Taylor & Francis
Taylor & Francis Group

Engineering Optimization

ISSN: 0305-215X (Print) 1029-0273 (Online) Journal homepage: http://www.tandfonline.com/loi/geno20

Optimal design of water distribution networks by a discrete state transition algorithm

Xiaojun Zhou, David Y. Gao & Angus R. Simpson

To cite this article: Xiaojun Zhou, David Y. Gao & Angus R. Simpson (2016) Optimal design of water distribution networks by a discrete state transition algorithm, Engineering Optimization, 48:4, 603-628, DOI: 10.1080/0305215X.2015.1025775

To link to this article: http://dx.doi.org/10.1080/0305215X.2015.1025775

Published online: 22 Apr 2015.

Engineering Optimization, 2016
Vol. 48, No. 4, 603–628, http://dx.doi.org/10.1080/0305215X.2015.1025775

Optimal design of water distribution networks by a discrete state transition algorithm

Xiaojun Zhou[a*], David Y. Gao[b] and Angus R. Simpson[c]

[a]*School of Information Science and Engineering, Central South University, Changsha, PR China;* [b]*School of Science, Information Technology and Engineering, Federation University Australia, Victoria, Australia;* [c]*School of Civil and Environmental Engineering, University of Adelaide, Adelaide, South Australia, Australia*

(*Received 16 April 2014; accepted 29 January 2015*)

In this study it is demonstrated that, with respect to model formulation, the number of linear and nonlinear equations involved in water distribution networks can be reduced to the number of closed simple loops. Regarding the optimization technique, a discrete state transition algorithm (STA) is introduced to solve several cases of water distribution networks. Firstly, the focus is on a parametric study of the 'restoration probability and risk probability' in the dynamic STA. To deal effectively with head pressure constraints, the influence is then investigated of the penalty coefficient and search enforcement on the performance of the algorithm. Based on the experience gained from training the Two-Loop network problem, a discrete STA has successfully achieved the best known solutions for the Hanoi, triple Hanoi and New York network problems.

Keywords: discrete state transition algorithm; water distribution network; intelligent optimization; NP-hardness

1. Introduction

Pipes, hydraulic devices (pumps, valves, *etc.*) and reservoirs are connected in a water distribution network in a complex manner. The physical behaviour of a looped network is governed by a set of linear and nonlinear equations, including continuity and energy equations, and head loss functions. The overall planning tasks to be performed in water distribution networks consist of three kinds of problem: layout, design and operation. Although these problems are not independent of each other, they can be formulated and solved separately from a technical point of view since each one can be considered as a parameter when others are being solved. In this work, the optimal design problem is focused upon.

The optimal selection of pipe diameters to constitute a water distribution network respecting certain pressure requirements has been shown to be an NP-hard problem (Yates, Templeman, and Boffey 1984), mainly for two reasons: nonlinear equations and discrete-valued diameters. A terribly clumsy method for designing pipe networks is by enumeration or complete trial and error (Gessler 1985). Traditional methods linearize and relax the problem, firstly to facilitate the use of linear and nonlinear programming, and then they have to round off the solution to the nearest

*Corresponding author. Email: tiezhongyu2010@gmail.com

discrete diameters (Alperovits and Shamir 1977; Morgan and Goulter 1985; Goulter, Lussier, and Morgan 1986; Kessler and Shamir 1989; Eiger, Shamir, and Ben-Tal 1994). Such algorithms cannot guarantee global optima and sometimes cause infeasible solutions. In the last few decades, intelligent optimization techniques, including genetic algorithms (Dandy, Simpson, and Murphy 1996; Savic and Walters 1997; Simpson, Dandy, and Murphy 1994), simulated annealing (Cunha and Sousa 1999), shuffled complex evolution (Liong and Atiquzzaman 2004), ant colony optimization (Zecchin *et al*. 2005 2006), harmony search (Geem 2006), particle swarm optimization (Montalvo *et al*. 2008), differential evolution (Vasan and Simonovic 2010) and some of their hybrids (Cisty 2010; Haghighi, Samani, and Samani 2011), have found wide application in this field. The advantages of using these stochastic algorithms are: (i) simple representation of a discrete-valued solution; (ii) independence of the problem structure to some extent; (iii) easy computation due to the use of only information about the objective function; and (iv) a high probability of gaining the global optimum or an approximate global optimum in a reasonable amount of time.

Recently, a new intelligent optimization algorithm, called the state transition algorithm (STA), was proposed by Zhou, Yang, and Gui (2011a,b 2012 2014), and it showed powerful performance in continuous function optimization. Yang *et al.* (2013) proposed a discrete STA for solving the travelling salesman problem, and the results demonstrated that it consumed much less time and had better search ability than the well-known simulated annealing and ant colony optimization. The goal of this article is to apply a discrete STA to the optimal design problem of water distribution networks.

This article is organized as follows. In Section 2, the optimization model of water distribution networks is established, including the objective function, decision variables and some constraints. In Section 3, the basic key elements of the discrete STA are introduced. The focus is on the intelligent operators of the discrete STA and a parametric study of the 'restoration probability' and 'risk probability'. How to deal with the constraints and the implementation of the discrete STA for the optimal design problem are illustrated in Section 4. In Section 5, several case studies are given. The Two-Loop network is mainly studied to investigate the influence of the penalty coefficient and search enforcement on the performance of the discrete STA. The experience gained is applied to other cases, and the results obtained by the proposed discrete STA as well as comparisons with other optimization algorithms are illustrated. Conclusions are given in Section 6.

2. Formulation of the water distribution network optimization model

For a given layout of pipes and a set of specified demand patterns at the nodes, the optimal design of a water distribution network is to find the combination of commercial pipe sizes which gives the minimum cost, subject to the following constraints:

- continuity of flow;
- head loss;
- conservation of energy;
- minimum pressure head.

2.1. *The objective function*

Considering that the pipe layout, connectivity and imposed minimum head constraints are known, in the optimal design problem of the water distribution network, the pipe diameters are the only decision variables. As a result, the objective function is assured to be a cost function of pipe diameters

$$\min_{D_j \in \Omega} f_{\text{obj}} = \sum_{j=1}^{\text{NP}} L_j c(D_j), \tag{1}$$

where Ω is a set of commercial pipe sizes, NP is the number of pipes, and L_j is the length of pipe j, which is known in this study. $c(D_j)$ indicates that, for every commercial pipe size, there is a corresponding cost per unit associated with it.

2.2. *Continuity equation*

Conservation of mass at nodes or junctions in a water distribution network yields a set of linear algebraic equations in terms of flows. At each node, flow continuity should be satisfied:

$$-\sum Q_{\text{in}} + \sum Q_{\text{out}} + \text{DM} = 0, \tag{2}$$

where DM is the demand at the node, and Q_{in} and Q_{out} are the flow entering and leaving the node, respectively.

2.3. *Head loss equation*

The head loss in a pipe in the water distribution network can be computed from a number of empirically obtained equations. Two commonly used equations are the Darcy–Weisbach head loss equation and the Hazen–Williams head loss equation. The general expression for the head loss in a pipe j located between nodes i and k is given by

$$H_i - H_k = r_j Q_j |Q_j|^{\alpha-1} = \omega \frac{L_j}{C^\alpha D_j^\beta} Q_j |Q_j|^{\alpha-1}, \tag{3}$$

where H_i and H_k are nodal pressure head at the end of the pipe at node i and k, respectively; r_j is called the resistance factor for pipe j; Q_j is the flow in pipe j; ω is a numerical conversion constant depending on the units used; L_j is the length of pipe j; C is the roughness coefficient; and α and β are coefficients.

In the International System of Units (SI), $\omega = 10.6744$ or $\omega = 10.5088$, $\alpha = 1/0.54 = 1.852$ and $\beta = 2.63/0.54 = 4.871$ are employed in this study using the Hazen–Williams formula.

Remark 1 In the existing literature references, different researchers have used different values of ω in the Hazen–Williams head loss equation; $\omega = 10.6744$ and $\omega = 10.5088$ are the most commonly used. For fair comparison, both these two values are used herein for the parameter ω.

2.4. *Energy equation*

Energy conservation equations around closed simple loops or between fixed head nodes along required independent paths in a network are nonlinear. Upon traversing a closed simple loop or a required independent path, the sum of pipe head losses around the loop or the path must be zero, which can be expressed as

$$\sum_{j \in L_s} \omega \frac{L_j}{C^\alpha D_j^\beta} Q_j |Q_j|^{\alpha-1} - \sum_{j \in L_s} \text{EL}_j = 0, \tag{4}$$

where L_s is the index of a pipe in a closed simple loop or a required independent path; EL_j is the hydraulic grade line at reservoir j.

2.5. *Minimum pressure head*

The minimum pressure head constraints at each node are given as follows:

$$H_i \geq H_{i\,\min}, \quad \forall i = 1, \ldots, \mathrm{NJ}, \tag{5}$$

where $H_{i\,\min}$ is known, and NJ is the number of nodes.

It can be seen that the optimal design of water distribution networks is a discrete optimization. In the next section, an optimization algorithm, called the discrete state transition algorithm, is introduced to solve the problem.

3. A brief review of the discrete state transition algorithm

Consider the following unconstrained integer optimization problem:

$$\min f(x), \tag{6}$$

where $x = (x_1, \ldots, x_n)$, $x_i \in \mathcal{I} \subset \mathbb{Z}^m$ (m is the number of choices for each x_i), $i = 1, \ldots, n$, and $f(x)$ is a real-valued function.

