Transparent water quality model induction in a reservoir affected by the rainfall-runoff process using a grammar-based genetic programming

JongHwa Ham

Korea Rural Research Institute

Timothy Yoon-Seok Hong (✉ hongt@lsbu.ac.uk)

London South Bank University

Research Article

Keywords: Water quality prediction model, Rainfall-runoff, Evolutionary model induction, Grammar-based genetic programming, Agricultural catchment

Posted Date: March 12th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-180459/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

1

2 **Transparent water quality model induction in a reservoir**

3 **affected by the rainfall-runoff process using a grammar-**

4 **based genetic programming**

5

6

7 JongHwa Ham¹

8 Yoon-Seok Timothy Hong^{2*}

9

10

11

12

13

14

15

16

17 1. Ph.D, Rural Research Institute, Korea Rural Community Corporation, #870 Haeandro,

18 Sangrok, Ansan, Gyeonggi 426-908, Republic of Korea, E-mail: drhaam@ekr.or.kr

19 2*. Corresponding author, Ph.D, London South Bank University, 103 Borough Road,

20 London, SE1 0AA, United Kingdom. E-mail: hongt@lsbu.ac.uk

21

ABSTRACT

Building a reliable water quality prediction model in the catchment is of importance both for understanding the process of these natural systems and providing a basis for water quality management decisions. Due to a rapid change of river flow during a Typhoon season in South Korea, water quality parameters in reservoirs are affected significantly within a short-time period by a rainfall-runoff process. Integrated conceptual hydrological and water quality models seem to be complicated to produce a good model calibration and prediction with reasonable generalizations under these dynamic condition of heavy rainfall events. As an alternative, this paper proposes an evolutionary model induction system based on grammar-based genetic programming (GBGP) to derive the transparent mathematical model for estimating the dynamic change of water quality parameters within a short-time period in an agricultural reservoir affected by the rainfall-runoff process during a typical Typhoon summer period.

Results showed that the GBGP system performed to evolve accurate water quality models, expressed in the form of explicit mathematical formulae which could predict the concentration and load of COD, SS, T-N, and T-P during the heavy rainfall event as a function of easily measurable rainfall, cumulative rainfall, and flow rate. The performance of the water quality models evolved by the GBGP was superior to ANN and optimized pollutant rating curve (PRC) model, showing that it has the lowest RMSE value. The transparent nature of water quality models evolved by the GBGP may allow inferences about underlying processes to be made. This work demonstrates that complex dynamic water quality change affected by the rainfall-runoff process in natural catchments can be successfully modelled through the use of GBGP system without costly or time-consuming tasks required in the conceptual modeling approach.

Keywords: Water quality prediction model; Rainfall-runoff, Evolutionary model induction; Grammar-based genetic programming; Agricultural catchment

1. INTRODUCTION

Developing a mathematical model for estimating a change of water quality concentration and load influenced by a rainfall-runoff process in the watershed is extremely important because it can provide a basis for water quality management decisions that can minimize water pollution risk to the public.

There are integrated hydrological and water quality models to estimate the water quality variables incorporating the rainfall-runoff process (Arnold et al., 2012; Rossman & Huber, 2016). Most of them are typically based on the mechanistic principles such as water balances, physical and chemical laws. The resulting models have complicated model structures and present the largest sources of uncertainty such as closure assumptions, unmodelled phenomena, empirical formulas, and uncertainty in the model input and structure.

Alternatively, unstructured data-driven input/output models using machine learning/searching algorithm in artificial neural networks has been applied to hydrological modeling (Abrahart et al., 2004) and water quality modeling (Chau, 2006) due to their simplicity and high levels of prediction accuracy. They only require the definition of relevant input and output variables and the more data are available, the more the model can be enhanced using the learning algorithm. However, the neural network-based models act in a “black-box” manner where the stored

knowledge cannot be easily interpretable and requires large amounts of data to generate an accurate prediction. The key weakness of these unstructured data-driven input/output model approach in artificial neural networks is that there is no coherence between the internal structure of the derived model and the structure of the real systems. Thus, the transparency of the derived model is low or missing.

In recent years, there has been growing interest in implementing automatic model induction systems, which involves automatic translation of a problem statement into a mathematical model using a computer, without requiring the development of a process-focused model, in diverse science and engineering fields. Automatic model induction systems heuristically search through a subset of all possible models given a set of arithmetical operators, mathematical functions, and variables (Hong et al, 2005). These systems aim to find the model that best represents observations. A diverse set of approaches to solving automatic model discovery problems have been developed. These include: Automatic programming (Barr and Geigenbaum 1982), inductive logic programming (Muggleton 1992; Lavrac and Dzeroski 1994), and evolutionary computational intelligence (Cramer 1985; Koza 1992). A particularly fast growing and interesting research topic in the automatic model induction field today is genetic programming (GP) which was created by John Koza (1992, 1994). Genetic programming is an evolutionary computation algorithm that uses a population-based search method that works on a population of individuals whose output is a potential model in the space of search to a given problem (Koza 1992). There are several successful applications of this GP technique to hydrological modeling (Whigham and Crapper, 2001; Hong and Rosen, 2002; Hong et al., 2005, Parasuraman and Elshorbagy, 2008; Selle and Muttill, 2011; Fallah-Mehdipour et al., 2013; Chadalawada et al., 2017; Heřmanovský et al., 2017; Liu and Shi, 2019; Sepahvand et al., 2019) and water quality modeling (Whigham and Recknagel 2001; Muttill

and Lee, 2005; Hong & Paik, 2012).

