Optimization of Water Distribution Systems by a Tabu Search Metaheuristic

Article · January 2000

DOI: 10.1007/978-1-4615-4567-5_16

5 authors, including:

A. Fanni
Università degli studi di Cagliari

Saverio Liberatore
Università degli studi di Cagliari

Giovanni M. Sechi
Università degli studi di Cagliari

Paola Zuddas
Università degli studi di Cagliari

Optimization of water distribution systems by a Tabu Search metaheuristic

A. Fanni (*), S. Liberatore (**), G.M. Sechi (**), M. Soro (*) and P. Zuddas (**)
(*) DIEE - University of Cagliari, Italy, *(**) DIT - University of Cagliari, Italy*

Key words: Tabu-Search ; Water distribution networks ; Design optimization.

Abstract: A Tabu Search optimisation technique is proposed for designing, planning and maintaining water distribution systems. As design and maintenance of pipe networks for water supply distribution require high costs, achieving the highest level of performance of existing networks at minimum costs is mandatory. The problem involves setting a lot of variables, as location and diameters of new pipes, operations on existing pipes, and so on. The domain of variables is discrete in nature, due to the fact that pipes are available with unified dimensions. Furthermore, the objective function to be minimised, i.e., the total cost of the plant, is non linear, non differentiable, highly ill-conditioned, and presents a huge amount of local minima. Recently, increasing attention has been paid to heuristic optimisation techniques, such as genetic algorithms (GA), simulated annealing (SA), and tabu-search (TS) for large combinatorial optimisation problems. In particular, GA has been applied to the problem of designing and maintaining water distribution networks. Results show good performance of the GA in terms of objective function values, but high computation time. One of the most promising approaches to combinatorial optimisation problems is the TS metaheuristic, that showed flexibility and effectiveness in a lot of applications. The aim of this paper is to present a TS based algorithm to the design of water distribution systems, and to demonstrate its validity in this field.

1. INTRODUCTION

A water distribution network is a system of hydraulic elements connected together to transfer predefined flow-rates of water from sources to demand centres. It has proved particularly useful to adopt a graph structure as a topological support to the model optimisation. Nodes can represent sources and demand centres as well as control elements in the network. Arcs represent connection pipes between nodes. Besides other well-known advantages, a graph structure allows the use of highly efficient data structure in the optimisation process.

In this paper we consider the optimal design problem which has been extensively investigated in many research studies in past decades (*Alperovits and Shamir, 1977; Quindry et al., 1981; Kessler and Shamir, 1989; Fujiwara and Khang, 1990; Eigher et al., 1994*). Many investigative methods and solution techniques have been proposed and tested for this problem, but it seems generally accepted that, with general optimisation tools, it is not possible to solve the design problem with real-size systems complexity.

An adequate approach to the design and maintenance optimisation of pipe networks for the water supply distribution should consider the non-linear relations between head-losses in each pipe, its diameter, length, and hydraulic property. The literature on optimisation procedures of pipe networks is extensive.

The mathematical models describing this kind of problem are typically characterised by difficulties in handling features, such as discrete variables, large dimensions and often the absence of those mathematical features that allow the development and convergence of most classical resolution techniques, such as descent algorithms or gradient projection methods.

The non-convexity of the problem reduces the convergence properties at the local level, and its non-differentiability requires adopting non-differentiable optimisation techniques, such as subgradient or ε-gradient ones. When used in decomposition techniques, the latter are very efficient computationally if the mathematical model shows a special structure (*Eigher et al., 1994*).

Interior point methods are particularly efficient in large-scale problems, but mostly in linear constraint models.

On the other hand, using techniques based on duality, it is difficult to evaluate the gap between the obtained solution and the global solution of the original problem (*Ben-Tal et al.,1994*).

Several resolution methods have been developed to solve non-linear, non-smooth problems requiring a heavy computational effort, if sufficient adherence is required (*Fujiwara and Khang, 1990*).

Specifically to solve the water distribution optimisation problem, several resolution techniques have been implemented. The traditional method is one of trial and error guided by single iteration results and user-experience. Linear Programming (LP) approximation or procedures using LP in partition models have also been developed for this problem. Standard non-linear optimisation procedures (such as the generalised reduced gradient technique) frequently identify only local optima for this kind of problem.

Over the past years a metaheuristic approach based on the genetic algorithm (GA) methodology has been developed for pipe-network optimisation (*Murphy and Simpson, 1992; Dandy et al., 1993; Simpson et al., 1994; Murphy et al., 1993; Dandy et al., 1996; Liberatore et al., 1998*). An extensive analysis of GA application to water system design and comparison of commercial optimizers has been reported in a recent work by *Lippai et al.,(1999)*. Even if improved versions of GA obtained using evolved representation codes for the network elements, for powered fitness evaluation, and for more appropriate GA operators, the computational time needed to reach near-optimal configurations remains the most limiting problem in the approach to real water distribution systems.

In this paper, an alternative metaheuristic approach based on Tabu-Search (TS) technique has been implemented and tested on a well-known test-problem given by *Gessler (1985)*. The results have been compared with those reported in the literature.

2. PROBLEM FORMULATION

The goal of the problem is to find the lowest cost network that can satisfy demands under hydraulic and continuity equation restrictions and penalties, imposed to ensure predefined hydraulic heads in node subsets.

