Chaos-directed genetic algorithms for water distribution network design: an enhanced search method

S. N. Poojitha\textsuperscript{1} · V. Jothiprakash\textsuperscript{1} · Bellie Sivakumar\textsuperscript{1}

Accepted: 26 February 2022 / Published online: 28 March 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

Abstract
The design of a water distribution network (WDN) is an ever-challenging problem. The formulation and application of optimization techniques for WDN design have been an important area of research. Recently, the introduction of chaos theory-based evolutionary algorithms (EAs), in addition to traditional random-based ones, has elevated the scope for further improving the performance of EAs. The present study proposes a chaos-directed genetic algorithm (CDGA) by incorporating chaos ergodicity in GA mechanics for the optimal design of WDNs by introducing two novel frameworks, namely non-sequential approach and sequential approach. In improving the search efficacy of GA, the influence of chaotic systems with high-dimensionality maps is also explored when compared to the low-dimensionality maps. Considering four widely studied WDN benchmark problems ranging from 8 to 454 dimensions, the performance of the proposed GA and CDGA models is evaluated. The results show that the CDGA models outperform GA with better search efficacy, requiring fewer function evaluations to locate the optimal solution. In addition, the CDGA models are found to outperform other optimization techniques reported previously to handle these benchmark problems. Based on the results obtained, the study suggests the use of the chaotic system with other bio-inspired techniques to further improve their searchability and, thus, their computational efficiency.

Keywords Water distribution network · Optimization · Genetic algorithm · Chaos-based evolutionary algorithms · Ergodicity · Non-sequential and sequential approaches

List of symbols

| Symbol | Description |
|--------|-------------|
| $C_j$  | Cost of $j$th pipe |
| $CDS$  | Commercially available diameter set |
| $CP$   | Crossover point |
| $CS$   | Chaotic sequence |
| $D$    | Genes or decision variable |
| $d_j$  | Diameter of $j$th pipe |
| $DL_S$ | Chaotic sequence data length allocated for a single generation using SA |
| $F_f$  | Fitness function |
| $G_s$  | Generation size |
| $H_i$  | Head available at the $i$th node |
| $H_{min}$ | Minimum head requirement at the $i$th node |
| $L$    | Lower bound of the decision variable |
| $l_j$  | Length of $j$th pipe |
| $M$    | Number of genes that undergo mutation operation |
| $MC$   | Mutation chromosome |
| $MV$   | Mutation variable |
| $N_{CD}$ | Number of commercially available pipe diameter options |
| $N_D$  | Number of decision variables |
| $N_n$  | Number of nodes |
| $PC$   | Parent chromosome |
| $P_c$  | Crossover probability |
| $P_m$  | Mutation probability |
| $PM$   | Penalty multiplier |
| $P_r$  | Number of chromosomes retained during truncation operator |
| $P_s$  | Population size |
| $r$    | Uniformly distributed random number |
| $SV$   | Swapping variable |
| $U$    | Upper bound of the decision variable |

\textsuperscript{1} Department of Civil Engineering, Indian Institute of Technology Bombay, Powai, Mumbai, Maharashtra 400076, India

\textsuperscript{b}@iitb.ac.in

Stochastic Environmental Research and Risk Assessment (2022) 36:3377–3393
https://doi.org/10.1007/s00477-022-02200-7
1 Introduction

Water distribution networks (WDNs) play a key role in socio-economic development. They are generally large complex systems made up of different hydraulic components interacting with each other in nonlinear ways. The design of WDNs aimed at accomplishing hydraulic, functional, and economic aspects is often a highly challenging problem. Its major design objective is to choose a hydraulically feasible, economical set of network pipe diameters. The network pipes, being the design variables and each having various commercially available diameter options, result in different design possibilities that become exhaustive with the increase in the size of the network (network pipes). In addition, the design problem has to deal with the domains with discrete feasible solution sets in the exhaustive search space, which is intractable. As a result, the WDN design problem is familiar as a combinatorial optimization problem. Further, it belongs to a class of nondeterministic polynomial-time hard problems (Reca and Martinez 2006).

During the past several decades, an enormous amount of research has been carried out for WDN design. Many efficient optimization techniques, ranging from the scale of classical optimization techniques to heuristic and metaheuristic search techniques, have been developed (e.g., Alperovits and Shamir 1977; Gessler 1985; Lansey and Mays 1989; Dandy et al. 1996; Ezzeldin et al. 2014; Zheng et al. 2014; Fallah et al. 2019). In recent decades, applications of metaheuristic optimization techniques, such as genetic algorithm (GA), differential evolution (DE), particle swarm optimization (PSO), harmony search (HS), cuckoo search (CS), and crow search algorithm (CSA), for WDN optimal design have gained significant momentum due mainly to their versatility and ease in application. Among the many metaheuristic optimization techniques, GA is one of the most widely used techniques (e.g., Simpson et al. 1994; Gupta et al. 1999; Vairavamoorthy and Ali 2000; Zyl et al. 2004; Kadu et al. 2008; Haghhighi et al. 2011; Mora-Melia et al. 2013; Johns et al. 2014).

Genetic Algorithm (GA) is a bio-inspired metaheuristic evolutionary algorithm (EA). It is a random yet structured algorithm, whose evolutionary process in the traditional sense is governed by the randomly generated initial population and its operators that work on the probabilistic rules and a random phenomenon (Goldberg and Kuo 1989). Although GA generally works well, it has certain drawbacks. For instance, (i) the occurrence of an optimal solution is skeptical at every trial considered, as the search procedure is a mere dependence of random walks in a random direction (Savic and Walters 1997; Keedwell and Khu 2005); and (ii) with an increase in the number of decision variables, GA becomes computationally expensive and has a high chance of stagnating at local optima (Ingu and Takagi 1999; Reca and Martinez 2006; Cheng et al. 2008; Zheng et al. 2011; Ali et al. 2018). Due to these drawbacks, in recent decades, many attempts have been made to improve GA’s searchability and computational efficiency.

