International Journal of
Operations Research

International Journal of Operations Research Vol. 2, No. 1, 15−20 (2005)

An Application of Grammatical Evolution for Reservoir Inflow Prediction and Operation

L. Chen* and P. K. Chiang

Department of Civil Engineering, Chung Hua University, Hsin Chu, Taiwan 30067, R.O.C.

Received March 2004; *Revised June 2004*; *Accepted January 2005*

***Abstract*—**The study describes a Grammatical Evolution (GE) system and applies it to establish the inflow prediction model of Der-Ji Reservoir in central Taiwan. GE is a new computing architecture in the area of optimization. It provides system identification in a transparent and structured way; a fittest function type of input-output relationship will be obtained automatically from this method. A multi-regressive (MR) method and a GE model were fitted to the inflow data series and their performances were compared in the dry year. The results indicate that this new model, GE, is better than traditional MR in all criteria. Then the real-time reservoir operation policy was developed through the genetic algorithms (GAs) and rule curves operation and their performance was compared. It was found that the GA model releases had the best objective function value.

***Keywords*—**Grammatical evolution, Real-time reservoir operation, Genetic algorithm

1. INTRODUCTION

Water is becoming a scarce resource as a result of the growing demand for its use in various purposes. Therefore, reservoir operation forms an important part in water resources development. Real time reservoir operation concerns the optimal operation of an existing reservoir system. A real time optimization model generally is operated on the basis of forecasted information. Forecasts of streamflow and other input deteriorate with time (Yeh, 1985). Systems analysis, which involves use of optimization, simulation, and other decision-making techniques, is a set of powerful tools to solve reservoir operation problems (Jain et al., 1999).

Traditional optimization techniques including linear programming (LP) and dynamic programming (DP) have been used to solve the reservoir operation problems. Yeh (1985) presents a comprehensive in-depth state-of-the-art review of reservoir-operation models, with a strong emphasis on optimization techniques. Generalized computer codes are available for solving LP problems, but the strict linear form of LP does limit its applicability (Wurbs, 1993). Nonlinear properties of a problem can be readily reflected in a DP formulation. However, the usefulness of DP for multireservoir systems is limited by the huge demand that it can induce on computational resources. The choice of methods depends on the characteristics of the reservoir system being considered, on the availability of data, and on the objectives and constraints specified. Most of these models, however, are valid only for simplified reservoir systems. Genetic Algorithms (GAs) have received much attention for their potential use as optimization techniques for complex systems recently (Chen, 2003a).

Evolutionary algorithms have been used with much success for the automatic generation of programs. In particular, genetic programming (GP) has enjoyed considerable popularity and widespread use (Chen, 2003b, c). Unlike GP, grammatical evolution (GE) does not perform the evolutionary process on the actual programs, but rater on variable-length binary strings. A mapping process is employed to generate programs in any language by using the binary strings to select production rules in a Backus-Naur form (BNF) grammar definition. The result is the construction of a syntactically correct program from a binary string which can then be evaluated by a fitness function (O'Neill and Ryan, 2001).

The average annual rainfall in Taiwan is abundant, i.e., 2500 mm, as compared to the global average of 970 mm. However, over 75 percent of the annual rainfall occurs during the wet season (from May to October). Typhoons usually occur in July, August, and September, bringing much of the needed rainfall for the coming dry season (from November to April), during which an enormous amount of water is provided for irrigating rice paddies. Such a significant season variation in annual rainfall makes reservoir operation complicated. During a drought, water rationing and water share reallocation for different established users are common remedial measures (Cheng et al., 2000). There, a driest year 1964 is chosen as inflow data to establish the real-time operation of Der-Ji Reservoir in central Taiwan. The main purpose of this paper was attained through the following steps:

(1) The multi-regressive (MR) method and GE approach were used to model reservoir inflows. The inflows were forecast by both approaches and the results were

* Corresponding author email: lichen@chu.edu.tw

compared.

(2) The real-time reservoir operation policy was developed through GA and rule curves operation and their performance was compared.