In this paper we refer to a simplified problem of pipe network design for a pressure system (without pump-stations) with the following main features:

- The network demands are known and configured as node-outflows;
- The continuity of flow must be maintained at all nodes in the pipe-network;
- The head loss in each pipe-arc is a known function of the flow in the pipe, its diameter, length and hydraulic properties of the pipe:
- At each node minimum and maximum pressure head limitations must be satisfied;
- Diameter constraints may be applied to the pipes.

In the network, the existing pipes (that have known diameters) as well as new pipes can be taken into account. For each pipe-arc, different possible states are examined, i.e. the possibility of leaving exactly the same pipe-arc (LEAVE), that of cleaning the existing pipe (CLEAN), that of adding a new pipe-arc to the existing one (DUPLICATE) and that of installing a new pipe-arc (NEW).

The general constraint equations considered for a given demand pattern are as follows:

Continuity at each node:

$$\sum_{j=1,N_j} Q_j + q_i = 0 \qquad i = 1, N_i \qquad (1)$$

where Q_j represents the flow in each of the N_j pipes connected to node i, and q_i the demand at node i.

Head-loss equation (Hazen-Williams):

$$H_{i_1} - H_{i_2} = \frac{kL_j (Q_j)^{\beta_j}}{(C_j)_j^{\alpha} (D_j)^{n_j}} \qquad i_1 , i_2 = 1 , N \qquad (2)$$

where i is the node-head, L_j the length of pipe, C_j the roughness coefficient and D_j the pipe diameter.

Minimum pressure head constraint:

$$H_i \geq H_i^* \qquad i = 1, N_i \qquad (3)$$

where H_i^* is the node hydraulic-head that must be guaranteed.

Bounds on pipe diameters:

$$D_{\min} \leq D_i \leq D_{\max} \qquad i = 1, N_i \qquad (4)$$

where the minimum diameter refers to the existing diameter in the case of duplication.

The purpose is to optimize a non-linear objective function (o.f.), i.e. the total cost needed for constructing new pipes, or cleaning or duplicating existing ones. For the last two cases, we used the "equivalent diameter" approach (*Simpson et al., 1994; Liberatore et al., 1998).* Moreover, implementing the GA and TS procedures, equations *(3)* are relaxed on the o.f. as penalty components depending on whether the network satisfies the minimum pressure constraints. After the generation of an initial network configuration, the procedure performs a hydraulic analysis of the pipe network, resolving the non-linear system given by equations *(1)* and *(2).* Node pressure differences from target values are then used in the o.f. to compute penalty costs.

The o.f. assumes the general form:

$$\min \sum_{j\in R_1} C_{1j} Lj + \sum_{j\in R_2} C_{2j} L_j + \sum_{j\in R_3} C_{3j} L_j + \sum_{j\in R_4} C_{4j} Lj + \\ + \sum_{i\in N^*} C_{5i} (H_i^* - H_i)^\gamma \tag{5}$$

where the first term refers to the maintenance of old pipes, the second to cleaned pipes, the third to the duplicate set, the fourth to new pipes and the last to hydraulic head differences.

As mentioned previously, the design must consider discrete elements of the system, and an adherent formulation of the problem may generate non-linear and non-convex models. Traditional (but still used today) design methods are based on trial and error and are guided by single iteration results and by user-experience. Recent works based on application of GA, may consider design of new pipes as well as duplication and maintenance rehabilitation.

For the water distribution systems design problem, metaheuristic algorithms seem to have several advantages with respect to other mathematical programming techniques, as they can be implemented without heavy *a-priori* requirements such as convexity and differentiability in o.f. and constraints. Thanks to their capacity to manage discrete variables they can deal directly with the alternatives available (commercial diameters, cleaning, duplication, etc.). Each alternative consists of a set of discrete and organised strings that are usually coded using predefined rules.

Since starting from initial pipe-network configurations, GA procedures only use o.f. or other fitness information, the hydraulic-head constraint violation must be penalised to allow the genetic algorithm to reach a feasible solution as a final optimum.

As previously referred, GA optimisation procedures have been extensively illustrated and applied by many authors and synthetically consist of the following steps:

- A "population" of possible network configurations is proposed by the genetic or the initialisation procedure.
- Continuity (1) and head-loss (2) equations are solved retrieving pressure-heads at nodes.
- For each configuration the economic o.f. evaluations are added to the penalty evaluation caused by target node pressure violation.
- GA uses the fitness information to generate a new population.

As will be shown in the following paragraphs, GAs do not guarantee that the global optimum will be reached, though tests indicate that using well-calibrated procedure parameters such as the population replacement parameter, optimal or near-optimal solutions are almost always obtained. Nevertheless, computational time needed to reach near optimal configurations remains the most limiting problem applying GAs to real water systems. This fact is mainly due to the particularly large number of o.f. evaluations that for each system configuration requires an *a-posteriori* resolution of the non linear system given by the hydraulic equations *(1)* and *(2)*.

3. TS METAHEURISTIC

Tabu-Search (TS), whose main concepts are collected in *Glover (1994)*, and *Glover and Laguna (1997),* was originally formulated by *Glover (1989), (1990)*. As a matter of fact, TS is usually defined as a meta-strategy or a metaheuristic *(Glover and Laguna, 1997; Hertz and De Werra, 1990)*, in that rather than one heuristic method, it is believed to be a main strategy that guides several subordinated heuristics to produce solutions beyond those normally generated by the search for a local optimum.