3.1. *The framework of the discrete state transition algorithm*

If a solution to a specific optimization problem is described as a state, then the transformation to update the solution becomes a state transition. Without loss of generality, the unified form of discrete state transition algorithm can be described as

$$\left.\begin{aligned} x_{k+1} &= A_k(x_k) \bigoplus B_k(u_k) \\ y_{k+1} &= f(x_{k+1}), \end{aligned}\right\} \tag{7}$$

where $x_k \in \mathbb{Z}^n$ stands for a current state, corresponding to a solution of a discrete optimization problem; u_k is a function of external states; $A_k(\cdot)$, $B_k(\cdot)$ are transformation operators, which are usually state transition matrices; $\bigoplus$ is an operation admissible to operate on two states; f is the cost function or evaluation function.

As an intelligent optimization algorithm, the discrete state transition algorithm has the following five key elements.

(1) *Solution representation*. In the discrete STA, special representations are chosen, that is, the permutation of the set $\{1, 2, \ldots, n\}$, which can be easily manipulated by some intelligent operators discussed below. The reason that the operators are called 'intelligent' is due to their geometric properties (swap, shift, symmetry and substitute), and an intelligent operator has the same functional feature for different representations. A big advantage of such representations and operators is that, after each state transformation, the newly created state is always feasible, avoiding the difficulty of rounding off a continuous solution as in other cases.

To illustrate the advantages of the solution representation, the following unconstrained discrete optimization problem is considered:

$$\min f(x) = (x_1 - 1)^2 + (x_2 - x_1^2)^2 + (x_3 - x_2^2)^2, \quad x_i \in \{a_1, a_2, a_3\}, \ i = 1, 2, 3,$$

where $\{a_1, a_2, a_3\}$ can be integers, for instance $\{a_1, a_2, a_3\} = \{-1, 2, 3\}$, or can be discrete values, for instance $\{a_1, a_2, a_3\} = \{0.8, 1.2, 1.5\}$. In the discrete state transition algorithm, the indices $\{1, 2, 3\}$ are used for solution representation, since there exists a one-to-one relationship to $\{a_1, a_2, a_3\}$, for example $\hat{x} = (1, 1, 2)^{\mathrm{T}}$ is used to represent $x = (a_1, a_1, a_2)^{\mathrm{T}}$.

Then, the intelligent operators are applied to the representative solution. Suppose the current solution is $\hat{x} = (1, 1, 2)^{\mathrm{T}}$, after the following swap transformation matrix, then

$$\begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix},$$

which is also a feasible solution, representing $x = (a_2, a_1, a_1)^{\mathrm{T}}$.

However, in other cases, if the floating point numbers are used for solution representation, for instance, two parents $x = (1.0, 1.0, 2.0)$, $y = (1.0, 2.0, 2.0)$, then by arithmetic crossover in the genetic algorithm, the offspring may be $z = 0.5(x + y) = (1.0, 1.5, 2.0)$, which is infeasible, and a round-off procedure is necessary to remedy the intermediate solution.

(2) *Sampling*. When a transformation operator is exerted on a current state, the next state is not deterministic, that is to say, there are possibly different choices for the next state. It is not difficult to imagine that all possible choices will constitute a candidate set, or a 'neighbourhood'. Then the transformation is executed several times, which is called search enforcement (SE), on the current state for sampling a candidate set or 'neighbourhood'. Sampling is a very important factor in state transition algorithms since it only uses some samples (the number of samples is SE) to represent the 'neighbourhood', which can reduce the search space and avoid enumeration. For example, suppose the current solution is $\hat{x} = (1, 2, \ldots, n)^{\mathrm{T}}$, under the swap transformation of exchanging two random positions there will be $[n \times (n-1)]/(2 \times 1)$ possible candidate solutions in total. However, in discrete state transition algorithms, the value of search enforcement (SE) is fixed at less than 100. If n is large compared to $0.5(n^2 - n)$, then SE is quite small and can thus reduce the search space and avoid enumerating all possible solutions.

(3) *Local exploitation and global exploration*. In optimization algorithms, it is quite significant to design good local and global search operators. Local exploitation can guarantee high precision of a solution and convergent performance of an algorithm, and global exploration can avoid getting trapped into local minima or prevent premature convergence. In discrete optimization, it is extremely difficult to define a 'good' local optimal solution due to its dependence on a problem's structure, which leads to the same difficulty in the definition of local exploitation and global exploration. Anyway, in the discrete state transition algorithm, a small change to the current solution by a transformation is defined as local exploitation, while a big change to the current solution by a transformation is defined as global exploration.

The big difference between different optimization algorithms is the local and global operator designs. In discrete state transition algorithms, state transformation matrices are the main difference.

For example, the swap transformation matrix $A_k^{\mathrm{swap}}(m_a)$, it has the function of exchanging m_a random positions. Suppose that the current solution is $\hat{x} = (1, 2, 3, 4, 5, 6)^{\mathrm{T}}$. If $m_a = 2$, it may have

$$\begin{pmatrix}1\\2\\6\\4\\5\\3\end{pmatrix} = \begin{pmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&0&0&0&1\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&1&0&0&0\end{pmatrix} \times \begin{pmatrix}1\\2\\3\\4\\5\\6\end{pmatrix}.$$

In this case, the elements in positions 3 and 6 are exchanged. Since there is a small change to the current solution, the swap transformation with $m_a = 2$ is called a local search.If $m_a = 4$, it may have

$$\begin{pmatrix}3\\2\\1\\4\\6\\5\end{pmatrix} = \begin{pmatrix}0&0&1&0&0&0\\0&1&0&0&0&0\\1&0&0&0&0&0\\0&0&0&1&0&0\\0&0&0&0&0&1\\0&0&0&0&1&0\end{pmatrix} \times \begin{pmatrix}1\\2\\3\\4\\5\\6\end{pmatrix}.$$

In this case, the elements in positions 1, 3, 5 and 6 are exchanged. Since there is a big change to the current solution, the swap transformation with $m_a = 4$ is called a global search.

(4) *Self learning and regular communication*. State transition algorithms behave in two ways, one is individual-based, the other is population-based, which is certainly an extended version. The individual-based state transition algorithm focuses on self learning, in other words, with emphasis on the operators' designing and dynamic adjustment (details of which are given in the following). Undoubtedly, communication between different states is a promising strategy for state transition algorithms, as indicated by Zhou, Yang, and Gui (2012). Through communication, states can share information and cooperate with each other. However, how to communicate and when to communicate are key issues. In continuous state transition algorithms, an intermittent exchange strategy was proposed, which means that states communicate with each other at a certain frequency in a regular way.

(5) *Dynamic adjustment*. This is a potentially useful strategy for state transition algorithms. In the iteration process of an intelligent algorithm, the fitness value can decrease sharply in the early stages, but can stagnate in the late stages, due to the static environment. As a result, some perturbation should be added to activate the environment. In fact, dynamic adjustment can be understood and implemented in various ways. For example, the alternative use of different local and global operators is dynamic adjustment to some extent. Then, the search enforcement can be changed, the cost function can be varied, the dimension can be reduced, *etc*. Of course, the probability of accepting a bad solution is another dynamic adjustment, which is widely used in simulated annealing (SA). In SA, the Metropolis criterion (Metropolis *et al*. 1953) is used to accept a bad solution: the probabiltity p is given by

$$p = \exp\left(\frac{-\triangle E}{k_{\mathrm{B}} T}\right), \tag{8}$$

where $\triangle E = f(x_{k+1}) - f(x_k)$, k_{B} is the Boltzmann probability factor, and T is the temperature to regulate the process of annealing. In the early stages, the temperature is high, and it has a big probability of accepting a bad solution, while in the late stages the temperature is low, and it has very small probability of accepting a bad solution, which is the key point to guarantee convergence. It can be seen that the Metropolis criterion has the ability to escape from local optimality, but on the other hand, it will miss some 'good solutions' as well.

In this study, the individual-based STA with a dynamic adjustment strategy called 'risk and restoration in probability' is focused upon, and the main process of the dynamic discrete STA is shown in the following pseudocode:

```
repeat
    [Best,fBest] ← swap(fcn,Best,fBest,SE,n,m_a)
    [Best,fBest] ← shift(fcn,Best,fBest,SE,n,m_b)
    [Best,fBest] ← symmetry(fcn,Best,fBest,SE,n,m_c)
    [Best,fBest] ← substitute(fcn,Best,fBest,SE,set,n,m_d)
    if fBest < fBest* then                          ▷ greedy criterion
        Best* ← Best
        fBest* ← fBest
    end if
    if rand < p_1 then                              ▷ restoration in probability
        Best ← Best*
        fBest ← fBest*
    end if
until the maximum number of iterations is met
```

As for detailed explanations, the swap function in above pseudocode is given as follows, for example:

```
State ← op_swap(Best,SE,n,m_a)
[newBest,fnewBest] ← fitness(funfcn,State)
if fnewBest < fBest then                            ▷ greedy criterion
    Best ← newBest
    fBest ← fnewBest
else
    if rand < p_2 then                              ▷ risk in probability
        Best ← newBest
        fBest ← fnewBest
    end if
end if
```

From the pseudocodes, it can be seen that, in the discrete STA, the 'greedy criterion' is adopted overall to keep the incumbent 'Best*', and in particular cases, a bad solution 'Best' is accepted in each inner state transformation with a probability p_2, and at the same time the historical 'Best*' is restored in the outer iterative process with another probability p_1. The probability of accepting a bad solution strategy aims to escape from local optima, while the 'greedy criterion' and 'restoring the historical best solution in probability' are to guarantee good convergence. The 'risk and restoration in probability' strategy in the dynamic discrete STA will help to improve the global search ability.