2. GENETIC ALGORITHMS

Recently, there has been an increasing interest in solving optimization problems. The genetic algorithms (GAs) is one of the most promising techniques in that domain and has received a great deal of attention regarding optimizing complex systems. The GA is essentially a Darwinian natural selection process, which combines an artificial survival of the fittest with natural genetic operators (Holland, 1975). Through the genetic evolution method, an optimal solution can be found and represented by the final winner of the genetic evolution. The GA is an iterative procedure, which maintains a population of individuals that are candidate solutions to specific domain. During each generation, the individuals in the current population are rated for their effective evaluations, and a new population of candidate solutions is formed using specific genetic operators such as reproduction, crossover, and mutation (Grefenstette, 1986). Then, Goldberg (1989) and Davis (1991) reviewed many important applications of GAs.

In the reservoir operation system fields, GAs have been demonstrated as powerful optimization approaches but there are few references in the literature. Esat and Hall (1994) applied a GA to the four-reservoir problem. They concluded GAs have potential in water resources optimization and that significant savings could be achieved in both memory and execution times. Olivera and Loucks (1997) used GAs to develop operating policies for multi-reservoir systems, and concluded that GAs are practical and robust methods, which could lead to effective operating policies. Chang and Chen (1988) applied real-coded GA for rule-based flood control reservoir management. The results show that the real-coded GA perform better in terms of efficiency and precision than the binary-coded GA. Wardlaw and Sharif (1999) demonstrated that using GAs can provide a robust and acceptable solutions for a four reservoir deterministic problem. Further, they could acquire the known global optimum. Sharif and Wardlaw (2000) presented multi-reservoir systems optimization using GAs. They compared with discrete differential dynamic programming (DDDP) that GA results are very close to the optimum, and the technique appears to be robust. Chen (2003a) applied the real-coded GA for the optimization of long-term reservoir in Taiwan. The results indicate that the real-coded GA with several revised operators significantly improves the performance of system, and could be expected to be very efficient for other highly nonlinear systems.

3. GRAMMATICAL EVOLUTION

GE has been applied to all manner of automatic programming problems, from symbolic regression, to C programs, or generation of graphical objects. The common view of Genetic Programming is that, given a particular problem statement, a program that satisfied the fitness function is to be generated. GE is an evolutionary automatic programming type system, that uses a combination of a variable length binary string genome and a BNF (Backus-Naur Form) grammar to evolve interesting structures. GE presents a unique way of using grammars in the process of automatic programming. Variable-length binary string genomes are used with each codon representing an integer value where codons are consecutive groups of 8 bits. The integer values are used in a mapping function to select an appropriate production rule from the BNF definition, the numbers generated always representing one of the rules that can be used at that time. This technique draws inspiration from the overlapping genes phenomenon exhibited by many bacteria, viruses, and mitochondria that enables them to reuse the same genetic material in the expression of different genes (Elseth and Baumgardner, 1995).

3.1. Backus-Naur form

BNF is a notation for expressing the grammar of a language in the form of production rules (Naur, 1963). BNF grammars consist of terminals, which are items that can appear in the language, e.g., +, -, etc., and nonterminals, which can be expanded into one or more terminals and nonterminals. A grammar can be represented by the tuple { N, T, P, S }, which N is the set of nonterminals, T the set of terminals, P a set of production rules that maps the elements of N to T, and S is a start symbol that is a member of N. When there are a number of productions that can be applied to one particular N, the choice is delimited with the '|' symbol.

Below is an example BNF, where

N = { expr, op, pre_op }

T = { Sin, +, -, /, X, 1.0, (,) }

S = <expr>

And P can be represented as

```
(1) <expr> :: = <expr><op><expr>        (0)
              | (<expr><op><expr>)      (1)
              |<pre-op> (<expr>)        (2)
              | <var>                   (3)

(2) <op> :: = +                         (0)
            | -                         (1)
            | /                         (2)
            |*                          (3)

(3) <pre-op> :: = Sin                   (0)
                | Cos                   (1)
                | Tan                   (2)
                | Log                   (3)

(4) <var> :: = X                        (0)
             | 1.0                      (1)
```

In GE, the BNF definition is used to describe the output language to be produced by the system, i.e., the compilable code produced will consist of elements of the terminal set T. As the BNF is a plug-in component of the system, it means that GE can produce code in any language thereby giving the system a unique flexibility. Therefore, the C language is used for plugging in to GE easily in this study.