South Korea's typhoon-related rainy season during the months of June to August leads rivers to easily peak and overflow. Due to a rapid change of river flow to a reservoir, water quality parameters in reservoirs are affected significantly within a short-time period by a rainfall-runoff process during a Typhoon season. Under these dynamic condition of heavy rainfall events, integrated conceptual hydrological and water quality models seem to be complicated to produce a good model calibration and prediction with reasonable generalizations where water quality parameters can be changed significantly within a short-time period due to heavy rainfall events. As an alternative of complex, integrated conceptual hydrological and water quality models, we propose a new automatic model induction system based on grammar-based genetic programming (GBGP) to derive the transparent inference model for estimating the dynamic change of water quality in the reservoir affected by the rainfall-runoff process during a heavy rainfall summer period. Given a set of arithmetical operators, mathematical functions and variables, the GBGP is designed to perform an automatic model induction of water quality prediction models that best fit observed data using a context-free grammar via the mechanism of genetics and natural selection. Unlike the back-box model in artificial neural networks, the inducted water quality models by the GBGP technique are structured by the transparent representation of evolved models in the form of multivariate mathematical formulae explicitly. The performance of our proposed modeling methodology is compared to the performance of a multilayer perceptron (MLP) neural network trained with the backpropagation (BP) learning algorithm, optimized pollutant rating curve (PRC) model, and multiple linear regression (MLR) model.

2. METHODOLOGY

2.1 Grammar-based genetic programming (GBGP)

Grammar-based genetic programming (GBGP) (Whigham 1995; O'Neill and Ryan 2001; McKay et al. 2010) is a variant of GP technique and uses a grammar system to derive possible models that best fit observed data. The grammar system in the GBGP is designed to represent syntactical constraints and to incorporate background knowledge in order to guide the model evolution process (Hong & Paik, 2012). In our proposed GBGP system, a context-free grammar (CFG) (Hopcroft and Ullman, 1979; Chomsky, 1986) is implemented to derive possible models by incorporating arithmetic operators, use-defined function, and variables proposed by Whigham (1995).

2.1.1 Context-free grammar (CFG)

Figure 1 is an example of a context-free grammar. A context-free grammar consists of a four tuple, $GR = \{Start, P, N, T\}$ where *Start* is the designated start *nonterminal* symbol, *P* is a set of productions, and *N* and *T* is a set of '*non-terminals*' symbols and a set of '*terminal*' symbols, respectively. In Figure 1, the start symbol *Start* in a grammar GR refers to the nonterminal used to start generating the expressions from the grammar. In the GBGP system, a context-free grammar contains a finite set of non-terminals *N* each representing sets of expressions that can appear in the model equations. A set of terminals *T* is represented by general arithmetic or logic operation, external inputs and constants, and zero argument functions. The expressions represented by the nonterminals are described in terms of nonterminals and terminals. As seen in Figure 1, the productions *P* are rules connecting the Non-terminals among themselves and to other terminals and represent the space of possible models that may be created during the evolution process (Whigham and Crapper, 2001; Hong & Paik, 2012). The productions P in

the grammar are of the form $X \rightarrow Y$, where X is a member of nonterminal N and Y is any combination of symbols from $\{\text{terminal } T \cup \text{nonterminal } N\}$. In Figure 1, productions of the form $Eq \rightarrow \langle Eq \rangle + \langle Eq \rangle$ and $Eq \rightarrow \langle Eq \rangle - \langle Eq \rangle$ is expressed using the distinctive symbol $|$, as $Eq \rightarrow \langle Eq \rangle + \langle Eq \rangle \mid Eq \rightarrow \langle Eq \rangle - \langle Eq \rangle$. The terminal $Constant \in T$ is used to denote a constant parameter in an equation that has to be fitted to the give data. The non-terminal $Variable \in N$ is to denote any variables (e.g., X, Y, Z) from the input domain. The grammar determines how model expressions can be formed (e.g., what functions and variables can be part of an expression) (Hong and Paik, 2012).

2.1.2 Generating model trees from the grammar

Figure 2 shows an example of a model derivation tree representing the mathematical equation $X + (3.8Y)$ that can be produced by grammar GR. In Figure 2, the connection points are called ‘nodes’. According to their position in the tree, the symbols in the nodes of a tree are divided into two types, terminals T and nonterminals N (Whigham 1995).

The model derivation trees are built using rules in grammar. When a new model derivation tree is built, the GBGP starts with a tree containing just one node. This node contains the *Start* symbol which is a non-terminal. The model derivation tree should satisfy the constraints that every vertex has a label: 1) interior nodes have labels from the set of non-terminals N and 2) leaf nodes have labels from the set of non-terminals N . The rules in the grammar tell us that a certain non-terminal given on the left-hand side of the rule can be replaced by a certain small tree containing a mix of terminals and nonterminals given on the right-hand side of the rule. However, the root will always be a non-terminal and the leaves will always be terminals in Figure 2 (Hong and Paik, 2012). This results in a population of possible models from which models are selected and modified using evolutionary search processes to produce better fitting

models.

