OPTIMAL DESIGN OF PIPE NETWORK BY AN IMPROVED GENETIC ALGORITHM

HOSSAM A. A. ABDEL-GAWAD

Irrigation and Hydraulic Department, Faculty of engineering,
Mansoura University, Egypt

ABSTRACT

In this paper several alternative formulations of a genetic algorithm (GA) for pipe network optimization have been developed. This has been done with a view to presenting fundamental guidelines for implementation of the approach to practical problems. Alternative representation of selection, crossover, and mutation schemes are considered. A FORTRAN Program created to solve these different alternatives. It is concluded that the most promising/improved genetic algorithm approach for optimal design of pipe network problem comprises real-value coding, tournament selection, uniform crossover, and modified uniform mutation. Results are presented comparing the performance of different configurations of GA's and the traditional/simple GA for the New York City water supply expansion problem. The lowest-cost feasible solution in far fewer generations yet presented in the literature has been obtained through the improved GA, in respect to both of earlier solutions of GA and linear/nonlinear optimization approaches. Thus improved GA approach developed in this researcher can be considered as the most satisfactory robust formulation of GA's, in spit of its easiness for application to complex systems. It has potential as a powerful alternative to stochastic dynamic programming approaches.

INTRODUCTION

The distribution networks are an essential part of all water supply systems. The cost of this portion of any sizable water supply scheme amount to more than 60% of the entire cost of the project. Also, the energy consumed in a distribution network is more than 80% of the total energy consumption of the system [Sarbu I., 1997]. The high investment and maintenance costs associated with both new water distribution networks and the expansion of existing have led hydraulic engineers to take great interest in mathematical methods to find their optimal design, that is, the minimum cost network. Over the last 3 decades a large number of optimization models have been developed. Most of these models start from a given layout of the hydraulic elements (pipes, valves, tanks, etc.) and a set of specified demand patterns to find the minimal cost network. Two different optimization approaches are applied to solve these models.

The first one is the deterministic approach which consists of linear programming LP [Fujiwara O., 1987; Sonak V.V., 1993; Sarbu I., 1997], and nonlinear programming NLP [Lansey E. K., 1989; Djebedjian B., 2000]. Pipe network optimization problem has a high degree of nonlinearly due to dependence of unknown pipe discharges, which satisfying the nodal demands and hydraulic constraints, on unknown pipe diameters. The general idea of LP is to relax this non-linearity by assuming a particular flow pattern (or removing one pipe from each loop). Then, applying LP to obtain the gradient of the objective function GOF and best pipe diameters. Using this GOF, the flow pattern must be revised. The two stage iterative procedure continued until no further reduction in cost occurred. In the other hand, in NLP the unknown pipe discharges are represented as a function of the unknown pipe diameters by satisfying the hydraulic conditions. Then pipe network optimized to get the minimum cost under hydraulic and design constraints. In general, the main drawback of deterministic approach is the nature of optimized domain conformation. Sonak V.V. (1993) proved that this domain is convex in general between the points represent minimum values of objective function corresponding to sequential assumptions of zero flow rate for one selected pipe from every loop, the optimized domain/surface between these points are represented by a concave surfaces. Thus a multiple local minimum created between these surfaces at the preceding mentioned points. Thus the optimized domain is a concave-convex. Depending on starting point/solution of optimization the search -which always stick with the surface of optimized domain and moves down- will be trapped at the nearest local minimum. So, the final solution achieved by the deterministic approach is always depending on the start point and the complexity of the optimized domain/ number of unknowns. To alleviate this bad behavior of deterministic methods the optimization procedure must be repeated with different starting points, the lowest local minimum achieved may coincide with the global one.

The second optimization approach is the stochastic one including both of simulated annealing method [Cunha M.C., 1999], and genetic algorithm [Simpson A.R., 1994; Dandy G.C., 1996; Savic D.A., 1997; Reis L.F.R., 1997; Montesinos P., 1999]. The main advantage of stochastic approach is the randomness procedure during the search which enables the optimization process to jump or release from any trapped local minimum to a lowest one, thus increasing probability of catching the global one. Difficulties adherent with deterministic approach due to the necessity of accurate calculation for the first/second derivatives for the objective function are avoided through the stochastic approach. Also, stochastic approach can deal directly with discrete/commercial sizes of pipe diameters, in contrast to deterministic approach which deals with continuous parameters/pipe diameters, and converges final solution to nearest commercial pipe size which weaken the solution.