3.2. Mapping process

The genotype is used to map the start symbol onto terminals by reading codons of 8 bits to generate a corresponding integer value from which an appropriate production rule is selected by using the following mapping function:

Rule = (codon integer value)

MOD

(number of rules for the current nonterminal) (1)

Considering the following rule, i.e., given the nonterminal op, there are four production rules to select from:

```
(2) <op> :: = +    (0)
           | -     (1)
           | /     (2)
           | *     (3)
```

If we assume the codon being read produces the integer 6, then

6 MOD 4 = 2

would select rule (2) /. Each time a production rule has to be selected to map from a nonterminal, another codon is read. In this way, the system traverses the genome.

Consider the individual in Figure. 1.

(1) First, concentrating on the start symbol <expr>, we can see that there are four productions to choose from. To make this choice, we read the first codon from the chromosome and use it to generate a number. This number will then be used to decide which production rule to use according to equation (1) in BNF. Thus, we have 200 MOD 4 = 0, meaning we must take the zeroth production so that <expr> is now replace with

<expr><op><expr>.

(2) Continuing with the first <expr>, i.e., always starting from the leftmost nonterminal, a similar choice must be made by reading the next codon vale (160) and again using the given formula we get 160 MOD 4 = 0，i.e., rule (0). The leftmost <expr> will now be replaced with <expr><op><expr> to give

<expr><op><expr><op><expr>.

(3) Again, we have the same choice for the first <expr> by reading the next codon value 206, the result being the application of rule (2) to give

<pre-op>(<expr>)<op><expr><op><expr>.

(4) Now, the leftmost <pre-op> will be determined by the codon value 96 that gives us rule (0), which is <pre-op> becomes Sin. We have the following:

Sin(<expr>)<op><expr><op><expr>

(5) The next codon will determine what <expr>，This is 27 MOD 4 = 3, i.e., rule (3). It is a <var> to give

Sin(<var>)<op><expr><op><expr>

(6) The next codon then determines what value <var>，which has two possible production rules, shall take. This is 72 MOD 4 = 0, i.e., rule (0)，which turns out to be X. We now have

Sin(X)<op><expr><op><expr>

(7) The next codon will determine what <op>，will become, so we have 107 MOD 4 = 3，which gives a *, and the resulting expression is

Sin(X)*<expr><op><expr>

(8) The mapping continues until eventually we are left with the following expression:

Sin(X)*Cos(X)+1.0

3.3. An example- symbolic regression

Symbolic regression problems involve finding some mathematical expression in symbolic form that represents a given set of input and output pairs. The aim is to determine the function that maps the input pairs onto the output pairs. The particular function examined is

$$f(x) = x^3 + x^2 + x$$

with the input values in the range [-1…1].

The grammar used in this problem is given below

N= { expr, op, pre_op }
T= { Sin, Cos, Tan, Log, +, -, /, *, X, 1.0, (,) }
S= { expr }

And P can be represented as

```
(1) <expr> :: = <expr><op><expr>         (0)
              | ( <expr><op><expr>)      (1)
              |<pre-op> (<expr>)         (2)
              | <var>                    (3)

(2) <op> :: = +                          (0)
            | -                          (1)
            | /                          (2)
            | *                          (3)

(3) <pre-op> :: = Sin                    (0)
                | Cos                    (1)
                | Tan                    (2)
                | Log                    (3)
```

11001000	10100000	11001110	01100000	00011011	01001000	01101011

⇩

200	160	206	96	27	72	107	62	21	55	88	100	203	41

Figure 1. Example individual expressed as integers. Integer values are generated by converting the 8-bit binary number that is each codon into its corresponding integer value.