As shown in Figure 2, the following model derivation tree steps, which defines the mathematical expression for $X+(3.8Y)$:

$$\begin{aligned} \text{Start} &\Rightarrow \langle Eq \rangle \Rightarrow + \langle Eq \rangle \langle Eq \rangle \Rightarrow + \langle Variable \rangle \langle Eq \rangle \\ &\Rightarrow + \langle X \rangle \langle Eq \rangle \Rightarrow + \langle X \rangle * \langle Eq \rangle \langle Eq \rangle \Rightarrow + \langle X \rangle * \langle Variable \rangle \langle Eq \rangle \\ &\Rightarrow + \langle X \rangle * \langle Y \rangle \langle Eq \rangle \Rightarrow + \langle X \rangle * \langle Y \rangle \langle Constant \rangle \Rightarrow + \langle X \rangle * \langle Y \rangle \langle 3.8 \rangle \quad (1) \end{aligned}$$

In Figure 2, the height of the model derivation tree is five (equal to the length of the path from the root node (the *Start* symbol) to the leaf nodes *Y*). The height of the model derivation tree as the maximum depth of a path from the root and a leaf node should be limited to prevent the evolution of a model with too many parameters.

2.1.3 Optimization of model derived

Generally, the models generated by the GBGP system may include a number of internal model parameters, thus the optimization process of these model parameters is required to prevent the possibility of removing a potentially good model due to poor model parameters. (Whigham and Crapper, 2001; McKay et al. 2010). In our work, the parameters evolved in the models are optimized by the Powell's algorithm (Powell, 1964).

2.1.4 Evaluating derived models

A crucial first step for evolving any model under the GBGP is to design a fitness function which determines how well an evolved model is able to solve the problem. In the GBGP, the

fitness is based on the error between the model inducted by genetic programming and the actual data. The performance of each model inducted is tested against a set of fitness cases.

The error functions such as the root mean square (RMSE) or R -square have been used as a fitness measure. Each model of the population is evaluated once for each of the data points in the training data and the results are compared to the observed values in the training period. A fitness value for each model of the population is then calculated by an error function. This fitness value is used to select parent models to generate the new models for the next generations.

Several selection methods have been suggested in the literature (Goldberg and Deb, 1991) and a fitness-proportionate selection has been used in this work. In the fitness- proportionate selection, a model (individual) is selected with a probability proportionate to its fitness. From Koza (1992), if $f(S_i(t))$ is the non-zero fitness of model S_i at time t , then the probability that model S_i will be selected from a population of P for breeding is:

$$\frac{f(S_i(t))}{\sum_{j=1}^P f(S_j(t))} \quad (2)$$

This method is described by Goldberg (1989).

2.1.5 Creating subsequent generations using Selection, Crossover, and Mutation

Once an individual (a model) has been selected based on its fitness value from the current population, three genetic operators (crossover, mutation, and direct reproduction) may be applied to modify models during the evolution. The choice of each operator is probabilistic, with crossover probability P_c ; mutation probability P_m ; and reproduction probability $P_r (= 1 - P_c - P_m)$ (Hong et al., 2005).

The function of the crossover operator is to generate new or offspring models from two parent models by combining information extracted from the parents. This operator tries to combine parts of two models in order to create a superior model. In the GBGP system, this is done by selecting a random sub-tree from within each of the parent models, then swapping them over. Figure 3 illustrates two trees in the parent models before and after the crossover operation. This small difference between parents and offspring models, shown in Figure 3, is vital for the GBGP and allows selection pressure to drive the evolution of the population. This crossover process is done in such manner as to follow the grammar. The operation of mutation allows new models to be created by making random changes to models about to enter the new generation. In the GBGP system, a single model is chosen and a random sub-tree within it is replaced with a new, freshly generated sub-tree. That model then is placed in the new population (Figure 4). The direct reproduction operator directly copies a model unaltered from the parent population to the next generation.

2.1.6 Model evolution procedure in the grammar-based genetic programming (GBGP)

The following steps show the model evolution procedure in the grammar-based genetic programming (GBGP) (Figure 5) (Hong and Paik,2012):

1. Initialization: Generate an initial population of P models using context-free grammar.
Generation $K=0$.
2. Optimize each model (variables and constants) in the population.
3. Execute and evaluate the fitness of each model in the population.
4. Genetic loop: Repeat until termination criterion is met (maximum generation, $K=max$).
 - 4-a. Generate a new population by crossover, and mutation of models

241 4-a-i. Select two models (parents) based on their fitness for breeding to the next
 242 generation.
 243 4-a-ii. Probabilistically perform crossover to randomly swap sub-trees of the two parents
 244 to two child models.
 245 4-a-iii. Probabilistically perform mutation to modify a random sub-tree of each child
 246 model.
 247 4-a-iv. Repeat steps 4-a-i and 4-a-ii to generate a total of P new offspring models.
 248 4-a-v. Old models from the parent population are replaced by new models generated.
 249 4-b. Go to next generation: $K+1$.