Murphy and Simpson (1992) were the first to apply GA for the optimal design of WDNs, followed by Simpson et al. (1994). These studies used binary encoding to represent the decision variables (network pipe diameters). Later, Dandy et al. (1996) and Savic and Walters (1997) improved the binary GA using gray coding. Vairavamoorthy and Ali (2000) formulated a real-coded GA to alleviate the redundant states of variables associated with binary coding. The other GA models developed for designing WDNs include: GA with enhanced operators (Montesinos et al. 1999; Reca and Martinez 2006; Johns et al. 2014), messy GA with building block filtering process (Wu and Simpson 2001), hybrid GA models (Zyl et al. 2004; Cisty 2010; Haghhighi et al. 2011), GA fed with a healthier initial population (Vairavamoorthy and Ali 2005; Keedwell and Khu 2005), modified-GA with a critical path scheme (Kadu et al. 2008), pseudo GA with integer coding scheme (Mora-Melia et al. 2013), GA with a novel decision variable representation (Cimorelli et al. 2020), and many others.

While various such approaches have and continue to be formulated, attempts to combine the concepts of chaos theory (see Lorenz (1963) and May (1976) for early studies on the theory) with the metaheuristic optimization techniques emerged in the meantime in diverse fields of engineering. Such attempts elevated the scope for improving the performance of evolutionary algorithms (EAs); see, for example, Mozaffari et al. (2018) for a detailed investigation on the effects of using different chaotic maps on the diversification or intensification features of EAs. In the field of water resources engineering, Yuan et al. (2002), Cheng et al. (2008), Arunkumar and Jothiprakash (2013) have been some of the studies on chaos-based GA models.

Despite the chaos theory-based attempts and outcomes to enhance the effectiveness and efficiency of EAs, the use of a chaotic approach has not been, to the best of our knowledge, explored for the optimal design of WDNs. However, it is important to note that the chaotic maps evolve through every possible state within a specific range depending on the system characteristics (Ammaruekarat and Meesad 2011). Their dynamic evolution through all the possible states guides the search mechanism to explore the different potential regions of search space, keeping the chance of reaching the global point of interest high. This
advantage of chaotic maps is beneficial in optimization and, thus, provides a great opportunity to completely leverage it in WDN optimization studies through its introduction in GA. In this sense, the present study offers a novel approach of replacing GA’s entire random phenomenon with chaos ergodicity to enhance its searchability and computational efficiency.

In most of the chaos theory-based water resources-related optimization models discussed above, the Logistic series was used. Further, in the studies where chaos maps were used in more than two operators, Logistic sequences with different initial conditions were used. As chaotic maps are highly sensitive to initial conditions, the chaos ergodicity may not be employed for the entire qualitative search when the operators are driven with different chaotic dynamics (i.e., with different initial conditions). In this regard, to retain the chaos ergodicity throughout the optimization process and analyze its effects on the convergence properties of GA, the present study uses the chaotic map with the same initial characteristics throughout the optimization process. Essentially, the present study differs from the previous ones in how the chaos ergodicity is simulated in GA random search mechanisms. Thus, the Chaos-directed Genetic Algorithm (CDGA) models are formulated, with the introduction of two novel frameworks for chaotic sequence (i.e., time series) allocation. They are (i) the non-sequential approach (NSA), where the chaotic sequence is pre-allocated at every random phase of GA, for the entire generation size $G_i$; and (ii) the sequential approach (SA), where, over generations, the chaotic sequence is assigned successively at every random phase of GA.

Further, it is important to note that a chaotic system with high dimensionality may allow a more diversified search when compared to a chaotic system with low dimensionality, especially in terms of the exploration in the search space (i.e., greater search effectiveness). Whether this is indeed true has not been examined thus far, since most of the studies that have attempted a chaotic search for the design of water resources-related optimization problems have essentially used the Logistic time series, which has very low dimensionality (correlation dimension is about 0.50) (Grassberger and Procaccia 1983). To address this issue, the present study explores slightly more complex chaotic systems with higher dimensionality. To this end, two other chaotic systems, namely the Henon map (Henon 1976) (with a dimension of 1.22) and the Lorenz map (Lorenz 1963) (with a dimension of 2.06), are considered.

Thus, overall, considering the increasing momentum in exploring the metaheuristic algorithms with an innate stochastic component for WDN design (Suribabu 2010; Babu and Vijayalakshmi 2013; Ezzeldin et al. 2014; Mora-Melia et al. 2015; Sheikholeslami and Talatahari 2016; Moosavian and Lence 2019; Jain and Khare 2021), the present study offers a novel approach for improving their computational efficiency by introducing NSA and SA for replacing the algorithm’s random phenomenon with chaotic force. As mentioned earlier, the widely studied GA is considered in this regard, and CDGA models are formulated. Their effectiveness is investigated by considering three well-studied benchmark problems: the Two-loop network (TLN) (Alperovits and Shamir 1977), the Bakryan network (BRN) (Lee and Lee 2001), and the Goyang network (GYN) (Kim et al. 1994). Further, the scalability of the best model is validated using the large-sized WDN, the Balerma irrigation network (BIN) (Reca and Martinez 2006).

