A coupled model tree–genetic algorithm scheme for flow and water quality predictions in watersheds

Ami Preis ¹, Avi Ostfeld *

Faculty of Civil and Environmental Engineering, Technion – I.I.T, Haifa 32000, Israel

Received 6 September 2007; received in revised form 7 November 2007; accepted 9 November 2007

SUMMARY
The rapid advance in information processing systems along with the increasing data availability have directed research towards the development of intelligent systems that evolve models of natural phenomena automatically. This is the discipline of data driven modeling which is the study of algorithms that improve automatically through experience. Applications of data driven modeling range from data mining schemes that discover general rules in large data sets, to information filtering systems that automatically learn users’ interests. This study presents a data driven modeling algorithm for flow and water quality load predictions in watersheds. The methodology is comprised of a coupled model tree–genetic algorithm scheme. The model tree predicts flow and water quality constituents while the genetic algorithm is employed for calibrating the model tree parameters. The methodology is demonstrated through base runs and sensitivity analysis for daily flow and water quality load predictions on a watershed in northern Israel. The method produced close fits in most cases, but was limited in estimating the peak flows and water quality loads.

Introduction

Hydrological phenomena are complex processes to describe constituting air, soil, and water, interacting on different spatial and temporal scales. Two basic approaches exist to model hydrological phenomena: physically based and data driven modeling.

A physically based approach requires descriptions of the system’s input, the physical laws which govern its behavior, and boundary and initial conditions. Symbolically this can be expressed as:

\[ Y(t) = \Phi[X(t)] \]

where \( t \) = time, \( X(t) \) = input, and \( \Phi[X(t)] \) = the system transformation of its input \( X(t) \) to its output \( Y(t) \). The characteristics of \( \Phi \) classify the system properties: linear, non-linear, lumped, distributed, deterministic, stochastic, etc. Physically based modeling concentrates on constructing \( \Phi \) using the physical understanding of the system. Contrary to that data driven modeling seeks to construct \( \Phi \) empirically by
learning the "data behavior" of sets of $X(t)$ and $Y(t)$. A data driven modeling approach can obviously be considered only if sufficient data is available.

The methodology developed and demonstrated in this study is data driven. It is a coupled model tree–genetic algorithm (MT–GA) scheme for predicting flow and water quality constituents in watersheds. The model tree (Quinlan, 1993) constructs linear rules through learning, while the genetic algorithm (Holland, 1975) tunes the model tree parameters. The method is demonstrated through base runs and sensitivity analysis for daily flow and water quality load predictions on a watershed in northern Israel.

### Literature review

Data driven modeling is the study of computer algorithms which improve automatically through experience. The most utilized techniques of data driven modeling are artificial neural networks, model trees, fuzzy-rule based systems, and support vector machines.

Since the last decade data driven modeling in both hydrology and water resources research is gaining an increasing interest. This section reviews and classifies the literature on data driven modeling for watershed hydrology into: real time flood forecasting, rainfall–runoff modeling, infilling missing data, coupled data driven–evolutionary optimization models, and water quality prediction models.

### Real time flood forecasting


### Nomenclature

- $D_t$ total annual precipitation (mm/year),
- $D_{t-\Delta t}$ accumulated precipitation (mm) at $\Delta t$ days prior to day $t$,
- $D_i^{t-\Delta t}(x_i^t)$ accumulated effective precipitation (mm) at $\Delta t$ days prior to day $t$,
- $d_i$ precipitation (mm/day) at day $t$,
- $G(k)$ the $k$-th generation,
- $k$ generation counter,
- $L_t$ water quality load at day $t$ (kg/day),
- $M_t$ month of year corresponding to day $t$,
- $m$ number of days considered,
- $N_t^{\text{din}}$ total nitrogen daily load (kg/day),
- $N_t^{\text{max}}$ maximum total nitrogen daily load (kg/day),
- $n$ number of precipitation domain partitions for the flow model,
- $n'$ number of precipitation domain partitions for the water quality model,
- $P_t^{\text{din}}$ total phosphorus daily load (kg/day),
- $P_t^{\text{max}}$ maximum total phosphorus daily load (kg/day),
- $Q_t$ flow at day $t$ ($m^3/day \times 10^3$),
- $O_t^{\text{max}}$ maximum daily flow ($m^3/day \times 10^3$),
- $T_{\text{ave}}(t-\Delta t)$ average air temperature ($^\circ C$) at $\Delta t$ days prior to day $t$,
- $T_t$ air temperature ($^\circ C$) at day $t$,
- $T_t^{\text{w}}$ week of year average temperature corresponding to day $t$,
- $t$ time,
- $W$ total source annual watershed water quality yield (kg/year),
- $W_t$ source watershed water quality yield (kg/day),
- $X(t)$ input,
- $x_i$ $i$-th coefficient calibrated by a genetic algorithm for the flow model,
- $x'_{i}$ $i$-th optimal $x_i$ coefficient,
- $x_{ij} [D_{t-\Delta t}] d_i$ effective daily precipitation,
- $\beta_j$ $j$-th coefficient calibrated by a genetic algorithm for the water quality model,
- $\beta'_{ij}$ $j$-th optimal $\beta_j$ coefficient,
- $\Phi [X(i)]$ the system transformation of its input $X(t)$ to its output $Y(t)$.