250

251 **2.2 Pollutant rating curve (PRC)**

252 Over the years, researchers have proposed several rating curves to determine the average
 253 relationship between discharge and pollutant load (Asselman, 2000; Overleir, 2004; Crowder
 254 et al., 2007). A typical rating curve consists of a graph or equation, relating pollutant discharge
 255 or concentration to stream discharge, which can be used to estimate pollutant load from the
 256 flow record. The PRC generally represents a functional relationship of the form:

257

$$258 \qquad L = aQ^b \qquad (3)$$

259

260 where Q is stream discharge and L is either pollutant load. Values of a and b for a particular
 261 stream are determined from data via a linear regression between $(\log L)$ and $(\log Q)$.

262

263 **3. WATER QUALITY MODEL DERIVATION USING A GRAMMA-BASED** 264 **GENETIC PROGRAMMING**

3.1 Study area and rainfall-runoff monitoring data

In South Korea, over 50% of pollutant load inflows to reservoirs are affected by rainfall-runoff processes, particularly during the rainy summer period. The Idong reservoir catchment in Korea is a typical example of this case. The Jein sub-watershed area (Figure 6) is 1,636ha (16% of Idong reservoir watershed) and is composed of 68% forested area, 22% agricultural area and 5% residential area. The data set measured in 1 or 2 hour intervals was collected from the monitoring network installed in the catchment during the heavy rainfall events. The data set contained 15 variables such as rainfall, cumulative rainfall, river flow rate, pH, electronic conductivity (EC), BOD, COD, SS (suspended solid concentration), $\text{NH}_4\text{-N}$, $\text{NO}_3\text{-N}$, total nitrogen (T-N), $\text{PO}_4\text{-P}$, and total phosphorus (T-P).

3.2 Gramma-based genetic programming (GBGP) setup

3.2.1 Parameter setting for the GBGP system

Table 1 shows the GBGP system setup parameters which were used to derive the water quality model. A population in the GBGP is the number of programs that the system will evolve. The larger the population, the more difficult a problem that will be solved and the longer the run will take. A population of 500 candidate models was created and tested in each run. The model evolution process was carried out for a maximum 50 generations (Table 1). The function set F and the maximum depths of model tree are two important control parameters that affect both the search space and the complexity of models in the algorithm.

To ensure that the created programs did not grow in an unbounded fashion, a limit is placed on the maximum depth of any program created by crossover. This parameter is referred to as Max. depth of model tree in Table 1 and reflected the maximum depth of any program that was

expected to be required to produce a model. The maximum depth of a model tree was limited to 6 to prevent the evolution of a model with too many parameters.

The probability of crossover and mutation was 80 and 10%, respectively. Hence, the direct reproduction probability was 10%, meaning that 10% of the population was passed unchanged into each subsequent generation. This ensured that good models are not prematurely discarded from the population (Hong et al., 2005)

The RMSE was used as the fitness function to evaluate the performance of each model in a population during the training phase and as a measure of generalization capability for the testing data. Therefore the GBGP evolution proceeded by minimizing the RMSE value to each model.

3.2.2 Context-free grammar and functional set

The grammar, GR, used by GBGP to develop the water quality models allowed general mathematical functions to be evolved, and was defined in Figure 7. In terminals T , the mathematical functions available as part of the evolved expression were the standard arithmetic operators, plus the inverse function, natural logarithm, power function and logarithm to an arbitrary base (Figure 7 and Table 1).

The terminal symbol *Constant* is used as an index to an external array of numbers to get the value of the node. It represents an independent variable or real value in a model. The Powell's optimization algorithm was used to find the specific value for this *Constant* node for which the value of the RMSE for this expression is minimal.

The $Var(Constant)$ retrieved the value of one of the independent variables for the current data point. The independent variable was selected depends on the value of $Constant$ optimized by the Powell's algorithm. If the first independent variable in the data set provided was a rainfall, $Var(I)$ represents a rainfall value at current time. As mentioned earlier, a number I was optimized by the Powell's optimization algorithm.

The exponential function was represented by the terminal string "Exp". The production $Eq \rightarrow Exp(Eq)$ allowed the exponential function to be included with any part of the evolved mathematical expression. $Eq \rightarrow Eq^{Constant}$ also allowed the power function of $x^{Constant}$ to be included with any part of the evolved mathematical model expression. In the grammar GR, the terminal $Constant$ is a random floating point number which is generated for each occurrence of $Constant$ when the initial population is created. These random "Constants" were potentially optimized once the initial generation of a solution was created.

The terminal symbols eT and $Rain$ represented the elapsed time and rainfall, respectively. The cR and Q terminals represented the cumulative rainfall and river flow rate, respectively.

3.3 Result from predicting water quality concentration by GBGP derived model

In this study, the modeling aim of GBGP was to develop the mathematical inference models in the form of explicit mathematical formulae for predicting the concentration (mg/L) and load (kg/d) of COD, SS, T-N, and T-P during the rainfall event as a function of easily measurable R (rainfall), cR (cumulative rainfall), Q (flow rate), and eT (elapsed time)

The specific water quality monitoring program in an agricultural catchment was undertaken to

analyze the impact of rainfall-runoff process on water quality parameters in a reservoir during a rainy summer period associated with rainfall events in 2016 and 2019 year. The data sets were split into two sets for modeling purposes: (1) the training data set (August 2016); and (2) the testing data set (July 2019). The training data set was used to derive the models and the testing data set was used to evaluate the models evolved by the GBGP system to show how well evolved models generalized the unseen data set. A multivariate model was first evolved to predict the COD concentration. A series of ten runs of the GBGP system was performed with different randomly generated initial models so that each run took a different genetic loop to evolve the better performing models for predicting COD concentration.

