

OPTIMIZACIJA NAČRTOVANJA VODOOSKRBNIH SISTEMOV Z UPORABO GENETSKIH ALGORITMOV OPTIMIZATION OF DESIGN OF WATER DISTRIBUTION SYSTEMS USING GENETIC ALGORITHMS

Angus R. SIMPSON

V članku je podrobno predstavljen relativno nov pristop k načrtovanju vodooskrbnih sistemov. Optimizacijska tehnika, ki jo imenujemo genetski algoritmi, je uporabljena za spreminjanje in izboljšanje populacije rešitev. Le zelo zelo majhen del celotnega obsega rešitev je treba raziskati, da bi našli rešitve z nizkimi stroški, ki zadovoljujejo kriterije načrtovanja. Genetski algoritmi (GA) uporabljajo operatorje selekcije, križanja in mutacije na skupini nizov, ki predstavljajo odločitvene spremenljivke, ki jih je treba določiti kot del načrtovalnega procesa. Ti operatorji omogočajo procesom genetskih algoritmov, da hitro poiščejo poceni optimalne rešitve. Obsežna testiranja procesa optimizacije s pomočjo genetskih algoritmov na dejanskih načrtih omrežij so pokazala, da so GA učinkoviti pri iskanju poceni rešitev. Ta tehnika je konsistentno našla cenejše rešitve kot poizkusi in pogreši pristop, ki ga načrtovalci običajno uporabljajo. Poleg tega je ta tehnika učinkovitejša in preprostejša za uporabo kot tradicionalne optimizacijske tehnike.

Ključne besede: vodovodni sistemi, genetika, genetski algoritmi, metode optimiranja, oblikovanje, obnova, obratovanje sistema

This paper presents details of a relatively new approach to the design of water distribution systems. An optimization technique called genetic algorithms is used to evolve an improving population of designs. Only a very, very small proportion of the total possible search space needs to be explored in order to find low cost solutions that satisfy all of the specified design criteria. The genetic algorithm (GA) uses operators of selection, crossover and mutation applied to a set of strings that represents the decision variables to be selected as part of the design process. These operators enable the genetic algorithm process to quickly seek out low cost optimal solutions. Many tests of the application of the genetic algorithm optimization process to real-life network designs has shown that the GA is effective at finding low cost solutions. The technique has consistently found lower cost solutions than the trial-and-error simulation approach typically used by design engineers. In addition, the technique is more effective and easier to apply than traditional mathematical optimization techniques.

Key words: water distribution systems, genetics, genetic algorithms, optimization methods, design rehabilitation, network operation

1. INTRODUCTION

Water distribution systems provide communities with adequate water for drinking, domestic household usage, garden use, fire fighting and irrigation. Simulation of water distribution systems using computer models has reached a mature stage of development. Many elements can be modelled, including pipes, tanks, pumps and valves. A set of non-linear equations is solved within the simulation model. Many sophisticated and extremely capable models are available commercially or from government agencies (for example, EPANET, Stoner Synergiee, WaterCAD, Cybernet and Piccolo). Excellent user interfaces plus links to geographical information systems (GIS) are provided with many of these simulation models. These models permit simulation of the peak hour demand loading case as well as extended period simulation and fire demand loading

cases. A trial-and-error approach is usually adopted in using simulation models for the design of water distribution systems. The user chooses all of the sizes of the various components based on engineering judgement and experience (e.g. diameters of pipes, whether a pipe should be cleaned or not, locations of tanks, optimal normal water reservoir or tank operating elevations, pump location and size and pump operation schedules). The selected combination of sizes of elements is then run in the computer simulation model for all of the design demand loading cases. The results, including low and high-pressure areas, are analysed and then changes are made to the sizes of elements in the simulation model. The simulation model is then re-run for all the demand loading cases in an attempt to find lower cost solutions that satisfy the design criteria. This process is repeated over and over.

Optimization of water distribution systems has two main design objectives. The first is to minimise the sum of the capital costs plus the operating costs of the water distribution system. The second is to satisfy the design criteria. The most common design criteria are to satisfy the minimum allowable pressure constraints. There have appeared many papers on the traditional mathematical optimization of water distribution systems in the research literature since the mid-1960s (for example, linear, dynamic and non-linear programming). However, there has been virtually no transfer from the world of academia to the professional practice of the techniques developed in these technical publications. The focus of this paper is the application of genetic algorithm optimization to the design of water distribution systems. Genetic algorithms appear to have the potential to be extremely useful in engineering practice in the optimization of the design and operation of water distribution systems.