(4) <var> :: = X (0)
| 1.0 (1)

The fitness for this problem is given by the sum of square errors, taken over 20 fitness cases, of the error between the evolved and target functions. Several parameters of GA are described as follows: the population size = 400, crossover rate = 0.8 and mutation rate = 0.01. The results demonstrate that GEGA could obtain the optimal function type, $f(x) = x^3 + x^2 + x$, within sixty generations, shown in Fig. 2. GE was successful in finding correct solution to the problem described here. The same problem was tackled using a standard GP (Chen, 2003b, 2003c). In this case, GE outperforms GP on this problem from around the fifteenth generation.

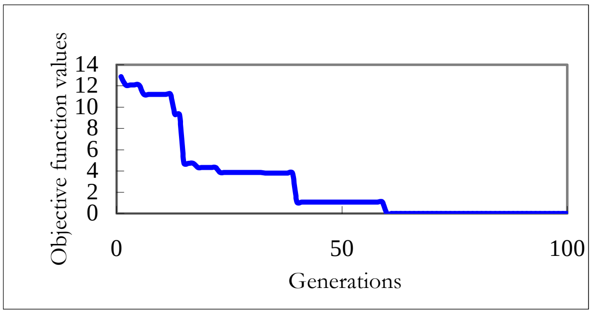

Figure 2. Objective function values of GE on the symbolic regression problem.

4. A CASE STUDY IN TAIWAN

The Der-Ji reservoir, which was completed in 1974 and has an efficient storage capacity of 169*106 m3, is one of the major storage reservoirs in central Taiwan. The hydropower plant at Der-Ji has a generating capacity of 234 MW. This reservoir is a multi-purpose reservoir for hydroelectric power generation, agricultural irrigation, water supply, flood control, and recreation. The primary water use in the basin is hydroelectric power generation.

4.1. Forecast through GEGA modeling

Historical ten-day (the traditional time period of reservoir operation in Taiwan) inflows to the reservoir for a period of 40 years (1959-1998), i.e. 1440 data were used for modeling. According to the correlation analyses, the first (t-1), second (t-2), third (t-3), 36th (t-36) ahead ten-day inflow and the average of inflow were chosen as the input variables. These five input variables were shown in Table 1.

The main consideration of objective function of the inflow prediction model is to minimizing the mean absolute error (MAE). This MAE for the ten-day periods is defined as follows:

$$\text{MAE} = \frac{1}{N}\sum_{t=1}^{N} |\hat{Q}_t - Q_t| \quad (2)$$

Where Q_t : the actual inflow at time t

$\hat{Q}_t$: the predicted inflow at time t

N : the total number of time steps (N = 1440)

t : time steps (ten-day)

To compare with traditional multiple regression (MR), the same input variables were used to construct the model, shown as follows.

$$\hat{Q}_t = b_0 + b_1 Q_{t-1} + b_2 Q_{t-2} + b_3 Q_{t-3} + b_4 Q_{t-36} + b_5 Q_{avg} \quad (3)$$

where $\hat{Q}_t$: the predicted inflow at time step t

Q_{t-1} : the actual inflow at time step t-1

Q_{t-2} : the actual inflow at time step t-2

Q_{t-3} : the actual inflow at time step t-3

Q_{t-36} : the actual inflow at time step t-36

Q_{avg} : the average of actual inflow of time step t

t: time step (ten-day)

b_0, b_1, b_2, b_3, b_4 and b_5: the coefficients

Table 1. The correlations between output and input variables

Input variables (ten-day)	correlations
The first ahead	0.5414
The second ahead	0.3114
The third ahead	0.2102
The 36th ahead	0.1576
The average	0.4669

4.2. Simulation results

The historical flow of the driest year 1964 was considered as a test example in this study. Several parameters within GE are shown in Table 2. Through 8000 generations, the optimal equations of each time steps are obtained. The MAE of GE is 34.512; and the MAE of MR is 105.774. Obviously, the result of GE is much better than the traditional method. The predicted inflows by using GE and MR as well as the actual inflows of the year 1964 are shown in Fig. 3 and 4. Several criteria including root mean squared error (RMSE), percentage absolute error (PAE), coefficient of correlation (CC) and coefficient of efficiency (CE) are compared deeply with these two models, shown in Table 3. It is indicated that the performance of GE is better than MR in all these five criteria.