It was observed that despite different initial conditions all runs were able to evolve a good model after 25 generations. The model evolution process was carried out for 50 generations. The best performing model evolved by the GBGP system are presented in Table 3.

The best performing model was selected at Run 7. The following model is discovered as the best performing model of SS prediction using the grammar GR:

$$SS = 0.231 \times Q \times \left(54.26 * \left(R^{2.325} \right) + R^{1.0568} \times \left(1378 + Q^{-0.8949 \times eT} \right) \right) \quad (4)$$

The best model evolved from Eq.(4) using only eT , R , and Q variables has a RMSE of 19.22 mg/L and R^2 of 0.97 shown in Table 2. It was observed that there was no overfitting condition for GBGP model, meaning that there was no significant difference between training and testing result with good model generalizations. Therefore, the average value of training and test result for RMSE and R^2 is shown in Table 2. Figure 8 shows the measured and predicted SS concentration by Eq.(4) on both the training set and testing set. A visual comparison with the

measured SS concentration indicated that Eq.(4) has captured the dynamic pattern of SS concentration and performed well over the full range data with good prediction and generalization.

The same simulation procedure as was applied for SS concentration to evolve a process model for the T-N concentration. The same GBGP parameter settings and grammar GR (Figure 8) were used. The best performing model for T-N concentration was defined as follows:

$$T - N = 2.024 - 6.76 \times 10^{-7} \times cR^Q - 0.5485 \times eT^{0.08235} - 0.1698 \times cR^{0.1994} \quad (5)$$

Figure 8 shows the measured and predicted T-N concentration in the effluent by Eq.(5) on both the training and testing data set. A RMSE of the best evolved inference model by Eq. (5) for predicting T-N concentration is computed as 0.09 mg/L, respectively.

The best performing models for the prediction of T-P evolved by GBGP was:

$$T - P = \frac{(0.9 \times Q + 0.00001 \times eT)}{(0.96297 \times cR + 0.03703 \times eT)} \times (0.15393 \times R + 0.84607 \times Q) + \frac{4.7127 \times (0.42244 \times Q + 0.57 \times cR)}{870.83 \times (0.95665 \times Q + 0.04335 \times eT)} \quad (6)$$

Table 2 shows a RMSE and R^2 for the model evolved from Eq.(6). Figure 8 displays the measured T-P and the prediction of the best model evolved from Eq. (6).

The following model was discovered as the best performing model for predicting COD concentration:

$$COD = \frac{117.32}{0.67cR + 0.3eT} + \frac{(935 \times (0.24R + 0.75Q))}{173.9 \times (0.547Q + 0.45eT)} \quad (7)$$

The model evolved from Eq.(7) had a very satisfactory performance with a RMSE of 0.55 mg/L and R² of 93%.

A multi-layer perceptron (MLP) trained with the backpropagation learning algorithm (MLP-BP) provided a comparison with the models evolved by the GBGP system by using the same inputs and output data. We employed a genetic algorithm (Goldberg, 1989) to design the architecture of the MLP-BP. The optimized MLP-BP model for predicting water quality concentrations employed the same 4 inputs, one hidden layer with 5 neurons, 3 of which used a sigmoid transfer function and 2 of which use a hyperbolic transfer function. Multiple linear regression model was also applied with same 4 inputs for the comparison purpose. Statistical measures of model accuracy (RMSE and R²) for the three modelling approaches are summarized in Table 2. It was observed that like GBGP model, there was no overfitting condition for MLP-BP model with reasonable model generalizations.

Models evolved by the GBGP system and the MLP-BP models performed well on predicting water quality concentration of SS, T-N, T-P, and COD over the full data range and gave reasonable generalizations as shown in Figure 8 and 9. Although models generated from both systems produced very similar model accuracy, the model evolved by GBGP performed slightly better than one evolved by the MLP-BP. The best evolved models from Eqs.(4)–(7) show that multivariate models are created in a nonlinear fashion. This demonstrates that a highly nonlinear relationship between dependent variables (water quality variables) and independent variables (hydrological variables) exist in this catchment. For this reason, Table 2

and Figure 8 show that the MLR models do not follow the measured pattern of water quality concentration and perform poorly when compared to the GBGP models and MLP-BP models.

3.4 Result from predicting water quality load

The estimation of water quality load in terms of kg/d was undertaken due to its importance in reservoir water quality management. To predict the water quality load of COD, SS, T-N, and T-P affected by the rainfall-runoff process, the pollutant rating curve (PRC) method was also applied for the comparison with GBGP, ANN, and MLR. The constants (a and b) in Eq.(3) of PRC model was optimized to provide the best prediction accuracy.

The best evolved models from the GBGP system and best optimized models of PRC are presented in Table 4. The prediction results from four modelling approaches (GBGP, ANN, MLR, and PRC) for predicting the water quality load of COD, SS, T-N, and T-P on both training set and testing set are displayed in Figure 10.