3.2. *The representation, local and global operators*

In the discrete STA, an index of commercial size is used as a representation of a solution to the optimal design problem. For example, if there are eight pipes and for each pipe there are three choices, then the details of four special geometric operators are defined as follows.

(1) *Swap transformation*

$$x_{k+1} = A_k^{\text{swap}}(m_a)x_k, \tag{9}$$

where $A_k^{\text{swap}} \in \mathbb{Z}^{n\times n}$ is called the swap permutation matrix, m_a is a constant integer called the swap factor to control the maximum number of positions to be exchanged, while the

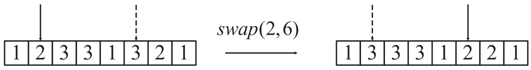

Figure 1. Illustration of the swap transformation.

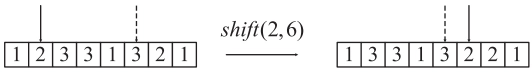

Figure 2. Illustration of the shift transformation.

positions are random. If $m_a = 2$, the swap operator is regarded as local exploitation, and if $m_a \geq 3$, the swap operator is regarded as global exploration. Figure 1 gives the function of the swap transformation graphically when $m_a = 2$.

$$\begin{pmatrix}1\\3\\3\\3\\1\\2\\2\\1\end{pmatrix} = \begin{pmatrix}1&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&1&0&0&0&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\end{pmatrix} \times \begin{pmatrix}1\\2\\3\\3\\1\\3\\2\\1\end{pmatrix}$$

(2) *Shift transformation*

$$x_{k+1} = A_k^{\text{shift}}(m_b)x_k, \tag{10}$$

where $A_k^{\text{shift}} \in \mathbb{Z}^{n\times n}$ is called the shift permutation matrix, m_b is a constant integer called the shift factor to control the maximum length of consecutive positions to be shifted. Note that the position selected to be shifted next, and the positions to be shifted, are chosen randomly. Similarly, shift transformation is regarded as local exploitation or global exploration when $m_b = 1$ or $m_b \geq 2$, respectively. To clarify, if $m_b = 1$, position 2 is set to be shifted after position 6, as described in Figure 2.

$$\begin{pmatrix}1\\3\\3\\1\\3\\2\\2\\1\end{pmatrix} = \begin{pmatrix}1&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\\0&1&0&0&0&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\end{pmatrix} \times \begin{pmatrix}1\\2\\3\\3\\1\\3\\2\\1\end{pmatrix}$$

(3) *Symmetry transformation*

$$x_{k+1} = A_k^{\text{sym}}(m_c)x_k, \tag{11}$$

where $A_k^{\text{sym}} \in \mathbb{Z}^{n\times n}$ is called the symmetry permutation matrix, m_c is a constant integer called the symmetry factor to control the maximum length of subsequent positions as centre. Note that the component before the subsequent positions, and the consecutive positions to be symmetrized, are all created randomly. Considering that the symmetry transformation can make a big change to the current solution, it is intrinsically regarded as global exploration. For instance, if $m_c = 0$, let position 3 be chosen, then the subsequent position or the centre

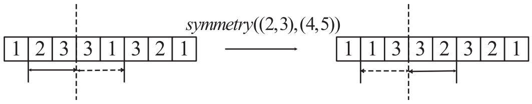

Figure 3. Illustration of the symmetry transformation.

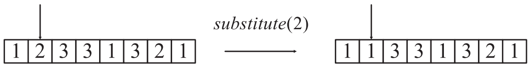

Figure 4. Illustration of the substitute transformation.

is $\{\varnothing\}$, and the consecutive positions $\{4,5\}$ with components $(3,1)$, and the function of symmetry transformation are given in Figure 3.

$$\begin{pmatrix}1\\1\\3\\3\\2\\3\\2\\1\end{pmatrix}=\begin{pmatrix}1&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&1&0&0&0&0\\0&0&1&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\end{pmatrix}\times\begin{pmatrix}1\\2\\3\\3\\1\\3\\2\\1\end{pmatrix}$$

(4) *Substitute transformation*

$$x_{k+1}=A_k^{\text{sub}}(m_d)x_k+B_k^{\text{sub}}(m_d)u_k, \tag{12}$$

where $A_k^{\text{sub}}, B_k^{\text{sub}} \in \mathbb{Z}^{n\times n}$ are called substitute permutation matrices, m_d is a constant integer called the substitute factor to control the maximum number of positions to be substituted. Note that the positions are created randomly. If $m_d = 1$, the substitute operator is regarded as local exploitation, and if $m_d \geq 2$, the substitute operator is regarded as global exploration. Figure 4 illustrates the function of the substitute transformation vividly when $m_d = 1$.

$$\begin{pmatrix}1\\1\\3\\3\\1\\3\\2\\1\end{pmatrix}=\begin{pmatrix}1&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\end{pmatrix}\times\begin{pmatrix}1\\2\\3\\3\\1\\3\\2\\1\end{pmatrix}+\begin{pmatrix}0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{pmatrix}\times\begin{pmatrix}0\\1\\0\\0\\0\\0\\0\\0\end{pmatrix}$$

Remark 2 Compared with the operators in other stochastic optimization algorithms, there are two main differences: one is that both local search operators and global search operators are designed purposely and specially for the discrete state transition algorithm; another is that the solution representation is much simpler and a new solution can be generated conveniently and flexibly by these local and global operators.

3.3. *Parameter selection*

In the state transformations, there are four factors to control the intensity between local search and global search. If the values of these factors are big, the four operators are all considered as global search. However, it is very difficult to decide for what values they should be taken as local search or global search. For simplicity and efficiency, as also demonstrated in Yang *et al.* (2013), the larger the values of these factors, the poorer the performance of the discrete STA, the swap, shift and substitute operators are taken as local search, and the symmetry operator is considered as global search; therefore, the values $m_a = 2, m_b = 1, m_d = 1$ and $m_c = 0$ are consistently set in the rest of this article.

On the other hand, the restoration probability p_1 and the risk probability p_2 play a significant role in the discrete STA. To select an appropriate combination of parameters (p_1, p_2), a Monte Carlo simulation study is arranged in the following.

Considering the following optimization problem:

$$\min_{x \geq 0} f(x) = x,$$

where a new candidate x_{k+1} is generated by the following update equation:

$$x_{k+1} = r_1.$$

Here, r_1 is a uniformly distributed random number over the interval $(0, 1)$.

The same 'risk and restoration in probability' strategy is used to accept a new solution as follows:

Initialize $x^* \leftarrow 0.5, f^* \leftarrow f(x^*), x_k \leftarrow x^*$
repeat
    $x_{k+1} \leftarrow r_1$
    if $f(x_{k+1}) < f(x_k)$ **then**
        $x_k \leftarrow x_{k+1}$
    else if $r_2 < p_2$ **then** ▷ risk in probability
        $x_k \leftarrow x_{k+1}$
    end if
    if $f(x_k) < f^*$ **then** ▷ greedy criterion
        $f^* \leftarrow f(x_k)$
    end if
    if $r_3 < p_1$ **then** ▷ restoration in probability
        $x_k \leftarrow x^*$
    end if
until the maximum number of iterations is met

Here, r_2 and r_3 are uniformly distributed random numbers over the interval $(0, 1)$ and $f^* = f(x^*)$ is historically the best solution.

Various groups of (p_1, p_2) are tested for the experiment, in which the maximum number of iterations is 1e3, and 1e4 runs are carried out for each group. The experimental results are shown in Table 1. Without loss of generality, the group $(p_1, p_2) = (0.1, 0.1)$ is adopted in this article for the following study due to its good performance and simplicity.

Remark 3 It can be seen that the Monte Carlo simulation study for the optimization problem can be considered as a general case for the selection of p_1 and p_2, and the group $(p_1, p_2) = (0.1, 0.1)$ can also be applied to other optimization problems with the 'risk and restoration in probability' strategy.

Table 1. A Monte Carlo simulation study.

p_1	p_2 0.1	0.3	0.5	0.7	0.9
0.1	**9.9254e−4 ± 9.8274e−4**[a]	0.0010 ± 9.8255e−4	0.0010 ± 0.0010	**9.9027e−4 ± 9.9498e−4**	**9.7423e−4 ± 9.8163e−4**
0.3	0.0010 ± 9.9359e−4	0.0010 ± 0.0010	0.0010 ± 0.0010	**9.8736e−4 ± 9.8159e−4**	9.9254e−4 ± 0.0010
0.5	0.0010 ± 9.9755e−4	**9.9249e−4 ± 9.7547e−4**	0.0010 ± 0.0010	0.0010 ± 0.0010	0.0010 ± 0.0010
0.7	0.0010 ± 0.0010	0.0010 ± 0.0010	0.0010 ± 0.0010	**9.9456e−4 ± 9.8730e−4**	**9.9609e−4 ± 9.9991e−4**
0.9	0.0010 ± 9.9637e−4	0.0010 ± 9.9667e−4	**9.8258e−4 ± 9.8261e−4**	0.0010 ± 0.0010	0.0010 ± 0.0010

[a]Indicates mean ± standard deviation.

