Durrant. win Gamma: A non-linear data analysis and modelling tool with applications to flood prediction. PhD thesis, Department of Computer Science, Cardiff University, Wales, UK. (2001)

[5] D. Evans. Data Derived Estimates of Noise Using Near Neighbour Asymptotics. PhD Thesis, Department of Computer Science, University of Cardiff, UK. (2002).

[6] D. Evans and A. J. Jones. A proof of the gamma test. *Proc. R. Soc. Ser. A* 458 (2027), 2759–2799. (2002).

[7] V. Vapnik. The Nature of Statistical Learning Theory. Springer-Verlag, New York, 1995.

[8] C. C. Chang and C. J. Lin. LIBSVM : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1--27:27 (2011).

[9] W. S. McCulloch and W. Pitts. A logical calculus of the ideas imminent in nervous activity. *Bulletin of Mathematical Biophysics*. Vol. 5, 115–33 (1943).

[10] F. Rosenblatt. Priciples of neurodynamics: perceptrons and the theory of brain mechanics. Spartan. (1962).

[11] P. Embrechts, C. Klüppelberg, and T. Mikosch. Modelling Extremal Events for Insurance and Finance, 4th ed. Berlin, Germany: Springer-Verlag, (2003).

[12] A. Singhee, J. Wang, B. H. Calhoun and R. A. Rutenbar, Recursive Statistical Blockade: An Enhanced Technique for Rare Event Simulation with Application to SRAM Circuit Design, *Proceedings of the 21st International Conference on VLSI Design*, p.131-136

Optimisation of Water Treatment Works using Static and Dynamic Models with an NSGAII Genetic Algorithm

Roger Swan[1], John Bridgeman[1] and Mark Sterling[1]

Abstract. This paper applies a genetic algorithm to static and dynamic models of a case study water treatment works to find near optimal designs. The mechanisms of these models, their calibration and accuracy are described. The models were used with stochastic data representative of conditions observed at the works and the NSGAII genetic algorithm was applied to minimise the size of the works and the failure likelihood. The dynamic model was found to predict more conservative designs than the static model. The genetic algorithm was found to require greater calibration to identify near-optimal solutions efficiently.

1 INTRODUCTION

Traditionally, the uncertainties in water treatment works (WTW) design have been accounted for by empirical data or past experience instead of scientific understanding and rigorous empirical analysis. Computational modelling of WTWs can offer a means by which designers and operators can assess the likely impact of raw water quality, works design and operational conditions on final water quality. This helps to provide an indication of the likelihood of failure to meet water quality standards under conditions not previously experienced.

Existing commercial programs are available for WTW simulation but they do not have the ability to assess WTW reliability under stochastic conditions. General static WTW models were assessed by Gupta and Shrivastava [1] who found that stochastic conditions produced more conservative designs than deterministic conditions when a genetic algorithm was used to identify cost effective solutions that achieved performance goals.

This paper will compare the optimal designs identified by a genetic algorithm for a case study site modelled using static and dynamic models under stochastic conditions. Dynamic models are reasoned to be superior as key processes, such as filtration, are dynamic by nature.

2 CASE STUDY WORKS

At the works, water is abstracted from a lowland reach of river through bar and band screens before being impounded in a reservoir. Water to be treated is divided into two treatment streams, one of which has hopper bottomed clarifiers (HBC) and the other dissolved air flotation (DAF) clarifiers. In both streams the water has a ferric sulphate coagulant added before flocculation and clarification. Post clarification, the waters are blended together prior to being filtered through dual media (anthracite/sand) rapid gravity filters. The water then passes through a balance tank, to reduce the fluctuations in flow that are caused by the backwashing of the filters, before being treated by granular activated carbon (GAC) adsorbers.

Chlorine dosed upstream of the contact tank is controlled by a feedback loop that is dependent on the free chlorine concentration entering and exiting the contact tank. The works has a reported maximum capacity of 60 Ml/d (2500 m^3/h).

[1]Dept. of Civil Engineering, University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK
Email: {rxs884, j.bridgeman, m.sterling}@bham.ac.uk

3 MODEL DESCRIPTION

3.1 Turbidity suspended solids relationship

It was assumed that there is a 2:1 ratio between total suspended solids (mg/l TSS) and turbidity (NTU) as used in OTTER [2] and as suggested by Binnie et al. [3] where no other data are available.

3.2 Coagulation and flocculation

It is assumed that all of the iron in the ferric sulphate coagulant form iron hydroxide which precipitates out of solution. A stoichiometric calculation (see Equation 1) results in the amount of suspended solids added by ferric sulphate as being 1.9 g/g Fe^{3+}. This assumption is expected to be approximately true if TOC-coagulant formations are negligible, doses of coagulant are greater than 2 mg/l as Fe^{3+} and pH is such that iron has a low solubility (4.0 - 11.0) [4]. This calculation method agrees with that reported by Warden (1983) as reported by WRc [2] and Binnie et al.[3].

$$Fe^{3+} \Rightarrow Fe(OH)_3 \quad (1)$$

Flocculation is assumed to be effective resulting in flocs of an appropriate size for DAF treatment [5]. Other treatability characteristics of the water that are affected by coagulation and flocculation are taken into account by calibration constants in the process models.

3.3 HBC clarification

The static model uses a plug flow first degree decay equation to calculate the reduction in suspended solids. The dynamic model is similar to that presented in Head et al. [6] and uses the same calibrated parameters except for the flocculation efficiency factor. Others modifications of the Head et al. model are the use of two continuous stirred tank reactors (CSTRs) and the removal of the settlement removal mechanism, due to its insignificance.

3.4 DAF clarification

Removal of solids in the static model was calculated using a plug flow first degree decay equation as presented in Edzwald [7] for attachment in the contact zone. Complete removal of flocs attached to bubbles is assumed due to the high retention times observed.

The dynamic model uses a perfectly mixed model again based on Edzwald [7] where attachment occurs throughout the tank and again removal of flocs attached to bubbles is complete.

3.5 Filtration

The removal of solids by filtration was modelled in both models by the Bohart & Adams method [8]. Backwashing is assumed to be ideal removing all solids. Filter ripening is modelled using an initial exponential increase in the attachment coefficient as carried out in the OTTER WTW program [2]. The ripening period and initial attachment efficiencies were adjusted to be representative of works data.

In the dynamic model, clean bed headloss was calculated using the Kozeny-Carman equation with adjustment made for solids build up using the method presented in Adin & Rebhun [9]. The static model differs from the dynamic model by using an empirical headloss equation based on flow and run time, and uses scheduled backwashes only.

3.6 GAC

Due to lack of adsorption removal efficiency measurements recorded at the WTW, GAC adsorption has not been modelled. Suspended solids (SS) removal was not modelled as this was not the process's principle purpose at the works where it is a tertiary treatment. Dependency on GAC for SS removal would have resulted in an increased need to backwash which would disrupts the mass transfer zone and cause premature breakthrough of adsorbable pollutants [10].

3.7 Chlorination

Chlorine is dosed upstream of the contact tank and any instantaneous demand is assumed to have occurred prior to entry. Chlorine decay was modelled by first order decay. In the static model plug flow was assumed and in the dynamic model the flow through the contact tank was modelled as a number of CSTRs identified by the relationship between t_{10}:$t_{theoretical}$ efficiency and the number of representative CSTRs presented in Denbigh and Turner [11]. The t_{10}:$t_{theoretical}$ efficiency was found from tracer tests carried out by the works employees.

The magnitude of disinfection by-product formation is measured using trihalomethanes (THM) as a reference pollutant. A method which relates the formation of THMs to the consumption of chlorine is applied, as presented in Brown et al. [12], using a conservative formation parameter (50 µg/l THM per mg/l free chlorine consumed).

4 FAILURE CONDITIONS

The failure criteria identified were chosen to represent performance that the existing WTW occasionally experienced. Where applicable these conditions are stricter than legislative limits so the probability of them occurring is statistically more significant. The works failure likelihood is the fraction of the time that at least one of the failure criteria, shown in

Table **1**, were observed

Failure parameter	Failure condition
Blended turbidity post clarification	> 1 NTU
Filtered turbidity	> 0.1 NTU
Individual filter headloss	> 1m
Flow through filter	No flow
Free chlorine concentration multiplied by contact time (CT)	< 60 mg.min/l
Trihalomethanes (THM)	> 25 µg/l
Flow through DAF clarification	> 300 m^3/h
Flow through HBC clarification	> 200 m^3/h

Table 1. Failure conditions

5 ACCURACY OF MODELS

5.1 Works data

The models were both calibrated using quarter hourly data collected from in line sensors from the beginning of January 2012 (n=2604). Calibration values were found by minimising the summative error squared of output values when compared to works data. The accuracy of the models was validated by applying observed works values of river water temperature, flow through the works and reservoir turbidity for the end of January 2012 (n=2604). The accuracies of the models are comparable to quality assurance checks carried out and so are considered to be sufficient.

5.2 Stochastic data

5.2.1 Representative probability distribution functions and sampling rates of input data

To analyse the performance of the works using Monte-Carlo methods, probability distribution functions (PDFs) representative of the conditions observed in January 2012 had to be identified. The input parameters observed did not accurately approximate to parametric distributions, such as normal or log normal, and so they were described using non-parametric general distributions. The variation from daily average turbidity was also described in the same way.

The frequencies at which these distributions were sampled were chosen based on best visual representation of the observed works time series data as reported in

Table **2**.

Parameter	Sample rate	Samples in 10,000 hours
Combined DAF and HBC flow	4 hours	2500
Reservoir turbidity	168 hours (1 week)	60
Deviation from daily mean turbidity	4 hours	2500
River temperature	168 hours (1 week)	60
Contact tank inlet concentration	Quarter hourly	40000

Table 2. Sample rate of distributions

5.2.2 Modelled and observed output data

Probability distributions of outputs from the static and dynamic models were produced from a validation selection of works data and two stochastic (10000 hour) datasets. The modelled output distributions were analysed and found to be comparable to the observed distributions with the following notable observations. The static model was susceptible to high frequency variations in input parameters resulting in higher than observed variations in output parameters. This was due to plug flow assumptions. The dynamic model was found to dampen high frequency changes in raw water quality appropriately due to the application of CSTRs to represent partial mixing within processes. The effect of this partial mixing is particularly evident in the filter headloss and chlorine decay models. The dynamic and static models as calibrated overestimated the turbidity removal ability of the HBC and DAF processes.

5.2.3 Modelled and observed failure likelihood

The failure likeihood predicted by both models for observed and stochastic data was assessed and is shown in Figure 1. The lower than observed failure rates for both models is due to the over prediction of solids removal by the HBC and DAF processes. It can be seen that although the dynamic model did not experience any failure events when works data was applied that its prediction of failure events when using stochastic data was compariable to the static model and of the same magnituse as that observed. The dynamic model predicted failure due to headloss exceedance as was observed whereas the static model predicted failure due to unobserved inadequate disinfection (CT) and disinfection by-product formation (THMs) exceedance.

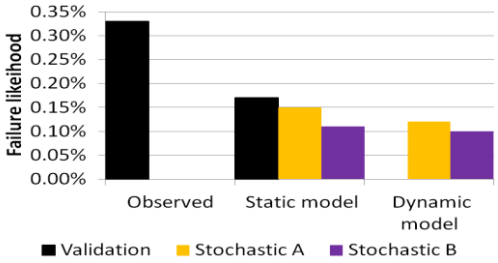

Figure 1. Failure likelihood

6 OPTIMISATION OF DESIGN

For operating conditions defined as being comparable to those observed at the case study WTW a near-optimal design was sought with the multiple objectives of minimum surface area and minimum failure likelihood. Five decision variables with ten or eleven possible integer values (0 or 1, to 10) were used.