The employed methodologies were essentially two: an adaptive memory and a "sensitive" exploration; both typify the method. It is a search process that, in order to explore the solution space domain, is guided by a logic that takes into account the past dynamic trend. The system practically exploits one of its memories in an attempt to avoid being trapped in attraction basins, or better, in order to direct the search towards domain areas believed to be more promising. This is achieved thanks also to suitable strategic decisions that condition future searches, since they are assisted by the knowledge so far acquired of the configuration space. This occurs through the introduction of limitations to the search process guaranteeing that domain regions characterised by strong attraction areas will be overcome. These limitations generally operate with the direct exclusion of search alternatives classified as prohibited, that is tabu, that may or may not play a role in the dynamics, according to adequate criteria that add weight to the sensitivity of the exploration.

In order to give a description of the work carried out by TS, one should reconsider the general approach to an optimisation problem, as described by *Glover (1989), (1990), (1993),*

$$\min C(\mathbf{x}) \mid \mathbf{x} \in X \text{ in } R_n \tag{6}$$

where $\mathbf{x} \subseteq R_n$, n being the dimension of the problem (number of variables). The objective function $C(\mathbf{x})$ may or may not be linear, and it may include penalty functions to guarantee that a certain type of imposed constraints are met. The condition $\mathbf{x} \in X$ summarizes the constraints on components of the vector $\mathbf{x}$, that except for a few special strategic variations will be maintained at each stage of the search. In many contexts of interest, it will also be required of some or all the components of $\mathbf{x}$, to assume discrete values. In some cases the problem of interest will not be formulated explicitly as said above, but *(6)* will represent a modified form of the original problem. Each $\mathbf{x} \in X$ possesses an associated neighbourhood $N(\mathbf{x}) \subset X$, and each solution $\mathbf{x}' \in N(\mathbf{x})$ can be reached by $\mathbf{x}$ by means of an operation called *move*. A move, s, therefore, is an operation of X that, applied to one of its elements, $\mathbf{x}$, transforms it into another element, $\mathbf{x}'$, that belongs to the neighbourhood of the same $\mathbf{x}$. If we assume $S(\mathbf{x})$ to be the set of moves that may be applied to $\mathbf{x}$, we can write:

$$\mathbf{x}' = s(\mathbf{x}), \text{ with } s(\mathbf{x}) \in S(\mathbf{x}). \tag{7}$$

The modality in which a move is defined may vary considerably depending on the characteristics of the problem under consideration. Generally it consists of the change of one or more *attributes* (or components) of the solution vector, $\mathbf{x}$, according to certain pre-established rules.

The metaheuristic TS guides the search in the solution space by creating and exploiting adequate memory structures, while allowing the process not to be trapped in local minima.

For this reason, a dynamic memory structure, called tabu list (*tabulist*), is created and maintained. It is made up of the set of the latest moves, or alternatively solutions (the size of the tabu list is called tabu-tenure and will be named *TT*).

3.1 Representation of solutions

As regards the issue considered in this work, the solution is represented by an array whose size coincides with the number of variables. During the information decoding phase, the sequence of decisional variables is recorded in a special array, and therefore the index in the array unequivocally identifies an arc of the net.

The diameter of each arc may be selected depending on the availability of commercial size pipes. The domain of each variable therefore can be assumed to be in biunivocal correspondence with a subset of whole numbers corresponding to these values. A dynamic allocation register will determine the domain of each variable, thus rendering the information available to the entire programme, since each variable may have a different domain. A possible solution could have the following form:

$$\mathbf{x} = \{1,4,8,5,2,2,1,1\}$$

The problem therefore, in brief, is a search for the optimal combination of elements in the numeric string, **x,** that will minimise the cost of the net. Obviously, the o-f. depends only on the decisional variables, that is on the elements of the array. In conclusion, it follows that the variable to be treated in the algorithm is the same **x**.

3.2 Definition of the move and neighbourhood

As previously suggested, defining the move is strategic in setting the search operated by TS. It is obvious, moreover, that the composition of the set making up the neighbourhood of the generic solution will depend on the choice of the move.

In this work, the choice falls on *cartesian* moves. A *cartesian* move is the variation of one single variable in the neighbourhood of the current position, in a *range* that defines its size.

In this paper, we tested static and dynamic variations of the range, and thus of the neighbourhood. In the following we report the results obtained with the dynamic technique, since it afforded better results both in terms of quality of the solutions and in terms of computational effort.

The exploration is done by varying only the components of the current solution one at a time while leaving the others unaltered.

3.3 Memory structures

In this implementation, we used a Short Term Memory (STM) that, by definition, keeps track of the attributes of the moves carried out in the recent past of the search, and a Long Term Memory (LTM). In this implementation, the STM actually does not record tabu moves, but entire configurations, while the LTM records the occurrence or transition frequency of the visited configurations. To implement the memory structure in this work, we used a *hashing* technique. A further advantage of this choice is that of saving the transition frequency of each solution visited by using the same memory also as a tabu list. *Hashing* is a method that assigns addresses directly to the records contained in a table by performing arithmetical transformations on the *keys* that express the solutions, in order to obtain the addresses of the records in the table. If the keys are integers in the interval between 1 and N, the record can be saved with the i key in the i-th position in the table, thus allowing immediate access through the value of the key *(Sedgewick, 1990)*. More particularly, we used *open hashing (Aho et al.,1987)*, where the index of an element of the *hash* table is extracted directly from the saved information. The size of the primary array is a parameter of TS established as a function not only of the size of the problem, which can be very large as will be seen, but also as a function of the greatest number of planned o.f. evaluations.