Table 2. The parameters setting of GE

Population size	1000
Length of codon	8
Length of individual	160
Rate of crossover	0.8
Rate of mutation	0.01

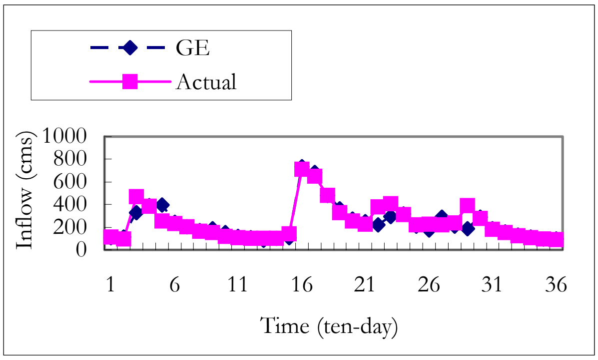

Figure 3. Inflow prediction by GE.

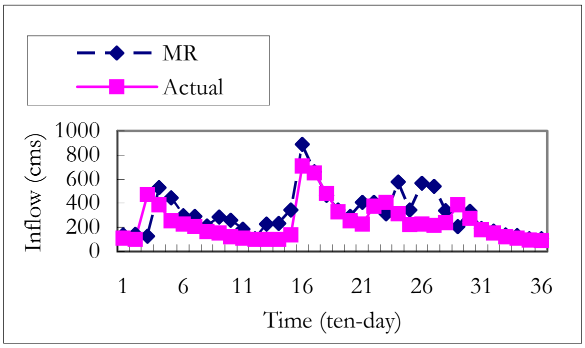

Figure 4. Inflow prediction by MR.

Table 3. The deep comparisons between GEGA and MR

Criteria	GEGA	MR
MAE	34.512	105.774
RMSE	60.868	143.498
CC	0.923	0.724
PAE	0.859	0.568
CE	0.842	0.120

4.3. Development of reservoir operation policy

The genetic algorithm (GA) was used to find the optimal reservoir releases. It constructed the real-time operation model, which combined with the inflow prediction by GE. There are 36 decision variables within GA in one year. The ten-day water supply demand target of this reservoir is in year 2021. The objective function of the GA was to maximize energy production subject to typical system constraints and minimize the deficit of water supply. Therefore, the objective function chosen was expressed as follows (Guo, 1996; Chio, 2000).

$$\text{Max } Z = \sum_{t=1}^{n} CP \times ENE_t - CS \times A^{SH_t} \times WSH_t \qquad (4)$$

where

t: 1, 2, 3,……………,n

n: 36 (ten-day)

CP: the coefficient of net benefit per hydropower generation (2NT dollars/KWH)

ENE_t: the hydropower generations of Der-Ji Reservoir at time t (KWH)

CS: the loss benefit per water shortage (4NT dollars/m^3)

A: a constant (10)

SH_t: the percent of water shortage at time t

WSH_t: the water shortage at time t (m^3)

The water balance of the reservoir system is considered as the system constraint. The other constraints are described as follows. Water levels at any period must be higher than the minimum level (intake elevations), and below flood control level or other limitations. All diversion facility and power-plant equipment capacity limitations in the system must be satisfied.

The real time operation was optimized with genetic algorithms (GAs). A GA incorporating ME selection and BLX-0.5 crossover (Chen 2003a), is concluded to produce the best results. The process of generating and evaluating decision parameters is repeated until no further improvement in performance is obtained. The main parameters which control the GA are shown in Table 4.