4. Implementation of the discrete STA

The above discrete STA is essentially for unconstrained discrete optimization problems. To realize the optimal design of water distribution networks, some constraints have to be dealt with. For the equality constraints on continuity of flow and conservation of energy, there exist some hydraulic analysis software packages, such as Epanet (Rossman 2000) and Kypipe (Wood 1980), in which the continuity and energy constraints are automatically satisfied. Considering that the continuity equations are linear, some pipe flows can be first fixed as known to solve the linear equations and then substituted into the energy equations, which can reduce the computational complexity of solving continuity equations (linear equations) and energy equations (nonlinear equations) simultaneously. The details can be found in the Two-Loop network, as shown in Section 5.1. It is not difficult to imagine that the number of nonlinear equations equals that of the simple closed loops or required independent paths in a network, and then the Newton–Raphson method is used to solve the nonlinear equations.

For minimum pressure head constraints, the most commonly used technique is the penalty function method, adding a penalty term when the corresponding constraint is violated. For example, the following scheme:

$$f_{\text{penal}} = pc \sum_{i=1}^{\text{NP}} \max\{0, H_{i\,\min} - H_i\}^{\rho}, \tag{13}$$

where pc is the penalty coefficient, and ρ is normally 1 or 2 ($\rho = 1$ in this study). Finally, the total cost is

$$f_{\text{cost}} = f_{\text{obj}} + f_{\text{penal}}. \tag{14}$$

A brief description of the steps using the discrete STA is given in the following.

Step 1: Create an initial *Best* solution. Generate a group of candidate solutions randomly (the size is the search enforcement, SE) and then select the fittest solution according to f_{cost}. Let $Best^* = Best$ and store $Best^*$.

Step 2: Update the *Best*. Use swap transformation to generate a group of candidate solutions on the basis of *Best*. If the fittest of the candidate solutions is better than *Best* according to f_{cost}, then accept the fittest solution as *Best*; otherwise, accept the fittest solution as *Best* with probability p_2. Similar procedures are adaptable to shift, symmetry and substitute transformations.

Step 3: Update the $Best^*$. The $Best^*$ is updated only when *Best* is better than $Best^*$.

Step 4: Restore the *Best*. The *Best* is restored to $Best^*$ with probability p_1.

Step 5: Go back to repeat Step 2 until the maximum number of iterations is met.

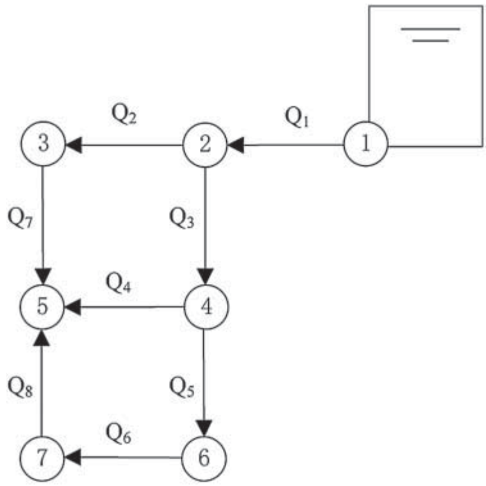

Figure 5. The Two-Loop network.

Remark 4 It should be noticed that once a solution is given, then the flow in each pipe is determined by solving the nonlinear equations, and then whether the minimum pressure head is satisfied can be evaluated and the corresponding penalty term to each head pressure constraint can be decided.

The proposed discrete state transition algorithm is programmed in MATLAB® R2010b (version 7.11.0.584) on an Intel® Core™ i3-2310M CPU @ 2.10 GHz in a Windows 7 environment. In the next section, some case studies are presented to investigate the performance of the proposed approach.

5. Case studies

The performance of the proposed discrete STA is investigated by three well-known water distribution networks, namely, the Two-Loop network, the Hanoi network and the New York network. First, a detailed study of the Two-Loop problem is given to show that the network system with eight unknowns governed by six linear equations and two nonlinear equations can be reduced to only two unknowns governed by two nonlinear equations. This Two-Loop case is also fully trained to study the effect of penalty coefficients and search enforcement on the performance of the algorithm. Based on the experience gained from the case, the known best solutions for the other two networks are also achieved by the algorithm.

5.1. *The Two-Loop network*

The layout of the Two-Loop network is shown in Figure 5. There is a single reservoir with a 210 m fixed head and eight pipes, each 1000 m long. The node data and cost data are given in Tables 2 and 3, and the minimum acceptable pressure requirements are all 30 m above ground level. The Hazen–Williams coefficient C is assumed to be 130 and $\omega = 10.5088$ for the Two-Loop network.

In this case study, an illustrative procedure is given to show how to reduce the complexity of solving the linear and nonlinear equations. The flow continuity equations of the Two-Loop

Table 2. Node data for the Two-Loop network.

Node	Demand ($m^3 h^{-1}$)	Ground level (m)
1	− 1120.0	210.00
2	100.0	150.00
3	100.0	160.00
4	120.0	155.00
5	270.0	150.00
6	330.0	165.00
7	200.0	160.00

Table 3. Cost data for the Two-Loop network.

No.	Diameter (in.)[a]	Cost (\$/m)	No.	Diameter (in.)[a]	Cost (\$/m)
1	1	2	8	12	50
2	2	5	9	14	60
3	3	8	10	16	90
4	4	11	11	18	130
5	6	16	12	20	170
6	8	23	13	22	300
7	10	32	14	24	550

[a] 1 in. = 2.54 cm.

network are given as follows:

$$\begin{cases} -Q_1 + Q_2 + Q_3 + \mathrm{DM}_2 = 0 \\ -Q_2 + Q_7 + \mathrm{DM}_3 = 0 \\ -Q_3 + Q_4 + Q_5 + \mathrm{DM}_4 = 0 \\ -Q_7 - Q_8 - Q_4 + \mathrm{DM}_5 = 0 \\ -Q_5 + Q_6 + \mathrm{DM}_6 = 0 \\ -Q_6 + Q_8 + \mathrm{DM}_7 = 0. \end{cases} \tag{15}$$

Assume that Q_4, Q_6 are fixed, then

$$\begin{cases} Q_1 = \mathrm{DM}_2 + \mathrm{DM}_3 + \mathrm{DM}_4 + \mathrm{DM}_5 + \mathrm{DM}_6 + \mathrm{DM}_7 \\ Q_2 = \mathrm{DM}_3 + \mathrm{DM}_5 + \mathrm{DM}_7 - Q_4 - Q_6 \\ Q_3 = \mathrm{DM}_4 + \mathrm{DM}_6 + Q_4 + Q_6 \\ Q_5 = \mathrm{DM}_6 + Q_6 \\ Q_7 = \mathrm{DM}_5 + \mathrm{DM}_7 - Q_4 - Q_6 \\ Q_8 = Q_6 - \mathrm{DM}_7. \end{cases} \tag{16}$$

The energy conservation equations can be formulated as

$$\begin{cases} r_3 Q_3 |Q_3|^{\alpha-1} + r_4 Q_4 |Q_4|^{\alpha-1} - r_7 Q_7 |Q_7|^{\alpha-1} - r_2 Q_2 |Q_2|^{\alpha-1} = 0 \\ r_5 Q_5 |Q_5|^{\alpha-1} + r_6 Q_6 |Q_6|^{\alpha-1} + r_8 Q_8 |Q_8|^{\alpha-1} - r_4 Q_4 |Q_4|^{\alpha-1} = 0 \end{cases} \tag{17}$$

and the head loss equations

$$\begin{cases} H_2 = \text{Head} - r_1 Q_1 |Q_1|^{\alpha-1} - G_2 \geq H_{2\min} \\ H_3 = H_2 - r_2 Q_2 |Q_2|^{\alpha-1} - G_3 \geq H_{3\min} \\ H_4 = H_2 - r_3 Q_3 |Q_3|^{\alpha-1} - G_4 \geq H_{4\min} \\ H_5 = H_4 - r_4 Q_4 |Q_4|^{\alpha-1} - G_5 \geq H_{5\min} \\ H_6 = H_4 - r_5 Q_5 |Q_5|^{\alpha-1} - G_6 \geq H_{6\min} \\ H_7 = H_6 - r_6 Q_6 |Q_6|^{\alpha-1} - G_7 \geq H_{7\min}, \end{cases} \tag{18}$$

where G_i $(i = 2, \ldots, 7)$ is ground level.

Remark 5 It should be noted that there are six linear equations and two nonlinear equations involving eight unknown flows in the original network system. However, by fixing Q_4, Q_6 as assumedly known variables, the above procedures show that it is only necessary to solve the nonlinear system (17) with two unknowns (Q_4, Q_6). That is to say, the number of linear and nonlinear equations involved in water distribution networks can be reduced to the number of closed simple loops.

For the Two-Loop network, a diameter has to be selected for each pipe, and for each pipe there is a choice of 14. It is not difficult to imagine that when choosing a numerical order (No.), it corresponds to an exact diameter. That is the reason why the discrete STA uses the permutation $\{1, 2, \ldots, n\}$ as its decision variables and all the intelligent operators are operated on a certain permutation.