The idea of simulated annealing came from observing how thermodynamic system solve problem, it is an appropriate method in case of existing approximately similar local minimums. If the global minimum is much better than all other local ones, optimized domain must be discovered by an ensemble of unknown parameters/(starting points) to explore more parts of the optimized space. GA that represents the process of natural selection in biology can achieve this goal [Gershenfeld N., 1999]. The study of genetic algorithms originated in mid 1970s by Holland and has developed into a powerful optimization approach. Goldberg (1989) gives excellent introductions to GA. There have been several application of GA to water resources problems, e.g., optimal reservoir system operation [Wardlaw R., 1999], optimal cleanup of aquifer/groundwater [Aly A.H., 1999], optimal design of groundwater monitoring systems [Cieniawski S.E., 1995], groundwater management [McKinney D.C., 1994].

PROBLEM FORMULATION

Given the layout of hydraulic elements and a set of specified demands at nodes, the optimal design of a looped network for gravity systems is defined by the set of pipe sizes which results in the minimum investment cost. Thus objective function f is assumed as a cost function of pipe diameters and lengths:

$$f(D_1, \ldots\ldots, D_n) = \sum_{i=1}^{N} c(D_i, L_i) \qquad (1)$$

where $c(D_i, L_i)$=cost of pipe i with diameter D_i and length L_i (L); and N = total number of pipes in the system. The objective function is to be minimized under the following constraints. For each junction node (other than the source) a continuity constraint should be satisfied, Then:

$$\sum Q_{i,in} - \sum Q_{i,ex} = Q_{i,nod} \qquad \forall\, i \in NN \qquad (2)$$

where $\sum Q_{i,in}$ = flow into the junction (M^3 / L), $\sum Q_{i,ex}$ =flow out of the junction (M^3 / L), and $\sum Q_{i,nod}$ =demand at the junction (M^3 / L), NN= number of junctions/nodes. For each of the basic loops in the network the energy conservation can be written as:

$$\sum_{i=1}^{NL} h_j = 0 \qquad (3)$$

where NL = number of pipes for the studied loop and h_j = head loss through any pipe j (L). The Hazen-Williams formula will be used to represent the head losses as:

$$h_j = w \frac{L_j}{C_j^{\beta} D_j^{\gamma}} Q_j \left| Q_j \right|^{B-1}, \qquad \forall \quad j \in N \qquad (4)$$

where w = numerical conversation constant (depending on used units); C = roughness coefficient (depending on material of pipe network); β *and* γ =regression coefficients. Design constraints as size of pipes, velocity and pressure bounds/(maximum and minimum limits) can be represented by the following equations:

$$D_{\min} \le D_j \le D_{\max} \qquad \forall\; j \in N \qquad (5)$$

$$V_{\min} \le V_j \le V_{\max}, \qquad \forall j \in N \qquad (6)$$

$$P_{\min} \le P_i \le P_{\max}, \qquad \forall i \in NN \qquad (7)$$

where V = velocity through pipes (*L/T*), P = pressure head (L).

Starting from any pipe set for the network under condition, Eq. (5), and initial estimate for pipe flows satisfying, Eq. (2), a steady state hydraulic analysis of the pipe network design is performed to assess both of hydraulic and design feasibility for the proposed system. This hydraulic analysis involves the prediction of the flows/velocities in the pipes and pressure heads at the nodes of the pipe network under steady state conditions. The hydraulic analysis method adopted in this study uses the Netwon-Raphson technique applied to the set of simultaneous nonlinear algebraic equations in terms of the unknown flow corrections around the loops [Watters G.Z., 1984].

SIMPLE GENETIC ALGORITHM (SGA)

A GA is a search algorithm based upon the mechanism of natural selection, derived from the theory of natural evolution. GAs simulate mechanisms of population genetics and natural rules of survival in pursuit of the ideas of adaptation. Indeed this led to a vocabulary borrowed from natural genetics [Gen, M., (2000)]. The brief idea of GA is to select population of initial solution points (strings/chromosomes) scattered randomly through the optimized space, then converge to better solutions by applying in iterative manner the following three processes (reproduction/selecting, crossover and mutation) until a desired criteria for stopping is achieved. The steps in using SGA for pipe network optimization are repeated here for completeness [Dandy, G.C., 1996]:

1. ***Generation of initial population.*** The GA randomly generates an initial population of coded strings representing pipe network solutions of population size equal to *NP*. Every coded string is similar to the structure of a chromosome of genetic code. Consider a coded string consisting of 21 coded substrings each of 4 binary bits. This coded string of 84 binary bits may, represent a design for a simple pipe network consisting of 21 pipes. Each 4-bit substring represents one of the 16 possible choices of pipe size for one of the 21 pipes in the network. A selected mapping between coded substrings and the design variables associates the artificial genetic code with a pipe network design. Each bit position in any string takes on a value of either 0 or 1. Any of the *NP* strings through the random starting population represents a possible combination of pipe sizes and thus represents different configuration of a pipe network.
2. ***Computation of network cost.*** The GA considers each of the *NP* strings in the population in turn and computes the total material and construction cost (objective function).
3. ***Hydraulic analysis of each network.*** A steady state hydraulic network solver computes the heads and discharges under the specified demand for each of the network designs in the population. The actual nodal pressures and pipe velocities are compared with the corresponding minimum and maximum limits. Any violation to any of the pressure and/or velocity is noted.
4. ***Computation of penalty cost.*** Chromosomes may be generated that fail to meet hydraulic constraints. Such chromosomes could be excluded from subsequent participation in the evolutionary process, but this may lead to useful genetic material being lost that disrupts the GA process and in effect requires many additional generations. Alternative to exclusion the GA assigns a penalty cost for each pressure/velocity violation. Different philosophies may be used to represent the penalty function. Some researches used a penalty function proportion to the degree of violation; therefore some strings attached very low fitness due to one violation which causes high degree of violation, and consequently a low chance of selection in the next population. This led to loss some features about the optimized space. In this paper constant penalty value was associated to any violation. This procedure more sensitive to number of undesired diameters selected.
5. ***Computation of total network cost*** as the sum of the network cost (step 2) plus the penalty cost (step 4).

6. ***Computation of the fitness.*** The fitness of each string is taken as a function of the total network cost. In general, GA computes the fitness for each proposed network in the current population as the inverse of the total network cost from (step5).
7. ***Generation of a new population using the selection operator.*** The GA generates/reproduce new members of the next generation by a selection scheme. SGA produce the next generation by creating a biased roulette wheel where each current string in the population has a roulette wheel slots sized in proportion to its fitness. Thus probability of choosing any string *i*, p_i, is:

$$p_i = f_i / \sum_{j=1}^{j=NP} f_j \tag{8}$$

   where f_i is the fitness of string *i* determined in (step6).
8. ***The crossover operator.*** Crossover is the partial exchange of bits/genes between two parent strings to form two offspring strings. Crossover occurs with some specified probability of crossover p_c for each pair of parent strings selected in (step 7). To perform one-point crossover, a crossover operator exchanges the bits/genes after the crossover point between the two-selected parent strings.
9. ***The mutation operator.*** Mutation is an important process that permits new genetic material to be introduced to a population. Mutation occurs with some specified probability of mutation p_m for each bit/gene in the strings, which have undergone crossover. The bitwise complement mutation operator changes the value of the bit to the opposite value (i.e., 0 to 1 or 1 to 0).
10. ***Production of successive generations.*** The SGA replaces the new generation with the old one and repeats (steps 2 to 9). This process usually increases the performance/fitness of the stings, and consequently converges the search algorithm to an optimal solution.

AN IMPROVED GENETIC ALGORITHM FORMULATION

Traditionally GAs have used binary coding. Standard binary coding of variables permits large jumps in variable values between generations, which can lead to difficulty in converging to good solution. This can overcome to some extent through the use of Gray coding in which the binary representation of two adjacent variable values changes by only one binary digit. An alternative approach to formulation of the GA is to use a representation appropriate to the components of the problem. Therefore, in this work real-value chromosomes have been used. That is a discrete real values between upper and lower limits are used instead of using a binary or Gray codes to represent the decision variables. This choice saves the computation time required for mapping the coded strings to the acceptable space of solution for the decision variables. Thus, every gene represents a decision variable and has a value lie between upper and lower limits of this variable. Using real-value representation reduces lengths of chromosomes and logically decreases number of iterative solutions required for convergence to an optimal solution, as a result in decreasing dimension of the optimized space due to shortest strings produced by real-value code.

Selection is the procedure by which chromosomes are chosen for participation in the reproduction process. The two most important issues in the genetic search process that greatly affected by the selection process are population diversity and selective pressure. Population diversity is maintained by exploiting good strings, while still exploring the rest of the search space. It is influenced by the degree to which the best strings are favored which may be termed selective pressure. A high selective pressure may lead to a rapid convergence- but convergence to suboptimal solution- while a low selective pressure may result in a greater number of generations being required before convergence to an acceptable solution is achieved. In general, the easiest way of selection is to create a biased roulette wheel as described in (step 7). A drawback of this type of selection is the existing of some extraordinary strings in the starting population, consequently a very low probability of selection are attached to these strings, giving a higher chance for good strings to overflow the search process. So, instead of exploring through the whole optimized space GA will be restricted to finite spaces corresponding to the zones containing good strings. Also, at the end of the search process most of strings have a similar finesses, so more sensitive selection scheme must be used. In this research the tournament selection was used. In tournament selection a group of strings are chosen at random from the population and string with the high fitness is selected for reproduction. This procedure is repeated until appropriate number of strings is selected for the new generation. As the number of strings used in the tournament selection increases, the selective pressure increases, so two strings are used in this work.