6.1 Objectives and decision variables

Surface area was used as a cost function due to the lack of available data for costing WTWs based on their size. It was reasoned that where the same processes were used that a smaller WTW would equate to a lower cost design. The optimisation assumed that process units of the same size as those present at the case study works could be applied from typically a minimum of zero up to a maximum of ten units (Filtration and disinfection had a minimum value of one unit). For consistency of approach the flow ratio between DAF and HBC processes streams was variable in 10% increments and the size of the contact tank was variable in 400 m^3 increments up to a maximum of ten units. The performance of different designs was assessed by their likelihood of failure over 10000 hours using stochastically generated conditions as described in section 5.2. The optimisation was completed by using the NSGAII genetic algorithm over as many generations as was possible (for the more computationally more demanding dynamic model) on a single node (dual-processor 8-core (16 cores/node) 64-bit 2.2 GHz Intel Sandy Bridge E5-2660 worker nodes with 32 GB of memory) on the University of Birmingham's BlueBEAR computer cluster over 48 hours.

6.2 Genetic Algorithm

Genetic Algorithms (GAs) are the most popular group of techniques known as evolutionary algorithms. All of these techniques apply the theory of natural selection to identify near optimal solutions. A random group of initial parent solutions are produced and offspring solutions are produced by applying crossover and mutation functions. Together these parent and offspring solutions make up the first generation of solutions. These solutions have their performance assessed and then a second generation of parent solutions is identified. This process continues with the intention that as the number of generations increases the performance of the population should improve.

The Non-dominated sorting genetic algorithm II (NSGAII) [13], a type of second generation multi objective evolutionary algorithm (MOEA) was used to identify the near optimal solutions.This method finds and preserves the best solutions through the use of an elite preserving operator, application of a fast algorithm to sort non-dominated fronts and use of a two level ranking method to assign effective fitness to solutions. Solutions are first ranked by Pareto set and then by crowding so that dissimilar solutions are preferential. NSGAII was used due to its good diversity preservation in comparison to other GAs [13-14] and due to its ability to identify Pareto fronts in both constraint and non constraint problems (which allowed flexibility of approach). A drawback to the NSGAII is that it allows non dominated solutions to replace each other which over time can result in deterioration of the population's performance and therefore not guarantee convergence. The NSGA was programmed in Matlab, based on a description by Deb et al. [13]. The genetic operators applied were tournament selection, simulated binary crossover (SBX) [15] and polynomial mutation operators [16]. The NSGAII internal parameters used in this study are shown in Table 3. These have been chosen based on previous use by Sharifi [17] in a water based application and expert opinion [18] for the current problem. They will eventually be identified through a sensitivity analysis of their impact on quality of solutions identified.

Parameter	Value
Maximum number of generations	500
Population size	50
Crossover probability	0.9
Crossover distribution index	10
Mutation probability	0.05
Mutation distribution index	20

Table 3. NSGAII Internal parameters

7 RESULTS

The genetic algorithm was applied to both models for seven generations and the following graphs (Figure 2 and Figure 3) show the smallest design, for each generation, that failed less than 1% of the time when run for 10000 hours with stochastic data. Figure 4 shows the evolution of the entire population of solutions for the dynamic model. For the dynamic model, by the twentieth generation 78% of solutions were for the minimum size solution possible.

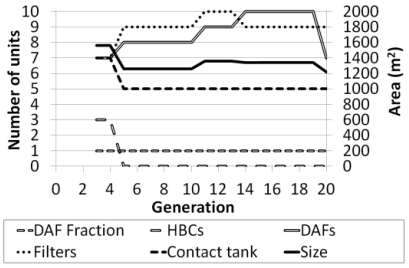

Figure 2. Evolution of near optimal static solution

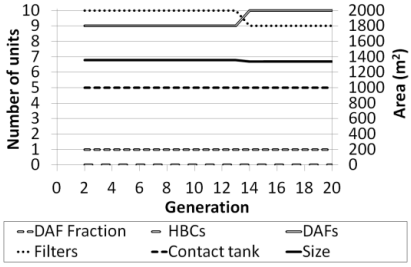

Figure 3. Evolution of near optimal dynamic solution

The static solution identified, after twenty generations, had a failure likelihood of 0.2% whereas the dynamic model experienced no failure events.

The computationally less demanding static model achieved 289 generations in the allotted time. The smallest solution identified in this time was 1180 m^2 which was not conserved.

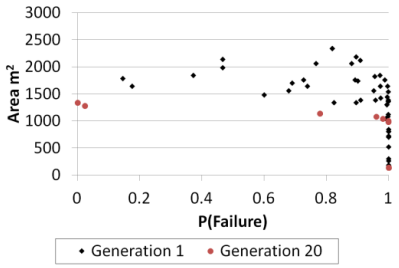

Figure 4. Populations of dynamic model solutions

8 ANALYSIS

After twenty generations the static model identified a smaller area solution (1220 m^2), with below 1% failure likelihood, than the dynamic model (1340 m^2). The smaller size of the static solution can partly be explained by its higher failure likelihood but the results are in agreement (in terms of relative sizes) with the best known solutions for this problem (static 1060 m^2, dynamic 1160 m^2). These results suggest that the dynamic model tends to identify more conservative estimates of works design than the static model.

The static model's smallest area solutions showed greater evolution throughout the GA's application than the dynamic model's. The small amount of evolution in dynamic model's solutions, the inability of the GA to identify the best known solution for the static model after 289 generations and the early dominance of minimal possible size solutions are indicative of premature convergence of the solution population.

Greater calibration of NSGAII internal parameters is required to ensure near-optimal solutions are identified for both models with acceptable computational demands.

ACKNOWLEDGEMENTS

We would like to thank the University of Birmingham for the funding of this research through the Postgraduate Teaching Assistantship Scheme.

REFERENCES

[1] Gupta AK, Shrivastava RK. Reliability-constrained optimization of water treatment plant design using genetic algorithm. *J Environ Eng-ASCE*. 2010;136(3):326-34.

[2] WRc. *WRc Otter version 2.1.3 user documentation*2002.

[3] Binnie, Kimber, Smethurst. *Basic water treatment*. 3rd ed: Royal Society of Chemistry; 2006.

[4] Tseng T, Edwards M. Predicting full-scale toc removal. *Journal / American Water Works Association*. 1999;91(Compendex):159-70.

[5] Edzwald JK, Walsh JP, Kaminski GS, Dunn HJ. Flocculation and air requirements for dissolved air flotation. *J Am Water Works Ass*. 1992 Mar;84(3):92-100.

[6] Head R, Hart J, Graham N, editors. Simulating the effect of blanket characteristics on the floc blanket clarification process. *Proceedings of the 1996 IAWQ/IWSA Joint Group on Particle Separation, 4th International Conference on the Role of Particle Characteristics in Separation Processes, October 28, 1996 - October 30, 1996*; 1997; Jerusalem, Isr: Elsevier Science Ltd.

[7] Edzwald JK. Chapter 6: Dissolved air flotation in drinking water treatment. In: Newcombe G, Dixon D, editors. *Interface science in drinking water treatment: Theory and applications*: Elsevier; 2006.

[8] Bohart GS, Adams EQ. Some aspects of the behavior of charcoal with respect to chlorine. *Journal of the American Chemical Society*. 1920 Mar;42:523-44.

[9] Adin A, Rebhun M. Model to predict concentration and head loss profiles in filtration. *J Am Water Works Ass*. 1977;69(8):444-53.

[10] Crittenden J, Trussell, Hand, Howe, Tchobanoglous, Harza MW. *Water treatment: Principles and design*: John Wiley; 2005.

[11] Denbigh KG, Turner JCR. *Chemical reactor theory: An introduction*: Cambridge University Press; 1971.

[12] Brown D, Bridgeman J, West JR. Predicting chlorine decay and thm formation in water supply systems. *Reviews in Environmental Science and Bio-Technology*. 2011;10(1):79-99.

[13] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*. 2002 Apr;6(2):182-97.

[14] Laumanns M, Thiele L, Deb K, Zitzler E. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary Computation*. 2002 Fal;10(3):263-82.

[15] Deb K. An efficient constraint handling method for genetic algorithms. *Comput Meth Appl Mech Eng*. [Article]. 2000;186(2-4):311-38.

[16] Subbaraj P, Rengaraj R, Salivahanan S. Enhancement of combined heat and power economic dispatch using self adaptive real-coded genetic algorithm. *Applied Energy*. 2009 Jun;86(6):915-21.
[17] Sharifi S. Application of evolutionary computation to open channel flow modelling: *University of Birmingham*; 2009.
[18] Anonymous Reviewer. *Malwas - submission review comments*. 2013. Received by Author 18/02/2013.

RAPIDS: Early Warning System for Urban Flooding and Water Quality Hazards

Andrew P Duncan[1], **Albert S Chen**[1], **Edward C Keedwell**[1], **Slobodan Djordjević**[1] **and Dragan A Savić**[1]

Abstract. This paper describes the application of Artificial Neural Networks (ANNs) as Data Driven Models (DDMs) to predict urban flooding in real-time based on weather radar and/or raingauge rainfall data. A time-lagged ANN is configured for prediction of flooding at sewerage nodes and outfalls based on input parameters including rainfall. In the absence of observed flood data, a hydrodynamic simulator may be used to predict flooding surcharge levels at nodes of interest in sewer networks and thus provide the target data for training and testing the ANN. The model, once trained, acts as a rapid surrogate for the hydrodynamic simulator and can thus be used as part of an urban flooding Early Warning System (EWS). Predicted rainfall over the catchment is required as input, to extend prediction times to operationally useful levels. Both flood-level analogue and flood-severity classification schemes are implemented. An initial case study using Keighley, W Yorks, UK demonstrated proof-of-concept. Three further case studies for UK cities of different sizes explore issues of soil-moisture, early operation of pumps as flood-mitigation/prevention strategy and spatially variable rainfall. We investigate the use of ANNs for nowcasting of rainfall based on the relationship between radar data and recorded rainfall history; a feature extraction scheme is described. This would allow the two ANNs to be cascaded to predict flooding in real-time based on current weather radar Quantitative Precipitation Estimates (QPE). We also briefly describe the extension of this methodology to Bathing Water Quality (BWQ) prediction.

Keywords. ANN, early warning system, flood risk, machine learning, neural network, nowcasting, prediction, rainfall, urban flood.

1 INTRODUCTION

Recent studies [1], [2] have documented the increased frequency and likelihood of extreme precipitation events. In the UK, the existing installed base of combined drainage systems is huge. This means that a large proportion of urban rainfall runoff is immediately mixed with effluent, increasing the potential public health risks from urban flooding. Even flooding from separate storm sewers is in any case destructive and costly. An ageing network and increasing urbanisation further exacerbate these problems. Therefore models are required, which can provide predictions of location, severity and/or risk of flooding. In order to be operationally useful, these need to provide 2+ hour lead-time [3] and be able to operate rapidly in real-time.

Hydrodynamic simulators are used as standard to model the response of Urban Drainage Networks (UDNs) to rainfall events. However, especially for large UDNs, these can be slow and computationally expensive. A faster surrogate method is sought, which would permit modelling of very large networks in real-time, without unacceptable degradation of accuracy. However, if actual rainfall is used as input, the predictive ability of such models is limited by the Time of Concentration (ToC) for the sewer network, with the possibility of flooding at any node commencing from zero time onwards, following the start of precipitation. In practice, ToC would normally be of the order of minutes, rather than hours for all but the downstream sections of the very largest UDNs.