Rainfall–runoff modeling


Infilling missing data


Coupled data driven—evolutionary optimization models


Water quality prediction models

Amenu et al. (2007) improved the traditional artificial neural network back-propagation feed-forward method by applying the adaptive minimal resource allocation network methodology. The model was applied to an agricultural watershed in central Illinois for predicting daily runoff and nitrate—nitrogen concentration. Shrestha et al. (2007) developed a fuzzy-rule based model for simulating watershed scale nitrate transport. The model uses simulated flows generated by WASHETH (Schulla and Jasper, 2001) upon which a fuzzy-rule based nitrate transport model was built using simulated annealing. Sahoo et al. (2006) used artificial neural networks to assess flash floods and their attendant water quality parameters using measured data of a Hawaii stream.

Among the models reviewed above, the study of Shrestha et al. (2007) is the closest to this work. This work differs from Shrestha et al. (2007) in using a data driven model for quantifying both the quantity and quality sections and in employing a genetic algorithm to scale model parameters. The two studies are alike in using heuristics (i.e., a fuzzy-rule based model versus a model tree) and evolutionary optimization (i.e., simulated annealing for deriving the fuzzy-rules versus a genetic algorithm for model tree calibration).

Methodology

The methodology is comprised of a coupled model tree—genetic algorithm scheme partitioned into two interconnected sub models for flow and water quality load prediction. This section is built of three sub sections: description of model trees, genetic algorithms, and the proposed methodology.

Model tree (MT)

A model tree is a data driven algorithm (Quinlan, 1993) built of a rule-based predictive structure using a top–down induction approach. The tree is fitted to a training data set by splitting the data into homogeneous subsets based on the data attributes. Thereafter, the tree is constructed with all training cases being predicted by the tree leaves (i.e., each leaf is a linear regression model which predicts continuous values for the numerical attributes). The tree is then pruned bottom–up and transformed into a set of if–then rules which simplify the tree structure, and thus improves its ability to classify new instances.

The predictive ability of the tree is measured using a correlation coefficient for the training and validation data sets.
The correlation coefficient equals one in case of a complete fit between measured values and model tree predictions.

**Genetic algorithm (GA)**

Genetic algorithms (GAs) (Holland, 1975) are heuristic combinatorial search techniques that imitate the mechanics of natural selection and natural genetics of Darwin’s evolution principle. The basic idea is to simulate the natural evolution mechanisms of chromosomes (represented by string structures), involving: selection, crossover, and mutation. This is accomplished by creating a random search technique that combines survival of the fittest among string structures with a randomized information exchange.

A typical form of a genetic algorithm involves three main stages: 1. *Initial population generation*: the genetic algorithm generates a bundle of strings (termed population, or generation), with each string (chromosome) being a set of values of the decision variables/optimization parameters. 2. *Computation of strings fitness*: the genetic algorithm evaluates each string’s fitness (i.e., the value of the objective function that corresponds to each string). 3. *Construction of new generation*: the genetic algorithm establishes the next generation by performing: selection, crossover, and mutation, where: selection involves the process of choosing chromosomes from the current population for reproduction according to their fitness values; crossover — partial exchange of information between pairs of strings; and mutation — a random change in one of the strings locations. The genetic algorithm parameters are the population size, the mating and mutation rates, and the number of generations.