LTM is also used efficiently to make a diversification strategy operational by introducing penalties in the objective function, that depend on the occurrence frequency of pejorative solutions compared to the current one.

Moreover, an aspiration criterion for each global objective has been introduced *(Glover, 1994; Glover and Laguna, 1997).*

3.4 Intensification

The essential aim of intensification is to guide the search in previously visited, promising regions of the domain, in order to explore them thoroughly. The implemented intensification procedure becomes operative as soon as the quality of the results in a certain lapse of iterations comes below a fixed threshold. This threshold, therefore, becomes a further parameter of TS and requires calibration. A number of tests have led to establish 50 as the optimal value for this parameter. This means that after 50 iterations without reaching a new optimum, the search will return to the last found optimum. If no improvements are made, the same procedure will be repeated after another 50 iterations since the last intensification.

3.5 Reactive search

A number of tests have shown that the choice of *TT* is strategic if a good performance of the algorithm is to be reached. We have therefore chosen to implement a so-called reactive kind of TS *scheme (Battiti and Tecchiolli, 1994).*

In this variant of TS, *TT* is dynamical, and its optimal value is estimated automatically from the algorithm by means of a retroactive assessment of the search history. The reaction mechanisms are related to the transition frequency of the current solution. These allow an increment of the *TT* as a result of a few consecutive repetitions of solutions, while in the event of total absence of repetitions, the reaction mechanism gradually reduces the value of the tabu tenure to a prefixed minimum value.

In the event of the frequency being greater than unity, an increment of the *TT* proportional to its minimum value is obtained; the occurrence frequency value of the current solution adds weight to the variation.

There are two possible reaction mechanisms. To give an exhaustive explanation of the need for such mechanisms, the evolution of the search process in combinatorial optimisation can be compared to the theory of dynamic systems. In searching for the global minimum, the local minima are *attractors* of the system dynamics for the steep descent strategy. Nevertheless, *limit cycles* (or *closed orbits*) are also possible, in which the trajectory indefinitely repeats a sequence of states. In TS, cycles are discouraged. But there is a third negative possibility that becomes very important in optimisation problems, i.e. the case in which the search trajectory is *confined* to a limited portion of space. In the theory of dynamic systems, this phenomenon is described by the concept of *chaotic attractors*. *Chaotic attractors* are characterised by a "contraction of areas" in such a way that also the trajectories originating in different initial conditions will always be compressed in a limited area. Moreover, they so much depend on initial conditions, that different trajectories, that should converge towards one and the same optimum, do in fact diverge.

A technique is efficient if, besides avoiding limit cycles and the repetition of the same configurations, it discourages chaotic attractors.

Referring to the presentation contained in (*Battiti and Tecchiolli, 1994*), we have a situation in which variation of *TT* is left to two constants that govern two different types of variation: a long term decrease in *TT* in which we use a variable indicating a movement proportional to the length of the definite cycle, and a fast reaction that increases *TT* while discouraging further repetition. After *K* immediate reactions, the geometric increment that relates the increment constant of *TT* to *K* is sufficient to break any limit cycle. The reaction is only related to the local properties of the search trajectory, but if the tabu tenure increased infinitely, it could be too high in the subsequent iterations, and would force the search excessively. A slower mechanism reduces *TT* after a certain number of iterations since the last change. This could be insufficient to avoid "chaotic trapping" of the trajectory in a limited area of the search space. Therefore a diversifying escape movement is used. This is triggered when the variable that counts the number of the most repeated solutions exceeds a fixed threshold value. The escape consists of a set of casual exchanges. In order to avoid an immediate return to regions of space that have already been explored, all the operated random exchanges become tabu.

What mainly distinguishes the proposed algorithm from the literature algorithms is the escape mechanism. Typically, to avoid a complex attractor, the system enters a phase of casual movements of duration proportional to the length of the defined cycle. In the presented algorithm, we have attempted to give a more deterministic variation law to these diversified movements.

If we consider a complete exploration of the neighbourhood of the current solution during the search, we generally obtain high quality solutions, to the detriment of a high computational burden on the part of the computer. In combinatorial problems, this is truer the wider the search space; the case in which the increase in the configuration space is due to an increase in the neighbourhood of each of the variables, rather than to an increase in the number of variables, is particularly negative. For this reason we find it useful in this work to combine TS with strategies that should impose an adequate range to the *cartesian* move.

This technique has been combined to the TS algorithm and has produced excellent results in the different cases studied.

3.6 Available code

The presented algorithm has been implemented using a general purpose Tabu Search (TS) tool called Universal Tabu Search (UTS) *(Fanni et al., 1998)*. The UTS algorithm is an original evolution of TS that combines the most interesting and effective search techniques conceived by various authors with new ideas and strategies, aiming to satisfy simplicity and versatility requirements. The code presents a user-friendly interface really simple to use and sufficiently powerful and versatile to be applied to constrained optimization problems. In particular the customization of the code to water system design problems has been easy and rapid. The code is implemented in ANSI C/C++ and it runs on different platforms like PC-WINDOWS and UNIX machines. The proposed source code is available to use and to validate on other water system design problems by sending a request to the e-mail address: fanni@diee.unica.it. A further possible development will be to interface the code with a commercial pipe network simulator to enlarge the possibility to use the proposed approach.