The rest of the paper is organized as follows. Section 2 discusses GA, CDGA models, optimization framework, sensitivity analysis results, and the benchmark problems. With the computational results compiled in Sects. 3, 4 highlights the key points observed and the scalability of the best CDGA model. Finally, Sect. 5 presents the major conclusions drawn from the study.

## 2 Methodology

### 2.1 Genetic algorithm (GA) model with random search phenomenon

Genetic Algorithm simulates the Darwinian principle of genetic inheritance (Goldberg and Kuo 1989). In GA, the evolution of the population takes place over generations by refining them through selection (reproduction), crossover, and mutation operators. Though there are different GA operator mechanisms, the present study considers the real-coded GA in its traditional form with truncation selection mechanism, single-point crossover, and bitwise mutation operators. The detailed mechanism of the real-coded GA in its traditional form is presented in Supplementary Material S1. Here, the main motive is to simulate the chaos ergodicity in the traditional form of GA and compare its performance with the other improved versions of GA and metaheuristic optimization techniques reported in the literature to handle the WDN optimization problem.

In the present case of WDN design, the main objective is to minimize the design cost subjected to the constraint of the minimum head criterion. There are different constraint handling methods, with each having its advantages and disadvantages. In evolutionary computation and other optimization frameworks, penalty functions that penalize the infeasible solutions are widely used. The present study considers the static penalty function that is easy and simple to implement (Michalewicz and Schoenauer 1996). Thus, the fitness function, $F_j$, is defined as the sum of the pipe cost, $C$, and the static penalty function, as follows:
\[ F_f = \sum_{j=1}^{N_N} C_j(d_j, l_j) + \sum_{i=1}^{N} PM \times (\max(0, H_{\text{min}} - H_i))^2 \]  

where \( d \) and \( l \) are the diameter and length of pipes, respectively, \( N_D \) is the number of decision variables (number of network pipes to design optimally), \( N_n \) is the number of nodes, \( PM \) is the penalty multiplier (chosen through sensitivity analysis, discussed below), and \( H_{\text{min}} \) and \( H_i \) are the minimum head requirement and the head available at the \( i \)th node, respectively.

### 2.2 Chaos-directed genetic algorithm (CDGA) with chaotic search phenomenon

Chaotic systems are nonlinear deterministic systems that produce random-looking outputs. They are predictable in the short term due to their inherent determinism but not in the long term due to their sensitivity to initial conditions (Sivakumar 2009, 2017). In the present study, using chaotic maps, the chaos ergodicity is simulated in GA’s search mechanism to enhance its search capability and computational efficiency. The chaos-directed genetic algorithm (CDGA) models are formulated, replacing every random mechanism in GA with the chaotic force by assigning the chaotic time series. Two novel frameworks are introduced to assign the chaotic sequence to GA operators, namely (i) non-sequential approach (NSA) and (ii) sequential approach (SA). Figure 1 presents the detailed mechanisms of these two approaches. In NSA, the assignment of the chaotic sequence is over each operator involved with a random phenomenon for the entire generation size \( G_s \), i.e., the chaotic sequence is pre-allocated for the entire \( G_s \) (non-sequential over generations). In SA, the assignment of the chaotic sequence follows every operator with a random phenomenon in every iteration, i.e., instead of pre-allocation for entire generations, the sequential chaotic sequence assignment iterates over \( G_s \) (sequential over generations). The detailed procedure of both approaches concerning the GA operators is discussed as follows.

(i) For both of these approaches, using a chaotic sequence, \( CS \), the initial population is generated as follows:

\[
D_{ij} = L_{ij} + (U_{ij} - L_{ij}) \times CS_k \\
i = 1, 2, \ldots, P_s; j = 1, 2, \ldots, N_D; k = 1, 2, \ldots, P_s \times N_D
\]

where \( D_{ij} \) is the \( j \)th decision variable of the \( i \)th chromosome, \( U \) and \( L \) are the upper and lower bounds (which are equal to the maximum and minimum count of the number of commercially available pipe diameter options, \( N(CD) \), \( P_s \) is the population size, and \( K \) is the length of the chaotic sequence required to generate the initial population and is equal to \( P_s \times N_D \). As the initial population is generated once over the entire \( G_s \), \( K \) is the same for both approaches.

(ii) Following the chaotic generation of the initial population, the best chromosomes are reproduced using the truncation selection mechanism. Then, to generate the offsprings, the crossover operation is executed for which the parent chromosomes, \( PC \), \( (PC1, PC2) \) are selected using chaos-based equations:

\[
PC1 = CS_{1} \times P_r; \quad PC2 = CS_{2} \times P_r; \\
i = (K + 1), \ldots, L_s \text{ or } L_s(G_s)
\]

where \( P_r \) is the number of chromosomes reproduced using the truncation scheme. Consequently, the length of the chaotic sequence allocated for generating \( PCs \) using NSA is \( L_{ns} = K + (P_s - P_r) \times G_s \) and using SA, which is the function of \( G_s \) is \( L_{ns}(G_s) = K + (P_s - P_r) + f(G_s) \). For \( G_s = 1, f(1) = 0 \), and for the last generation, \( G_s,l, f(G_s,l) = DL_s \times (G_s,l - 1) \), where \( DL_s \) is the data length of the chaotic sequence allocated for a single generation using SA.

