農業工程學報　第41卷第2期
中華民國84年6月出版

Journal of Chinese Agricultural Engineering
Vol. 41. No. 2, June 1995

遺傳機制原理應用於灌溉系統規劃之理念

A Genetic Algorithm Approach to Decision Support for Irrigated Systems Planning: An Overview

美國猶他州立大學生物及灌溉工程學系研究助理
郭　勝　豐
Kuo, Sheng-Feng

摘　要

遺傳機制原理是利用平行處理（Parallel Implementation）原則來找到全面性的最佳值（Global optimal）。它不同於傳統的線性或非線性規劃，是利用點到點（point to point）原則，僅能找到局部的最佳值。台灣農業灌溉用水約佔全年水資源的70％，如何做好灌溉不足的最好途徑。遺傳機制原理可嘗試應用於處理複雜的灌溉系統規劃，以便有效的應用水資源調配，並求得最佳的經濟效益。

關鍵詞：遺傳機制，複製，交叉，變種，總體最佳值，局部最佳值。

ABSTRACT

Genetic Algorithm is used the parallel implementation to find out the global optimal in all of the data. It is different from the traditional linear and non-linear planning which are the relation between point to point, it only find out local optimal from the whole data. There is 70 %of water resources used to the agriculture, how to save the agricultural water in water management which transfers to the another purpose use of the water, it is the best way to use the theory of Genetic Algorithm and maybe get the best economic efficiency in water distribution.

Keywords: Genetic algorithm, Reproduction, Crossover, Mutation,Global optimal, Local optimal.

Introduction

Greater attention is being given to water management in irrigated systems planning due to increasing water scarcity. Also, it is generally expected that an increasing water supply will be required to meet growing irrigation demands into the next century; therefore, the relatively large amount of water used by irrigation is often targeted for water saving by governmental agencies. For example, the annual water utilization is about 184 * 10^6 m^3 in Taiwan. The annual water was about 123 * 10^6 m^3 for irrigation,16 * 10^6 m^3 for industry, 19 * 10^6 m^3 for municipality,and 26 * 10^6 m^3 for others. The irrigation water was as high as 67%of annual water use. At the time of scarcity water in Taiwan, it is the most important work to improve the irrigation water management to save water for other purposing.

Irrigation system planning is a typical optimization problem because it includes complicated components such as crop, soil, weather, and water supply. At the beginning of each year, irrigation manager should have irrigation programs and optimal cropping pattern to maximize the net benefit and water use efficiency for the irrigated systems. Linear or dynamic optimization is the traditional tool for irrigation managers to make this kind of decision.

Genetic Algorithm (GAs) is a new tool for optimization problem. Goldberg(1989) concluded genetic algorithms are different from more normal optimization and search procedures in four ways:(1) GAs work with a coding of the parameter set, not the parameters themselves (2) GAs searches from a population of points,not a single point (3) GAs use objective function information, not derivatives or other auxiliary knowledge (4) GAs use probabilistic transition rules, not deterministic rules. Also , Goldberg (1989) stated that the genetic algorithms(GAs) is an example of a search procedure that uses random choice as a tool to guide a highly exploitative search through a coding of a parameter space. Using random choice as a tool in a direct search process seems strange at first, but nature contains many examples.

Genetic Algorithms (GAs) is a new field for irrigation engineering but GAs should be a useful tool to optimize the complicated components in irrigation systems planning. Although less literatures direct related to irrigation planning,GAs has quickly been applied to number of optimization problems (Wentzel et al.,1994; Fahmy, 1994; McKinney, 1994; Wang, 1991). Wentzel (1994) used GAs to optimize the pipe network pumping strategy in New Mexico State University. Fahmy (1994) used GAs to economic optimization of river management. Wang(1991)introduced a genetic algorthm for function optimization and applied to calibration of a conceptual rainfallrunoff model for data from a particular catchment. McKinney (1994) incorporated GAs with groundwater simulation model to solve the three groundwater management problems: maximum pumping from an aquifer; minimum cost water supply development; and minimum cost aquifer remediation.McKinney also stated that the formulation of the method is straightforward and provides solutions which are as good as or better than those obtained by linear and nonlinear programming. Constraints can be incorporated into the formulation and do not require derivatives with respect to decision variables as in nonlinear programming.

From the previous statement, it would be interesting to use GAs to optimize irrigated systems planning to use water efficiency and maximize the net benefit from them.

Genetic Algorithms and An Simple Example

John H. Holland at the University of Michigan is recognized as the pioneer of genetic algorithms. Holland's (1975) book, Adaptation in Natural and Artificial Systems, established the basic mathematical theory of genetic algorithms. In1989, Goldberg's book, Genetic Algorithms in Search, Optimization, and Machine Learning, was the most complete and comprehensive work for a textbook or a self-study guide on GAs field.