As seen in Table 5, the best evolved models from the GBGP and MLP-BP system performed better than the PRC and MLR models. However, the model evolved by GBGP performed slightly better than the MLP-BP models.

The SS load prediction is the most important task in reservoir water quality management in Korea as SS load inflows to reservoir are significantly influenced by the rainfall-runoff process, particularly during the rainy summer period. From Table 5 and Figure 10, it is clear that the MLP and PRC models do not provide the reasonable model accuracy to

predict the SS load to reservoir during the rainy period. Both the models evolved by GP and the MLP-BP model were more accurate and produced better predictions. They have captured phenomena that cause the high SS load inflow to reservoir during the rainy period. The SS load model evolved by the GBGP system produced the best predictive performance among the four modelling techniques.

The best evolved models in Table 4 show that multivariate models are created in a complex nonlinear fashion as a function of flow rate, Q . This means the SS load cannot be predicted easily by the simple linear like PRC model.

4. CONCLUSION

The evolutionary model induction system based on grammar-based genetic programming (GBGP) was proposed and applied to evolve process models from a set of mathematical functions with a context-free grammar for predicting the water quality in the reservoir affected by the rainfall runoff process in a heavy rainfall summer period.

Results have shown that the evolved water quality models by the GBGP system successfully derives accurate water quality models to predict the concentration and load of COD, SS, T-N, and T-P during the rainfall event as a function of easily measurable rainfall, cumulative rainfall, and flow rate. The performance of the water quality models evolved by the evolutionary process model induction system is superior to the performance of ANN, optimised PRC and MLR model, demonstrated by the good generalization result with the lowest RMSE value.

Our paper demonstrates that unlike other black-box modeling techniques such as neural networks, the explicit multivariate mathematical representation of evolved models in the GBGP provides a better understanding of water quality variables affected by rainfall-runoff process by allowing analysis of the resulting models and identifying important inputs variables. Our result shows that the GBGP system can work most efficiently where the possible model is unknown or partially known and the understanding of the resulting model is important.

Further work is planned to show how the conceptual-type models can be derived so that models evolved by the GBGP system may provide better conceptual hydrological information combined with known conceptual hydrological knowledge. This can be achieved by creating a new context-free grammar in the GBGP system through hybridizing known conceptual hydrological functions or models.

ACKNOWLEDGEMENTS

This work was supported by Korea Rural Research Institute, National Korea Rural Community Corporation, Republic of Korea.

DECLARATIONS

- The authors (JongHwa Ham and Yoon-Seok Timothy Hong) approved the contents in this manuscript and agreed to submit our paper to Water Resources Management.
- The field data collection and lab analysis, and modelling application were performed by JongHwa Ham. The main modeling system design and modelling application was undertaken by Yoon-Seok Timothy Hong.

- Both authors contributed to produce the draft manuscript including the result interpretation. The final manuscript has been read and agreed by the authors.
- The authors declare that this manuscript has not been published and is not under consideration for publication elsewhere.
- The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.
- This work was supported by Korea Rural Research Institute, National Korea Rural Community Corporation, Republic of Korea.
- Some data that support the findings of this study are available from the corresponding author upon reasonable request under the permission of Korea Rural Research Institute.

REFERENCES

- Abrahart, R.J., Kneale, P.E., See, L.M. (2004) Neural Networks for Hydrological Modelling. Taylor & Francis Group plc., London, UK.
- Arnold, J.G., Moriasi, D.N., Gassman, P.W., Abbaspour, K.C., White, M.J., Srinivasan, R., Santhi, C., Harmel, R., Van Griensven, A., Van Liew, M.W. (2012) *SWAT: Model use, calibration, and validation*. Transactions of the ASABE, 55, 1491-1508.
- Asselman, N.E.M. (2000) Fitting and Interpretation of Sediment Rating Curves. *Journal of Hydrology* 234, 228-248.
- Barr, A., Geigenbaum, E. (1982) Automatic Programming. Los Altos. Handbook of Artificial Intelligence, William Kauffman.

506 Chadalawada, J., Havlicek, V., & Babovic, V. (2017) A Genetic Programming Approach to
 507 System Identification of Rainfall-Runoff Models *Water Resources Management*, Vol.
 508 31, 3975–3992.

509 Chomsky, N. (1986) Knowledge of language: Its Nature, Origin and Use. Preager Publishers,
 510 New York.

511 Cramer, N.L. (1985) A representation for the adaptive generation of simple sequential
 512 programs. In Grefenstette, JJ Editor, *Proceedings of an Int. Conf. on Genetic Algorithms*
 513 *and the Applications*, Pittsburgh PA: Carnegie-Mellon University, 183-187.

514 Crowder, D.W., Demissie, M., Markus, M. (2007) The Accuracy of Sediment Loads When
 515 Log-transformation produces Nonlinear Sediment Load-Discharge Relationships.
 516 *Journal of Hydrology* 336, 250-268.

517 Fallah-Mehdipour, E., Haddad, O., Orouji, H. & Mariño, M. A. (2013) Application of
 518 Genetic Programming in Stage Hydrograph Routing of Open Channels *Water Resources*
 519 *Management*, Vol. 27, 3261–3272.

520 Goldberg, D.E. (1989) Genetic algorithms in search, optimization, and machine learning,
 521 Addison-Wesley, Reading, MA.