Three types of crossover operation are used: 1) one-point crossover, 2) two point crossover, and 3) uniform crossover. For the three crossover operations only one child are created from two selected parents. In one point crossover, a crossover point is selected uniformly at random through the length of the string. Summation of genes lies before crossover point for the first parent and that after crossover point for the second parent creates the new child/string. In two point crossover, genetic material between two positions chosen at random along the length of chromosome is exchange. Uniform crossover operates in individual genes of the selected string, rather than on blocks of genetic material, and each gene is considered in turn for crossover or exchange.

Two approaches for mutation are used. First approach: uniform mutation that permits the value of a gene to be mutated randomly within its feasible range of values. Second approach, modified uniform mutation that permits modification of a gene by a specified amount, which may be either positive or negative, one diameter larger/smaller was selected.

Last modification applied to SGA is the replacement strategy. Replacing the new generation by the old may cause to loose fitter strings included in the old generation. So the principle of tournament selection is used also to the replacement policy. As in the selection mechanism, a group of strings are chosen at random from the old population the worst string are exchange with a new string from the new generation. This process is repeated successively for all strings in the new generation. This strategy ensures that the best performing chromosome of the previous generation are stored in the current population, so the elitist strategy is preserved.

CASE STUDY: THE NEW YORK CITY WATER SUPPLY TUNNELS PROBLEMS

A number of studies in pipe network optimization have examined the expansion of the New York water supply system. In the last decade several researches are concentrated in using GA to solve this problem [Dandy, G.C., 1996; Savic, D.A., 1997; Montesinos, P., 1999]. The common objective of the studies was to determine the most economically effective design for additions to the then-existing system of tunnels that constituted the primary water distribution system of the city of New York (Fig. 1). The same input data, e.g., existing pipe data, discrete set of available diameters, and associated unit pipe costs, were used in this study as shown in table (1). The Imperial system of units was used to enable easy comparison with previous studies. Because of age and increased demands the existing gravity flow tunnels have been found to be inadequate to meet the pressure requirements (at nodes 16, 17, 18, 19, and 20) for the projected consumption level. The proposed method of expansion was the same as in previous studies, i.e., to reinforce the system by constructing tunnels parallel to the existing tunnels. For 15 available diameters, i.e., 16 possible decisions including the do nothing option and 21 pipes to be considered for duplication, the total solution space is $16^{21} = 1.93 \times 10^{25}$ possible network designs. For all new and existing pipes a Hazen-Williams roughness coefficient equal to 100 is considered. In this study $w,\ \beta\ and\ \gamma$ are taken equal to 4.727, 1/0.54, and 2.63/0.54 respectively. Maximum flow velocity in any pipe restricted to upper level equal to 8.2 *feet/s*. Penalty is considered as a constant value for each violation equal to 170 millions $.

RESULTS AND DISCUSSION

A comparison between different formulations of modified GAS is represented in Fig. (2). The presented results are based on constant number of population *NP* equal to 100 with constant number of generations equal to 200. Due the randomness behavior of the GA process, different seed numbers lead to different results. So, to explore the behavior of different GA formulations/ alternatives in more sensitive way, the whole results are averaged for ten sequential runs with different starting seeds for the same search/formulation processes. Figure (2) shows the behavior of different alternatives considered in this research; those are the modified uniform or uniform random mutation plus any one of the three types of crossover. Probability of Crossover is considered as a constant value equal to 0.5 for the six GA alternatives, while probability of mutation represented by a wide range from 0.01 to 0.3. From the results it can be noticed that high average cost (for the ten different seed numbers used to start the runs) are obtained for the six GA alternatives when the probability of mutation are small than 0.07. Minimum average cost are achieved for probability of mutation between [0.08 to 0.1] for all GA alternatives, while GAs that used modified uniform mutation gave a better response for a longer range of mutation ration, from [0.08 to 0.3], especially in case of using two-point and uniform crossover. Best response obtained by using the modified uniform mutation in combination with uniform crossover, so this improved GA formulation used for the remaining of the paper.