2. AN OVERVIEW OF GENETIC ALGORITHMS

A genetic algorithm (GA) is a powerful population-orientated search algorithm based upon mechanisms of natural selection and genetics (Holland 1975, Goldberg 1989). Application of genetic algorithms to the design of water distribution systems was pioneered at the University of Adelaide, South Australia in the early 1990s (Murphy and Simpson 1992). "Survival of the fittest" relentlessly drives the genetic algorithm towards improving the design of a water distribution network. The genetic algorithm selects, combines and manipulates possible solutions in a search for the lowest cost network. The operators within a genetic algorithm search are analogous to survival in nature, reproduction and the combination of chromosomes in the search of the best adaptation in natural systems. GAs are not constrained by requirements such as continuity, derivative existence, unimodality or linearity which are often necessary in other traditional mathematical optimization techniques. An essential element of GA optimization as applied to pipe networks is the ability to simulate the water distribution system in a hydraulic solver. The genetic algorithm optimizer and the hydraulic simulator are separate, but are linked by the transfer of information between the two.

In applying the genetic algorithm (GA) technique to the particular design problem being optimized, a GA deals with a population of solutions. Decision variables that are to be selected as part of the design in a piped distribution design problem are coded as strings in the genetic algorithm technique. Choices for each decision variable are provided. For example, a new pipe may be selected as one of four pipe diameters of say, for example, 150, 200, 250 and 300 mm. Table 1 shows 8 pipe diameter choices. The genetic algorithm also requires cost information (material cost plus installation), as well as head loss characteristics of the pipe (e.g. relative roughness height or Hazen-Williams C value).

Table 1. A decoding lookup table.

Nominal Diameter (mm)	Actual or Internal Diameter (mm)	Binary Coding	Integer Coding	Roughness height (mm) for Darcy-Weisbach friction factor	Unit Cost (\$/m)
150	151	000	1	0.25	49
200	199	001	2	0.25	63
250	252	010	3	0.25	95
300	305	011	4	0.25	133
375	384	100	5	0.25	171
450	464	101	6	0.25	220
500	516	110	7	0.25	270
600	615	111	8	0.25	330

A genetic algorithm starts with a population of strings representing pipe network designs, and thereafter generates successive populations of strings. The GA relies on the diversity of the initial population to move generation by generation in a parallel fashion towards improved solutions. One of the advantages of a genetic algorithm is the robustness of the technique.

2.1 COMBINATORIALLY LARGE DESIGN PROBLEMS

The number of possible options in designing a water distribution system is usually unbelievably enormous. The New York City tunnels problem (Schaake & Lai 1969) involved 21 decision variables representing a new pipe in parallel to each of the 21 existing pipes. For 16 allowable choices for each for the duplications (including zero diameter), the number of possible combinations is:

$$16^{21} = 2^{84} = 1.934 \times 10^{25}$$

Even if a computer were able to evaluate 1 million different design combinations per second, complete enumeration of every possible alternative for the New York City tunnels problem would take 613 billion years. Thus complete enumeration of the entire search space to find the global optimum solution is usually infeasible even for small design problems. As a result, effective optimization techniques are required in order to be able to quickly search for low-cost designs. Most water distribution systems problems involve search space sizes that are much bigger than the example given for the New York City tunnels problem outlined above.

2.2 STEPS IN USING GENETIC ALGORITHMS FOR NETWORK OPTIMIZATION

The following steps summarise an implementation of a genetic algorithm for optimizing the design of a water distribution network system (based on Simpson et al. 1993; Simpson et al., 1994):

1. Develop a coding scheme to represent the decision variables to be optimised, and the corresponding lookup tables for the choices for the design variables.
2. Choose the form of the genetic algorithm operators; e.g. population size (say N=100 or 500); selection scheme - tournament selection or biased Roulette wheel; crossover type - one-point, two-point or uniform; and mutation type - bit-wise or creeping.