Goldberg (1989) stated that a "simple genetic algorithms (SGA)" is composed of three operators:(1) reproduction (2) crossover (3) mutation. A "roulette wheel" idea is used in the three operators of SGA. Reproduction is an operator that individual strings are copied according to their objective values. Copying strings according to their fitness value means that strings with a higher value have a higher probability of contributing one or more offspring in the next generation (Goldberg, 1989). At the time of mating pool has been filled, the crossover is performed with the pair to develop the new strings of next generation,if the random number is less than the probability of crossover (P_c). Next,a crossover position is selected at random. Both "parent" strings are broken at this position and exchange the portion old string beyond this position with their partner. The two "child" strings produced by this operation become members of the next generation. Finally, the mutation operator is performed on a bit-by-bit basis by changing its value from either 0 to 1 or 1 to 0, if the random number is less than the probability of mutation (p_m). The mutation normally plays a secondly role in the GAs because mutation is not beneficial and are allowed to occur at only a very modest rate.

It is probably wise to do a small "hand calculation" as an example to describe the idea of SGA. Wentzel(1993) described a simple example to consider the SGA process as follows. Consider one simple objective function:f (x)=-0.25 $\times^{2x}$ 8 x +1 where x is an integer value on the range from 0 to 31.Our problem is to maximize the objective function. Our SGA operators will be reproduction and crossover only, and the mutation rate will be assumed too small to play an insignificant role in the problem. The select four population size and the initial values of x are chosen at random as 18, 4, 24, and 31. The first column of Table 1 shows these four values. These four values can be encoded into binary strings and are shown in the second column of Table 1. The fitness values of each string is calculated based on the objective function, along with the sum fitness, average fitness, and maximum fitness are shown in the third column.The probabilities for each string are calculated by individual fitness divided by the sum fitness and are shown in the fouth column. The number of times that we would expect each string to appear in the mating pool is the population size (i.e. 4) times the selection probability. These values are shown in the fifth column. Column six shows the actual number of each strings be selected into the mating pool and the mating pool is shown in column seven. The crossover operator is performed to create the next generation of strings at the time of mating pool is assembled. Column eight and nine show the partners for each strings and the crossover sites for each set of parents by randomly. To understand how crossover is achieved, we consider the first set of parents(i.e. first and third string)in the mating pool and the second possible position is chosen as the break point by randomly. Before crossover, the strings look like as follows (where "¦" is the break position):

1 0 ¦ 0 1 0
0 0 ¦ 1 0 0
^ ^ ^ ^
Breakpoints 1 2 3 4

After crossover, the child strings look like as follows:

1 0 ¦ 0 1 0
0 0 ¦ 1 0 0

Table 1: Simple Genetic Algorithm Example, First Iteration (Wentzel, 1993)

(1) i	(2) First Population	(3) f(i)	(4) Selection Probability	(5) Expect Number	(6) Actual Number	(7) Mating Pool	(8) Partner	(9) Break Point	(10) Second Generation	(11) i	(12) f(i)
18	10010	64	0.42	1.70	2	10010	3	2	10100	20	61
4	00100	29	0.19	0.76	1	10010	4	3	10000	16	65
24	11000	49	0.33	1.30	1	00100	1	2	00010	2	16
31	11111	8.75	0.06	0.23	0	11000	2	3	11010	22	56
	sum	150.75	1.00	3.99	4						198
	avg	37.69	0.25	1.00	1						49.5
	max	64	0.42	1.70	2						65

Table 2: Simple Genetic Algorithm Example, Second Iteration

(1) i	(2) First Population	(3) f(i)	(4) Selection Probability	(5) Expect Number	(6) Actual Number	(7) Mating Pool	(8) Partner	(9) Break Point	(10) Second Generation	(11) i	(12) f(i)
20	10100	61	0.31	1.23	2	10100	4	2	10010	18	64
16	10000	65	0.33	1.31	1	10100	3	4	10100	20	61
2	00010	16	0.08	0.32	0	10000	2	4	10000	16	65
22	11010	56	0.28	1.13	1	11010	1	2	11100	28	29
	sum	198	1.00	3.99	4						219
	avg	49.5	0.25	1.00	1						54.8
	max	65	0.33	1.31	2						65

The new population of strings is shown in column ten. The decoded x values and fitness values for these strings are shown in column eleven and twelve,respectively. By comparing the fitness values of initial (i.e. column 3) and(i.e. column 12) second population in Table 1, it is obvious that the sum,average, and maximum string fitness have improved from 150.75 to 198, 37.69 to49.5, and 64 to 65, respectively. Continuing the previous procedure, Table 2 shows the process and results of the second iteration. Also, the sum, average,and maximum fitness values have improved. The simple genetic algorithm is working from the simple example.

Genetic Algorithms and Irrigation Systems Planning

The typical research can divided into three major activities. The first can involve the design of a main model with a user-friendly interface for operation the mode. The second activity can develop a genetic algorithm submodel (GAM) to search the optimal combinations such as water supply, cropping pattern and related areas for the irrigated systems. The third process can develop an irrigation simulation submodel (ISM) to cooperate with the genetic algorithm submodel to estimate crops' relative yield, water requirement, and allocation water to each command area if the water demand is greater than supply. The specific procedures of three activities are described below.