Minimum cost of construction achieved through the whole runs equal to 38.79 millions $, with suggestion to construct a parallel pipes to the old network as: number of pipe (diameter in inch) [15(120), 16 (84), 17(96), 18(84), 19(72), and 21(72)]. The same results are obtained by both of [Dandy, G.C., 1996; Montesinos, P., 1999]. Anyway, in this research the improved GA reaches to the optimal solution using smaller number of string evaluations [less than 4,500 to 9,000], instead of 44,280 for Montesinos (1999), and [96,750 to 151,400] for Dandy (1996).

Figures (3, 4, 5) show the effect of probability of mutation on number of global solutions achieved in 10 runs, average cost, and average number of string evaluations respectively. It is clear that as the probability of crossover increases to 1.0 the algorithm behaves badly and vises versa. Maximum number of global optima achieved in 10 sequential runs is 5, at probability of crossover equal to 0.5 and probability of mutation equal to 0.14. In general there is at least on global minimum always achieved, in ten runs, for any probability of mutation and crossover = 0.5.

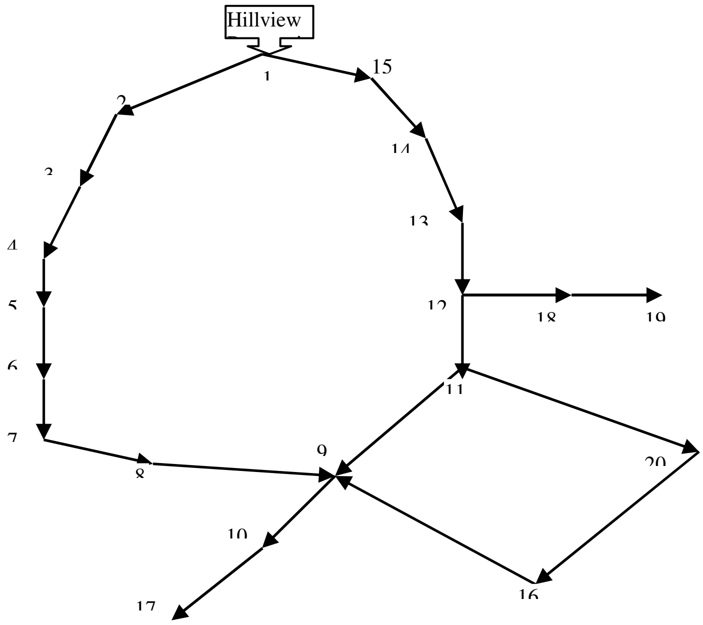
CONCLUSION

Different alternatives for GA are presented in this paper. For the example network studied, the best formulation composed from uniform crossover, modified uniform mutation, constant value of penalty and finally tournament strategy for both of selection/reproduction and replacement steps. The best GA formulation reached to optimal solution in far fewer generations than any previous GA. One significant advantage of the best formulation of GA is its robustness for wide range of operators [crossover and mutation].

REFERENCES

1. Aly, A.H., and Peralta, R.C., (1999), "Optimal Design of Aquifer Cleanup System Under Uncertainty Using a Neural Network and Genetic Algorithm", Water Resources Research, Vol. 35, No. 8, pp. 2523-2532.
2. Cieniawski, S.E., and Eheart, J.W., Ranjithan S., (1995), "Using Genetic Algorithms To Solve a Multiobjective Groundwater Monitoring Problem", Water Resources Research, Vol. 31, No. 2, pp. 399-412.
3. Cunha, M.C., and Sousa, J., (1999), "Water Distribution Network Design Optimization: Simulated Annealing Approach", Journal of Water Resources Planning and Management, Vol. 125, No. 4, pp. 215-221.
4. Dandy, G.C., Simpson, A.R., and Murphy, L.J., (1996), "An Improved Genetic Algorithm for Pipe Network Optimization", Water Resources Research, Vol. 32, No. 2, pp. 449-458.
5. Djebedijan, B., Herrick, A., and Abou-Rayan, M., (2000), "An Investigation of The optimization of Portable Water Network", Fifth International Water Technology Conference IWTC, Alexandria, Egypt, pp.61-70.
6. Fujiwara, O., Jenchaimahakoon, B., and Edirisinghe, N.C.P., (1987), "A Modified Linear Programming Gradient Method for Optimal of Looped Water Distribution Network", Water Resources Research, Vol.23, No. 6, pp. 977-982.