(iii) To perform crossover operator, the crossover point, \( CP \), is chosen as follows:

\[
CP_i = CS_m \times (N_D - 1); m = (L_{ms} + 1) or (L_m(G_s) + 1), \ldots, M_{ms} \text{ or } M_m(G_s)
\]

where \( M_{ms} = L_{ms} + (0.5 \times (P_s - P_r)) \times G_s \) is the length of chaotic sequence required using NSA, and \( M_m(G_s) = L_m(G_s) + 0.5 \times (P_s - P_r) \) using SA.
2.3 Optimization and simulation model

For the optimal design of WDNs, information on the layout of the network, pipe material, commercially available diameter set (CDS), demand, and minimum pressure head requirement at the demand nodes are generally established a priori. With this information, the main objective of the WDN design is to choose an economical set of diameters for the pipes in the network that ultimately meets the nodal demands at demand nodes. The fitness function, $F_p$, is presented in Eq. (1). Here, for every possible solution proposed by the optimization algorithm, the pressure head at every demand node is checked to meet the minimum head criterion. Constraint violation, if any, is penalized using the static penalty function, as presented in Eq. (1). Further, Eq. (2) ensures the selection of network pipe diameters from the CDS. The other constraints, such as balancing the energy and mass balance equations, are taken care of by the simulation model. The present study considers the EPANET software that works on the global gradient method, designed by Rossman (2000), as the simulation model to simulate the actual hydraulic conditions of the WDN. Thus, the design algorithm for WDN is the combination of an optimization algorithm and a simulation model. The codes for both the GA and the CDGA models are written in MATLAB and are linked to the EPANET software using the MATLAB-EPANET interface.

To test the computational efficiency of the GA and CDGA models, three widely studied benchmark networks, namely Two-loop network (TLN), Bakryan network (BRN), and Goyang network (GYN), are considered. Further, to validate the scalability of the best of the proposed models, a large-sized WDN, namely the Balerma irrigation network (BIN), is employed.

### 2.4 Parameter sensitivity analysis

The parameters of GA play a vital role in the convergence properties of the algorithm. An optimally chosen parameter set along with well-defined search guidance drives the algorithm towards global convergence. In the present study, a sensitivity analysis is performed to select the optimal value for the parameters. While performing the sensitivity analysis, the scale over which the parameters vary is chosen carefully. A too small or too large value of any parameter may not result in an efficient or effective search. For instance, for the population size, $P_s$, a too low value saturates the population soon without proper exploration of the search space; a too large value makes the population redundant, making the search highly random. To overcome these problems, $P_s$ is varied in a moderate range of 50 to 500 for the small to medium-sized problems (TLN, BRN, and GYN) and 500 to 2500 for the large-sized network (BIN).

The crossover probability, $P_c$, explicitly reflects the percentage of new chromosomes (offsprings) generated. A too-small $P_c$ means not much new information generation to explore the new areas in the search space; a too-high $P_c$ loses useful information with the best fit chromosomes (an effective guiding medium). Thus, a moderate $P_c$, varying from 0.5 to 0.7, is considered. Once the $P_c$ is fixed, the truncation probability is considered as $(1 - P_c)$. Although the randomized crossover operator pays off the search in unexplored areas, there is always a chance of overindulgence and missing important information (Goldberg and
Kuo 1989). Therefore, to induce diversity in the population, a small percentage of the probability of mutation \( P_m \) (compared to \( P_c \)) in the range of 0.01 to 0.1, as suggested by Savic and Walters (1997) and Kadu et al. (2008), is employed.

Further, with the penalty multiplier, \( PM \), a too high value makes the search stringent, resulting in loss of useful information with the infeasible solutions. On the contrary, a too low value of \( PM \) may concentrate the search around the infeasible regions. As a result, the study evaluates for a moderate value of \( PM \) from the sensitivity analysis. Once the optimal values for all these parameters are ascertained from the sensitivity analysis, the generation size, \( G_s \), is selected based on the convergence requirement of the algorithm. It is important to note that the above parameters are problem-specific and model-specific. Therefore, the sensitivity analysis is carried out separately for each model, with each benchmark problem considered in the present study.

### 2.5 Benchmark Problems

The TLN is arguably the most widely used small-sized gravity-fed network. It is an 8-dimensional problem with two loops, first introduced by Alperovits and Shamir (1977). The BRN is a small-sized rehabilitation network that is 9-dimensional (Lee and Lee 2001). The third network, GYN, is a pumped, medium-sized, 30-dimensional problem with nine loops (Kim et al. 1994). Details of these three benchmark problems, in terms of the number of design parameters, \( N_D \), number of commercially available diameters, \( N_CD \), solution space available (i.e., number of possible combination of pipe sets from the commercially available data set), and the minimum residual pressures, \( H_{min} \), required at the demand nodes, are presented in Table 1. The layout and other hydraulic details of these networks are available in Geem (2006). The fourth network is BIN, which is a large-sized network with 454 pipes and 443 demand nodes. Each pipe of BIN has ten commercially available diameter options, resulting in a solution search space of \( 1.0 \times 10^{154} \). A pressure head of 20 m at demand nodes is desirable for BIN to deliver the designed demands. The complete details of BIN are available in Reca and Martinez (2006).

### 3 Computational results

#### 3.1 Results for Two-loop network (TLN)

##### 3.1.1 GA model

At first, the sensitivity analysis for TLN is performed using the GA model. The optimal parameter set obtained from the sensitivity analysis is presented in Table 1. With this optimal parameter set, the GA model is successful in converging to an optimal feasible design cost of $419,000 by exploring only a small percentage of the solution space, i.e., \( 2.033 \times 10^{-4} \), with minimum function evaluations (MFEs) of 3,000, requiring an average computation time of 21.556 s. Besides the best results of GA, Table 1 also includes the results of the other models considered in the present study, comparing their computational efficiency.