4. BENCHMARK CASES

In order to test the TS optimisation technique, we have examined a case study presented by *Gessler (1985)* and considered also by other authors (*Simpson et al., 1994; Liberatore et al., 1998)*. The system is shown in Figure 1; the solid lines represent existing pipes, while the dashed lines are the new pipes to be dimensioned. The *Gessler* problem considers eight different pipe sizes (commercial diameters) available for new pipes, while existing pipes may be left as they are cleaned or duplicated with a new pipe. Moreover, two supply resources are available, and three demand patterns and the associated minimum pressure heads must be satisfied. Elevation, pipe length, roughness, and demand patterns are given as well as pipe cost and available diameters. The *Gessler* problem can also be used to compare solutions obtained with different approaches. With commercial non linear optimisation software, only near-optimal solutions were obtained, and there is a problem approximating the obtained pipe-size up or down to the nearest available commercial diameter.

Starting from *Gessler* scheme*,* two design problems, described in the following, will be used as benchmarks to test the performance of the proposed TS algorithm.

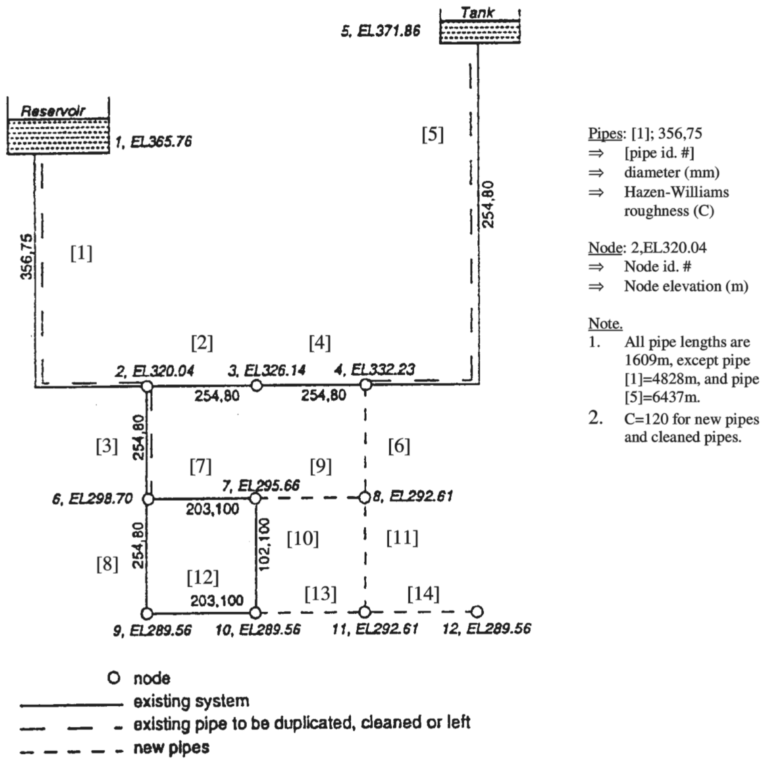

Figure 1. Gessler network scheme

4.1 First water scheme

As already stated, the first water scheme is the one proposed by *Gessler (1985)* and referred to by *Simpson et al.(1994)* and *Liberatore et al. (1998)*. The given possibilities are to plan *ex-novo* five new pipes as well as of choosing among options that include cleaning, duplication or maintaining three existing pipes.

This water scheme is associated with three demand patterns that are reported in Table 1 and include two cases of extraordinary operation of the system as a result of fires.

Table 1. Demand patterns associated to Gessler network

Node	Demand Pattern 1		Demand Pattern 2		Demand Pattern 3	
	Demand (L/s)	Minimum Head (m)	Demand (L/s)	Minimum Head (m)	Demand (L/s)	Minimum Head (m)
2	12.62	28.18	12.62	14.09	12.62	14.09
3	12.62	17.61	12.62	14.09	12.62	14.09
4	0	17.61	0	14.09	0	14.09
6	18.93	35.22	18.93	14.09	18.93	14.09
7	18.93	35.22	82.03	10.57	18.93	14.09
8	18.93	35.22	18.93	14.09	18.93	14.09
9	12.62	35.22	12.62	14.09	12.62	14.09
10	18.93	35.22	18.93	14.09	18.93	14.09
11	18.93	35.22	18.93	14.09	18.93	14.09
12	12.62	35.22	12.62	14.09	50.48	10.57

In *(Gessler, 1985)* eight possible alternatives are provided for the existing arcs and six for the new arcs. With a system having these characteristics we obtain a total of 3981312 possible combinations. In *(Gessler, 1985)*, the use of a selective enumeration is proposed to achieve a dramatic reduction of the search space. The technique is exclusively based on the experience of the designer, and is therefore a trial and error search of the global optimum. The lowest cost configuration found with this technique leads to an objective function value of 1833700 $.

Simpson et al., (1994) perform an exhaustive search of the 3981312 possible combinations. The 30 lowest cost solutions obtained, that meet the imposed constraints, are reported in Table 2. We note two global minima with a different configuration (solutions no.1 and no.2 in Tab. 2), as well as the presence of a certain number of different configurations with the same cost. Besides the complete enumeration, the same problem is solved in (*Simpson, 1994)* with the optimisation package *GINO (Liebman et al., 1986)* but relaxing a few constraints, such as the discrete nature of the variables, leading to poorly accurate solutions especially for small size pipes. Furthermore, this does not guarantee that the obtained solution meets the hydraulic constraints.