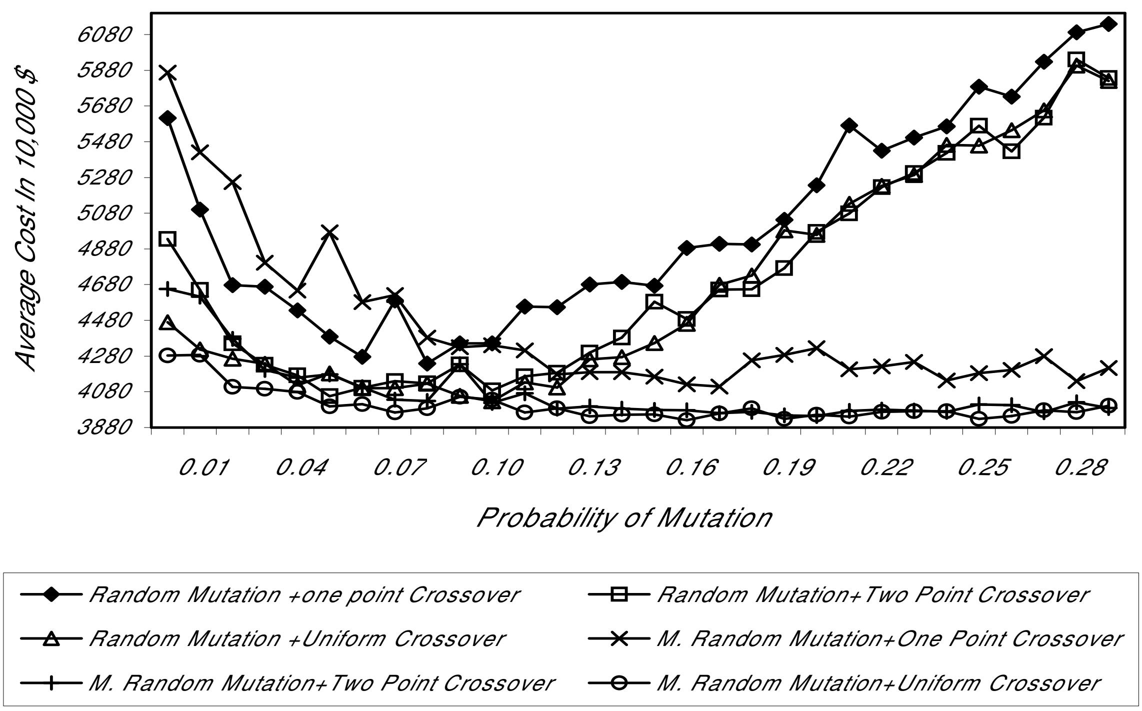
7. Gen, M., Cheng, R., (2000), "Genetic Algorithms and Engineering Optimization", Wiley-Inter-science Publication, p. 495.
8. Gershenfeld, N., (1999),"The Nature of Mathematical Modeling", Cambridge University Press, p. 344.
9. Goldberg, D.E., (1989), "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley, Reading, Mass.
10. Lansey, K.E., and Mays, L.W., (1989), "Optimization Model for Water Distribution System Design", Journal of Hydraulic Engineering, Vol. 115, No. 10, pp. 1401-1418.
11. McKinny, D.C., and Lin M.-D., (1994), "Genetic Algorithm Solution of Groundwater Management Models", Water Resources Research, Vol. 30, No. 6, pp. 1897-1906.
12. Montesinos, P., Garcia-Guzman, A., and Ayuso, J.L., (1999), "Water Distribution Network Optimization Using a Modified Genetic Algorithm", Water Resources Research, Vol. 35, No. 11, pp. 3467-3473.
13. Sarbu, I., (1997), "Optimal Design of Water Distribution Networks", Journal of Hydraulic Research, Vol. 35, No. 1, pp. 63-79.
14. Savic, D.A., and Walters, G.A., (1997), "Genetic Algorithms for Least-Cost Design of Water Distribution Networks", Journal of Water Resources Planning and Management, Vol.123, No. 2, pp.67-77.
15. Simpson, A.R., Dandy, G.C., and Murphy, L.J., (1994), "Genetic Algorithm Compared to Other Techniques for Pipe Optimization", Journal of Water Resources Planning and Management, Vol. 120, No. 4, pp. 423-443.
16. Sonak, V.V., and Bhave, P.R., (1993), "Global Optimum Tree Solution for Single-Source Looped Water Distribution Networks Subjected to Single Loading Pattern", Water Resources Research, Vol. 29, No. 7, pp. 2437-2443.
17. Wang Q.J., (1991), "The Genetic Algorithm and Its Application to Calibrating Conceptual Rainfall-Runoff Models", Water Resources Research, Vol. 27, No. 9, pp. 2467-2471.
18. Ward, R., and Sharif, M., (1999), "Evaluation of Genetic Algorithms for Optimal Reservoir System Operation", Journal of Water Resources Planning and Management, Vol. 125, No. 1, pp. 25-33.
19. Watters, G.Z., (1984), "Analysis and Control of Unsteady Flow in Pipelines", An Ann Arbor Science Book, pp. 349.