A genetic algorithm pseudo code may take the following form:

1. **Initialization**
   1.1 Set the generation counter \( k = 0 \);
   1.2 Generate an initial population \( G(0) \);
   1.3 Evaluate \( G(0) \).
2. **Main Scheme**
   Repeat
   2.1 Set: \( k = k + 1 \);
   2.2 Generate \( G(k) \) using \( G(k - 1) \);
   2.3 Evaluate \( G(k) \);
   Until stopping conditions are met.

In this study the following GA operations are used: **Selection** — using weighted random pairing; where the better the fitness of a chromosome, the higher is its likelihood to be selected as a parent. **Crossover** — using the one point crossover method where the offspring is a linear combination of its two parents. **Mutation** — through randomly altering one of the chromosome’s parameter values. **Elitism** — the best chromosome in each generation is moved unchanged to the next.

**The proposed model tree—genetic algorithm (MT–GA) methodology**

A schematic description of the proposed model is shown in Fig. 1. The method is comprised of two interconnected schemes of similar structure for predicting flow and water quality loads, respectively.

The prediction of flows (top of Fig. 1) is accomplished through a model tree whose input effective precipitation time series classifier is calibrated by a genetic algorithm. The prediction of water quality loads (bottom of Fig. 1) is attained through using the calibrated effective precipitation time series and by adjusting a vector of parameters which define the fraction of the source water quality loads contributing to the watershed outlet.

**Flow prediction**

The flow prediction model is comprised of a coupled model tree—genetic algorithm scheme. The model tree has three classifiers: (1) \( z_i \cdot [D_{t - \Delta t}] \cdot d_t \) = effective daily precipitation, where \( z_i, 0 \leq z_i \leq 1 \) (i = 1, …, n) = i-th coefficient calibrated by a genetic algorithm, \( n \) = number of precipitation
domain partitions, $D_{t - \Delta t} = \text{accumulated precipitation (mm)}$ at $\Delta t$ days prior to day $t$, and $d_t = \text{precipitation (mm/day)}$ at day $t$; (2) $T_{\text{ave}, t - \Delta t} = \text{average air temperature (°C)}$ at $\Delta t$ days prior to day $t$; and (3) $M_t = \text{month of year corresponding to day } t$.

For a given precipitation daily time series $d_t$ ($t = 1, \ldots, m$), where $m$ is the number of days considered, number of precipitation domain partition $n$, a set of coefficient values $x_i, 0 \leq x_i \leq 1$ ($i = 1, \ldots, n$), a time interval $\Delta t$, time series of the average daily temperatures at $\Delta t$ days prior to each day $t$, and the month associated with each day, a model tree is constructed using CUBist (cubist@www.rulequest.com). The fitness of the model tree is evaluated against the training data used to construct it and on an independent validation data set. The resulted fit which is a maximum of 2: a correlation coefficient of 1 for a complete fit of the training data set plus 1 for a complete fit of the validation data set, defines the fitness of the model tree. The genetic algorithm uses the generated fits of the model trees (i.e., a model tree for each set of coefficients $x_i$), crossover, and mutation to construct the next generation of strings of candidate $x$ coefficients. The genetic algorithm stops if either a predefined number of generations was attained or if no improvements of the model tree fits were observed at some subsequent generations. The output of the flow model is a set of optimal coefficients $x^*_{i}(i = 1, \ldots, n)$.

Observations

1. The flow prediction model parameters are the vector of coefficients $x_i$, the number of precipitation domain partitions $n$, and the time interval $\Delta t$ prior to day $t$. Finding the appropriate parameter values is in general case dependent. Justification of the parameter values selected for this study is provided through sensitivity analysis, as described in the example application section.

2. The rationale for using the accumulated precipitation prior to day $t$ for predicting the effective rainfall at day $t$ is associated with the upper soil water content. It is assumed that as the assembled precipitation prior to day $t$ increases, so is its impact on the effective rainfall at day $t$. This was verified through the example application as: the accumulated precipitation prior to day $t$ increased, so did its corresponding optimal $x_i$ coefficient.

3. The reasoning for partitioning the accumulated precipitation prior to day $t$ to a set of discrete domains stems from the need to constrain as of computational efficiency, the dimensionality of the genetic algorithm search space.