Next, an empirical study of the the Two-Loop network by the proposed discrete STA is conducted to investigate the influence of the remained parameters, namely the search enforcement (SE) and the penalty coefficient (*pc*). SE is set to be 0.5, 1, 2, 3 and 4 times the dimension of the decision variable. Considering that the average cost times the average pipe length is 1.0335e5 and the average of the minimum pressure heads is 30, the order of magnitude for *pc* is set at 1e4. In this situation, *pc* is fixed at 1e4, 2e4, 4e4, 8e4 and 1e5, or increases from 1e4 to 1e5 in a linear way. The maximum number of iterations is set at 2e2, and a total of 20 runs are executed for each group of search enforcement SE and penalty coefficient *pc*.

As can be seen from Table 4, for a fixed SE, the search ability is declining as the *pc* increases, but the feasibility rate increases simultaneously with the *pc*. For a fixed *pc*, the search ability is increasing as the SE increases from four to eight but declining as the SE increases any more. When the *pc* varies in the iterative process, the performance is not the best but much more satisfactory than a constant one to some extent. By observation, it can be seen that setting SE to be the dimensionality of the decision variable is a good choice, and in this setting environment, $pc = 2\text{e}4$ is a good penalty coefficient. Figure 6 gives the iterative curves of the best solutions gained when SE = 8 and $pc = 2\text{e}4$, respectively. It should be emphasized that best solution is $419,000 in all cases.

Remark 6 Under the circumstance, the minimum function evaluations to achieve the best known solution is 2048, which takes up 0.000,1387% of all possible combinations ($14^8 = 1.4758\text{e}9$).

Table 5 gives the best solutions gained by various algorithms, and it can be seen that the STA can achieve the best known solution in this case. It should be noted that the same solution was also achieved by GAs (Savic and Walters 1997), SA (Cunha and Sousa 1999) and HS

Table 4. An empirical study of the Two-Loop network.

SE	*pc*					
	1e4	2e4	4e4	8e4	1e5	1e4 → 1e5
4	4.2978e5 ± 1.4882e4(55%)[a]	4.3631e5 ± 1.3394e4(85%)	4.5184e5 ± 2.3575e4(95%)	4.5063e5 ± 1.7400e4(95%)	4.4190e5 ± 1.6121e4(95%)	4.4368e5 ± 1.8563e4(90%)
8	4.2195e5 ± 1.4853e4(65%)	**4.3181e5 ± 1.3870e4(85%)**	4.3526e5 ± 1.2721e4(90%)	4.3577e5 ± 1.2903e4(95%)	4.4085e5 ± 1.5853e4(90%)	4.3620e5 ± 1.5702e4(80%)
16	4.2682e5 ± 1.2946e4(75%)	4.3340e5 ± 1.5347e4(80%)	4.3410e5 ± 1.2004e4(90%)	431, 550 ± 1.4406e4(100%)	4.3458e+5 ± 1.4992e4(90%)	433, 600 ± 1.4207e4(100%)
24	4.2380e5 ± 1.2756e4(75%)	4.3193e5 ± 1.2898e4(95%)	4.3555e5 ± 1.5049e4(90%)	440, 950 ± 1.4417e4(100%)	432, 600 ± 1.5398e4(100%)	4.3073e5 ± 1.3252e4(95%)
32	4.2686e5 ± 1.5549e4(55%)	4.3046e5 ± 1.5523e4(80%)	4.3376e5 ± 1.3900e4(95%)	451, 400 ± 5.4648e4(100%)	4.3600e5 ± 1.7731e4(85%)	4.3305e5 ± 1.4657e4(95%)

[a]Indicates the percentage of feasible solutions.

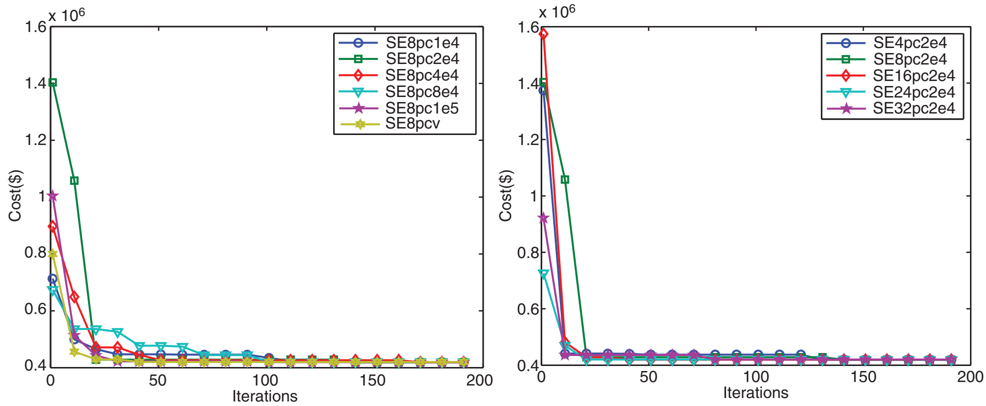

Figure 6. Iterative curves of best solutions when SE = 8 and *pc* = 2e4 for the Two-Loop problem, respectively.

Table 5. Solutions for the Two-Loop network.

Pipe	(Alperovits and Shamir 1977)	(Goulter, Lussier, and Morgan 1986)	(Kessler and Shamir 1989)	STA (fixed)	STA (variable)
1	20	20	18	18	18
	18	18			
2	8	10	12	10	10
	6		10		
3	18	16	16	16	16
4	8	6	3	4	4
	6	4	2		
5	16	16	16	16	16
		14	14		
6	12	12	12	10	10
	10	10	10		
7	6	10	10	10	10
		8	8		
8	6	2	3	1	1
	4	1	2		
Cost($)	497,525	435,015	417,500	419,000	419,000

Table 6. Pressure heads for the Two-Loop network (unit: m).

Node	(Alperovits and Shamir 1977)	(Goulter, Lussier, and Morgan 1986)	(Kessler and Shamir 1989)	STA (fixed and variable)
2	53.96	54.30	53.26	53.24
3	32.32	33.19	30.08	30.49
4	44.97	44.19	43.64	43.44
5	32.31	32.32	30.10	33.78
6	31.19	31.19	30.08	30.43
7	31.57	31.57	30.09	30.54

(Geem 2006) with function evaluations at 250,000, 70,000 and 5000, respectively. Although the solution in Kessler and Shamir (1989) is even better, it should be noted that it involves segmental pipes. The pressure heads for the Two-Loop network obtained by various algorithms are given in Table 6.

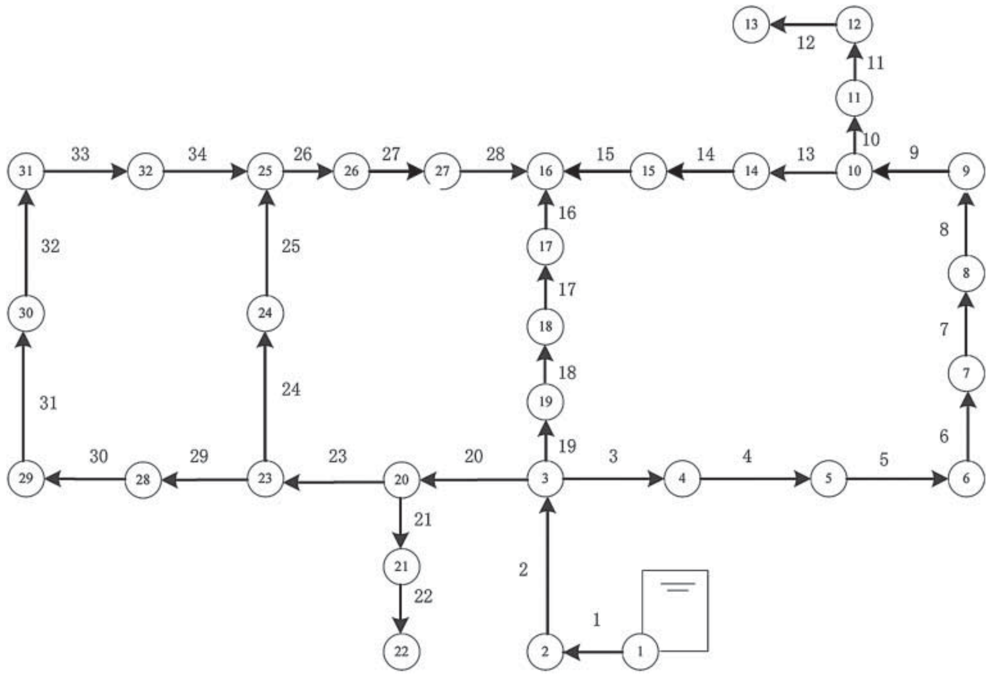

Figure 7. The Hanoi network.

Table 7. Cost data for the Hanoi network.

No.	Diameter (in.)	Cost ($/m)
1	12	45.726
2	16	70.400
3	20	98.387
4	24	129.333
5	30	180.748
6	40	278.280

5.2. *Hanoi network*

The layout of the Hanoi network is given in Figure 7. There are 32 nodes, 34 pipes and 3 loops in this network system. At node 1, there exists a reservoir with a 100 m fixed head. The cost data, and the pipe and node data are given in Tables 7 and 8, respectively. The minimum acceptable pressure requirements at all nodes are also fixed at 30 m and the Hazen–Williams coefficient C is assumed to be 130 as well.