3. Choose values for the genetic algorithm parameters (e.g. crossover probability - p_c ; mutation probability - p_m ; penalty cost factor K).
4. Select a seed for the random number generator.
5. Randomly generate the initial population of water distribution network designs.
6. Decode each string in the population by dividing into its sub-strings and then determining the corresponding decision variable choices (using the lookup tables).
7. For the decoded strings, compute the network cost of each of the designs in the population.
8. Analyse each network design with a hydraulic solver for each demand loading case to compute network flows, pressures and pressure deficits (if any).
9. Compute a penalty cost for each network where design constraints are violated.
10. Compute the fitness of each string based on the costs in steps 7 and 9; often taken as the inverse of the total cost (network cost plus penalty cost).
11. Create a mating pool for the next generation using the selection operator that is driven by the "survival of the fittest."
12. Generate a new population of designs from the mating pool using the genetic algorithm operators of crossover and mutation.
13. Record the lowest cost solutions from the new generation.
14. Repeat steps 6 to 13 to produce successive generations of populations of designs - stop if all members of the population are the same.
15. Select the lowest cost design and any other similarly low cost designs of different configuration.
16. Check if any of the decision variables have been selected at the upper bound of the possible choices in the lookup table. If so, expand the range of choices and re-run of genetic algorithm.
17. Repeat steps 4 to 16 for, say, ten different starting random number seeds.
18. Repeat steps 4 to 17 for successively larger and larger population sizes.

Some of the main steps in the genetic algorithm process are now described in more detail.

2.2 CODING SCHEMES

A genetic algorithm coding scheme is required for each of the design variables within the water distribution system to be selected as part of the design. Examples of design variables that may need to be selected include:

1. Diameter, material and class of new pipes.
2. Diameter, material and class of duplicate pipes (that is – pipes in parallel to existing pipes).
3. The possibility of cleaning or eliminating existing pipes.
4. Possible locations of source pumps and/or booster pumps.
5. Sizes of pumps (that is –the rated head and rated discharge)
6. The operating schedule for pumps.
7. Possible locations of storage tanks.
8. Sizes and normal operating levels for the tanks.
9. Possible locations of pressure regulating valves (for example, pressure reducing valves, pressure sustaining valves, flow control valves)
10. Pressure settings for the pressure regulator valves.

Each decision variable to be selected is coded within a finite-length string. As an example, consider 8 new pipes (3 possible duplicate pipes and 5 new pipes) that are to be sized using genetic algorithm optimization (for example, the 8-dashed pipes as shown in Figure 1 of the Two-Reservoir Network). A three-bit substring is used to represent each decision variable for each pipe whose diameter is to be optimally sized. Thus, the lookup table as shown in Table 1 for each pipe will contain 8 different diameters. Note that in Figure 1 all the pipe sizes have been assigned the largest pipe size, and as a consequence, this would lead to the largest possible cost of the network. If all the

pipes were minimum size pipes represented by sub-string bit combinations of 000, then the least expensive network would result.

In Table 1, eight choices for pipe [1] are provided. For a binary coding system, the number of choices must be a multiple of 2^n where $n = 1, 2, 3, 4, \dots$ etc. (i.e. 2, 4, 8, 16, etc.). A different decoding table may be specified for each of the decision variables, if necessary. It can be seen that binary coding is somewhat restrictive when 2^n choices must be provided. An alternative to binary coding is integer coding that allows the number of choices to be any integer number (e.g. 3, 10, 15 etc.). Integer codings are also shown in Table 1.

A string or chromosome for the design problem in Figure 1 is made up of the 8 sub-strings for the three existing pipes (that can either be cleaned or duplicated – pipes [1], [4] and [5]) - and the 5 new pipes – pipes [6], [8], [11], [13] and [14].

2.3 THE RANDOMLY GENERATED INITIAL POPULATION

Once the coding representation has been selected, an initial population of, say 100 strings or designs, is generated randomly. The binary coding scheme is used for the 24-bit problem in Figure 1. As an example, consider a population of size $N = 4$ as shown in Table 2. Each bit position takes on either a zero (0) or one (1). These four strings of length 24-bits each may be randomly generated by the toss of a coin: 0 for heads and 1 for tails. In a computer code, a uniform random number generator would be used with a range of 0.0 and 1.0. For a random number less than 0.5, the bit is set to a value of zero, while for random numbers greater than or equal to 0.5, the bit is set to a value of one.