Therefore prediction of rainfall is a requirement to achieve the lead-times sought. Many papers have been written on rainfall nowcasting methods from radar rainfall images [3–11]. A novel machine-learning based approach to this is currently at an early stage of development within the Centre for Water Systems.

2 APPROACH USED ('RAPIDS')

As part of University of Exeter's research under Work Package 3.6 of the Flood Risk Management Research Consortium Phase 2 (FRMRC2) [12] project , we developed the 'RAdar Pluvial flooding Identification for Drainage System' (RAPIDS) using ANN's to predict flooding in sewer systems. This was described in our paper [13] and was further developed for an UKWIR-funded joint industry / University of Exeter project [14] in which three case studies were carried out for UDN's in South London, Portsmouth and Dorchester, with promising results.

The RAPIDS software (currently in MATLAB) includes two programs: RAPIDS1, which addresses the need for a faster surrogate for hydrodynamic simulators as well as classifier models for flood and other hydrological parameters, and RAPIDS2 (under development), which aims to provide nowcasting for rainfall over the catchment containing the modelled UDN. It is hoped to be able to demonstrate the cascading of these two systems to provide the required urban flood predictive model.

The RAPIDS1 program is based on a lagged-input, 2-layer, feedforward Artificial Neural Network (ANN), used to relate incoming rainfall data to the extent of flooding present at each node in the UDN. It has the same number of output neurons as sewerage nodes of interest – i.e. there is no requirement to model nodes identified from hydrodynamic modelling as never flooding, making an immediate computational saving. The ANN architecture is varied to establish an optimum. The supervised training regime uses either backpropagation of error quasi-Newton gradient-descent or NSGA-II [15] Evolutionary Algorithm method. A moving time-window approach is implemented whereby lagged time-series signals (e.g. rainfall intensity, cumulative rainfall, soil moisture, pump states, tidal levels etc) are provided in parallel over the time-window as inputs to the ANN. If no direct observation data is available for the UDN to be modelled, output target signals for training and evaluation of ANN model performance are provided from the flood-level, volume or flow hydrographs generated by

[1] Centre for Water Systems, University of Exeter, Harrison Building, North Park Road, Exeter EX4 4QF, UK. Email: {apd209, A.S.Chen, E.C.Keedwell, S.Djordjevic, D.Savic}@exeter.ac.uk

hydrodynamic simulator outputs for each sewerage node to be modelled. This only needs to be done for the training dataset of rainfall events. The trained ANN thus aims to generate the same hydrographs for new rainfall events as would the UDN itself, based on having learned and generalised the (non-linear) relationship between the provided input signals and observed or simulator-generated targets. Figure 1 illustrates the architecture of the RAPIDS1 system to predict sewer network outputs. The target signals selected are the flood levels at each sewerage node at a time-step that corresponds to the desired prediction lead-time (i.e. up to network ToC).

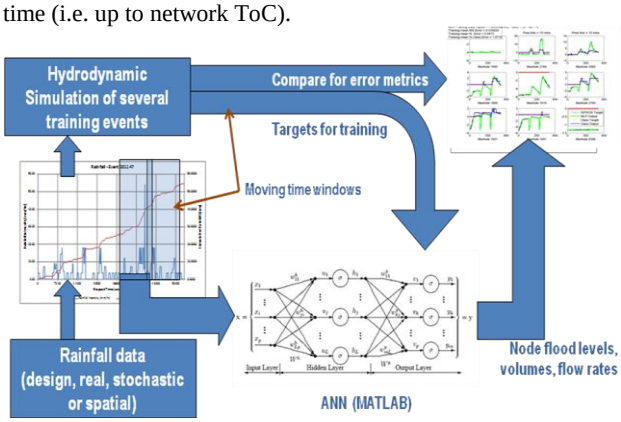

Figure 1. Architecture of RAPIDS1

Event profile data arrays of the input-signals are prepared for use as the time-series input to the ANN as illustrated in Figure 2. In line with best practice, all input data are normalised.

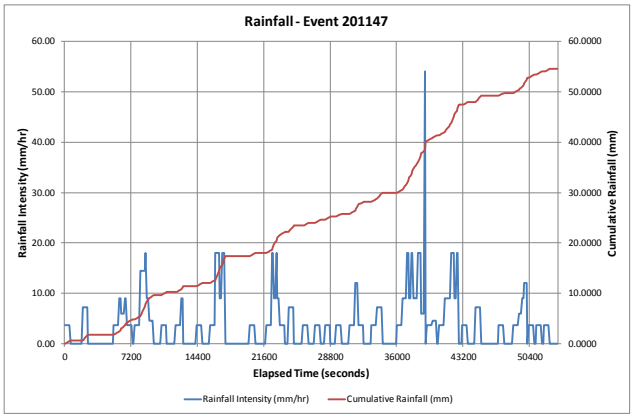

Figure 2. Selected ANN Input signals for a typical rainfall event

A selection of (historic) rainfall events is needed for the training dataset. These need to be representative of the envelope of likely intensities and rainfall totals for the future events to be modelled. If sufficient of these are not available, existing events can be augmented by factorally increasing rainfall intensity and modelling resulting target hydrographs using a hydrodynamic simulator.

Rainfall radar images are sourced from the UK Met Office NIMROD system [16], [17], which produces a composite 1km resolution Quantitative Precipitation Estimate (QPE) image covering the whole UK, every 5-minutes. A live RSS feed is available on request. Historic data images (from April 2004 to present) are available for download from [18]. Treatment of radar QPE images 1km pixel-by-pixel by an ANN is computationally prohibitive since, for example, for a 3-h prediction there would be 36-images, each with at least 3602-pixels (allowing for a maximum storm advection velocity of 60 km/h). This would potentially require ~5×10^6 neurons (at 1-neuron per pixel). Therefore features are extracted from the rain echoes in each time-step and associated with features from previous time-steps. These can then be applied to the inputs of an ANN as time-series signals. The feature extraction approach proposed is similar to Discrete Wavelet Transforms (DWT) using Haar wavelets [19], but using different sized grids depending on the proximity to the catchment being modelled. The mean rainfall for the whole area is evaluated; then residuals of mean rainfall over each sub-grid square are computed: see Figure 3. Standard deviations show that information is contained at all spatial scales [20].

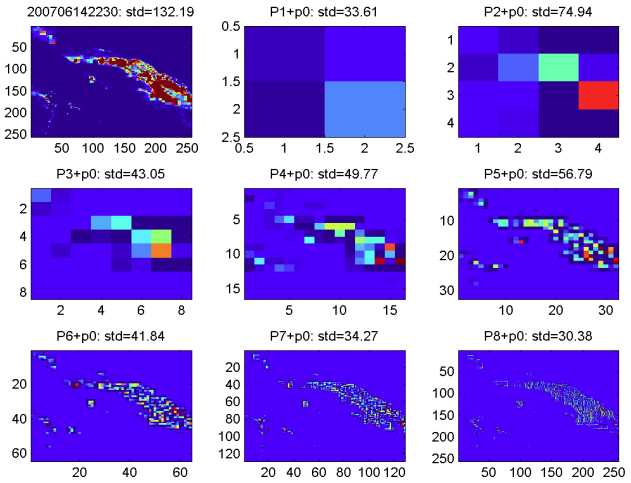

Figure 3. RAPIDS2: Rainfall Event 2007-06-14 – QPE snapshot at 22:30 showing original image (top left) and feature extraction of residuals at finer grid resolutions (128 to 1 km)

The extracted residuals from multiple images over the duration of each event become time-series signals, which can be applied as input signals to ANNs: see Figure 4.

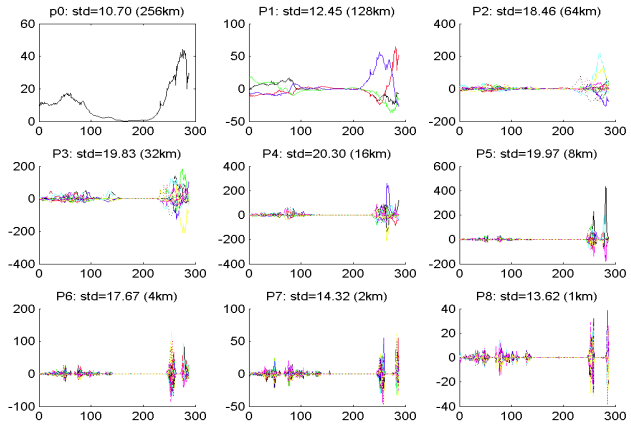

Figure 4. RAPIDS2: Rainfall Event 2007-06-14 – Time-series ANN input signals over 24-hours at spatial resolutions as shown; x-axis is radar image no.; y-axis is Δ rainfall intensity in mm/hr.

It is proposed to implement a similar time-windowed ANN framework as for RAPIDS1. Target rainfall for training and evaluating the ANN is derived from the rainfall intensities in grid squares covering the required catchment containing the UDN to be modelled, advected into the future by the required prediction period.

In summary, the proposed methodology is to cascade the two stages together (RAPIDS2 providing predicted rainfall, which can be applied to RAPIDS1 inputs) and thus provide flood predictions for each node of interest in the UDN, hopefully with operationally useful lead-times of 2+ hours.

3 CASE STUDIES

An initial "proof-of-concept" case study for RAPIDS1 was conducted as part of FRMRC2. An ANN with 123-outputs was used to model the Stockbridge sub-section of the combined rain/wastewater drainage system for the town of Keighley, West Yorkshire, containing 122 manholes and one combined sewer overflow (CSO). Design rainfall was used. The neural network gave a floating-point estimate of the level of flooding at each node. However, this level of accuracy is unlikely to be required for flood-warnings. Therefore a classification scheme to provide predictions of flood severity was implemented by post-processing ANN outputs. Results were reported in [13].

Under the UKWIR-funded joint-industry Real-time Machine Learning (RTM) project [14] the following 3 case studies were implemented, in a two-stage project to evaluate effectiveness in different sized catchments under different conditions; stage 1 used design rainfall and stage 2 used real rainfall:

Dorchester: small urban catchment (6km^2); evaluation of the significance of use of soil moisture as ANN input.

Portsmouth: medium urban catchment (30km^2); island location; tidal effects; need for pumping; evaluation of effectiveness of ANN models to provide early starting of pumps – as a flood-mitigation / prevention strategy.

Crossness (South London): large urban catchment (230km^2); evaluation of model effectiveness using spatially varying rainfall as ANN inputs.

In order to allow all partners to present results consistently, the MS Excel-based 'HydroMAT' model analysis tool was developed to provide automated assessment of ANN output using a number of metrics[2] including those recommended in [21]. Results below (Figures 7-9) were assessed using this tool.

4 RESULTS & DISCUSSIONS

Figure 5 shows average ANN training times of around 115 seconds for the 123-node network used in the FRMRC case-study. Fifteen 6-hour events (rainfall + runoff) were used for training. In comparison, hydrodynamic simulation for each took approx 240 seconds (total 3600 seconds). Once the ANN was trained, however, test run times were of the order of 0.1 seconds for each 6-hour event (Figure 6). Figures 7-9 illustrate the reporting of metrics provided by the HydroMAT tool; Figure 7 shows a typical spread of NSEC values over a 20-node sample for a single test rainfall event; Figure 8 compares ANN-generated hydrograph with the target hydrograph for a single node for a single test event; Figure 9 shows flood severity classification matrix for peak flood depths for a 20-node sample for a single event. This compares target classifications (rows A to C) with ANN-generated classifications (columns A to C). It also shows a colour-coded assessment of 3 'Accuracy bands'.