4. The classifier of the average air temperature at $\Delta t$ days prior to day $t$ is a surrogate to Evapotranspiration. The month of the year maps the watershed general conditions (e.g., land use and agricultural management practices, air humidity, radiation, etc.).

Water quality load prediction

As in the flow prediction model, the water quality load prediction model is comprised of a model tree coupled with a genetic algorithm. The model tree has the following classifiers: (1) $\beta^*_j[D^*_{t - \Delta t}(x^*_i)]W_t = \text{effective water quality daily load (kg/day), where: } \beta^*_j, 0 \leq \beta^*_j \leq 1$ ($j = 1, \ldots, n'$) = $j$-th coefficient calibrated by a genetic algorithm, $n'$ = number of precipitation domain partitions ($n' \neq n$), $D^*_{t - \Delta t}(x^*_i) = \text{accumulated effective precipitation (mm)}$ at $\Delta t$ days prior to day $t$ (i.e., the sum of effective rainfall $\Delta t$ days prior to day $t$ calculated using the $x^*_j$ coefficients received from the flow prediction model), $W_t = \text{source watershed water quality yield (kg/day)}$ for day $t$ as given in Eq. (2) below, (2) $T_t = \text{air temperature (°C)}$ at day $t$; and (3) $M_t$.

$$W_t = \frac{d_t}{d_t} W$$

where $D_t = \text{total annual precipitation (mm/year)}$, $W = \text{total source annual watershed water quality yield (kg/year)}$.

For a given source watershed water quality yield time series $W_t(t = 1, \ldots, m)$, number of precipitation domain partition $n'$, a set of coefficient values $\beta^*_j, 0 \leq \beta^*_j \leq 1$ ($j = 1, \ldots, n'$), a time series of the accumulated effective precipitation (mm) at $\Delta t$ days prior to day $t$, time series of daily temperatures $T_t(t = 1, \ldots, m)$, and the month associated with each day, a model tree is constructed using Cubist. As in the flow prediction model, the fitness of the model tree is evaluated against the training data used to construct it and on an independent validation data set. The resulted fit which is a maximum of 2:1 for a complete fit for the training data set plus 1 for a complete fit for the validation data set, defines the fitness of the model tree. The genetic algorithm uses the generated fits of the model trees (i.e., a model tree for each set of coefficients $\beta^*_j$), crossover, and mutation to construct the next generation of candidate coefficients. The genetic algorithm stops if either a predefined number of generations was attained or if no improvements of the model tree fits were observed at some subsequent generations. The output of the water quality load prediction model is a set of optimal coefficients $\beta^*_j$ ($j = 1, \ldots, n'$).

Observations

1. The water quality load prediction model parameters are the vector of coefficients $\beta^*_j$, and the number of precipitation domain partitions $n'$ ($n' \neq n$). The time interval $\Delta t$ prior to day $t$ is assumed to be the same as in the flow prediction model.

2. The rationale for using the accumulated effective precipitation prior to day $t$ for predicting the effective water quality daily load is associated with the flow in the watersheds governed by the effective rainfall.

3. The daily air temperature at day $t$ acts as a surrogate to water temperature which governs water quality reactions (e.g., absorption on soil). The rationale for using $M_t$ as a classifier is similar to that described for the flow prediction model.

4. The daily source water quality watershed yield as given in Eq. (2) is an assumption for scaling down the annual water quality yield to potential daily water quality yields. Other formulations or assumptions can be used.

5. The daily effective rainfall ties the flow prediction and water quality load prediction models.
Example application

The model was applied to Meshushim watershed, a sub-watershed within the Lake Kinneret (Sea of Galilee) watershed (Fig. 2).

The Lake Kinneret watershed is about 2730 km² (2070 km² in Israel, the rest in Lebanon), inhabiting about 200,000 people, organized in twenty five municipalities, and three cities (the Israeli part).