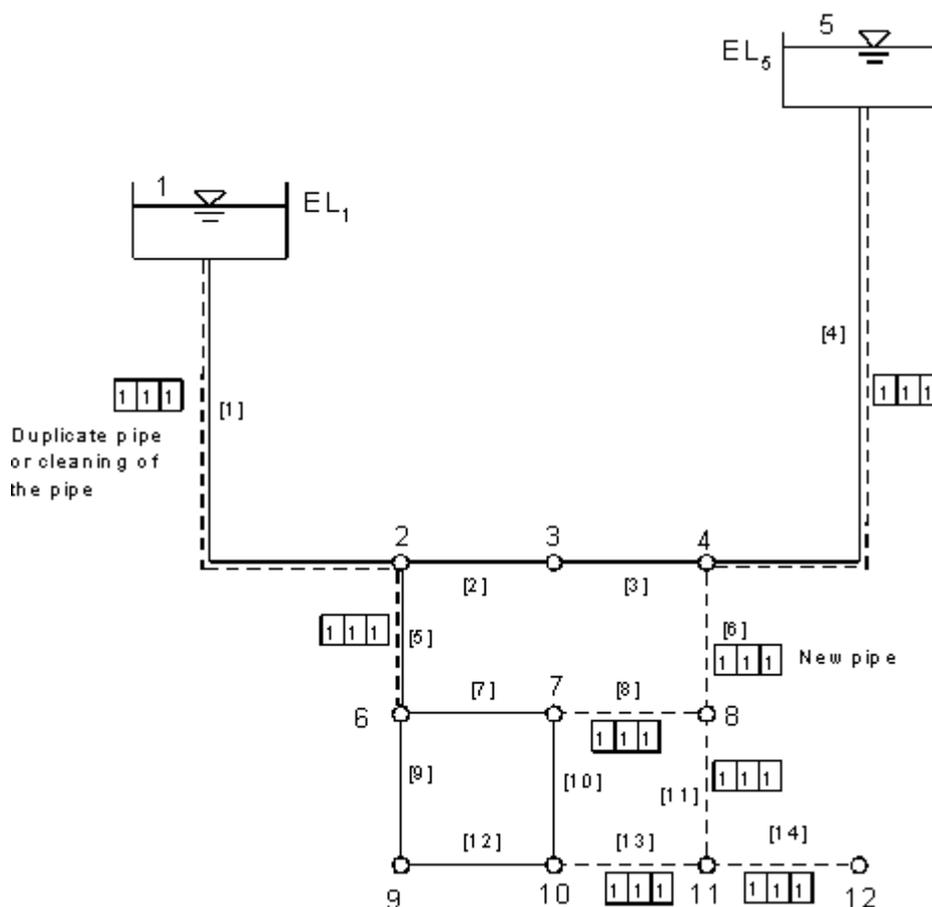


Figure 1. A pipe network showing genetic algorithm coded sub-strings.

2.4 COMPUTATION OF FITNESSES OF THE STRINGS

For each string in the randomly generated initial population, the cost of each component in the water distribution system represented by the string is computed and summed up in order to determine a total network cost. Annual operating costs can be taken back to a net present value to be added onto the capital costs. Each network is then simulated in a hydraulic solver for each of the demand loading cases (peak hour, fire and extended period simulation). A penalty cost (a pseudo-cost or fictitious-cost) is added to the network cost if any of the design constraints are not satisfied. Other design criteria may include maximum allowable flow velocities and water quality design criteria such as chlorine residual requirements. The total cost is the network cost plus the penalty cost. The fitness is taken as a function of the total cost; for example, the inverse of the total cost. Once the fitnesses of each string in the initial population have (subject/verb) been determined, the genetic algorithm operators are applied to the strings representing the water distribution system designs in order to generate a new population of solutions.

In the general case of a model for an existing system, a calibrated model would need to be developed. Genetic algorithms have been used to carry out steady state calibration.

3. GENETIC ALGORITHM OPERATORS

The operators within a genetic algorithm provide probabilistic transition rules to guide the search (Goldberg, 1989). A simple genetic algorithm has three operators including:

- Reproduction or selection to create a mating pool to base the next generation on.
- Crossover.
- Mutation.

There are many variations of these operators.

Table 2. Example of an initial population of size 4.

Pipe Number	[1]	[4]	[5]	[6]	[8]	[11]	[13]	[14]
String number 1	100	010	011	111	110	001	000	101
Diameter (mm)	384	252	305	615	500	199	151	464
String number 2	110	100	101	010	001	110	111	000
Diameter (mm)	516	384	464	252	199	516	615	151
String number 3	000	000	110	100	010	111	111	100
Diameter (mm)	151	151	516	384	252	615	615	384
String number 4	101	011	110	001	001	111	100	100
Diameter (mm)	464	305	516	199	199	615	384	384

3.1 THE SELECTION OR REPRODUCTION OPERATOR

The selection or reproduction operator selects a mating pool for the next generation from the current population of pipe network designs based on the fitness values of strings in the previous generation. The selection process determines which members of the current population will have copies (usually more than one for the fitter of the strings in the population) in the next generation.