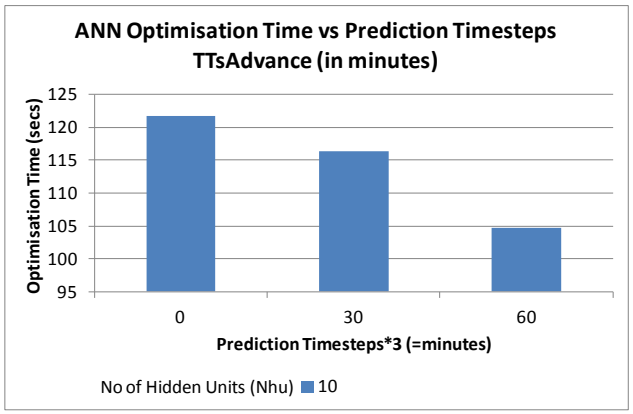

Figure 5. RAPIDS1 – typical 123-node ANN training times for FRMRC study.

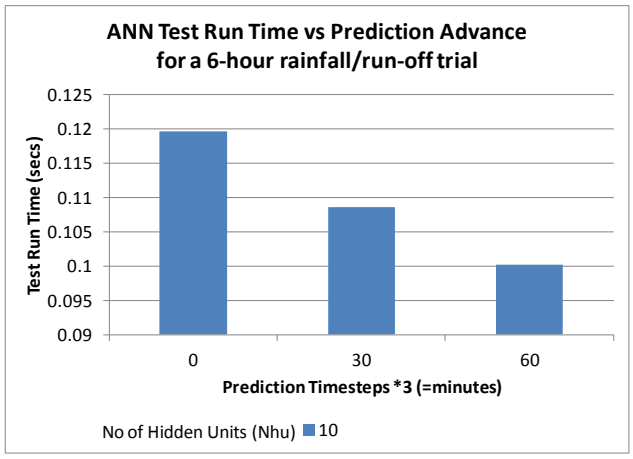

Figure 6. RAPIDS1 – typical 123-node ANN test times for FRMRC study.

In summary, results for UKWIR case studies demonstrated the following:

(Dorchester): Use of soil moisture levels (NAPI) as ANN input demonstrated a small improvement in model performance, but this was probably not sufficient to offset additional costs of data gathering, preparation and application to ANN model.

[2] Nash-Sutcliffe Efficiency Coefficient (NSEC); RMSE-Observations Standard Deviation Ratio (RSR) ; Percentage Bias (PBIAS) ; Total Volume Error (TVE) ; ANN Normalised Root Mean Square Deviation (NRMSD); % Samples in Limits - All Nodes; Amplitude Error of Hydrograph Peak ; Timing Error of Hydrograph Peak; R-Squared - All Nodes; Pearson Correlation Coefficient - All Nodes; ANN Output vs Target X-Y Plot (ATXY) - Single Node; ANN Output & Target Hydrographs - Single Node; Confusion Matrix for Peak Flood Depth Categories; Confusion Matrix for Flood Positives & Negatives; Confusion Matrix Accuracy Band summary analysis

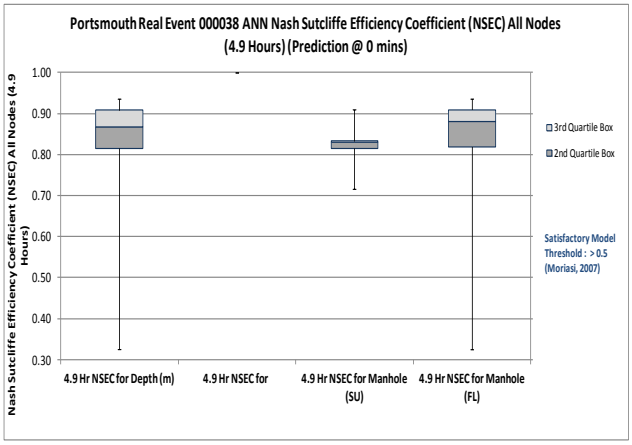

Figure 7. RAPIDS1 – typical spread of ANN output NSEC scores over 20-nodes for a single real rainfall event (Portsmouth case study)

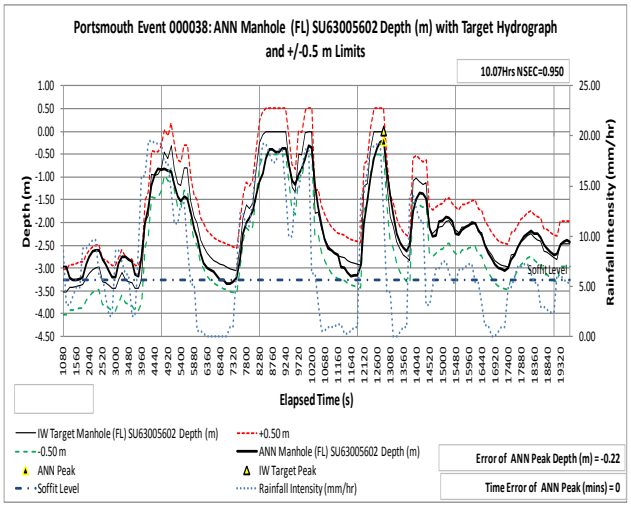

Figure 8. RAPIDS1 – typical target hydrograph and ANN response for a single manhole and rainfall event (Portsmouth catchment)

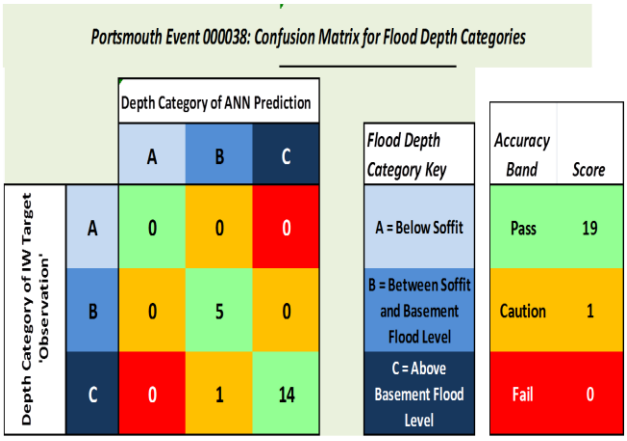

Figure 8. RAPIDS1 – typical classification matrix for three peak flood depth categories (A|B|C) at 20-sewer nodes, for a single rainfall event (Portsmouth catchment). Colour-coded accuracy bands for all nodes are also shown.

(Portsmouth): Use of ANN models were demonstrated successfully to prevent flooding in the 'Morass' area of Portsmouth, when used as a trigger for early initiation of pumping at the Eastney pumping station.
(Crossness): Results for the entire 230km^2 catchment using 23 raingauges as ANN input were poor. Spatial rainfall input worked best when applied to smaller areas (4-5 raingauges subcatchments). Further work is needed.

Work on RAPIDS2 rainfall nowcasting is at too early a stage to present results beyond those shown in Figures 3-4 for the proposed feature extraction approach; the methodology is still under development.

5 CONCLUSIONS & FUTURE WORK

Results for RAPIDS1 show that ANNs can provide a very significant speed improvement over conventional hydrodynamic simulators without excessive degradation in performance. They can moreover be used for flood severity classification. The RAPIDS1 method presents opportunities for automated generation of flood alarms / warnings right down to the individual sewer node, including potentially for networks of considerable size, without being computationally expensive.

However, flood prediction based on actual rainfall alone cannot provide operationally useful lead-times. Instead, prediction is limited in the worst case by the ToC for each node (typically <30 min). However, possibilities for stand-alone use of ANNs for rainfall nowcasting are being explored through a process of radar rainfall echo feature extraction and feature time-series prediction using ANNs (RAPIDS2). More work is needed to determine the value of this approach.

Extending prediction time to operationally useful values of 2+ hours could potentially be achieved by using Met Office rainfall prediction products in place of RAPIDS2.

Assuming that RAPIDS2 achieves satisfactory results, the possibility of cascading the two systems to provide flood-level prediction at manholes based on live radar rainfall images will be tested.

The RAPIDS1 package has been written to allow tailoring to other catchments and water-related EWS requirements to be readily achieved. At present a version of RAPIDS1 is being adapted to early warning of bathing water quality exceedances to comply with the EU directive [22], using a variety of ANN input parameters; principally antecedent rainfall over the catchment.

Acknowledgment. The research reported in this paper was conducted as part of the FRMRC2 and the UKWIR RTM projects, with support from the Engineering and Physical Sciences Research Council, the Department of Environment, Food and Rural Affairs/Environment Agency Joint Research Programme, UK Water Industry Research, Office of Public Works Dublin, and Northern Ireland Rivers Agency. Data were provided by the British Atmospheric Data Centre, Environment Agency, Halcrow, HR Wallingford, Met Office, Mouchel, Ordnance Survey, Richard Allitt Associates and Yorkshire Water. Our thanks go to all the above organisations for their support.

REFERENCES

[1] S.-K. Min, X. Zhang, F. W. Zwiers, and G. C. Hegerl, "Human contribution to more-intense precipitation extremes," *Nature*, pp. 378–381, 2011.

[2] P. Pall, T. Aina, D. A. Stone, P. A. Stott, T. Nozawa, A. G. J. Hilberts, D. Lohmann, and M. R. Allen, "Anthropogenic greenhouse gas contribution to flood risk in England and Wales in autumn 2000," *Nature*, pp. 382–386, 2011.

[3] T. Einfalt, K. Arnbjerg-Nielsen, C. Golz, N.-E. Jensen, M. Quirmbach, G. Vaes, and B. Vieux, "Towards a roadmap for use of radar rainfall data in urban drainage," *Journal of Hydrology 299*, pp. 186–202, 2004.

[4] L. Li, W. Schmid, and J. Joss, "Nowcasting of Motion and Growth of Precipitation with Radar over a Complex Orography," *Journal of Applied Meteorology*, vol. 34, no. 6, pp. 1286–1300, Jun. 1995.

[5] W. F. Krajewski and J. A. Smith, "Radar hydrology: rainfall estimation," *Advances in Water Resources*, vol. 25, no. 8–12, pp. 1387–1394, Aug. 2002.

[6] N. E. Bowler, C. E. Pierce, and A. W. Seed, "STEPS: A probabilistic precipitation forecasting scheme which merges an extrapolation nowcast with downscaled NWP," *Quarterly Journal of the Royal Meteorological Society*, vol. 132, no. 620, pp. 2127–2155, 2006.

[7] S. Achleitner, S. Fach, T. Einfalt, and W. Rauch, "Nowcasting of rainfall and of combined sewage flow in urban drainage systems," *Water Science and Technology*, vol. 59, no. 6, p. 1145, 2009.

[8] A. N. A. Schellart, M. A. Rico-Ramirez, S. Liguori, and A. J. Saul, "QUANTITATIVE PRECIPITATION FORECASTING FOR A SMALL URBAN AREA: USE OF RADAR NOWCASTING," in *8th INTERNATIONAL WORKSHOP on PRECIPITATION IN URBAN AREAS*, St Moritz, CH, 2009, pp. 22–26.

[9] P. Wang, A. Smeaton, S. Lao, E. O'Connor, Y. Ling, and N. O'Connor, "Short-Term Rainfall Nowcasting: Using Rainfall Radar Imaging," *Eurographics ireland*, p. pp, 2009.