Table 2. Lowest cost solutions obtained by complete enumeration (Simpson et al., 1994)

N°	Cost ($)	Best final configurations							
		Pipe [1]	Pipe [3]	Pipe [5]	Pipe [6]	Pipe [9]	Pipe [11]	Pipe [13]	Pipe [14]
1	1'750'300	Leave	Leave	Dup 356	305	203	203	152	254
2	1'750'300	Leave	Leave	Dup 356	305	203	254	152	203
3	1'772'500	Leave	Leave	Dup 356	305	203	203	203	254
4	1'772'500	Leave	Leave	Dup 356	305	203	254	203	203
5	1'791'000	Leave	Dup 203	Dup 356	254	203	203	152	254
6	1'799'900	Leave	Clean	Dup 356	254	203	203	203	254
7	1'801'000	Leave	Leave	Dup 356	305	203	254	152	254
8	1'801'000	Leave	Leave	Dup 356	305	254	203	152	254
9	1'801'000	Leave	Leave	Dup 356	305	254	254	152	203

N°	Cost ($)	Best final configurations Pipe [1]	Pipe [3]	Pipe [5]	Pipe [6]	Pipe [9]	Pipe [11]	Pipe [13]	Pipe [14]
10	1'811'500	Leave	Leave	Dup 356	356	203	203	152	254
11	1'811'500	Leave	Leave	Dup 356	356	203	254	152	203
12	1'811'500	Leave	Leave	Dup 356	305	203	305	152	203
13	1'811'500	Leave	Leave	Dup 356	305	203	203	152	305
14	1'813'100	Leave	Dup 203	Dup 356	254	203	203	203	254
15	1'823'200	Leave	Leave	Dup 356	305	203	203	254	254
16	1'823'200	Leave	Leave	Dup 356	305	203	254	203	254
17	1'823'200	Leave	Leave	Dup 356	305	254	203	203	254
18	1'823'200	Leave	Leave	Dup 356	305	203	254	254	203
19	1'823'200	Leave	Leave	Dup 356	305	254	254	203	203
20	1'828'500	Leave	Clean	Dup 356	254	203	254	152	254
21	1'830'000	Leave	Dup 152	Dup 356	305	203	203	152	254
22	1'830'000	Leave	Dup 152	Dup 356	305	203	254	152	203
23	**1'833'700**	**Leave**	**Leave**	**Dup 356**	**305**	**203**	**305**	**203**	**203**
24	1'833'700	Leave	Leave	Dup 356	356	203	254	203	203
25	1'833'700	Leave	Leave	Dup 356	356	203	203	203	254
26	1'833'700	Leave	Leave	Dup 356	305	203	203	203	305
27	1'838'500	Clean	Leave	Dup 305	254	254	254	152	254
28	1'839'000	Clean	Dup 203	Dup 305	254	203	203	152	254
29	1'839'000	Leave	Clean	Dup 356	305	203	203	152	254
30	1'839'000	Leave	Clean	Dup 356	305	203	254	152	203

Moreover, *Simpson et al.(1994)* expand the search space by adding another two possible alternatives for the arcs being planned, bringing the size of the search space to a total of 16777216.

In the following we refer to this case as *Problem 1.*

In (*Simpson et al.1994)*, the same *Problem 1* is tackled after using a genetic algorithm. The programme is "launched" ten times using a maximum limit of 50000 function calculations as a stop criterion (that is 0.298% of the 16777216 possible combinations). The obtained solutions are reported in Table 3.

I n *(Liberatore et al., 1998)*, the same hydraulic scheme is used as benchmark case with the same constraints on the flow rate and hydraulic head at nodes, but besides providing for a reliability procedure of the system, the search space is extended to include, among the options for the existing arcs, replacing the pipeline completely and doubling the alternatives for the available. In the latter case, the total search space is $3{,}4358738 \cdot 10^{10}$ possible configurations.

Table 3. GA results obtained in Simpson et al. (1994) for Problem 1

Run n°	Best network cost ($) (diff. % from opt.)	O.F. evaluation	Position in Table 2
1	1'772'500 (1,27%)	29070	3
2	**1'750'300**	10350	1

Run n°	Best network cost ($) (diff. % from opt.)	O.F. evaluation	Position in Table 2
3	**1'750'300**	43740	2
4	1'811'500 (3,5%)	40860	10
5	**1'750'300**	17190	2
6	**1'750'300**	11070	2
7	**1'750'300**	10080	1
8	**1'750'300**	41490	2
9	**1'750'300**	12510	1
10	**1'750'300**	19890	1

In the following we refer to this problem as *Problem 2.*

This benchmark case was used in *(Liberatore et al., 1998)* to test the performance of a genetic approach to obtain the lowest cost network. The best results are reported in Table 4.

Table 4. GA results obtained in Liberatore et al. (1998) for Problem 2

Run n°	Best network cost ($) (diff. % from opt.)
1	1'992'131 (13,9%)
2	1'797'789 (2,8%)
3	1'841'028 (5,2%)
4	**1'750'300**
5	**1'750'300**
6	2'058'805 (17,7%)
7	**1'750'300**
8	**1'750'300**
9	**1'750'300**
10	**1'750'300**

4.2 Second water scheme

The need to check the reliability of the TS implemented algorithm and to verify its performances as part of *general purpose* tool for this class of problem, has led to the study of a second scheme that introduces an increment in the number of variables (new arcs). In this case we increased the arcs by 50%, i.e. from 8 to 12. By hypothesising a domain (or possibility of option) of 8 for each variable, as in, (*Simpson et al.1994),* we obtain a search space of $6,8719476 \cdot 10^{10}$ possible configurations.