Table 4. The parameters setting of GA in the real time operation

Population size	1000
Number of variables	36
Rate of crossover	0.8
Rate of mutation	0.01

The management of most reservoirs in Taiwan uses operating rule curves. These curves primarily guide the release of the reservoir system according to the current storage level and the time of year. The results of the optimal operation obtained from GA combined with inflow prediction by GE and the traditional rule curves operation were compared in Table 5. A reservoir behavior was prepared with important reservoir attributes, namely, water shortage index, maximum water deficit rate, total water release, total spill and total hydropower generations. It is shown that the objective function value of real time operation optimized by GA and GE is better than the result of original rule curves operation. In other words, the real-time operation could obtain more hydropower generations and less water shortage compare with the conventional operation.

Table 5. Comparisons between the Real Time and Rule Curve Operation

Items	Real time	Rule curve
Water shortage index in year (SI)	2.01	2.63
Maximum water deficit rate in ten-day（%）	76.12	76.89
Total water shortage in year (10^4 tons)	16,546.7	17,083.8
Total water release in year (10^4 tons)	96,272.6	94,946.7
Total hydropower generations in year (MWH)	209,740	193,268
Objective function value (10^6 NT dollars)	2036.5	-3947.8

5. CONCLUSIONS

This study used multi-regressive (MR) method, grammar evolution (GE), system simulation, and genetic algorithm (GA) for the operation of a single, multipurpose reservoir. The objective of this paper was to assess the application potential of the GE in attaining the reservoir operational objectives, compared with the conventional models. GE is a system that can produce code in any language with arbitrary complexity. The only inputs are a Backus-Naur Form (BNF) definition for the genotype-to-phenotype mapping process and a fitness function.

For reservoir inflow prediction, regression analysis and GE were used. It was found that the low flows are modeled better through the GE. Thus, GE can be effectively used in drought regulation by reservoirs. As for reservoir operation, the optimal releases were computed using GA. It was found that the GA model releases had the best objective function value. Thus, GA combined with GE is an effective tool for reservoir real-time operation.

ACKNOWLEDGMENTS

This paper is based on partial work supported by National Science Council, R.O.C. (Grant No. NSC 92-2313-B-216-001).

REFERENCES

1. Chang, K.S., Yeh, H.C., and Liou, C.Y. (2000). *Comparative Study of Drought Prediction Techniques for Reservoir Operation*, 36: 511-521.
2. Chen, L. (2003a). Real-coded genetic algorithm optimization of long-term reservoir operation. *Journal of American Water Resources Association*, 39: 1157-1165.
3. Chen, L. (2003b). A study of applying genetic programming to reservoir trophic state evaluation using remote sensor data. *International Journal of Remote Sensing*, 24: 2265-2275.
4. Chen, L. (2003c). A study of applying macro-evolutionary genetic programming to concrete strength estimation. *Journal of Computing in Civil Engineering*, ASCE, 17: 290-294.
5. Chio, S.H. (2000). *A Study of Real-time Reservoir Operation by Using Genetic Algorithms*, Master's thesis, Department of Civil Engineering, National ChungHsing University, Tai Chung, Taiwan.
6. Elseth, G.D., and Baumgardner, K.D. (1995). *Principles of Modern Genetics*. St. Paul, MN: West.
7. Guo, J.W. (1996). *The Optimal Modeling of Research for Der-Ji Reservoir Operation*, Master's thesis, Department of Civil Engineering, National ChungHsing University, Tai Chung, Taiwan.
8. Jain, S.K., Das, A., and Srivastava, D.K. (1999). Application of ANN for reservoir inflow prediction and operation. *Journal of Water Resources Planning and Management*, ASCE, 125: 263-271.
9. Naur, P. (1963). *Revised Report on the Algorithmic Language ALGOL 60*. Commun. ACM, 6: 1-17.
10. O'Neill, M. and Ryan, C. (2001). Grammatical evolution. *IEEE Transactions on Evolutionary Computation*, 5: 349-357.
11. Wurbs, R.A. (1993). Reservoir-system simulation and optimization models. *Journal of Water Resources Planning and Management*, ASCE, 119: 455-472.
12. Yeh, W.W.-G. (1985). Reservoir management and operation models: A state-of-the-art review. *Water Resource Research*, 21: 1797-1818.