##### 3.1.2 CDGA-I model

Following the GA model, the CDGA-I model (i.e., with the Logistic map) is considered for designing the TLN. Initially, to fix the fertility rate, \( a \), of the Logistic equation, the CDGA-I is run, varying \( a \) from 3.8 to 4.0 with a step size of 0.02 with nine different initial values \( X_0 \), varying from 0.1 to 0.9. With the NSA and SA, it is found that \( a = (3.94, 3.96, 3.98, 4.0) \) are efficient in yielding an optimal cost of $419,000. As the initial value \( X_0 \) of the Logistic equation can take any value in the range (0,1), and the fact that the chaotic system is sensitive to even a fractional change in the initial value, \( X_0 \) is taken randomly for the main analysis. Thus, the Logistic equation with \( a = (3.94, 3.96, 3.98, 4.0) \) and with a randomly chosen initial value is used to replace the random phases in GA in the two approaches, i.e., CDGA-I using NSA and CDGA-I using SA.

Keeping \( P_s \), \( P_c \), and \( P_m \) the same as that of the GA model, \( G_s \) is varied from 100 to 200, with a step size of 10. At \( a = 3.94 \), the Logistic sequence is very efficient in obtaining the optimal cost for every combination of \( G_s \) considered for both the approaches, as shown in Fig. 2a. With NSA, at \( G_s = 150, 200 \), the MFEs required are 2900, which is less than that of the random approach of GA (3,000). At \( a = 3.96 \), the optimal cost is obtained at every combination, except at \( G_s = \{140, 150\} \) of SA, as shown in Fig. 2b. The MFEs required to locate the optimal cost are less than GA model at \( G_s = 120 \), i.e., 2,900, using SA. At \( a = 3.98 \), the CDGA-I model excels in attaining an optimal cost with a much lesser number of MFEs, i.e., 1,600, at \( G_s = 130 \) using SA, as shown in Fig. 2c. At \( a = 4.0 \), the CDGA-I model converges to the suboptimum using NSA and SA at \( G_s = 120 \) and \( G_s = 100 \), respectively. At \( G_s = 120 \), using SA, compared to the GA model, fewer MFEs are required, i.e., 2,400 (see Fig. 2d).

A summary of the best results of the sensitivity analysis using CDGA-I models is presented in Table 1. Overall, the CDGA-I model is successful and efficient in exploring a small percentage of the solution space, i.e., \( 1.084 \times 10^{-4} \), to locate the optimal solution with the MFEs of 1,600, requiring significantly less computation time of an average 6.917 s when compared to the GA model.
Table 1: Overall review of benchmark problems considered in the study and their computational results using GA, CDGA models

| Parameter                  | TLN (GA) | CDGA-I | CDGA-II | CDGA-III | BRN (GA) | CDGA-I | CDGA-II | CDGA-III |
|----------------------------|----------|--------|---------|----------|----------|--------|---------|----------|
| Number of pipes            | 8        | 14     | 14      | 14       | 9        | 11     | 11      | 11       |
| Commercially available options | 89     | 14     | 11      | 11       | 9        | 15     | 15      | 15       |
| Solution space             | $14^8 \times (1.475 \times 10^6)$ | $14^8 \times (1.475 \times 10^6)$ | $11^9 \times (2.357 \times 10^9)$ | \(2.357 \times 10^9\) |
| Minimum residual pressure (m) | 30      | 30     | 30      | 30       | 30       | 30     | 30      | 30       |
| Population size \((P_s)\) | 102      | 102    | 53      | 53       | 102      | 102    | 102     | 102      |
| Probability of crossover \((P_c)\) | 0.5   | 0.5    | 0.7     | 0.7      | 0.5      | 0.7    | 0.7     | 0.7      |
| Probability of mutation \((P_m)\) | 0.02 | 0.02   | 0.03    | 0.08     | 0.035    | 0.06   | 0.06    | 0.09     |
| Generation size \((G_s)\) | 400      | 130    | 400     | 400      | 500      | 100    | 100     | 100      |
| Penalty multiplier \((PM)\) | $7.5 \times 10^4$ | $7.5 \times 10^4$ | $5 \times 10^5$ | $5 \times 10^5$ |
| Optimal feasible cost      | US$ 419,000 | US$ 419,000 | US$ 419,000 | US$ 419,000 |
| Minimum functional evaluations | 3000  | 1600   | 1664    | 32,700   | 2200     | 728    | 1040    | 6600     |
| Solution space explored (%) | $2.033 \times 10^{-4}$ | $1.084 \times 10^{-4}$ | $1.128 \times 10^{-4}$ | $2.216 \times 10^{-3}$ | $3.933 \times 10^{-5}$ | $4.411 \times 10^{-5}$ | $2.799 \times 10^{-5}$ |
| Average computation time (s) | 21.556  | 6.917  | 8.007   | 19.871   | 74.988   | 8.003  | 8.035   | 20.690   |