[10] S. THORNDAHL, T. Bøvith, M. R. Rasmussen, and R. S. Gill, "On comparing NWP and radar nowcast models for forecasting of urban runoff," in *Proceedings of IAHS symposium held in Exeter, UK, April 2011*, Exeter, UK, 2011, vol. 351, pp. 620–625.

[11] S. Liguori, M. A. Rico-Ramirez, A. N. A. Schellart, and A. J. Saul, "Using probabilistic radar rainfall nowcasts and NWP forecasts for flow prediction in urban catchments," *Atmospheric Research*, vol. 103, no. 0, pp. 80–95, Jan. 2012.

[12] "Flood Risk Management Research Consortium 2." 2012-2009.

[13] A. Duncan, A. S. Chen, E. Keedwell, S. Djordjevic, and D. A. Savic, "Urban flood prediction in real-time from weather radar and rainfall data using artificial neural networks," in *IAHS Red Book series no. 351*, Exeter, UK, 2011, vol. 351.

[14] R. Kellagher, "The Use of Artificial Neural Networks (ANNs) in Modelling Sewerage Systems for Management in Real Time: Volume 1 - UKWIR Main Report (12/SW/01/2)." UKWIR, 2012.

[15] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *Evolutionary Computation, IEEE Transactions on*, vol. 6, no. 2, pp. 182–197, 2002.

[16] B. W. Golding, "Nimrod: a system for generating automated very short range forecasts," *Meteorological Applications*, vol. 5, no. 1, pp. 1–16, 1998.

[17] N. BADC, "Met Office - Rain radar products (NIMROD)." http://badc.nerc.ac.uk, 2011.

[18] N. BADC, "BADC UKMO Nimrod Data," *British Atmospheric Data Centre*. 2013-2004.

[19] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. 41, no. 7, pp. 909–996, 1988.

[20] I. Tchiguirinskaia, D. Schertzer, C. T. Hoang, and S. Lovejoy, "Multifractal study of three storms with different dynamics over the Paris region," in *Proceedings of Weather radar and hydrology symposium, Exeter, UK*, 2011.

[21] D. N. Moriasi, J. G. Arnold, M. W. Van Liew, R. L. Bingner, R. D. Harmel, and T. L. Veith, "Model evaluation guidelines for systematic quantification of accuracy in watershed simulations," *Transactions of the Asabe*, vol. 50, pp. 885–900, 2007.

[22] European Commission, "Revised Bathing Water Directive (2006/7/EC)." European Commission, 15-Feb-2006.

Dynamically Integrated Project Portfolio Planning to Accommodate Asset Management Plan Cycles in the UK Water Industry

Alireza Pakgohar[1], David Z. Zhang[1], Sarah Ward[1]

Abstract. Project Portfolio Planning (PPP) is a hierarchical decision-making process that has been frequently studied in a number of sequential phases such as Project selection, contractor/resource selection, Project scheduling and rescheduling and Supply chain configuration. This paper proposes an integrated approach for modelling and simulating PPP systems to evaluate the behaviour of the portfolio as quickly as possible in response to demand changes and desires of counterpart supply chain. In this work an agent based architecture, is developed, wherein each portfolio resource is represented by an agent. Using autonomous agent methodology enables the resources to manage themselves efficiently and effectively. The agent-based modelling and interaction approach enables a multi layer supply chain of contractors to be allocated to the selected projects dynamically in an optimal manner. By using an agent based discrete event simulation, project selection, resource allocation and project scheduling can occur simultaneously. This paper introduces a novel methodology for integration of agent-based modelling, multi project planning/scheduling, rescheduling, identification of alternative portfolio counterpart supply chain configurations, simulation and analysis of the configurations within an integrated framework for UK water companies in particular, in order to accommodate the Asset Management Plan (AMP) cycle.

Keywords: Asset Management Planning, Project portfolio planning, Agent based modelling, Genetic algorithm

1 INTRODUCTION

Regulation and investment in the water sector in the UK is undertaken in 5 year cycles, known as the Asset Management Plan (AMP) cycle. Traditionally, the majority of investment by water companies is made at the beginning or towards the middle of the AMP cycle, as companies strive to achieve regulatory compliance by the end of the AMP cycle. Capital outlay then decreases as the cycle enters its final stage [1]. The oscillating cycles is adapted from ofwat's report [2] and depicted in Figure 1. It also covers an approved projected cycle plan.

[1]College of Engineering, Mathematics and Physical Sciences, University of Exeter, North Park Road, EX4 4QF, Exeter, UK
{A.Pakgohar, D.Z.Zhang,Sarah.Ward}@exeter.ac.uk

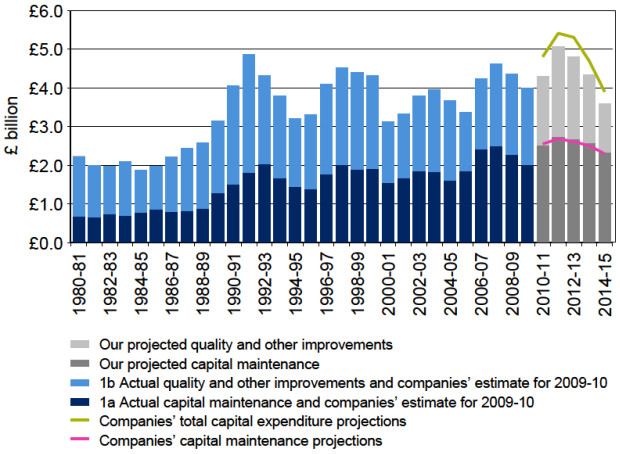

Figure 1. The AMP Cycle in the UK water sector [2]

Consequently, the AMP cycle has been criticised for creating a 'boom and bust' cycle, where expertise is recruited at the beginning of a cycle, once tenders have been won, and laid off at the end of a cycle [1], [3], [4] . This results in not only the loss of jobs, skills and experience (as employees move to the international job market), but also in a cyclical pattern of huge recruitment, laying off and retraining costs (in the region of £600M). Additional costs arise from inefficiencies in maintaining overheads, purchasing, stock control and borrowing. This subsequently impacts production, innovation and diversification due to uncertainties in planning, supply chain management and long-term business practices [4].

One of the major challenges for water companies is how to react rapidly and cost-effectively to dynamic variations in demand patterns derived from AMP cycles: in essence, how do they 'smooth' the impact of the cycle? Moreover changes in capabilities, capacities and desires of multi layer Water suppliers/contractors is another challenge in each AMP Cycle that causes turbulence in water specialist renewable resources. Therefore, the most difficult parts of the project portfolio planning processes in water companies and across different layers of their counterpart supply chain are related to how they can make strategic, tactical and operational decisions. These decisions include selecting contractors and configuring the supply chain, collaborating with them to achieve a plan/schedule for each individual project and rescheduling them if it is needed. Furthermore, project selection and resource restructuring are two major interrelated issues that first tire suppliers such as Atkins

and Balfour Beatty are involved with. However, in the long term, the other issue that will be raised in these companies to accommodate AMP is system adaptation. This adaptation could be partially done by developing training programmes for human resources as well as supply reconfiguration for external resources.

At the operational level, while each project manager is asked to select pre-assessed contractors and provide a schedule of the work packages based on the collaboration between traders in a decentralised manner, in tactical and strategic level, principal managers are facing the problem of how a decentralized decision could affect overall performance of the Water Company. How are the contractors affected by these sorts of decisions? What is the impact of these decisions on the total supply chain and its reaction to the market demand derived from AMP?

This paper sets out an investigation to address and highlight these challenges and design a framework for decision making processes. The aim is to understand how portfolio managers could dynamically and cost-effectively optimize the enterprise in order to cope with dynamic variations in demand patterns of the projects and the unpredictable reactions of the traders in tendering processes across the supply chain. It will also investigate how they could achieve partnership agreements among traders and what the effect would be on an enterprise's clients.

2 RESEARCH QUESTIONS

1. At all levels of the water supply chain hierarchy – both prime and sub-contractors: How to consider decisions for rescheduling and reconfiguration of resources across the hierarchies of a distributed project supply chain concurrently, with minimum change to existing contracting structures, to cope with unexpected delays in project progress?

2. How to design and optimise a project supply chain that is robust to changes and uncertainties (lead-time uncertainties, unexpected delays and events in the supply and logistic network) whilst being able to satisfy quality and delivery requirement at minimum cost, both at the contract formation and resource reconfiguration stages?

3. How to identify the optimum time and option for upgrading resource and skills matrix in a project oriented business in response to changing needs from the market? How does this feed into the human and organisational decisions of the business?

3 LITRATURE REVIEW

Complex project portfolio planning (PPP) has been a crucial part of a wide range of studies for more than 6 decades. According to the necessity of having a powerful model to cope with diverse aspects of PPP many scholars and practitioners proposed several hierarchical planning frameworks for decision making in PPP [5], [6],[7], [8].

The framework that has been presented by Hans et al. [6] includes operational, tactical and strategic levels of decision making. Figure 2 depicts this framework.

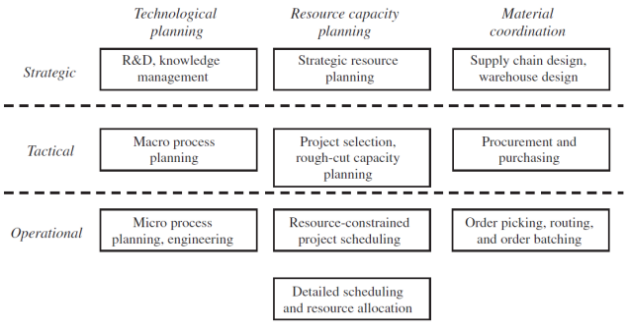

Figure 2. Integrated hierarchical multi-project planning [6]

At the operational level of this framework, there are ample studies for Resource-constrained project scheduling and resource allocation. There are two different approaches used by research undertaken in this area, including centralized and decentralized decision-making methods. The former approach has been considered across more than six decades. Pritsker et al. [9] presented the generic model of scheduling projects by introducing a super network consisting of multiple projects, Vercellis [10] proposed a "Lagrangean decomposition approach" for solving the PPP in multi-mode conditions. She relaxed two groups of constraints that interrelate among themselves across the projects, by introducing two set of multipliers and presented Lagrangean relaxation of the resource constrained multi project scheduling problem (RCMPSP). Gonçalves et al. [11] Proposed a heuristic approach for modelling and solving a multi project problem. In the stream of the centralized modelling, some scholars consider each single project among the portfolio independently so-called truly multi-project modelling. Kurtulus and Davis [12] could be considered the pioneers of modelling multiple projects by individual networking. They showed that this kind of modelling outperforms super network modelling. Further, Kumanan [13] proposed the use of a heuristic and a genetic algorithm (GA) for scheduling a multi-project environment. They consider multiple projects the activities of which can be performed in one of several modes where for performing an activity there are several options available. We were unable to find out any other papers in multi-mode multi-project planning and scheduling problem.

State-of-the-art approaches have started to consider the use of agent-based modelling and simulation, which has been applied in a range of research areas [14], [15]. Using the notion of agent based modelling, Knotts [16] for the first time proposed agent-based project scheduling. Under the umbrella of this idea, some scholars have modelled PPP by using a multi agent system as a part of artificial intelligence systems. The concept has been developed into that of decentralized multi project planning. In fact, the idea of decentralizing multi-project scheduling problem can be identified in 2003 when Lee et al. [17] claimed that, as a result of large improvement in technology of Internet and globalization of the business, multi-project firms performed more distributed organisationally and geographically. They believed that centralized project management, where all the projects are managed by a single manager, is no longer suitable for this environment. Therefore, they established the term "decentralized or distributed multiple projects (DMP)" environment and proposed the Decentralized Multi-project Scheduling Problem (DMPSP). DMPSP is a complex

combinatorial approach, which employed Multi-Agent Systems (MAS) to simulate the genuine multi-project problem. It was a distributed approach based on informational and geographical aspects in project portfolio organizations. Following this research, Confessore et al. [18] illustrated a Decentralized Resource Constrained Multi-project Scheduling Problem (DRCMPSP).