All selection methods rely on some form of the principle of "survival of the fittest", where more copies of strings with larger fitness are preferred as candidates to go into the mating pool for the next generation. Two commonly used operators are proportionate selection (also called Roulette wheel selection) and tournament selection. Tournament selection has been shown to be much more effective than proportionate selection (Simpson & Goldberg 1994). Proportionate selection can be improved by using fitness scaling (Murphy et al. 1993).

3.2 THE CROSSOVER OPERATOR

The crossover operator interchanges bit information between pairs of selected strings. Two strings are randomly selected from the mating pool (that has been created by applying the selection operator). The first decision to be made is whether the crossover operator will be applied to this pair of strings. This is dependent on the value of the probability of crossover (for example, $p_c = 0.7$). A random number is generated between zero and 1.0. If the random number (for example, 0.4672) is less than the probability of crossover, then the crossover operator is applied. If the random number (for example, 0.8324) is greater than the probability of crossover, then crossover is not applied and the two strings pass directly to the next generation from the mating pool.

There are different forms of crossover operator including:

1. One-point crossover.
2. Two-point crossover.
3. Uniform crossover.

In one point crossover, a crossover position along the strings is chosen randomly. The same position applies to both strings. The bits past the crossover point are swapped between the strings, as shown in Figure 3. The crossover operator results in an exchange of bits between strings.

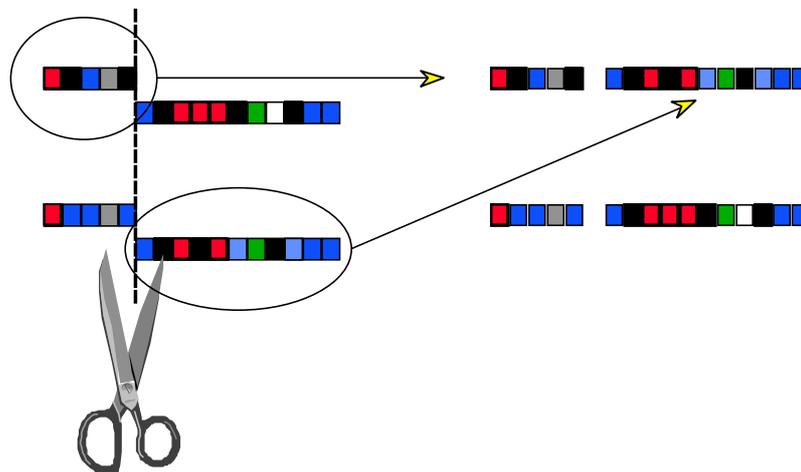


Figure 3. The crossover operator.

3.3 THE MUTATION OPERATOR

Mutation involves the occasional flipping of a bit from 1 to a zero or zero to a 1 (for a binary coding scheme). Mutation ensures no genetic material is irretrievably lost from one particular bit position. Each string in the new generation (resulting from the selection and crossover operators) is considered, in turn, for the application of the mutation operator. Each bit in the string is considered in turn. The probability of mutation is usually very small (for example, $p_m = 0.01$). A decision must be made as to whether the mutation operator will be applied to a particular bit. If a random number

between zero and one is generated (for example, 0.00523) and is less than the probability of mutation, then the mutation operator is applied. If the random number (for example, 0.367) is greater than the probability of mutation, then the bit is not mutated and the next bit along the string is then considered.

4. SUCCESSIVE GENERATIONS

For each new generation of solutions or designs that result by successive application of the selection, crossover and mutation operators, costs are calculated and hydraulic analyses are performed. The genetic algorithm process is usually continued for many generations. The lowest cost designs must be remembered along the way. The final result is a range of low cost piped network designs that are usually significantly cheaper than the cost of networks developed by designers using traditional trial-and-error techniques based on hydraulic computer simulation software.