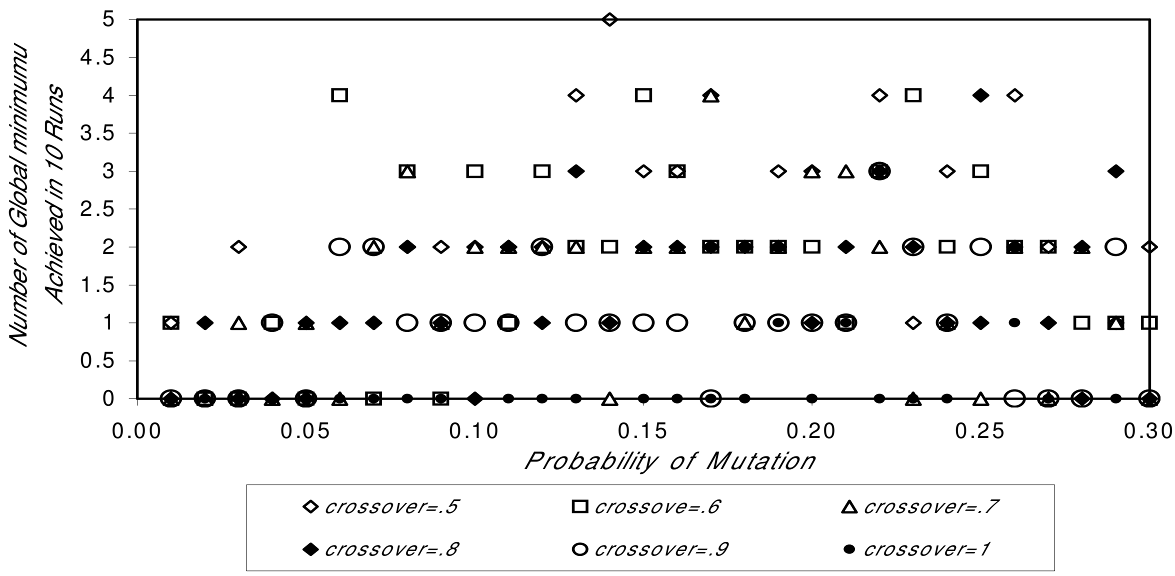
(Fig.1) Layout for the New York City Water Supply System

Table 1. Geometric Data, Nodal Requirement and Cost of Available Pipe Sizes for the New York Supply System.

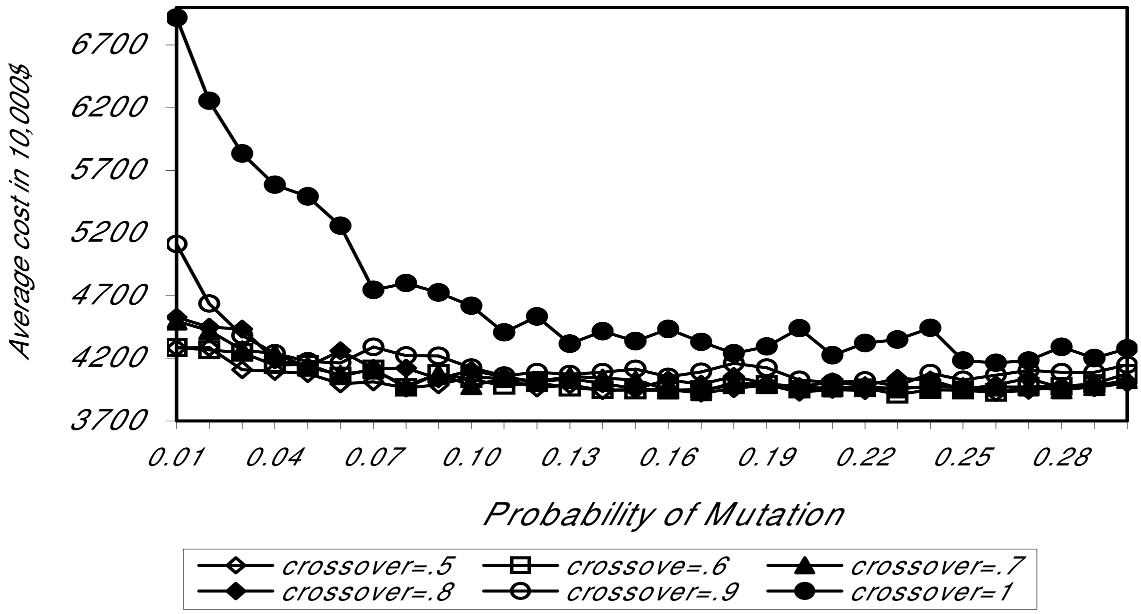
PIPE	START NODE	END NODE	DIAM-ETER, INCHES	LENGTH, FEET	NODE	DEMAND ft^3/s	MINIMUM HEAD, FEET	DIAM-ETER, INCHES	COST $ /FEET
1	1	2	180	11,600	1	-	300.0	0	0
2	2	3	180	19,800	2	92.4	255.0	36	93.5
3	3	4	180	7,300	3	92.4	255.0	48	134.0
4	4	5	180	8,300	4	88.2	255.0	60	176.0
5	5	6	180	8,600	5	88.2	255.0	72	221.0
6	6	7	180	19,100	6	88.2	255.0	84	267.0
7	7	8	132	9,600	7	88.2	255.0	96	316.0
8	8	9	132	12,500	8	88.2	255.0	108	365.0
9	9	10	180	9,600	9	170.0	255.0	120	417.0
10	11	9	204	11,200	10	1.0	255.0	132	469.0
11	12	11	204	14,500	11	170.0	255.0	144	522.0
12	13	12	204	12,200	12	117.1	255.0	156	577.0
13	14	13	204	24,100	13	117.1	255.0	168	632.0
14	15	14	204	21,100	14	92.4	255.0	180	689.0
15	1	15	204	15,500	15	92.4	255.0	192	746.0
16	10	17	72	26,400	16	170.0	260.0	204	804.0
17	12	18	72	31,200	17	57.5	272.8	-	-
18	18	19	60	24,000	18	117.1	255.0	-	-
19	11	20	60	14,400	19	117.1	255.0	-	-
20	20	16	60	38,400	20	170.0	255.0	-	-
21	9	16	72	26,400	-	-	-	-	-