The Meshushim watershed has an area of about 140 km². The soil consists of an upper layer of rocks, clay and sand. The land is mainly used for agriculture. Urban and industrial zones are located at the center of the area around the city of Katzerin. Rainfall in the basin is measured by Mekorot Company Kinneret Watershed Unit, and by the Israeli Meteorological service on a continuously basis, while monitoring of flow and water quality constituents (e.g., total nitrogen, total phosphorus, nitrate, ammonium, chloride, sodium, etc.) on a daily/weekly basis (Markel and Shamir, 2002). The database of water quality load sources used in this work is that constructed for the Israeli Water Commission by DHV MED (2001).

The data set used in this study spans the years of 1996–2001. 70% of the instances were chosen randomly for training for both the flow prediction and water quality load prediction models. The remaining 30% served for verification.

The training data set is used for calibrating the parameters of the methodology for both the genetic algorithm and the model trees. Once these parameters are set they are not altered any further. These parameters are used on the validation data set to test its performance.

Flow prediction

The results of a base run and sensitivity analysis for the flow prediction model for the Meshushim sub-watershed are summarized in Figs. 3–8.

Fig. 3 shows the outcome of a ‘zero run’ — a model tree construction without the genetic algorithm component, a base run, there sensitivity analysis runs, and a general analysis run.

The following can be seen from Fig. 3: (1) At the ‘zero run’, using the week of year average temperature corresponding to day \( t \) instead of \( T_{\text{ave}, t-M} \), the total correlation coefficient received was the lowest (1.80), and the number of classification rules (15) — the highest. (2) At the base run a total correlation coefficient of 1.91 corresponding to nine model tree classification rules was obtained. (3) Sensitivity
at sensitivity analysis 1 (SA1) only the precipitation (mm/day) at day \( t \), and the average air temperature (°C) seven days prior to day \( t \), served as the model tree classifiers. At SA2 the maximum number of model tree rules was constrained to five. At SA3 the number of precipitation domain partitions was reduced to five. The outcome of all three sensitivity analysis runs was a slight reduction in the total correlation coefficient: 1.90 compared to 1.91 at the base run. (4) At the general analysis run the modifications implemented at each of the sensitivity analysis runs were imposed simultaneously. This resulted in a total correlation coefficient of 1.90 as in SA1–SA3, but a major reduction in the computational effort: eight genetic algorithm generations to convergence compared to twenty at the base run and at SA1–SA3.

Fig. 4 presents a comparison between normalized measured and predicted daily flow values for the validation data set of 1997–98 for the base run. It can be seen from Fig. 4 that the model tree constrained to a maximum of five model tree rules

**Legend**

R0 = run 'zero' - a model tree construction without the genetic algorithm part (\( T_{\text{ave}, \Delta t}^{\text{w}} \) = week of year average temperature corresponding to day \( t \)); BR = base run; SA1 = sensitivity analysis 1; GAR = general analysis run; NA = not applicable; +, - = considered, not considered, respectively; = modification compared to base run; \( d_t \) = precipitation (mm/day) at day \( t \); \( T_{\text{ave}, \Delta t} \) = average air temperature (°C) at \( \Delta t \) days prior to day \( t \); \( M_t \) = month of year corresponding to day \( t \); \( n \) = number of precipitation domain partitions; \( \Delta t \) = duration (days) prior to day \( t \); \( P \) = population size; \( g \) = chromosome size; \( \gamma \) = mating rate; \( \eta \) = mutation rate; \( G_c \) = number of generations to convergence; \( R_T^2 \) = training correlation coefficient; \( R_V^2 \) = validation correlation coefficient; \( R_{\text{Total}}^2 \) = total correlation coefficient (i.e., \( R_T^2 + R_V^2 \)); MTR = number of model tree rules.

**Observations and comments**

1. The \( \alpha_i \) (\( i = 1, \ldots, 7 \)) obtained optimal values for the base run were in the range of 0.9687 to 0.123, where the values of 0.9687 and 0.123 corresponded to the highest and lowest weekly accumulated precipitation prior to day \( t \), respectively.

2. The computational time for one genetic algorithm generation on an IBM PC 3.6 GHz, 1GB of Ram was about 1 minute.

Figure 3  Flow prediction model base run and sensitivity analysis results.

![Figure 3](image)

**Legend**

\( Q_t \) = daily flow; \( Q_t^{\text{max}} \) = maximum \( Q_t \) among the measured and predicted data for 1997–98;

- measured and predicted, respectively.

Figure 4  Base run: comparison of normalized measured and predicted daily flows on validation data set for 1997–1998.