The models proposed by Lee et al. [17] and Confessore et al. [18] are based on modern electronic auctions for resource allocation and simple heuristics for scheduling activities.

Homberger [19] introduced a restart evolution strategy (RES) - a meta-heuristic approach- that could find the solution for resource constrained scheduling problem centrally as well as a MAS – that could solve DRCMPSP (solving the problem decentrally). He showed that "*decentralized MAS approach is competitive with a central solution using the RES*." Araúzo et al. [20] proposed a model for project selection and scheduling based on MAS as an integration of tactical level and operational level of decision making in project portfolio environment. This shows that MAS is a capable methodology for modelling and covering several layers of hierarchical planning in multi-project environments. However, as far as could be identified from the literature reviewed, none of the research on utilising a decentralized approach take into consideration multi-mode case as well as the dynamic nature of the portfolio [21].

At the strategic layer of Hans et al.'s framework [6], supply chain designing has been addressed. There is a number of studies in the field of project portfolio supply chain design/optimisation [22], [23], [24], [25]. Furthermore, Xue et al. [26, 27] introduced Multi Agent Systems for the construction supply chain in particular. These research aimed to facilitate communication between different parties involved in an Engineering, Procurement and Construction (EPC) enterprise. Although the research in the field of supply chain design indicate that the planning and scheduling of the project plays a critical role in all the processes of the project based organizations, it has not covered the integration of project planning with supply chain design and configuration.

In summary, it can be concluded that there is limited research that devoted to integration of different aspects of hierarchical PPP.

4 PROPOSED FRAMEWORK FOR PPP

In this research, we proposed an MAS for PPP that is able to cope with multi-project planning in multi mode condition and each decision is made by individual project manager in a decentralized way rather than taking all the decisions by a single portfolio manager. In addition, projects enter the portfolio in a dynamic way. Furthermore, in conformity with Araúzo et al. [20] and beyond, our proposed MAS architecture could address all three layers of decision making framework of Hans et al. [6] in an integrated way such that, while in the operational layer, decisions of resource allocations and project scheduling are made, in the tactical layer, the architecture could accommodate the decisions such as project selections and resource restructuring decisions. Finally, the decisions regarding long term resource planning such as supply chain reconfiguration and partnership agreement decisions will be accommodated at the strategic layer where multi layers of the framework work together as an integrated and dynamic system. Figure 3 depicts our dynamic integrated framework that has been adopted to demonstrate its applicability to the Water Supply Chain in the water industry in the UK.

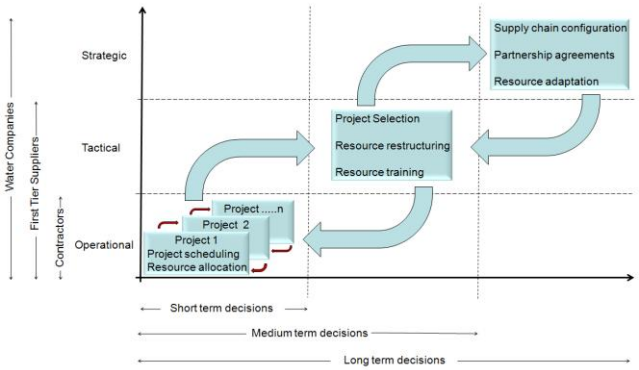

Figure 3. Dynamic Integrated PPP framework

5 PPP AGENT-BASED ARCHITECTURE

The concept is to simulate the behaviours of a complex project portfolio enterprise along with all of its internal and external resources in its counterpart supply chain by implementing a multi agent-system model. The model could enable enterprise managers to concurrently generate and evaluate alternative scenarios of planning and control using an agent-based bidding process. This framework and its architecture are able to integrate decision making processes at operational, tactical and strategic levels. The role and behaviour of the project and portfolio managers in this framework will be investigated with regard to contractor selection, communication to achieve planning, scheduling and rescheduling, if required.

Our proposed integrated decision platform would enable changes to be captured and scheduling, planning, configuration, restructuring and adaptation options to be created virtually and evaluated concurrently in a co-ordinated manner by continuous interactions amongst individual supply chain activities. This platform is based on a multi agent system to simulate and also enable the entire Water Supply Chain, including design and consultancy, construction and manufacturing companies that are involved in AMP cycles, to be optimised dynamically in an integrated way.

The architecture of the proposed MAS has been adapted from previous studies in the Exeter Manufacturing Enterprise Centre (XMEC) [28] [29], [30], to capture the feature of the PPP is demonstrated in Figure 4.

The major aim of the work presented in this paper is to provide a cost-effective approach that would enable the Water Industry to integrate and coordinate planning and control operations and improve awareness and responsiveness to the changes derived from AMP cycle.

For scheduling of each project, a coordinated iterative bidding process, inspired by the negotiation process between sellers and buyers has been proposed and tested in XMEC [21], [28].

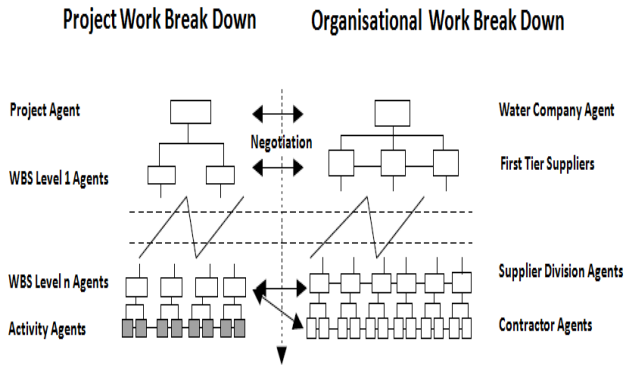

Figure 4. PPP agent network architecture

This process provides an optimal combination of resources for a project by implementing several bidding iterations. The process starts when a particular project with corresponding activities comes to the portfolio dynamically. All of the resources are informed, through their blackboard, when a new project enters the portfolio. The resources update their local times and their previous activities in their buffers according to the current time. Then the project agent initializes virtual prices for activities and minimum virtual profits for all resources. Iterative loops can then be started for achieving an optimal resource combination for conducting the project. For performing each activity of a project agent, resource agents evaluate corresponding cost and if the cost is more than the allocated virtual price, which means that there is no profit for them, they do not participate in the bidding process. However, if there is a reasonable profit, which means that the virtual minimum profit has been satisfied, they put forward the bid in the auction process. Therefore, in performing each activity there are several possible bids at each iteration. Having received bids from resources where their virtual profits have been satisfied, the project manager collects all of the bids for activities, evaluates the makespan of the project and chooses the best combination if and only if the collective plan satisfies the project deadline. The cost of the project is then calculated and stored in its blackboard in the first iteration. In the next iteration, virtual prices and minimum profits are updated by a GA that has been developed at the University of Exeter [28]. This is to encourage more resources to put forward their bids. It is clear that, by increasing virtual prices more resources will be involved in auction and vice versa. Therefore time-cost trade off will be evaluated at each iteration and virtual parameters updated to achieve the minimum cost that satisfies the due date of the project. In other words, the virtual prices for activities of each project and minimal virtual profits for a portfolio's resources act as governor for the bidding process, so that each resource achieves its local goal while the global goal of the portfolio agent may be achieved by calculating the minimum cost of each project.

The procedure is finished when a given number of iterations have been run. Finally, the project manager agent selects the minimum cost and its associated combination of resources.

The platform is implemented by a multi-layer hierarchical agent-based modelling and simulation architecture for modelling complex heterogeneous systems i.e. organisation along with its multi-tiers suppliers/contractors as well as Project work breakdown structure. The agent-based architecture facilitates the implementation and the execution of a hierarchical and optimally controlled agent-based bidding process including a method for identifying, simulating and evaluating system restructuring options to accommodate changes in AMP cycles. This approach attempts to improve the performance of the Water Industry and facilitate adaptation to the changes in the AMP Cycles. In this way, changes can be utilised as opportunities for further improvement and adaptations.

6 CONCLUSIONS

A dynamic integrated framework for hierarchical PPP has been proposed. Using MAS allows the system to simulate the water supply chain environment and find out the effect of each AMP transition. Based on the amount of turbulence in the water industry, what-if analyses and reconfiguration options could be evaluated to optimize the best supply chain configuration (long term period). In this framework, optimal project planning - as short term decisions - could be achieved by coordination between project agents and contractor agents. These agents negotiate collaboratively with each other to achieve a project schedule where minimum cost and no due date violation occurs. In other words, this paper introduces an evolutionary way of optimising distributed the water supply chain where Engineering, Procurement and Construction resources involved in projects are coordinated to offer best resource planning and supply chain structuring simultaneously.

7 ACKNOWLEDGEMENT

The authors wish to sincerely thank Arthur Thornton from Atkins Limited and Chris Binnie from Chartered Institution of Water and Environmental Management (CIWEM) who kindly shared their ideas and experiences and providing useful insights with regard to connecting XMEC's expertise to the UK Water Industry.

REFERENCES

[1] G. Pontin. Water: have we learned the lessons from AMP 4? In: *Impact, Magazine of the Association for Consultancy and Engineering.* March/April, p. 11.
http://www.acenet.co.uk/Documents/Files/Impact/Impact%2036/Water%2036.pdf [Accessed: 13-Jan-2013], (2011).

[2] Ofwat. Future water and sewerage charges 2010-15: Final determinations. Birmingham , [Online]. Available: http://www.ofwat.gov.uk/pricereview/pr09phase3/det_pr09_finalfull.pdf [Accessed: 13-Jan-2013], (2010).

[3] HM Threasury, Smoothing investment cycles in the water sector. *London*. (2012).

[4] P. Mullord. Maintaining a resilient and sustainable industry supply chain. In: *Institute of Water Annual Conference*. 17th May (2012).

[5] M. Speranza and C. Vercellis. Hierarchical models for multi-project planning and scheduling. *European Journal of Operational Research*, 64: 312–325, (1993).

[6] E. W. Hans, W. Herroelen, R. Leus, and G. Wullink. A hierarchical approach to multi-project planning under uncertainty," *Omega*, 35, no. 5:563–577, (2007).

[7] A.Can and G. Ulusoy. Multi-project Scheduling with 2-Stage decomposition. *Working Report, Sabanci University*. (2010).

[8] K. K. Yang and C. C. Sum. A comparison of resource allocation and activity scheduling rules in a dynamic multi-project

environment," *Journal of Operations Management*, 11: 207–218, (1993).

[9] A. Pritsker, B. Allan, L. J. Watters, and P. M. Wolfe. Multiproject scheduling with limited resources: a zero-one programming approach. *Management Science*, 16:93–108, (1969).

[10] C. Vercellis. Constrained multi-project plannings problems: A Lagrangean decomposition approach. *European Journal of Operational Research*. 7, (1994).

[11] J. F. Gonçalves, J. J. M. Mendes and M. G. C. Resende. A genetic algorithm for the resource constrained multi-project scheduling problem. *European Journal of Operational Research*. 189, no. 3: 1171–1190, (2008).

[12] I. Kurtulus and E. Davis. Multi-project scheduling: Categorization of heuristic rules performance. *Management Science*. 28, no. 2:161–172, (1982).