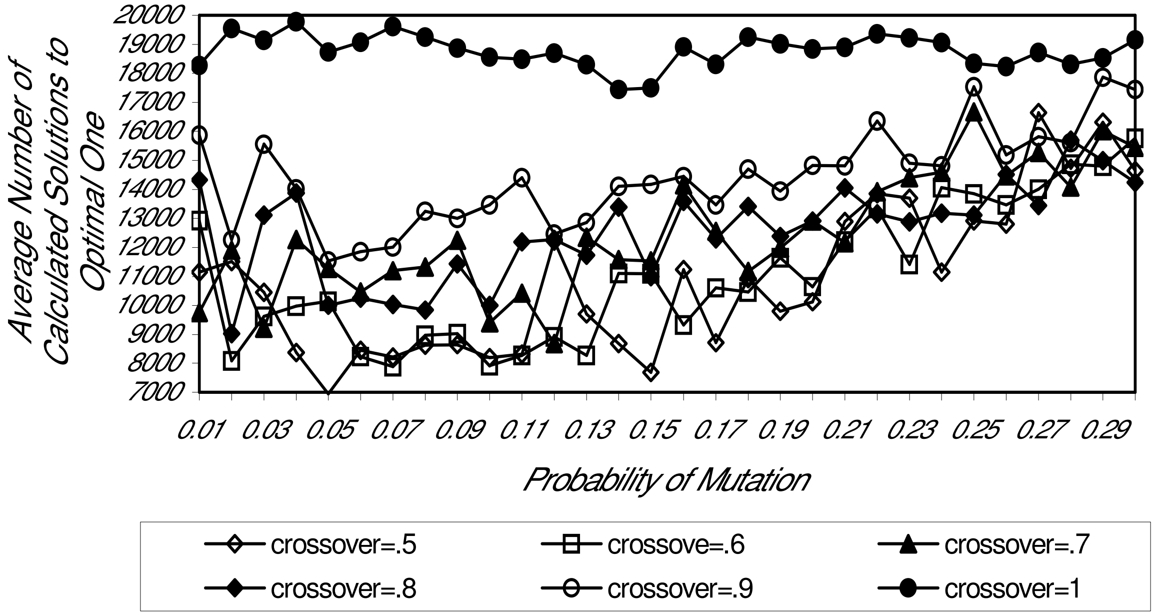
(Fig. 2) Effect of Probability of Mutation on Average Cost for 10 Runs on New York Pipe Network



(Fig. 3) Effect of Probability of Mutation on Number of Global Solution Achieved in 10 Runs



(Fig. 4) Effect of Probability of Mutation on Average Cost for 10 Runs on New York Pipe Network



(Fig. 5) Effect of Probability of Mutation on Average Number of Calculated Strings for 10 Runs to Optimal Solution