From the experience gained from the training of the Two-Loop network, the search enforcement SE does not affect the performance of the discrete STA explicitly, but the penalty coefficient *pc* plays a significant role in the search ability and the solution feasibility, and a good penalty coefficient can be evaluated by setting the order of magnitude equal to the average pipe length times the minimum pressure head.

For the Hanoi network, the search enforcement SE is set at 20, and the penalty coefficient *pc* is fixed at 4e4, or varies from 4e4 to 1e5 in a linearly increasing way. The maximum number of iterations is set at 1e3, and a total of 20 runs are executed for both fixed and variable *pc*.

Table 8. Pipe and node data for the Hanoi network.

Pipe	Length (m)	Pipe	length (m)	Node	Demand (m^3/h)	Node	Demand (m^3/h)
1	100	18	800	1	– 19,940	18	1345
2	1350	19	400	2	890	19	60
3	900	20	2200	3	850	20	1275
4	1150	21	1500	4	130	21	930
5	1450	22	500	5	725	22	485
6	450	23	2650	6	1005	23	1045
7	850	24	1230	7	1350	24	820
8	850	25	1300	8	550	25	170
9	800	26	850	9	525	26	900
10	950	27	300	10	525	27	370
11	1200	28	750	11	500	28	290
12	3500	29	1500	12	560	29	360
13	800	30	2000	13	940	30	360
14	500	31	1600	14	615	31	105
15	550	32	150	15	280	32	805
16	2730	33	860	16	310	–	–
17	1750	34	950	17	865	–	–

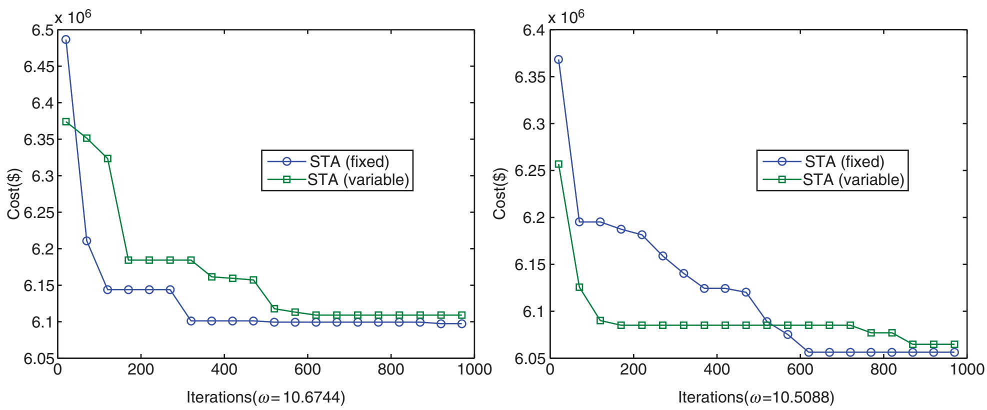

Figure 8. Iterative curves of the best solutions using the STA for the Hanoi problem when ω is 10.6744 and 10.5088, respectively.

Figure 8 gives the iterative curves of best solutions using the STA for the Hanoi problem when ω is 10.6744 and 10.5088, respectively.

Remark 7 Under the circumstances, the minimum number of function evaluations to achieve the best known solution is 23,240, which takes up 8.1114e−21% of all possible combinations ($6^{34} = 2.8651e26$).

Table 9 gives the best solutions gained by various algorithms, and it can be seen that, if $\omega =$ 10.5088, the STA with fixed *pc* can achieve the best known solution in this case at a cost of 6.056 million dollars, while the solution using the STA with variable *pc* obtains a solution at a cost of 6.065 million dollars. If $\omega = 10.6744$, the STA with fixed *pc* can achieve the best known solution at 6.097 million dollars, and it can obtain 6.109 million dollars with variable *pc*. Savic and Walters (1997) used the GA to obtain the solution with 1,000,000 function evaluations. The solution gained in Zecchin *et al.* (2006) using ACO need 100,000 function evaluations. Exactly the same solution was achieved by SA (Cunha and Sousa 1999) and HS (Geem 2006) as well, with the function evaluations of 53, 000 and 200, 000, respectively. It should be noted

Table 9. Solutions for the Hanoi network.

Pipe	(Savic and Walters 1997)	(Zecchin *et al.*, 2006)	(Haghighi, Samani, and Samani 2011)	STA(fixed) $\omega = 10.6744$	$\omega = 10.5088$	STA(variable) $\omega = 10.6744$	$\omega = 10.5088$
1	40	40	40	40	40	40	40
2	40	40	40	40	40	40	40
3	40	40	40	40	40	40	40
4	40	40	40	40	40	40	40
5	40	40	40	40	40	40	40
6	40	40	40	40	40	40	40
7	40	40	40	40	40	40	40
8	40	40	40	40	40	40	40
9	40	40	30	40	40	30	30
10	30	30	30	30	30	30	30
11	24	24	30	24	24	30	30
12	24	24	24	24	24	24	24
13	20	20	16	20	20	20	20
14	16	12	12	16	16	12	16
15	12	12	12	12	12	12	12
16	12	12	16	12	12	12	12
17	16	20	20	16	16	20	16
18	20	24	24	24	20	20	24
19	20	20	24	20	20	24	20
20	40	40	40	40	40	40	40
21	20	20	20	20	20	20	20
22	12	12	12	12	12	12	12
23	40	40	40	40	40	40	40
24	30	30	30	30	30	30	30
25	30	30	30	30	30	30	30
26	20	20	20	20	20	20	20
27	12	12	12	12	12	12	12
28	12	12	12	12	12	12	12
29	16	16	16	16	16	16	16
30	16	16	12	16	12	16	12
31	12	12	12	12	12	12	12
32	12	12	16	12	16	16	16
33	16	16	20	16	16	16	16
34	20	20	24	20	24	20	24
Cost($M)	6.073	6.134	6.190	6.097	6.056	6.109	6.065

that in GA (Savic and Walters 1997), SA (Cunha and Sousa 1999) and HS (Geem 2006), they used $\omega = 10.5088$, while in ACO (Zecchin *et al*. 2006), $\omega = 10.6744$ was adopted. The pressure heads for the Hanoi network obtained using the discrete STA with fixed and variable *pc* are given in Table 10.

5.3. *New York network*

The layout of the New York network is given in Figure 9. There are 20 nodes, 21 pipes and 1 loop in this network system. At node 1, there exists a reservoir with 300 ft fixed head. The New York problem is different from the other two cases, because pipes already exist in the old system. The common objective of this problem is to determine additional parallel pipes added to the existing ones to meet increased water demands while maintaining the minimum pressure requirements. The the cost data, and pipe and node data are given in Tables 11 and 12, respectively. The Hazen–Williams coefficient C is assumed to be 100 in this case.

For the New York network, the search enforcement SE is also set at 20, and the penalty coefficient *pc* is fixed at 2e6, or varies from 1e6 to 1e7 in a linearly increasing way. The maximum

Table 10. Pressure heads for the Hanoi network (unit: m).

Node	STA(fixed) $\omega = 10.6744$	$\omega = 10.5088$	STA(variable) $\omega = 10.6744$	$\omega = 10.5088$	Node	STA(fixed) $\omega = 10.6744$	$\omega = 10.5088$	STA(variable) $\omega = 10.6744$	$\omega = 10.5088$
1	100.00	100.00	100.00	100.00	17	33.56	30.51	38.00	33.20
2	97.14	97.17	97.14	97.17	18	49.94	44.29	44.89	50.16
3	61.64	61.99	61.64	61.99	19	55.08	55.90	58.68	55.37
4	56.90	57.23	57.08	57.34	20	50.53	50.89	50.43	50.90
5	51.02	51.31	51.42	51.56	21	41.18	41.57	41.07	41.59
6	44.82	45.07	45.49	45.48	22	36.01	36.42	35.90	36.44
7	43.36	43.61	44.10	44.06	23	44.41	44.73	44.21	44.76
8	41.63	41.85	42.48	42.37	24	39.23	39.03	38.91	39.07
9	40.25	40.44	41.20	41.02	25	35.98	35.34	35.56	35.40
10	39.23	39.40	37.39	37.01	26	32.25	31.44	31.58	31.53
11	37.67	37.85	35.83	35.45	27	31.20	30.15	30.22	30.29
12	34.24	34.43	34.68	34.30	28	35.76	39.12	35.60	39.15
13	30.03	30.24	30.46	30.10	29	31.06	30.21	30.94	30.26
14	35.61	35.49	34.64	33.66	30	30.10	30.47	30.01	30.52
15	33.87	33.44	30.86	32.17	31	30.58	30.75	30.13	30.80
16	31.61	30.36	30.38	30.53	32	31.84	33.20	31.41	33.26

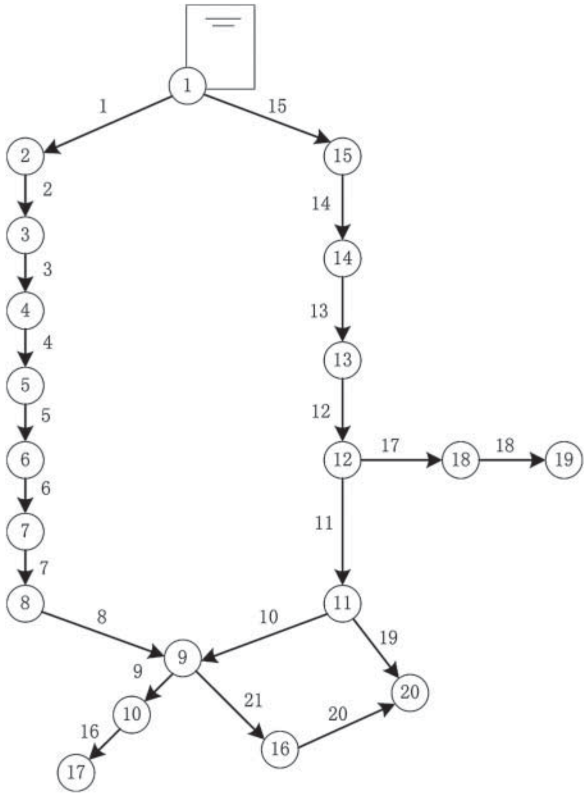

Figure 9. The New York network.