(A) Main Model

Traditional irrigation water management models do not provide a user-friendly interface. Furthermore, the results are usually displayed in numerical form through long printouts, which exacerbate the problem of data interpretation and decision making in the simulation environment. Thus, there is a need for an improved management model that would provide a user-friendly interface to help facilitate operation of the model. The main model can include a pull down menu, sample data file, data edit and help window. The top menu can include File (e.g. new,open, save), Edit (e.g. project data, weather data, crop data, soil data and economic data etc.), Run, Results (e.g. tablur, graphics, print) and Help. The user interface can let user friendly to operate the model. The sample data file can help the user run the model easily and the data editing capability can let the user input and change data easily; furthermore, the help window can help the user operate the model. At the time of "run"items be clicked by mouse or keyboard, the GAM submodel can be called and cooperate with the IGM submodel to begin simulation work. The results can be shown by Tables, graphics and print outs.

(B) Genetic Algorithms Submodel

The GAM submodel can compose of the following steps:(1) default data:population size, number of generations (iterations), probability of crossover(P_c) and mutation (p_m) (2) set string lengths to represent research problem (3) randomly generate an initial population of binary strings (4)decode binary strings to real number to represent parameters such as crop types and related area, conjunctive water supply etc. (5) call the ISM submodel to find the fitness value by objective function. During this step, the penalty method can used to transform the constrained problem in optimization into an unconstrained problem by associating a cost or penalty with all constraint variations as follows (Goldberg, 1989):

Maximize: $g(x)$

subject to: $h_i(x)>=0$ $i=1,2,\ldots,n$

where x is an m vector

We transform this to the unconstrained form:

Maximize $g(x)+r. \Sigma\, \Phi\, [h_i(x)]$

where Φ is penalty function

r is penalty coefficient

(6) continue subsequent generations in three steps of reproduction, crossover and mutation. The simulation procedures can repeat from step (4) to (6) and stop at the time to meet the maximum number of generation.

(C) Irrigation Simulation Submodel

The ISM submodel can be called by the GAM submodel many times to calculate the fitness values based on the objective function. First, the ISM submodel can calculate the crop evapotranspiration by Penman-Monteith method or other methods. The effective rainfall can be calculated by one of the following methods:fixed percentage, dependable rain, empirical formula, and USDA Soil Conservation Service Method (Smith, 1991). Second, the water demand for each crops can be summed together to determine the system demand based on the irrigation schedule types. If the system demand is greater than the water supply, water can be allocated to each command area by one of the following four methods (Prajamwong, 1994): (1)Proration (2) Full Proration (3) Equity (4) First come first serve. Third, the on farm soil water balance for each crops can be simulated on a ten-day or monthly interval to determine the relative crop yield as a function of the soil moisture. The relative yield equation recommended by the FAO Irrigation and Drainage Paper NO.33 (Doorenbos et al. 1979) can be used to calculate the"actual " crop yield as follows:

$$\frac{Y_a}{Y_P}=1-\sum_{i=1}^{NoStage}[K_{yi}*(1-\frac{ET_a}{ET_P})]$$

where Y_a is actual crop yield; Y_p is potential crop yield; K_{yi} is the crop reduction coefficient in each stage; ET_a is actual evapotranspiration;and ET_p is potential evapotranspiration. It is obvious that the relative crop yield will be influenced by the irrigation schedule types and water allocation.The crop yield reduction can be calculated by each stage and multiplied by together to get the seasonal relative crop yield. Finally, the objective function can be used to calculate fitness value to obtain the maximum net benefit and water use efficiency for the irrigated systems. The objective function can include many factors such as: crop and water price, crop relative yields,energy cost, fertilizer cost, labor cost etc.

The general program logic can show as follows:

```
User-friendly interface to operate model ;
Input population size, number of generation, probability of crossover and
mutation ;
Random initial population ;
for I=1 to number of generation
{
  for J=1 to population size
  {
    Decode binary strings to represent real number ;
    Call Irrigation Simulation Submodel to find fitness values ;
  }
  Three simple genetic algorithms operators: reproduction, crossover, and
  mutation ;
}
Show results by graphics, tables, or print outs ;
```

Conclusion

A genetic algorithm approach has been introduced to decision support for irrigation systems planning. As demonstrated here, GAs have four ways that different from normal optimization method:(1) GAs work with a coding of the parameter set, not the parameters themselves (2) GAs search from a population of points, not a single point (3) GAs use objective function information, not derivatives or other auxiliary knowledge (4) GAs use probabilistic transition rules, not deterministic urles. The penalty method can be incorporated to transform the constrained problem in optimization into an unconstrained problem by associating a cost or penalty with all constraint variations and do not require derivatives with respect to decision variables as in nonlinear programming. For complicated components of irrigation system planning problem, the parallel implementation of the GA with the irrigation simulation model will likely be required.

References

Dejong, K.A. (1975). *An analysis of the behavior of a class of genetic adaptive systems,* Ph.D. Dissertation,

（文轉第10頁）