4.1 DRAWBACKS OF GENETIC ALGORITHMS

GAs are expensive computationally, especially if the network being optimised is large. Run times of a couple of days are common for these large networks. In addition GAs need significant tuning of parameters. For large networks, no optimisation method can guarantee finding the global optimum solution. Despite these difficulties, GAs still offer the best hope for the adoption of optimisation techniques by industry. Hybrid tools may be a further development of the optimisation concepts that are presented here.

5. OUTCOMES OF THE GENETIC ALGORITHM PROCESS

There are a number of outcomes when genetic algorithm optimization is used. The most important outcome is the large cost savings that result. The GA process quickly narrows in on optimal low cost solutions. Offspring in the genetic algorithm process have genetic characteristics of the parents in the previous generation; however, progressively fitter and fitter solutions are produced. Genetic algorithm optimization has been applied to numerous real-life designs of water distribution systems resulting in savings of between 15 and 50% in capital and operating costs. In addition, the trial-and-error repetitive nature of the design process of simulation modelling is eliminated when genetic algorithm optimization is used. Optimization allows the engineer to focus on developing alternative solutions and options instead of having to fiddle with the sizes of elements in the simulation model. Another outcome is that a series of low cost solutions results from the genetic algorithm process. Non-quantifiable criteria can then be used to select the preferable design solution. Some solutions found by the GA optimization process may be just infeasible, and these may lead to significant cost savings for only a small deficit of pressure.

Genetic algorithm optimization can be applied to the design of the following systems:

- Urban water distribution systems.
- Piped off-farm irrigation systems.
- Design of new water distribution systems.
- Expansion of existing systems.
- Rehabilitation of existing systems.
- Optimizing the operation of existing systems.

6. CONCLUSIONS

Genetic algorithm optimization represents a significant advance in the approach to design and operation of water distribution systems. A procedure for applying GA optimization to the design of water distribution systems has been described in this paper. This technique is innovative and effective. The major advantage of GA optimization is its use of an embedded simulation model within the technique. All of the sophistication of the hydraulic simulation model can be used directly in the optimization technique. GA optimization is very flexible and is only limited by what the hydraulic simulation model can do. Very little usage has been made of traditional mathematical optimization techniques by the water industry. These techniques have been difficult to apply and require a high level of expertise by the user. In addition, traditional optimization techniques have difficulty in dealing with discrete decision variables. On the other hand, genetic algorithm optimization using operators of selection, crossover and mutation operators based on an analogy to natural genetics, is very effective at finding low cost optimal solutions. The GA process generates a range of different low cost solutions. Genetic algorithm optimization offers a practical way for the use of optimization by the water industry. Its implementation allows the designer to focus and develop alternative options or configurations, having the GA find low cost combinations of the components, and thus eliminate the trial-and-error usage of a simulation model.

ACKNOWLEDGEMENTS

The author would like to acknowledge the useful comments of the reviewers that have helped to improve this paper.

VIRI - REFERENCES

- Goldberg, D.E. (1989). *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley Publishing Company, Inc., 412pp.
- Holland, J.M. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor, Michigan.
- Murphy, L.J., Simpson, A.R. (1992). "Genetic algorithms in pipe network optimisation." *Research Report No. R93*, Department of Civil Engineering, The University of Adelaide, Australia 75pp.
- Schaake, J.C., Lai, D. (1969). "Linear programming and dynamic programming applications to water distribution network design". *Report No. 116*, Hydrodynamics Laboratory, Department of Civil Engineering, MIT, Cambridge, Massachusetts.
- Simpson, A.R., Goldberg, D.E. (1994). "Pipeline optimisation via genetic algorithms: from theory to practice", Proceedings, *Second International Conference on Water Pipeline Systems*, Edinburgh, Scotland, May, 309-320.
- Simpson, A.R., Dandy, G.C., Murphy, L.J. (1994). "Genetic algorithms compared to other techniques for pipe optimization". *Journal of Water Resources Planning and Management*, ASCE, July/August, 423-443.
- Simpson, A.R., Murphy, L.J., Dandy, G.C. (1993). "Pipe network optimization using genetic algorithms". In: Proceedings, *Water Resources Planning and Management Specialty Conference*, Seattle, Washington, May, 392-395.

OZNAKE - NOTATION

K	penalty cost factor;
N	population size;
n	exponent;
p_c	probability of crossover; and
p_m	probability of mutation.

Naslov avtorja - Author's Address

Angus R. SIMPSON

Department of Civil and Environmental Engineering, Adelaide University
Adelaide, Australia 5005

Email: asimpson@civeng.adelaide.edu.au