In the following we refer to this problem as *Problem 3.*

This expanded scheme is reported in Figure 2. Even in this example the demand patterns take into account the possibility of the network operating under extraordinary load demands.

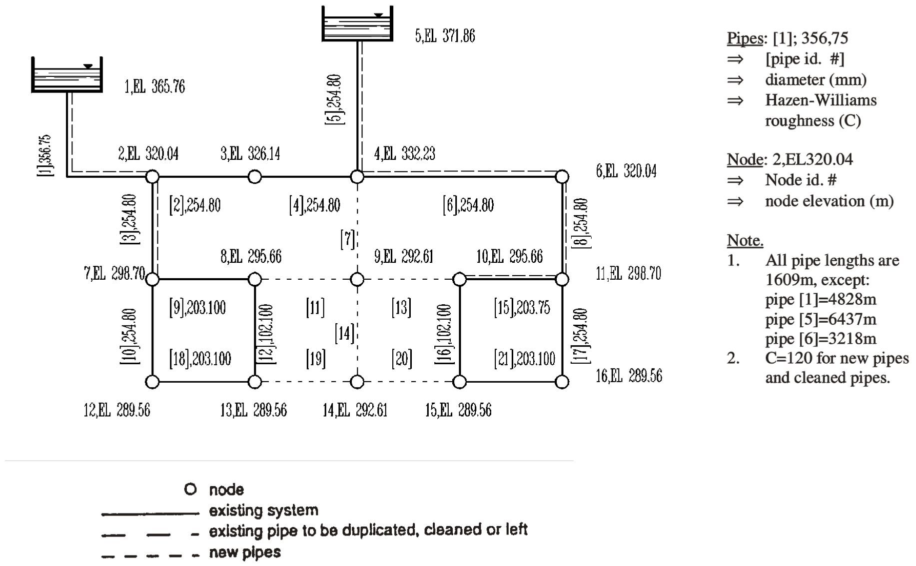

Figure 2. Extended Gessler network scheme

5. RESULTS

In this section we report the results obtained by applying TS to the benchmark cases and compare them both with those reported in the literature and summarised in the previous section, and with the results obtained applying a GA based on *PGA-Pack Library* (*Argonne National Laboratory, 1996)*. We report statistics over 100 runs, starting from randomly generated initial configurations.

The results are presented in several tables that report the following indications:

– The minimum cost function obtained;
– The reference of the found solution in Tables 1 (for *Problem 1* and *Problem 2)*
– The relative occurrence frequency of the optimum;
– The average number of iterations needed to assess the corresponding optimum;
– The number of o.f. evaluations needed to reach the optimum;

– The percentage shift from the optimum; in the case of *Problem 1*, the global optimum is known (equal to 1'759'300 $, it should also be remembered that it can be obtained with two distinct configurations, cf. Table 2). It is reasonable to consider, moreover, that this should remain the global optimum for the *Problem 2* with the settings made in *(Liberatore et al., 1998)*.

As can be observed from the results reported in Table 5 for *Problem 1*, TS finds its absolute optimum in 100% of the cases, with a very low number of function calculations.

Table 5. TS results (100 runs) obtained from Problem 1

Cost ($)	Position in Tables 2 and 3	% of success	Average iter n°	Average O.F. evals
1'750'300	2	15%	180	2760
1'750'300	1	85%	171	2609
Global optimum (1+2)		100%	172	2632

The proposed algorithm shows a remarkably better performance than any other described method. In particular: the cost of the optimum solution supplied by *Gessler (1985)* with selective enumeration (1'833'700, solution no. 23 Table 2) is 4.8% higher than the absolute optimum; with the additional disadvantage that reducing the search space, though according to correct guidelines, may lead to removing the global optimum. *Simpson et al. (1994)* complete enumeration leads to the global optimum, but it involves a huge number of o.f. evaluations, compared, for example, to the 2632 checks made by TS (0.016% of the total search space).

Moreover the comparison of our results for *Problem 1* with those obtained with GA in *(Liberatore et al., 1998*) shows that success percentage rises from 80% to 100%. Besides the clear difference in success percentage, the difference in the number of function calculations is particularly large: about 690% greater in GA.

A comparison with non-linear optimisation is a non-trivial task, since this approach (and, of course, the whole optimisation process) involves continuous variables. The solution must therefore be rounded off to available commercial diameters, and the obtained solution checked *a-posteriori* for hydraulic acceptability. In large size systems then, this rounding off process actually becomes a second optimisation problem.

Table 6 contains the results obtained by TS for *Problem 2*. The Table shows that also for this problem TS results are significantly better than those obtained using GA in *(Liberatore et al., 1998)*.

Table 6. TS results (100 runs) obtained for Problem 2

Cost ($)	% of success	Average iter n°	Average O.F. evals
1'750'300	100%	698	11753

The global optimum of the water network considered in *Problem 3* is not known and a complete enumeration is clearly prohibitive. Nevertheless, indications can be drawn on the characteristics that a generic optimum solution must have. A reliability check of the solution given by TS (see Table 7) has been carried out by the water system solver, *WaterCAD (Haestad Methods, 1998)*.