[13] S. Kumanan, G. Jegan Jose, and K. Raja. Multi-project scheduling using an heuristic and a genetic algorithm. *The International Journal of Advanced Manufacturing Technology*. 31, no. 3–4: 360–366, (2006).

[14] M. Wooldridge, *An Introduction to Multi-Agent Systems*, John Wiley & Sons Ltd., England, U.K., (2002).

[15] M. j. North and M. M. Charles, *Managing Business Complexity, Disccovering Strategic Solutions with Agent-Based Modeling and Simulation*, Oxford University Press, Inc. New York, (2007).

[16] G. Knotts, M. Dror, and B. C. Hartman. Agent-based project scheduling. *IIE Transactions*, 32, no. 5:387–401, (2000).

[17] Y. Lee, S. Kumara, and K. Chatterjee. Multiagent based dynamic resource scheduling for distributed multiple projects using a market mechanism. *Journal of Intelligent Manufacturing*, (2003).

[18] G. Confessore, S. Giordani, and S. Rismondo. A market-based multi-agent system model for decentralized multi-project scheduling. *Annals of Operations Research*, 150, no. 1:115–135, (2006).

[19] J. Homberger. A multi-agent system for the decentralized resource-constrained multi-project scheduling problem. *International Transactions in Operational research*. 14:565–589, (2007).

[20] J. A. Araúzo, J. Pajares, and A. Lopez-Paredes. Simulating the dynamic scheduling of project portfolios. *Simulation Modelling Practice and Theory*, 18, no. 10:1428–1441, (2010).

[21] A. Pakgohar and D. Zhang. Dynamic multi-mode multi-project scheduling problem: an agent-based approach. *Proceeding of 17th International Working Seminar on Production Economics*, Innsbruck, Austria, 1: 407–418, (2012).

[22] J. Gosling and M. M. Naim. Engineer-to-order supply chain management: A literature review and research agenda. *International Journal of Production Economics*. 122, no. 2:741–754, (2009).

[23] C. Hicks, T. McGovern, and C. Earl. Supply chain management: A strategic issue in engineer to order manufacturing. *International Journal of Production Economics*. 65, no. 2: 179–190, (2000).

[24] K. T. Yeo and J. H. Ning. Integrating supply chain and critical chain concepts in engineer-procure-construct (EPC) projects. *International Journal of Project Management*, 20, no. 4:253–262, (2002).

[25] K. T. Yeo and J. H. Ning. Managing uncertainty in major equipment procurement in engineering projects. *European Journal of Operational Research*. 171, no. 1: 123–134, (2006).

[26] X. Xue, X. Li, Q. Shen, and Y. Wang. An agent-based framework for supply chain coordination in construction. *Automation in Construction*. 14, no. 3:413–430, (2005).

[27] X. Xue, Y. Wang, Q. Shen, and X. Yu. Coordination mechanisms for construction supply chain management in the Internet environment. *International Journal of Project Management*, 25, no. 2:150–157,(2007).

[28] D. Z. Zhang, A. Anosike, and M. K. Lim. Dynamically Integrated Manufacturing Systems (DIMS)- A Multiagent Approach. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS*, 37, no. 5: 824–850, (2007).

[29] A. I. Anosike and D. Z. Zhang. Dynamic reconfiguration and simulation of manufacturing systems using agents. *Journal of Manufacturing Technology Management*. 17, no. 4:435–447, (2006).

[30] D. Z. Zhang, A. I. Anosike, M. K. Lim, and O. M. Akanle. An agent-based approach for e-manufacturing and supply chain integration. *Computers & Industrial Engineering*, 51, no. 2:343–360, (2006).

Modelling dynamic systems using a hybrid approach

Mateja Skerjanec[1], **Darko Cerepnalkoski**[2], **Saso Dzeroski**[2], **Boris Kompare**[1] **and Natasa Atanasova**[1]

Abstract. In this work, we present a hybrid approach to modelling dynamic systems, which combines a typical conceptual modelling approach with machine learning, enabling the construction of process-based models tailored to specific modelling requirements. The key element of the proposed methodology is a domain specific library capable of storing background knowledge about the modelling task. Here we present two such libraries, namely an aquatic ecosystem modelling library and a watershed modelling library. Both libraries are written in a formalism compliant with the equation discovery tool ProBMoT, which can automatically construct models from the components in the library, given a conceptual model specification and measured data. We applied the proposed modelling methodology to the Lake Bled to induce a phytoplankton model, and to the Ribeira da Foupana catchment to extract a semi-distributed hydrologic model. Finally, we present the outlook for the integration of both libraries under a common framework.

1 INTRODUCTION

Models of dynamic systems are often formulated in terms of basic processes that govern the dynamic behaviour of the observed system. Each basic process influences the change of one or more system variables, while the model of a basic process specifies the equations used to model its influence.

Equation discovery (ED) is an area of machine learning (ML) that aims to automatically discover a model stated as equations from measured data. In contrast to the conceptual, knowledge-driven approach, where the modeller formulates the equations based on theoretical principles, i.e., composes "white-box" models, ED, unless hybridized with some knowledge introduction procedures, composes so called "black box" models, where the structure of the model is not transparent and clear.

Quite a few successful attempts have been made towards the introduction of domain knowledge into the procedure of model discovery from data. Langley et al. [1] proposed a formalism, which uses generic processes to present the general domain modelling knowledge and specific processes to present the specific modelling task. This formalism is supported by the IPM system [2]. Todorovski and Dzeroski [3] introduced the concept of grammars in the system LAGRAMGE, where grammars were used to introduce the domain knowledge into the ED procedure and, in that way, limit the search space of all possible solutions. The development proceeded towards more general knowledge representation. Dzeroski and Todorovski [4] introduced the formalism for modelling knowledge libraries supported by LAGRAMGE 2.0. Using this formalism, Atanasova et al. [5] developed a library for modelling food webs in aquatic ecosystems, which was successfully applied for discovering models of lake ecosystems for the lakes Glumsø [6] and Kinneret [7].

ProBMoT [8] is a further development of the above systems in terms of the knowledge representation formalism, as well as, support for more advanced model optimization procedures. Using the ProBMoT formalism, two modelling libraries were developed, i.e., the upgraded library for aquatic ecosystems, initially introduced by Atanasova et al. [5], and a catchment modelling library developed anew.

In this paper, we present both libraries and their use in the automated modelling (AM) tool ProBMoT for model discovery. Although different in their structure, the libraries can be used within the same framework, which opens the potential of their integrative use, i.e., for integrating catchment models with aquatic food web models. Bellow we present the AM methodology, the libraries, two applications and conclusions.

2 AUTOMATED MODELING METHODOLOGY: ProBMoT

ProBMoT [8] automatically generates a set of viable process-based models of the system under study by using a library of domain knowledge and a user-specified conceptual model, which constrain the space of candidate model structures. The candidate models are transformed to equations and calibrated against measurements to obtain the best model of the system, i.e., the model that fits the measurements best (see Figure 1).

The background knowledge about the modelling task is captured into a domain specific library. The library represents a repository of template components, namely entities and processes that serve as building blocks for ProBMoT. Entities correspond to the actors of the observed system, while processes define the relationships among them. Each template component is specified as one data structure, which has its own unique name and a set of properties. The main goal of the templates is to capture some general knowledge that applies to many different entities or processes and can be reused when dealing with different specific tasks.

In order to apply the AM methodology to a specific case study, a conceptual model of the observed system has to be provided. The basic elements of the conceptual model are instances – specific entities and processes that follow the templates encoded in the library. From the AM point of view, the conceptual model represents a set of constraints that any specific model must obey. In contrast to the theoretical modelling approach, where a conceptual model represents a completely determined system with fully specified system variables and processes that relate these variables, here we can provide either a

[1] University of Ljubljana, Faculty of Civil and Geodetic Engineering, Slovenia. Email: {mateja.skerjanec, boris.kompare, natanasa.atanasova}@fgg.uni-lj.si

[2] Jozef Stefan Institute, Department of Knowledge Technologies, Slovenia. Email: {darko.cerepnalkoski, saso.dzeroski}@ijs.si

complete or an incomplete conceptual model of the observed system.

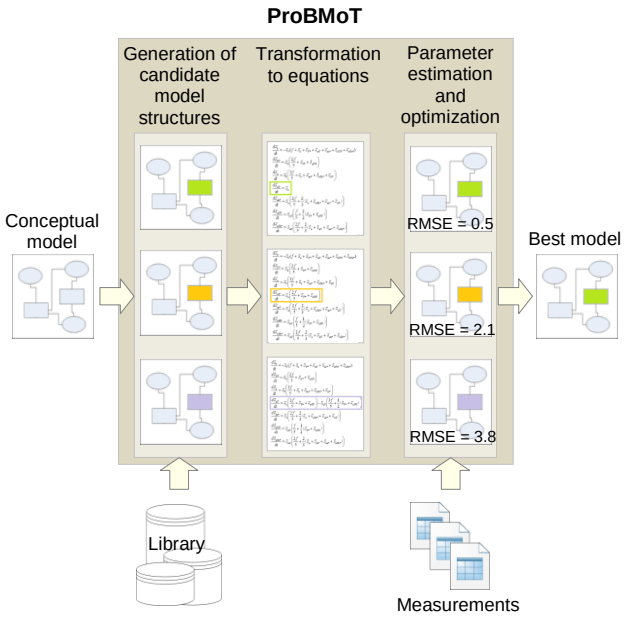

Figure 1. The proposed automated modelling methodology

In the case of complete conceptual model, we completely specify the entities and processes between them, defining a unique conceptualization of the system. At the level of mathematical models, each of the processes in the conceptual model can be described with different formulations, resulting in various mathematical models generated from a single complete conceptual model.

The proposed AM methodology allows the specification of a model beyond the level of a conceptual model. It can go to the level of mathematical formulations of the processes, and further, to the specification of the process parameters. In the case of complete specification of the mathematical formulations of the processes, the search space of candidate models is reduced to a single model structure. Furthermore, if the user provides the exact parameter values, the AM tool can skip both structure and parameter identification, and proceed directly to model simulation.

An incomplete conceptual model can represent multiple conceptualizations of the system, each contributing to the generation of at least one mathematical model. The incompleteness of the conceptual model can be determined at two levels. The processes that relate the variables can be given as general high-level process, where various process conceptualizations (with different complexities) can fulfil the requirements. The other option is to include an "empty" process among the alternatives, indicating that the process doesn't exist. In both cases, the AM tool needs to perform structure identification and parameter calibration.

This level of specification of the conceptual model directly influences the number of generated candidate models to be later optimized against measured data. The more the conceptual model structure is defined, the smaller the number of candidate models.

The model structure identification task is presented in Figure 1. In the first stage, using the components from the library, the equation discovery tool ProBMoT generates candidate model structures that adhere to the conceptual model. In the second stage, each model structure is translated into a set of algebraic and/or ordinary differential equations. The last stage consists of estimating the numerical parameters of the equations.

The numerical parameters are estimated as to minimize the discrepancy between the observed data and the model simulation. ProBMoT supports different objective functions for quantifying the discrepancy, such as root mean squared error (RMSE). Differential evolution [9], which is a non-linear metaheuristic optimization method, is used for estimating the parameters.

The result of the AM procedure is a list of candidate models with fully specified structure and parameter values, ranked according to their RMSE values.

3 DOMAIN LIBRARIES

Domain libraries are the key element to the entire AM procedure. In this section, we present two libraries for two related domains, i.e., a library for aquatic ecosystem modelling and a library for watershed modelling.