522 Goldberg, D. E., Deb, K. (1991) A comparative analysis of selection schemes used in genetic
 523 algorithms. In *Foundations of Genetic Algorithms*, Morgan Kaufmann Publishers, 69-93.

524 Heřmanovský, M., Havlíček, V., Hanel, M., Pech, P. (2017) Regionalization of runoff models
 525 derived by genetic programming. 547, 544-556.

526 Hong, Y.S.T., Paik, B.C. (2012) Inference model derivation with a pattern analysis for
 527 predicting the risk of microbial pollution in a sewer system. *Stoch Environ Res Risk Assess*
 528 26, 695–707.

529 Hong, Y.-S., Rosen, M.R. (2002) Identification of an urban fractured rock aquifer dynamics
 530 using an evolutionary self-organizing modelling, *J. Hydrology* 259, 89– 104.

531 Hong, Y.-S., White, P.A., and Scott, D.M. (2005) Automatic rainfall recharge model induction
 532 by evolutionary computational intelligence. *Water Resources Research* 41, W08422,
 533 doi:10.1029/2004WR003537.

534 Hopcroft, J.E., Ullman, J.D. (1979) Introduction to automata theory, languages and
 535 computation. Addison-Wesley Reading, MA.

536 Chau, K.-W. (2006) A review on integration of artificial intelligence into water quality
 537 modelling. *Marine Pollution Bulletin* 52: 726–733

538 Koza, J. R. (1992) Genetic Programming: On the Programming of Computers by Means of
 539 Natural Selection. The MIT Press, Cambridge, MA.

540 Koza, J.R. (1994) Genetic Programming II: Automatic Discovery of Reusable Programs.
 541 Cambridge MA: MIT Press.

542 Lavrac, N., and Dzeroski, S. (1994) Inductive Logic Programming: Techniques and
 543 Applications. Chichester: Ellis Horwood.

544 Liu , S. & Shi, H. (2019) A Recursive Approach to Long-Term Prediction of Monthly
 545 Precipitation Using Genetic Programming. *Water Resources Management*, Vol.
 546 33, 1103–1121.

547 McKay, R., Hoai, N.X., Whigham, P., Shan, Y., O’Neil, M. (2010) Grammar-based Genetic
 548 Programming: a survey. *Genetic Programming and Evolvable Machine* 11: 365-396.

549 Muggleton, S. (1992) Inductive Logic Programming. Academic Press, San Diego CA.

550 Muttill, N., Lee, J.H.W. (2005) Genetic programming for analysis and real-time prediction of
 551 coastal algal blooms. *Ecological Modeling* 189, 363-376.

552 O'Neill, M., Ryanm C. (2001) Grammatical Evolution. *IEEE Transactions on Evolutionary*
553 *Computation*, 5 (4), 349-358.

554 Overleir, A.P. (2004) Accounting for Heteroscedasticity in Rating Curve Estimates. *Journal*
555 *of Hydrology* 292, 173-181.

556 Parasuraman, K, Elshorbagy, A. (2008) Toward improving the reliability of hydrologic
557 prediction: Model structure uncertainty and its quantification using ensemble-based
558 genetic programming framework. *Water Resources Research*, 44, W12406,
559 DOI:10.1029/2007WR006451.

560 Powell, M.J.D. (1964) An efficient method for finding the minimum of a function of several
561 variables without calculating derivatives. *Comput. J.* 7, 152–162.

562 Rosssman, L.A., Huber, W. (2016) Strom Water Management Model Reference Manual
563 Volume III –Water Quality. US EPA Office of Research and Development, Washington
564 DC. EPA/600/R-16/093

565 Selle, B., Muttill, N. (2011) Testing the structure of a hydrological model using Genetic
566 Programming. *Journal of Hydrology* 39, 1-9.

567 Sepahvand, R., Safavi, H. & F. Rezaei, F. (2019) Multi-Objective Planning for Conjunctive
568 Use of Surface and Ground Water Resources Using Genetic Programming. *Water*
569 *Resources Management*, Vol. 33, 2123–2137.

570 Whigham, P.A. (1995) Inductive Bias and Genetic Programming. In Zalzala, AMA, editor,
571 *First International Conference on Genetic Algorithms in Engineering Systems:*
572 *Innovation and Application*, GALEZIA, The Institute of Electrical Engineer UK, 414,
573 461-466.

574 Whigham, P. A., and Recknagel, F. (2001) An inductive approach to ecological time series
575 modelling by evolutionary computation. *Ecological Modelling* 146, 275–287.

Whigham, P., Crapper, P. (2001) Modelling rainfall-runoff using genetic programming.
Math Comput Model 33(6), 707–721.

Table 1 GBGP parameters setting.