![Figure 4](image)
that predictions versus measurements were in general in good agreement, however less satisfactorily for the high flow records. This deficiency is attributed to the relative small data set used for training. An increase in the training data set will likely improve the model prediction accuracy.

Figs. 5–8 present sensitivity analysis for the parameter values selected for the base run. Fig. 5 shows the tradeoff between the total correlation coefficient results (i.e., training correlation coefficient + verification correlation coefficient) and the employed training percentage. It can be seen from Fig. 5 that a training fraction of 64% or more gave the best total correlation coefficient results. Fig. 6 shows the tradeoff between the total correlation coefficient outcome and the number of precipitation domain partitions. It can be seen from Fig. 6 that a number of precipitation domain partitions of five or more gave the best total correlation coefficient values. Fig. 7 shows the tradeoff between the total correlation coefficient values and the antecedent duration $\Delta t$ prior to day $t$. It can be seen from Fig. 7 that an antecedent duration of seven days gave the best total correlation coefficient outcome. Fig. 8 shows the tradeoff between the total correlation coefficient and the maximum number of model tree rules.

Rule 1
If $d_t \leq 12.5$ (mm/day) and $T_{ave, t-7} > 14.1$ (°C)  
Then $Q_{t+1}$ (m³/day $\times 10^3$) = $1.95 + 0.2 d_t$

Rule 2
If $d_t \leq 12.5$ and $T_{ave, t-7} \leq 14.1$  
Then $Q_{t+1} = 9.66 + 0.7 d_t - 0.5 T_{ave, t-7}$

Rule 3
If $d_t \leq 5.9$ and $T_{ave, t-7} \leq 11$  
Then $Q_{t+1} = -0.52 + 0.9 d_t + 0.7 T_{ave, t-7}$

Rule 4
If $5.9 < d_t \leq 12.5$ and $T_{ave, t-7} \leq 14.1$  
Then $Q_{t+1} = 21.36 + 0.5 d_t - 0.2 T_{ave, t-7}$

Rule 5
If $d_t > 12.5$  
Then $Q_{t+1} = 15.74 - 9.3 T_{ave, t-7} + 9.6 d_t$
coefficient results and the number of maximum model tree rules. It can be seen from Fig. 8 that at least nine model tree rules are required to maximize the total correlation coefficient. It can be seen from Fig. 8 that at least nine model tree rules are required to maximize the total correlation coefficient.

Fig. 9 is an example of a model tree outcome. The figure shows the five rules obtained for the general analysis run (GAR) (see Fig. 3). Note that rules 2 and 3 overlap for $d_t = 5.9$ (mm/day) and $T_{ave, t} = 11$ (°C). In such instances the model tree forecasts are averaged to arrive at a final prediction.

Water quality load prediction

Two water quality constituent loads were predicted: total nitrogen and total phosphorus. For the total nitrogen two sources were considered: grazing – 14,144 (kg/year), and surface runoff – 42,713 (kg/year), while for the total phosphorus: grazing – 2053 (kg/year), surface runoff – 2373 (kg/year), and residential-industrial – 503 (kg/year).

The results of a base run and sensitivity analysis for the total nitrogen and total phosphorus predictions are summarized in Figs. 10–13.

Figs. 10 and 11 summarize base runs and sensitivity analysis for the total nitrogen and total phosphorus predictions, respectively.

The following can be observed from Fig. 10: (1) At the base run a total correlation coefficient of 1.95 corresponding to thirteen model tree classification rules was obtained. (2) Sensitivity analysis: at SA1 only the source watershed water quality yield (kg/day) for day $t$, and the air temperature (°C) were considered.

Legend

BR = base run; SA1 = sensitivity analysis 1; +, - = considered, not considered, respectively; $\square$ = modification compared to base run; $W_t$ = source watershed water quality yield (kg/day) for day $t$; $T_t$ = air temperature (°C) at day $t$; $M_t$ = month of year corresponding to day $t$; $n'$ = number of precipitation domain partitions; $\Delta t$ = duration (days) prior to day $t$; $P$ = population size; $g$ = chromosome size; $\gamma$ = mating rate; $\eta$ = mutation rate; $G_c$ = number of generations to convergence; $R^2_T$ = training correlation coefficient; $R^2_V$ = validation correlation coefficient; $R^2_{Total}$ = total correlation coefficient (i.e., $R^2_T + R^2_V$); MTR = number of model tree rules.