Table 7. TS results (100 runs) obtained for Problem 3

Cost ($)	% of success	Average iter n°	Average O.F. evals
2'256'013	100%	91	2078

6. CONCLUSIONS

A TS metaheuristic is proposed to design water distribution systems. Mathematical model with an adequate level of adherence requires heavy computational effort if non-linear programming techniques are adopted. The complexity of real water distribution network problem grows with the necessity to consider non-smooth non-convex large-size problems and discrete variables. Metaheuristic techniques seem to overcome many of these difficulties. Genetic algorithms developed for pipe-network optimization, can manage this class of problem but the computation time needed to reach near-optimal configuration is a great limiting problem in using this approach in real water distribution systems.

TS metaheuristic seems to be viable method. As seen in the paper, the computational effort of the proposed implementation is limited, and it seems reasonable to use TS to solve more complex problems that normally arise in water system management such as network reliability, that dramatically increase the problem complexity.

7. REFERENCES

Aho A.V., J.E. Hopcroft, J.D. Ullman, (1987), *Data structures and algorithms*, Addison~Wesley Publishing Company.

Argonne National Laboratory (1996), *PGA - Pack Library. Parallel Genetic Algorithm,* Argonne, Illinois.

Alperovits E., U.Shamir, (1977), Design of optimal water distribution system, *Water Resour. Res., 13(6)*, 885-900.

Battiti R., Tecchiolli G., (1994), The Reactive Tabu-Search, *ORSA Journal on Computing,* Vol.6, n.2, 126-140.

Ben-Tal A., G. Eiger, Z. Gershovitz (1994), Global minimization by reducing the duality gap, *Math. Programming, 63,* 193-212.

Dandy G.C., A.R. Simpson, L.J. Murphy, (1993), A review of pipe network optimization techniques, *Proceedings of Watercomp '93, Melbourne, Australia.*

Dandy G.C., A.R. Simpson, L.J. Murphy, (1996), An improved genetic algorithm for pipe network optimization, *Water Resour. Res., 32(2),* 449-458.

Eiger G., U. Shamir, A.Ben-Tal, (1994), Optimal design of water distribution networks, *Water Resour. Res., 30(9),* 2637-2646.

Fanni A, S. Bibbo', A. Giua, A. Matta, (1998) A general purpose Tabu Search code: an application to Digital filters design, *IEEE Int. Conf. on Systems, Man, and Cybernetics*, San Diego (CA), (invited).

Fujiwara O., D.B. Khang, (1990), A two phase decomposition method for optimal design of looped water distribution networks, *Water Resour. Res., 26(4),* 539-549.

Gessler J., (1985), Pipe network optimization by enumeration, *Proc. ,Computer applications for water resources* – ASCE, New York, N.Y.,572-581.

Glover F., Laguna M., (1997) *Tabu Search,* Kluver Academic Publishers.

Glover F., (1989), *Tabu Search - Part I* ORSA Journal on Computing, Vol 1, n° 3, Summer 1989.

Glover F., (1990) *Tabu Search - Part II* ORSA Journal on Computing, Vol 2, n° 1, Winter 1990.

Glover F. (1993), *Tabu Search, modern heuristic techniques for combinatorial problems*, Colin R. Reeves (Ed.), 70-150, Blackwell Scientific Publications, Oxford.

Glover F., (1994), *Tabu Search fundamentals and uses*, University of Colorado, Boulder, 1994.

Haestad Methods (1998) *WaterCAD,* Waterbury, CT.

Hertz A., D. De Werra (1990) *The Tabu Search metaheuristic: how we used it*, Annals of Mathematics and Artificial Intelligence, 1 (1990) 111-121.

Kessler A., U. Shamir, (1989), Analysis of the linear programming gradient method for optimal design of water supply networks, *Water Resour. Res., 25(7),* 1469-1480.

Liberatore S., G.M. Sechi, P. Zuddas, (1998), A Genetic Algorithm approach for water system optimization, *AICE Annual Conference,* Milano- Italy.

Liebman J.S., L. Lasdon, L. Schrage, A. Waren, (1986), *Modeling and optimization with GINO,* The Scientific Press, Palo Alto, Ca.

Lippai I., J.P. Heaney, M. Laguna, (1999), Robust water system design with commercial intelligent search optimizers, accepted by *J.Computing in Civil Eng.*

Murphy L.J., A.R. Simpson, (1992), Pipe optimization using genetic algorithms, *Res. Rep.*93, p.95, Dept. of Civ.Eng., Univ. of Adelaide.

Murphy L.J., A.R. Simpson, G. Dandy, (1993), Design of a network using genetic algorithms, *Water, 20,* 40-42.

Quindry G.E., E.D. Brill, J.C. Liebman, A.R. Robinson, (1979), Comment on "Design of optimal water distribution system" by E Alperovits and U. Shamir, *Water Resour. Res., 15(6),*1651-1654.

Simpson A.R., G.C.Dandy, L.J.Murphy, (1994), Genetic algorithms compared to other techniques for pipe optimization *Journal of Water Resources Planning and Management,* ASCE Vol. 120, n° 4, 423-443.

Sedgewick R., (1990), *Algorithms in C* , Addison~Wesley Publishing Co.

Vanderbei R. (1994) Interior-point methods algorithms and formulations *ORSA Journal on Computing 6*, 32-34.