3.1 Aquatic ecosystem library

The aquatic ecosystems library comprises typical food-web modelling processes, such as nutrient uptake, growth, grazing, mortality, respiration, etc. The generic scheme of the processes and their relations with the system variables is given in Figure 2.

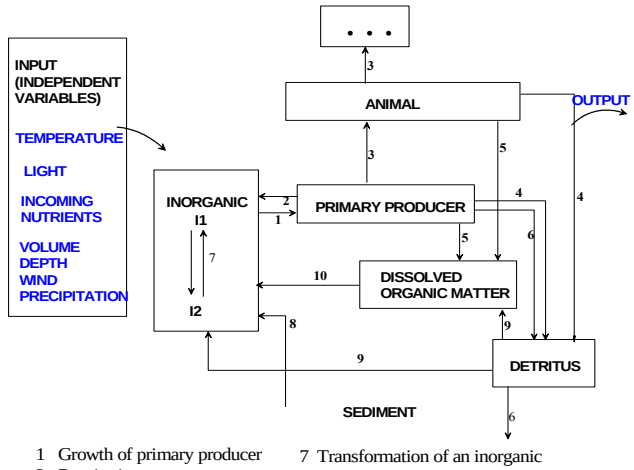

Figure 2. Generalized scheme of state variables (boxes) and relations or processes (arrows) in aquatic ecosystems, as captured in the modelling knowledge library

Each process is encoded with its alternative formulations. For example, the growth process has three alternatives: exponential,

logistic or food-limited equation. Furthermore, the limitation functions in the food-limited equations are formulated with several alternatives, such as the Monod model or the exponential limitation model. For more details, please refer to Atanasova et al. [5] and Cerepnalkoski et al. [8].

3.2 Watershed modelling library

The watershed modelling library represents the key element for automatic generation of semi-distributed watershed models. For the initial setup of the library, a watershed modelling concept similar to the one introduced by Haith and Shoemaker [10] was used, because it offers an acceptable level of complexity, taking into account all the basic watershed processes. The library allows us to sufficiently model hydrologic and constituent generation processes on a watershed scale (see Figure 3). The entities encoded in the watershed modelling library correspond to different pools within the water cycle, climate variables and various types of constituents (namely nutrients and sediment). Examples of processes include water fluxes, i.e., transfer processes that are involved in the water cycle, and constituent loadings.

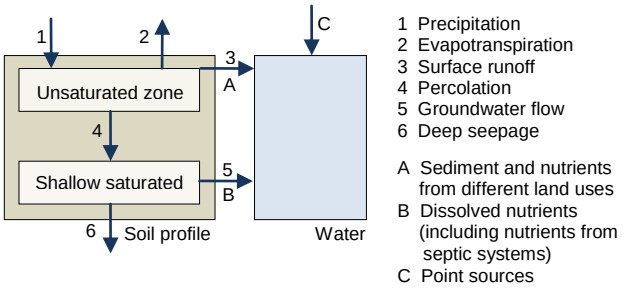

Figure 3. Processes (arrows) encoded in the watershed modelling library. Numbers indicate hydrologic processes, while letters are used to symbolize different constituent loadings.

4 APPLICATIONS

4.1 Lake Bled

The aquatic ecosystem library was used to induce a phytoplankton model for the Lake Bled. Lake Bled is a typical subalpine lake of glacial-tectonic origin. It occupies an area of 1.4 km^2 with a maximum depth of 30.1 m and an average depth of 17.9 m. The data set about the lake (obtained from the Slovenian Environmental Agency) comprises measurements of physical, chemical and biological parameters from 1995 to 2002 with a monthly frequency. The data used for modelling are as follows: temperature, light, dissolved inorganic nutrients in the lake (phosphorus, nitrogen and silica), total phytoplankton biomass, and the biomass of zooplankton species Daphnia hyalina.

Previous work on this data set included phytoplankton model discovery with LAGRAMGE 2.0 [5], which was successful in identifying different model structures for each year of the observed period, but failed to optimize a single structure for the entire period.

The model induction with ProBMoT was performed through the following steps: (1) preparing the domain knowledge about the lake by specifying a conceptual model (2) preparing the measured data for the optimization procedure, and (3) collecting and simulating the "best" model.

The conceptual model was prepared by specifying the entities for which we have measurements and the conceptual top-level processes appropriate for modelling phytoplankton dynamics: growth, respiration, mortality, sedimentation, and grazing by zooplankton. The entities involved are Nitrogen, Phosphorous, Silica, Phytoplankton and Zooplankton.

Given the data set and the modelling task specification ProBMoT improved the accuracy of the phytoplankton models previously discovered with LAGRAMGE 2.0 [8]. Further experiments are being developed towards the discovery of a model that would successfully simulate the entire period (not just one year), as well as to the discovery of a food-web model including equations for nutrients, phytoplankton and zooplankton. In Figure 4, we present the simulation of the model discovered on the data from year 2000.

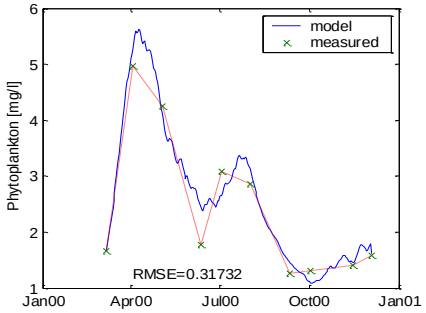

Figure 4. Simulation of the model discovered from the year 2000 data

4.2 Ribeira da Foupana

The watershed modelling library was used to generate a semi-distributed hydrologic model of the Ribeira da Foupana catchment, covering an area of 140 km^2. The study area included three experimental subcatchments, each composed of several functional units characterized by homogeneous land use. Altogether, seven functional units were considered for the entire catchment.

The conceptual model was prepared by specifying the entities and processes for each sub-compartment, corresponding to a single functional unit. Given the conceptual model and the watershed modelling library, the search algorithm generated 128 hydrologic models for the selected study area. This number of models was obtained because all seven sub-compartments could use one of the two alternative formulations for the calculation of the potential evapotranspiration (PET). Consequently, a number of alternatives (i.e., two) had to be raised to the power of the

number of sub-compartments (i.e., seven), resulting in 128 different candidate models.

Afterwards, the search space of candidate model structures was limited by introducing an additional constraint to the conceptual model, allowing the application of uniform evapotranspiration model structure for the entire experimental catchment. This resulted in just two candidate models, namely one using the Hammon PET equation [11] and the other using the Hagreaves PET equation [12] in all seven sub-compartments.

For the optimization (calibration) phase of the AM procedure, the daily values of the following independent variables were provided for each sub-compartment: precipitation, minimum, maximum and average temperature, solar radiation, daylight hours and saturated vapor pressure. The above mentioned data were obtained from the SNIRH[3] database for the meteorological station Malfrades and for the period of December 22, 1999 to December 21, 2000. During the optimization phase, the selected parameters (namely curve number, cover coefficient, groundwater recession constant and seepage constant) were automatically calibrated against measurements based on a comparison of the calculated outflows from the selected study area to the flows measured at the hydrological station Tenencia.

For each model the RMSE values were generated. The model with the lowest RMSE value was selected as the best hydrologic model for the selected study area. As it turned out, the model that used the Hargreaves equation (RMSE 3.625) performed slightly better than the model that used the Hammon equation (RMSE 3.665). Figure 5 shows the best model simulation results, namely the comparison between the measured and calculated outflows from the selected study area.

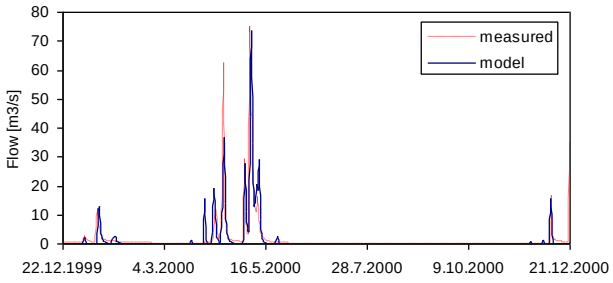

Figure 5. Simulation results: comparison between measured and calculated flows at the Tenencia hydrological station

5 CONCLUSIONS & FUTURE WORK

This paper demonstrates the use of domain modelling knowledge (in form of libraries) in the procedure of automated model discovery. Two related modelling domains, i.e., aquatic food web modelling and watershed modelling were encoded into libraries and integrated in the AM tool ProBMoT. The tool was successfully applied in two domains, i.e., the Lake Bled for discovery of phytoplankton model (part of a food web) and the catchment of Ribeira da Foupana for discovery of a hydrologic model.

Future work will be focused on the improvement of the existing models, i.e., the discovery of a long term phytoplankton equation and a more complete food web model in the case of Lake Bled. In the case of Ribeira da Foupana we foresee the calibration of the nutrient loading model. Further, options for integration of both libraries into a common framework will be explored and demonstrated on additional case studies, including related catchment and aquatic ecosystem, where the catchment model shall provide an input for the aquatic ecosystem food web.

[3] http://www.snirh.pt

ACKNOWLEDGEMENTS

The first author is part-financed by the European Union, European Social Fund, under contract number P-MR-08/23.

REFERENCES

[1] P. Langley, J. Sanchez, L. Todorovski, and S. Dzeroski. Inducing process models from continuous data. In: *Procs. of 19th ICML*, Sydney, Australia. pp. 347-354, (2002).

[2] W. Bridewell, P. Langley, L. Todorovski, and S. Dzeroski. Inductive process modeling. *Machine Learning*, **71**, 1-32, (2008).

[3] L. Todorovski and S. Dzeroski. Declarative bias in equation discovery. In: *Procs. of 14th ICML*, Nashville, TN, USA. pp. 376-384, (1997).

[4] S. Dzeroski and L. Todorovski. Learning population dynamics models from data and domain knowledge. *Ecol. Model.*, **170**, 129-140, (2003).

[5] N. Atanasova, L. Todorovski, S. Dzeroski, and B. Kompare. Constructing a library of domain knowledge for automated modeling of aquatic ecosystems. *Ecol. Model.*, **194**, 14-36, (2006).

[6] N. Atanasova, L. Todorovski, S. Dzeroski, and B. Kompare. Application of automated model discovery from data and expert knowledge to a real-world domain: Lake Glumsø. *Ecol Model.*, **212 (1-2)**, 92-98, (2008).

[7] N. Atanasova, S. Dzeroski, B. Kompare, L. Todorovski, and G. Gal. Automated discovery of a model for dinoflagellate dynamics. *Environ. Model. Softw.*, **26 (5)**, 658-668, (2011).

[8] D. Cerepnalkoski, K. Taskova, L. Todorovski, N. Atanasova, and S. Dzeroski. The influence of parameter fitting methods on model structure selection in automated modeling of aquatic ecosystems. *Ecol. Model.*, **245**, 136-165, (2012).

[9] R. Storn and K. Price. Differential Evolution - A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *J. Global Optim.*, **11**, 341-359, (1997).

[10] D.A. Haith and L.L. Shoemaker. Generalized watershed loading functions for stream flow nutrients. *Water Resour. Bull.*, **23**, 471-478, (1987).

[11] W.R. Hammon. Estimating potential evapotranspiration. *J. Hydraul. Div., Proceedings of the American Society of Civil Engineers*, **87**, 107-120, (1961).

[12] G.L. Hargreaves, G.H. Hargreaves, and J.P. Riley. Agricultural benefits for Senegal River Basin. *J. Irrig. Drain Eng.*, **111 (2)**, 113-124, (1985).