Figure 10  Total nitrogen water quality load prediction model base run and sensitivity analysis results.

Legend

BR = base run; SA1 = sensitivity analysis 1; +, - = considered, not considered, respectively; $\square$ = modification compared to base run; $W_t$ = source watershed water quality yield (kg/day) for day $t$; $T_t$ = air temperature (°C) at day $t$; $M_t$ = month of year corresponding to day $t$; $n'$ = number of precipitation domain partitions; $\Delta t$ = duration (days) prior to day $t$; $P$ = population size; $g$ = chromosome size; $\gamma$ = mating rate; $\eta$ = mutation rate; $G_c$ = number of generations to convergence; $R^2_T$ = training correlation coefficient; $R^2_V$ = validation correlation coefficient; $R^2_{Total}$ = total correlation coefficient (i.e., $R^2_T + R^2_V$); MTR = number of model tree rules.

Figure 11  Total phosphorus water quality load prediction model base run and sensitivity analysis results.
at day \( t \), served as the model tree classifiers. At SA2 the maximum number of model tree rules was constrained to five. At SA3 the number of precipitation domain partitions was increased to six. SA1 and SA2 resulted in a reduction in the total correlation coefficient of 1.94 and 1.92, respectively. At SA3 the total correlation coefficient remained unchanged, but the required computational effort increased substantially.

Similar observations can be seen for the total phosphorus load prediction, as shown in Fig. 11: a total correlation coefficient of 1.90 with 11 classification rules for the base run; reduction of the total correlation coefficient at SA1 and SA2; and the same total correlation coefficient for SA3, which required additional computational effort.

Figs. 12 and 13 show comparisons for the base run between measured and predicted values for normalized total nitrogen and total Phosphors, respectively for the validation data set of 1997–1998. It can be seen from both figures that predictions versus measurements were in general in good agreement, however less satisfactorily for the high loads. This deficiency, as observed for the flow prediction (see Fig. 4), is attributed to the relative small data set used for training. An increase in the training data set will likely improve the model prediction accuracy.
Conclusions

1. A coupled data driven (model tree)—evolutionary scheme (genetic algorithm) for flow and water quality load predictions in watersheds was developed and demonstrated. The model tree was used for prediction of flow and water quality loads while the genetic algorithm for calibrating the model tree parameters.

2. The employment of data driven modeling for simulation of complex physical systems is receiving an increasing interest as the result of the growing availability of data. Most of the developed schemes are for real-time flood forecasting and for rainfall—runoff modeling using artificial neural networks. There are almost no models which use model trees, and none which use model trees for water quality load predictions in watersheds. The advantage of using a model tree is in its outcome simplicity: a set of linear rules which can be easily accessed and potentially physically interpreted.

3. As in every data driven modeling implementation one of the most important challenges is to tie the data driven technique to the most important physical governing processes of the system. This task is extremely complex for simulating natural systems like flow and water quality behavior in watersheds. In this study it was assumed that the most important governing phenomena driving both the flow and the water quality loads is the effective rainfall.

4. The predictions received by the developed model were in general in good agreement with measurements. However, the model was less successful in predicting high flows and water quality loads. This is an inherent deficiency of a data driven technique whose accuracy is primarily dependent on the data set used for its training. The larger a data set, the higher likelihood of receiving better predictions. It is anticipated that increasing the number of training instances for the proposed model will improve its prediction accuracy.

5. Research challenges for extending this study are in more closely incorporating the system’s physics. This can be accomplished for example by using a physically based model in conjunction with a data driven modeling technique. An evolutionary algorithm can tie both, yet the computational effort expected to run such a setup is anticipated to be very high. The construction of such a framework is thus not straightforward; hence, additional amendments will need to be undertaken.

Acknowledgements

This work was funded by the Israeli Water Commission, and by the Technion Grand Water Research Institute (GWRI).

References

Abedini, M.J., Nasseri, M., 2004. Spatiotemporal rainfall forecasting via ANNS coupled with GA. In: Liong, Phoon, Babovic (Eds.), Sixth International Conference on Hydroinformatics. Published on CD.