Parameter	Value
Population size	500
Max. depth of model tree	6
Generation No.	50
Crossover probability	80%
Mutation probability	10%
Optimization algorithm	Powell method
Fitness measure	Minimize RMSE
Available functions	+, -, *, / inv(x), ln(x), x ^y , logxy, exp(x) Constant

Table 2. The RMSE (mg/L) of three models for predicting water quality concentration

		MLR	MLP-BP	GBGP
COD	RMSE	1.45	0.74	0.55
	R ²	0.54	0.88	0.93
SS	RMSE	72.84	27.72	19.22
	R ²	0.52	0.93	0.97
T-N	RMSE	0.19	0.09	0.09
	R ²	0.50	0.88	0.89
T-P	RMSE	0.05	0.035	0.023
	R ²	0.38	0.66	0.84

Table 3. Best evolved models from the GBGP system and optimized MLP models for predicting water quality concentration

		Best Model derived
COD	MLR	$2.2 - 0.001 \times eT + 0.095 \times R + 0.005 \times cR + 1.8 \times Q$
	GBGP	$\frac{117.32}{0.67cR + 0.3eT} + \frac{(935 \times (0.24R + 0.75Q))}{173.9 \times (0.547Q + 0.45eT)}$
SS	MLR	$13.68 + 1.118 \times eT + 10.87 \times R - 0.8 \times cR + 25.07 \times Q$
	GBGP	$0.231 \times Q \times (54.26 + (R + R^{2.325}) + R \times 0.0568^R \times (1378 + Q^{-0.8949 \times eT}))$
T-N	MLR	$1.357 - 0.005 \times eT + 0.003 \times R - 0.007 \times cR - 0.039 \times Q$
	GBGP	$2.024 - 6.76 \times 10^{-7} \times cR^Q - 0.5485 \times eT^{0.08235} - 0.1698 \times cR^{0.1994}$
T-P	MLR	$0.004 + 0.00001 \times eT + 0.002 \times R + 0.00000008 \times cR + 0.044 \times Q$
	GBGP	$\frac{(0.9 \times Q + 0.00001 \times eT)}{(0.96297 \times cR + 0.03703 \times eT)} \times (0.15393 \times R + 0.84607 \times Q) + \frac{4.7127 \times (0.42244 \times Q + 0.57 \times cR)}{870.83 \times (0.95665 \times Q + 0.04335 \times eT)}$

Table 4. Best evolved models from the GBGP system and optimized PRC models for predicting water quality load (kg/d)

		Best model derived
COD	GBGP	$(-1.038 \times eT + 1.984 \times R + 32.77) \times eT \times Q^2 - 117.1 \times Q^2 + 276.7 \times Q$
	PRC	$474.57 \times Q^{1.26}$
SS	GBGP	$(1230 + 3735 \times Q - cR \times Q \times (69.39 + Q)) \times (R + Q) \times Q$
	PRC	$6643.9 \times Q^1$
T-N	GBGP	$Q \times (115.1 - cR + 0.01786 \times Q \times eT \times (6.898 \times R + eT))$
	PRC	$85.079 \times Q^{0.884}$
T-P	GBGP	$\frac{(Q + Q^{-251.53 \times Q^{21.89}} + 0.3272 \times cR \times Q) \times eT \times Q}{eT + 30}$
	PRC	$6.081 \times Q^{1.596}$

eT : elapsed time, R : rainfall, cR : cumulative rainfall, Q : river flow rate

Table 5. The RMSEs (kg/day) and R^2 for predicting water quality load

		MLR	MLP-BP	GBGP	PRC
COD	RMSE	95	55	36	111
	R^2	0.95	0.97	0.99	0.94
SS	RMSE	5839	1376	1181	6917
	R^2	0.55	0.95	0.98	0.37
T-N	RMSE	10	7	4	11
	R^2	0.97	0.99	0.99	0.96
T-P	RMSE	3	1	1	3
	R^2	0.88	0.98	0.98	0.89

Figures

GR = {**Start**,
 N = { *Eq*, *Variable* }
 P = { **Start** $\rightarrow$ *<Eq>*
 Eq $\rightarrow$ *<Eq>* + *<Eq>* | *<Eq>* - *<Eq>* | *<Eq>* * *<Eq>* | *<Eq>* / *<Eq>*
 Eq $\rightarrow$ *<Variable>* | *<Constant>* |
 Variable $\rightarrow$ *<X>* | *<Y>* | *<Z>*
 Constant $\rightarrow$ *<0.5>* | *<2>* | *<3.8>*
 }

T = { + , - , * , / , X , Y , Z , Constant }

Figure 1

An example of the grammar (*Eq* represents the word equation).

1,2,3,4,5 denotes the depth of each part of the equation tree

Figure 2

Example of model derivation tree for the mathematic expression $X+3.8Y$.

Crossover

Figure 3

Crossover between model derivation trees

Mutation

X+5Y

New

X+exp(Z+1.5)

Figure 4

Mutation process in the model derivation tree

Figure 5

Structure of the model induction by the GBGP system.

Figure 6

Idong reservoir catchment watershed and Jein sub-catchment.

GR = {Start,
N = {Eq},
P = {
 Start → Eq
 Eq → Eq+Eq | Eq-Eq
 Eq → Eq*Eq | Eq/Eq
 Eq → Exp(Eq) | Eq^{Constant} | log(Eq)
 Eq → Var(Constant)
 Eq → Constant
 }
T = { +, -, *, /, Exp(x), x^y, log(y), eT, Rain, cR, Q, Constant}
}

Figure 7

A context-free grammar used for the water quality model evolution.

Figure 8

Observed and predicted water quality concentration

Figure 9

Observed versus predicted water quality concentration.

Figure 10

Observed and predicted water quality load using included flow data set.