| Parameter                  | GYN (GA) | CDGA-I | CDGA-II | CDGA-III |
|----------------------------|----------|--------|---------|----------|
| Number of pipes            | 30       | 30     | 30      | 30       |
| Commercially available options | 8       | 8      | 8       | 8       |
| Solution space             | $8^3 (1.237 \times 10^{27})$ | $8^3 (1.237 \times 10^{27})$ | $8^3 (1.237 \times 10^{27})$ | $8^3 (1.237 \times 10^{27})$ |
| Minimum residual pressure (m) | 15      | 15     | 15      | 15       |
| Population size \((P_s)\) | 53       | 53     | 53      | 53       | 102     | 102    | 102     | 102      |
| Probability of crossover \((P_c)\) | 0.5   | 0.5    | 0.7     | 0.7      | 0.5      | 0.5    | 0.5     | 0.5      |
| Probability of mutation \((P_m)\) | 0.02 | 0.05   | 0.035   | 0.08     | 0.035    | 0.035  | 0.035   | 0.035    |
| Generation size \((G_s)\) | 500      | 500    | 500     | 500      | 500      | 500    | 500     | 500      |
| Penalty multiplier \((PM)\) | $3 \times 10^7$ | $3 \times 10^7$ | $3 \times 10^7$ | $3 \times 10^7$ |
| Optimal feasible cost      | 177,009,557 KRW | 177,009,557 KRW | 177,009,557 KRW | 177,009,557 KRW |
| Minimum functional evaluations | 10,088 | 7,280  | 22,256  | 48,700   | 177,010,359 KRW | 177,010,359 KRW | 177,010,359 KRW | 177,010,359 KRW |
| Solution space explored (%) | $8.149 \times 10^{-22}$ | $5.881 \times 10^{-22}$ | $1.798 \times 10^{-24}$ | $3.937 \times 10^{-21}$ |
| Average computation time (s) | 27.352  | 27.002 | 25.985  | 111.278  |

© Springer
3.1.3 CDGA-II model

To check the impact of two-dimensional chaotic Henon map dynamics, the CDGA-II model is formulated. As the Henon map is two-dimensional \((x, y)\), the sequences of both dimensions are used. For a fixed \(G_s = 400\), and with \(P_s = \{53, 102\}\) and \(P_c = \{0.5, 0.7\}\), the CDGA-II model is efficient in obtaining the optimal solution at almost every value of \(P_m\) varying from 0.02 to 0.09 using both the NSA and SA. Figure 2e–l shows the results with the \(x\) and \(y\) sequences, respectively. Using \(x\) sequence, at \(P_s = 53\) with a combination of \(P_c = 0.5, 0.7\) (see Fig. 2e, f), the CDGA-II with NSA is successful in converging to the optimal solution with a lesser number of MFEs, 2340 at \(P_m = \{0.02, 0.03\}\), and 1,820, at \(P_m = 0.04\), respectively, which is less than that of the GA model (3,000). When \(P_s = 102\) is considered, at both \(P_c\) values (0.5, 0.7), for every combination of \(P_m\), the optimal solution is located at the cost of an additional number of MFEs when compared to that of the GA model (see Fig. 2g and h).

With the \(y\) sequence, for almost all the combinations, the optimal solution of \$419,000 is obtained (as shown in Fig. 2i–l). For the optimal convergence, with the combination of \(P_c = 0.5\), \(P_m = 0.02\) using SA (Fig. 2i), and \(P_c = 0.7\) and \(P_m = 0.03\) using NSA (Fig. 2i), a lesser number of MFEs, i.e., 1,924 and 1,664, respectively, is required when compared to the GA model (3,000). With the combination of \(P_c = 102\), at \(P_c = 0.7\) and \(P_m = 0.04\), using NSA, the optimal solution is located with fewer MFEs, i.e., 1,768 (see Fig. 2l). For every other combination of \(P_c\) and \(P_m\), the MFEs required are more when compared to the GA model (see Fig. 2k, l). Table 2 compiles the best results for the two approaches using the \(x\) and \(y\) sequences with the CDGA-II model. The main advantage of the CDGA-II model is that once the algorithm’s optimal parameter set is fixed, it excels in obtaining the optimal solution at every trial run with a 100% success rate.

The best results of the CDGA-II model in comparison with the other models are presented in Table 1. Thus, overall, the CDGA-II model with the non-sequential \(y\) sequence outperforms the GA model in terms of a high success rate and with better computational efficiency in exploring a small percentage of the solution space, i.e., \(1.128 \times 10^{-4}\), with an MFEs of 1,660 and with less average computation time of 8.007 s (see Table 1).

3.1.4 CDGA-III model

Following the CDGA-II, the CDGA-III model with the three-dimensional Lorenz map is considered. All three sequences of the Lorenz map in combination with the NSA

Fig. 2 Computational results for TLN using CDGA-I and CDGA-II models
Table 1. Best computational results using CDGA-II model for TLN, BRN, and GYN

| Benchmark problem | Different approaches to CDGAII | Minimum functional evaluations | Time of computation (s) | Optimal parameter set |
|-------------------|--------------------------------|-------------------------------|-------------------------|-----------------------|
| TLN               | Non-sequential x               | 1820                          | 8.972                   | \( P_s = 53, G_s = 400, P_m = 0.04, P_c = 0.7 \) |
|                   | Non-sequential y               | 1664                          | 8.007                   | \( P_s = 53, G_s = 400, P_m = 0.03, P_c = 0.7 \) |
|                   | Sequential x                   | 3016                          | 11.228                  | \( P_s = 53, G_s = 400, P_m = 0.05, P_c = 0.7 \) |
|                   | Sequential y                   | 1924                          | 9.852                   | \( P_s = 53, G_s = 400, P_m = 0.02, P_c = 0.5 \) |
| BRN               | Non-sequential x               | 1352                          | 8.071, 8.440            | \( P_s = 53, G_s = 100, P_m = \{0.06,0.07\}, P_c = 0.7 \) |
|                   | Non-sequential y               | 1248                          | 8.704                   | \( P_s = 53, G_s = 100, P_m = 0.1, P_c = 0.7 \) |
|                   | Sequential x                   | 1040                          | 8.035                   | \( P_s = 53, G_s = 100, P_m = 0.06, P_c = 0.7 \) |
|                   | Sequential y                   | 1196                          | 8.075                   | \( P_s = 53, G_s = 100, P_m = 0.1, P_c = 0.7 \) |
| GYN               | Non-sequential x               | 28,000                        | 67.775                  | \( P_s = 204, G_s = 300, P_m = 0.04, P_c = 0.7 \) |
|                   | Non-sequential y               | 22,256                        | 25.985                  | \( P_s = 53, G_s = 500, P_m = 0.035, P_c = 0.7 \) |
|                   | Sequential x                   | 26,400                        | 67.278                  | \( P_s = 204, G_s = 300, P_m = 0.035, P_c = 0.7 \) |
|                   | Sequential y                   | 29,000                        | 69.326                  | \( P_s = 204, G_s = 00, P_m = 0.03, P_c = 0.7 \) |