Table 11. Cost data for the New York network.

No.	Diameter (in.)	Cost ($/foot)	No.	Diameter (in.)	Cost ($/foot)
1	0	0.00	9	120	417.0
2	36	93.5	10	132	469.0
3	48	134.0	11	144	522.0
4	60	176.0	12	156	577.0
5	72	221.0	13	168	632.0
6	84	267.0	14	180	689.0
7	96	316.0	15	192	746.0
8	108	365.0	16	204	804.0

number of iterations is set at 2e3 and 1e3 for $\omega = 10.6744$ and $\omega = 10.5088$, respectively, and a total of 20 runs are executed for both fixed and variable *pc*. Figure 10 gives the iterative curves of the best solutions using the STA with fixed and variable *pc* for the New York problem when ω is 10.6744 and 10.5088, respectively.

Remark 8 Under the circumstances, the minimum number of function evaluations to achieve the best known solution is 5200, which takes up $2.6883\text{e}-19\%$ of all possible combinations $(16^{21} = 1.9343\text{e}25)$.

Table 13 gives the best solutions gained by various algorithms, and it can be seen that the STA with both fixed and variable *pc* can achieve the best known solution at a cost of 37.13 million

Table 12. Pipe and node data for the New York network.

Pipe	Length (ft)[a]	Diameters (in.)	Node	Demand (ft^3/s)[b]	Min Total Head (ft)
1	11,600	180	1	17.5	300.0
2	19,800	180	2	92.4	255.0
3	7,300	180	3	92.4	255.0
4	8,300	180	4	88.2	255.0
5	8,600	180	5	88.2	255.0
6	19,100	180	6	88.2	255.0
7	9,600	132	7	88.2	255.0
8	12,500	132	8	88.2	255.0
9	9,600	180	9	170.0	255.0
10	11,200	204	10	1.0	255.0
11	14,500	204	11	170.0	255.0
12	12,200	204	12	117.1	255.0
13	24,100	204	13	117.1	255.0
14	21,100	204	14	92.4	255.0
15	15,500	204	15	92.4	255.0
16	26,400	72	16	170.0	260.0
17	31,200	72	17	57.5	272.8
18	24,000	60	18	117.1	255.0
19	14,400	60	19	117.1	255.0
20	38,400	60	20	170.0	255.0
21	26,400	72	–	–	–

[a] 1 ft = 0.3048 m.
[b] 1 ft^3/s = 28.3168 L/s.

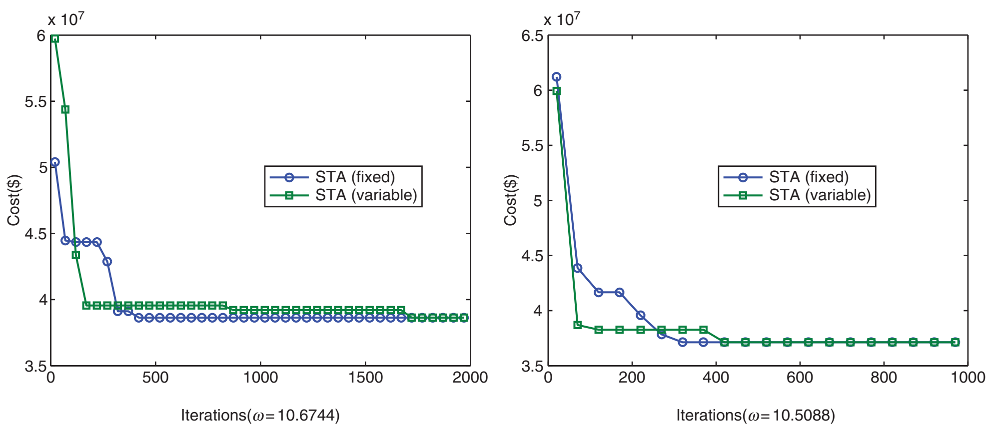

Figure 10. Iterative curves of the two best solutions using the STA for the New York problem when ω is 10.6744 and 10.5088, respectively.

dollars with $\omega = 10.5088$. As a matter of fact, the same solution with $\omega = 10.5088$ was also gained by GA (Savic and Walters 1997) with the function evaluations at 1000,000. It should be noted that $\omega = 10.5088$ in Gessler (1985) and Morgan and Goulter (1985), while $\omega = 10.6744$ in Dandy, Simpson, and Murphy (1996). The pressure heads for the New York network obtained by the discrete STA are given in Table 14.

5.4. *Triple Hanoi network*

The triple Hanoi network is an extension of the original Hanoi network, as shown in Figure 11. All the corresponding parameters for the nodes and lines in the triple Hanoi network are the same

Table 13. Solutions for the New York network.

Pipe	(Gessler 1985)	(Morgan and Goulter 1985)	(Dandy Simpson, and Murphy 1996)	STA(fixed) $\omega = 10.6744$	$\omega = 10.5088$	STA(variable) $\omega = 10.6744$	$\omega = 10.508$
1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0
7	100	144	0	144	108	144	108
8	100	144	0	0	0	0	0
9	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0
15	0	0	120	0	0	0	0
16	100	96	84	96	96	96	96
17	100	96	96	96	96	96	96
18	80	84	84	84	96	84	84
19	60	60	72	72	72	72	72
20	0	0	0	0	0	0	0
21	80	84	72	72	72	72	72
Cost($M)	41.80	39.20	38.80	38.64	37.13	38.64	37.13

Table 14. Pressure heads for the New York network using the STA (unit: m).

Node	$\omega = 10.6744$	$\omega = 10.5088$	Node	$\omega = 10.6744$	$\omega = 10.5088$
1	300.00	300.00	11	273.85	273.86
2	294.20	294.33	12	275.12	275.15
3	286.14	286.47	13	278.09	278.12
4	283.78	284.16	14	285.55	285.58
5	281.68	282.13	15	293.32	293.34
6	280.06	280.55	16	260.05	260.16
7	277.50	278.08	17	272.85	272.86
8	276.65	276.51	18	261.15	261.30
9	273.76	273.76	19	255.02	255.21
10	273.73	273.73	20	260.70	260.81

as in the original Hanoi network on all single parts except for the length of four pipes (pipes 1, 2, 35 and 68), the head in the reservoir, and the demand in node 3. The detailed changes in the triple Hanoi network are mentioned below: the head in the reservoir is changed to 105 m; the lengths of pipes 1 and 2 are 1 m and 1786.50 m, respectively; the length of both pipes 35 and 68 is 1641.69 m; and the demand of node 3 is zero.

Under these conditions, as demonstrated by Cisty (2010), the objective function of the triple Hanoi network can be formulated as

$$f_{\mathrm{TH}} = 3f_{\mathrm{H}} - 3L_1C_1 - 3L_2C_1 + (1 + 1786.5 + (2 \times 1641.69))C_1,$$

where f_{H} is the referenced optimal cost of the original Hanoi network, L_1, L_2 are the lengths of the first and second pipes in the original network, and C_1 is the unit price of the diameter 1016 mm.

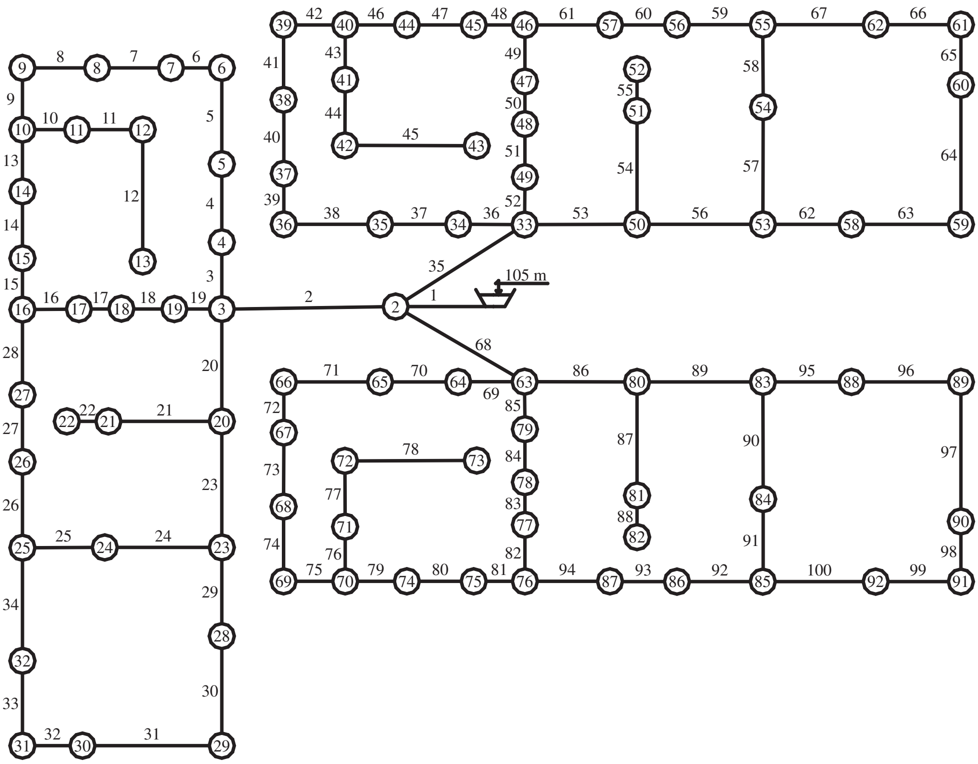

Figure 11. The Triple Hanoi network.