and SA are considered for the analysis. From the sensitivity analysis, for the NSA of the CDGA-III model using the S sequence, the optimal parameter set is \( P_s = 102 \), \( G_s = 400 \), \( P_m = 0.08 \), and \( P_c = 0.5 \). With this optimal parameter set, the algorithm converges to an optimal feasible cost of $419,000 with the MFEs as high as 32,700 exploring the \( 2.216 \times 10^{-3} \) percentage of solution space with a minimum average computation time, 19.871 s. For the remaining sequences of the Lorenz map, with both approaches, the CDGA-III model converges to the local optimum. Although the CDGA-III model converges to the optimal solution with more function evaluations than the GA model, the average computation time is less. Further, similar to that of the CDGA-II model, if the optimal parameter set is fixed, the success rate of the CDGA-III model is 100%, unlike the GA model.

Thus, from the sensitivity analysis results for TLN, it is evident that the CDGA models are efficient in decreasing the computational effort over the GA model with the random search phenomenon. Especially, the CDGA models with the Logistic map and the Henon map (i.e., CDGA-I and CDGA-II) are efficient in reducing the search space, and the Henon and Lorenz maps excel with a 100% success rate in reaching the optimum feasible cost of $419,000.

3.2 Results for Bakryan network (BRN)

3.2.1 GA model

The optimal parameter set ascertained from the sensitivity analysis for BRN using the GA model is presented in Table 1. From the results, it is observed that the GA model is successful in converging to a design cost of $903,620 (similar to the earlier studies by Lee and Lee (2001) and Geem (2006)). For this optimal convergence, the GA model explores \( 9.330 \times 10^{-5} \) percentage of solution space requiring 2,200 MFEs, with an average computation time of 74.988 s, as presented in Table 1.

3.2.2 CDGA-I model

The computational results of the CDGA-I model at different fertility rate, \( a \), values, for both the NSA and SA are presented in Fig. 3a–d. From the sensitivity analysis results, unlike GA, the CDGA-I model requires comparatively smaller population size, \( P_s \), and generation size, \( G_s \), for optimal convergence. Considering \( P_s \), \( G_s \), and crossover probability \( P_c \) as 53, 100, and 0.06, mutation probability \( P_m \) is varied from 0.05 to 0.1, with a step size of 0.01. At every \( a \) value and \( P_m \) combinations, the CDGA-I model is successful in reaching the optimal solution with less number of minimum functional evaluations (MFEs) as compared to that of the GA model (2,200), except for \( a = 3.96 \) at \( P_m = \{0.05,0.06\} \) of the NSA and at \( P_m = 0.08 \) of the SA (see Fig. 3b; see Fig. 3a, c, and d for results at \( a = 3.94, 3.98, \) and 4.0, respectively). Table 1 includes the summary of the best results. At \( a = 4.0 \) with \( P_m = 0.6 \), the CDGA-I model is computationally very efficient (Fig. 3d). Unlike the GA model, the optimal cost of $ 903,620 is achieved by exploring a minimal percentage of solution space \( 3.087 \times 10^{-5} \) in an average computation time of
8.003 s requiring fewer MFEs (728) compared to the GA model.

### 3.2.3 CDGA-II model

The computational results of the CDGA-II model with both the sequences and approaches, with varied $P_m$ values for fixed optimal values of $P_s = 53$, $G_s = 100$, and $P_c = 0.7$, using $x$ and $y$ sequences, are shown in Fig. 3e and f. With the $x$ sequence, for both the approaches at $P_m = \{0.06, 0.1\}$, and with the NSA at $P_m = \{0.03, 0.04, 0.07\}$, the optimal solution is achieved with fewer functional evaluations when compared to the GA model (see Fig. 3e). With the $y$ sequence, at $P_m = \{0.06, 0.09, 0.1\}$ for both the approaches and at $P_m = 0.08$ of the SA, the optimal solution is obtained with fewer functional evaluations compared to the GA model (Fig. 3f). As presented in Table 2, the SA outperforms the NSA. Notably, the sequential $x$ approach outperforms with much less computational effort exploring significantly less percentage of the solution space, i.e., $4.411 \times 10^{-5}$, requiring fewer MFEs (1,040) with an average computation time of 8.035 s (see Table 1).

### 3.2.4 CDGA-III model

From the sensitivity analysis results using the CDGA-III model, only the $S$ sequence with the SA successfully converges to the optimal cost of $903,620$. The optimal parameter set ascertained is $P_s = 204$, $G_s = 100$, $P_m = 0.09$, and $P_c = 0.7$. For the optimal convergence, the MFEs required are 6,600 with an average computation time of 20.690 s. Similar to the results obtained for TLN, the CDGA-III model excels in converging to the optimal solution for BRN with less average computation time but with more MFEs. It explores $2.799 \times 10^{-5}$ percent of the solution space for converging to the optimal solution, which is less than the total available solution space.