Table 15. Solutions for the Triple Hanoi network.

Network	GA	Cisty (2010)	Geem (2006)	STA
Hanoi	6,081,087	6,057,697	6,081,087	6,056,362
Triple Hanoi	19,269,160	18,394,255	18,839,302	18,369,692

It can be seen that the global optimal solution of the triple Hanoi network is dependent of the solution obtained from solving the original Hanoi network problem. By using the proposed discrete state transition algorithm, the optimal cost $f_{\mathrm{TH}} = 18{,}369{,}692.48$ can be obtained. The best results obtained by different optimization algorithms can be seen in Table 15. It can be seen that the discrete state transition algorithm can find the best optimal solution as well.

6. Conclusion

The complexity of the water distribution network comes from two aspects, one is the linear and nonlinear equations, which are commonly handled by a hydraulic solver to ensure that the continuity and head loss equations are satisfied automatically, the other difficulty is that the commercial pipe size is discrete, which has been proved to be NP-hard.

In this article, it is shown that the network system can be reduced to the dimensionality of the number of closed simple loops or required independent paths, which can reduce the computational complexity of solving linear and nonlinear equations simultaneously to a large extent.

To overcome the NP-hardness, a new intelligent optimization algorithm called the discrete state transition algorithm is introduced to find the optimal or suboptimal solution. There are four intelligent operators in the discrete STA, which are easy to understand and to implement. The 'restoration in probability' p_1 and 'risk in probability' p_2 strategy in the discrete STA is used to escape from local optima and increase the probability of capturing the global optimum.

At first, a Monte Carlo simulation is studied to investigate a good combination of p_1 and p_2, and it can be seen that $(p_1, p_2) = (0.1, 0.1)$ is a good choice. Then, an empirical study of the Two-Loop network focuses upon studying the network, and it can be seen that the penalty coefficient plays a significant role in the search ability and solution feasibility.

Based on the experience gained from the Two-Loop problem, the discrete STA has been successfully applied to the Hanoi, the triple Hanoi and New York networks, and the results show that the discrete STA can achieve the best known solutions with fewer function evaluations. The success of the discrete STA in the optimal design of water distribution networks has demonstrated that the discrete STA is a promising alternative in combinatorial optimization.

In the future, the proposed discrete state transition algorithm will be extended to large size water distribution networks by combining it with decomposition methods and the proposed approach will also be extended to multi-objective optimization problems involved in water distribution networks.

Acknowledgement

We would like to thank the anonymous reviewers for their valuable comments and suggestions that helped improve the quality of this paper.

Funding

This work was supported by the National Science Foundation for Distinguished Young Scholars of China [61025015], the Foundation for Innovative Research Groups of the National Natural Science Foundation of China [61321003] and the China Scholarship Council.

References

Alperovits, E., and U. Shamir. 1977. "Design of Optimal Water Distribution Systems." *Water Resources Research* 13 (6): 885–900.

Cisty, M. 2010. "Hybrid Genetic Algorithm and Linear Programming Method for Least-Cost Design of Water Distribution Systems." *Water Resources Management* 24 (1): 1–24.

Cunha, M. da Conceição, and J. Sousa. 1999. "Water Distribution Network Design Optimization: Simulated Annealing Approach." *Journal of Water Resources Planning and Management* 125 (4): 215–221.

Dandy, G. C., A. R. Simpson, and L. J. Murphy. 1996. "An Improved Genetic Algorithm for Pipe Network Optimization." *Water Resources Research* 32 (2): 449–458.

Eiger, G., U. Shamir, and A. Ben-Tal. 1994. "Optimal Design of Water Distribution Networks." *Water Resources Research* 30 (9): 2637–2646.

Geem, Z. W. 2006. "Optimal Cost Design of Water Distribution Networks Using Harmony Search." *Engineering Optimization* 38 (3): 259–277.

Gessler, J. 1985. "Pipe Network Optimization by Enumeration." *Computer Applications in Water Resources*, 572–581. New York: ASCE.

Goulter, I. C., B. M. Lussier, and D. R. Morgan. 1986. "Implications of Head Loss Path Choice in the Optimization of Water Distribution Networks." *Water Resources Research* 22 (5): 819–822.

Haghighi, A., H. M. V. Samani, and Z. M. V. Samani. 2011. "GA-ILP Method for Optimization of Water Distribution Networks." *Water Resources Management* 25 (7): 1791–1808.

Kessler, A., and U. Shamir. 1989. "Analysis of the Linear Programming Gradient Method for Optimal Design of Water Supply Networks." *Water Resources Research* 25 (7): 1469–1480.

Liong, S. Y., and M. Atiquzzaman. 2004. “Optimal Design of Water Distribution Network Using Shuffled Complex Evolution.” *Journal of The Institution of Engineers Singapore* 44 (1): 93–107.

Metropolis, N., A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. 1953. “Equation of State Calculations by Fast Computing Machines.” *Journal of Chemical Physics* 21 (6): 1087–1092.

Montalvo, I., J. Izquierdo, R. Pérez, and M. M. Tung. 2008. “Particle Swarm Optimization Applied to the Design of Water Supply Systems.” *Computers & Mathematics with Applications* 56 (3): 769–776.

Morgan, D. R., and I. C. Goulter. 1985. “Optimal Urban Water Distribution Design.” *Water Resources Research* 21 (5): 642–652.

Rossman, L. A. 2000. “Epanet 2: Users Manual, US Environmental Protection Agency.” EPA/600/R-00/057. Cincinnati, OH: National Risk Management Research Laboratory, Office of Research and Development. http://nepis.epa.gov/Adobe/PDF/P1007WWU.pdf.

Savic, D. A., and G. A. Walters. 1997. “An Improved Genetic Algorithm for Pipe Network Optimization.” *Water Resources Research* 123 (2): 67–77.

Simpson, A. R., G. C. Dandy, and L. J. Murphy. 1994. “Genetic Algorithms Compared to Other Techniques for Pipe Optimization.” *Journal of Water Resources Planning and Management* 120 (4): 423–443.

Vasan, A., and S. P. Simonovic. 2010. “Optimization of Water Distribution Network Design Using Differential Evolution.” *Journal of Water Resources Planning and Management* 136 (2): 279–287.

Wood, D. J. 1980. *Computer Analysis of Flow in Pipe Networks Including Extended Period Simulations: User’s Manual*. Lexington, KY: Office of Continuing Education and Extension of the College of Engineering of the University of Kentucky.

Yang, C. H., X. L. Tang, X. J. Zhou, and W. H. Gui. 2013. “A Discrete State Transition Algorithm for Traveling Salesman Problem.” *Control Theory & Applications* 30 (8): 1040–1046.

Yates, D. F., A. B. Templeman, and T. B. Boffey. 1984. “The Computational Complexity of the Problem of Determining Least Capital Cost Designs for Water Supply Networks.” *Engineering Optimization* 7 (2): 142–155.

Zecchin, A. C., A. R. Simpson, H. R. Maier, M. Leonard, A. J. Roberts, and M. J. Berrisford. 2006. “Application of Two Ant Colony Optimization Algorithms to Water Distribution System Optimization.” *Mathematical and Computer Modelling* 44 (5): 451–468.

Zecchin, A. C., A. R. Simpson, H. R. Maier, and J. B. Nixon. 2005. “Parametric Study for an Ant Algorithm Applied to Water Distribution System Optimization.” *IEEE Transactions on Evolutionary Computation* 9 (2): 175–191.

Zhou, X. J., C. H. Yang, and W. H. Gui. 2011a. “Initial Version of State Transition Algorithm.” In *International Conference on Digital Manufacturing and Automation (ICDMA)*, 644–647. Piscataway, NJ: IEEE.

Zhou, X. J., C. H. Yang, and W. H. Gui. 2011b. “A New Transformation into State Transition Algorithm for Finding the Global Minimum.” In *International Conference on Intelligent Control and Information Processing (ICICIP)*, 674–678. Piscataway, NJ: IEEE.

Zhou, X. J., C. H. Yang, and W. H. Gui. 2012. “State Transition Algorithm.” *Journal of Industrial and Management Optimization* 8 (4): 1039–1056.

Zhou, X. J., C. H. Yang, and W. H. Gui. 2014. “Nonlinear System Identification and Control Using State Transition Algorithm.” *Applied Mathematics and Computation* 226: 169–179.