Overall, similar to TLN, for BRN, the CDGA models are more efficient in decreasing the computational effort
than the GA model, especially the CDGA-I and CDGA-II models.

### 3.3 Results for Goyang network (GYN)

#### 3.3.1 GA model

Table 1 presents the optimal parameter set of GA evaluated for GYN through sensitivity analysis. An improved feasible design cost of 177,009,557 KRW is obtained, as reported by Poojitha et al. (2020). To locate the optimal design cost, the GA model explores $8.149 \times 10^{-22}$ percentage of the solution space with MFEs of 10,088 in an average computation time of 27.352 s.

#### 3.3.2 CDGA-I model

From the sensitivity analysis results, using the CDGA-I model, $G_s$ is the same as GA (50) with slightly higher $P_c$ and $P_m$ values (0.7, 0.05). Fixing these parameters, $P_s$ is varied from 50 to 300. The computational results using NSA and SA with different $a$ values are presented in Fig. 4a–d. For every value of $a$ considered in the study, at $P_s = 53$, with both the NSA and SA, the MFEs required to reach the optimal cost of 177,009,557 KRW falls below 10,088, i.e., less than the GA model (see Fig. 4a–d). For any $P_s$ value above 53, the MFEs required for locating the optimal solution is high, except for $P_s = 102$ at $a = 4.0$ using the SA (Fig. 4d); see Figs. 4a, b, and c for results obtained using $a = 3.94$, 3.96, and 3.98, respectively. As compiled in Table 1, at $a = 3.98$ with $P_s = 53$, the CDGA-I model is computationally efficient using the NSA. It explores only $5.881 \times 10^{-22}$ percentage of the solution space requiring far fewer MFEs (7280) with an average computation time of 27.002 s.

#### 3.3.3 CDGA-II model

Figure 4e–h presents the computational results of the CDGA-II model using NSA and SA with $x$ and $y$ sequences. They demonstrate the parameter combination where the CDGA-II model successfully converges to the optimal design cost of 177,009,557 KRW. From Fig. 4e–h, for every combination of parameters, the MFEs for optimal convergence are above 10,088, i.e., greater than the GA model. The best computational results concerning both the approaches and sequences are included in Table 2. Further, as presented in Table 1, though the CDGA-II model requires relatively more MFEs, with the best parameter set ($P_s = 53$, $G_s = 500$, $P_m = 0.035$, and $P_c = 0.7$), using the non-sequential $y$ approach, on average, the time needed for executing the algorithm is less (25.985 s) when compared to that of the GA and CDGA-I models by exploring $1.798 \times 10^{-21}$ percent of solution space. The only variation in the optimal parameter set from the GA model is $P_m$ and $P_c$, and from the CDGA-I model is $P_m$.

#### 3.3.4 CDGA-III model

From the sensitivity analysis results using both the approaches with the three sequences of the Lorenz map, the CDGA-III model fails to converge to an optimal cost of
177,009,557 KRW. When the \( P_s \) and \( G_s \) are further increased to check for the possibility of convergence to the optimal solution, the computational load increases, which negatively influences the computational efficiency. However, the CDGA-III model with the three sequences using NSA and SA successfully finds an optimal feasible design cost of 177,010,359 KRW, reported by Mora-Melia et al. (2015). Its variation from 177,009,557 KRW is just 0.00045 percent. Table 1 presents the best computational results of GYN using the CDGA-III model along with its optimal model parameters.

Thus, similar to the computational results for TLN and BRN, the CDGA-I model excels for GYN as well with enhanced computational efficiency over GA. Further, the CDGA-II and CDGA-III models excel over the GA and CDGA-I models regarding the average computation time (low) and success rate (high) in achieving the optimal cost at every possible trial executed.

3.4 Performance comparison of GA and CDGA models

The results on the application of the GA and CDGA models on the three benchmark problems (TLN, BRN, and GYN), whose dimensions vary from 8 to 30, suggest that introducing the chaos ergodicity in the GA search phenomenon acts as a carrier wave in decreasing the search space, inducing faster convergence to the optimal solutions. Figure 5 presents the convergence plots (left) (i.e., the evolution to the minimum design cost over the generations) and the bar charts (right) (comparing the MFEs) for every model considered in the present study. These plots correspond to the optimal parameter sets finalized through the sensitivity analysis as presented in Table 1.

For TLN, the GA model converged to an optimal cost of $419,000 at the 30th generation. Although the minimum cost obtained in the first generation with the CDGA-I model was $1,210,000, it soon converged to $419,000 at
the 16th generation, followed by the CDGA-II model and the CDGA-III model at the 32nd and the 327th generations, respectively (see Fig. 5a). The convergence plot for BRN is shown in Fig. 5c. The CDGA-I model converged faster to the optimal design cost of $903,620 at 15th generation, followed by the CDGA-II model at 20th, the GA model at 22nd, and the CDGA-III model at 33rd generations. For GYN, the CDGA-III model failed to converge to the optimal solution of 177,009,557 KRW. However, the CDGA-I model converged faster, followed by the GA and CDGA-II models at $G_s$ of 140, 194, and 428, respectively (see Fig. 5e). For all the three benchmark problems, Fig. 5b, d, f presents the computational efficiency of the GA and CDGA models in terms of MFEs. From Figs. 5b, d, f, of all the models, the CDGA-I model excels with faster convergence for the respective benchmark problems.

4 Discussion

4.1 Computational analysis of CDGA models

An interesting feature observed from the computational analysis using the CDGA-I model with the Logistic map is that for any value of $a$, at $X_0 = \{0.4, 0.6\}$, convergence remains the same, including the number of function evaluations. Further, at $a = 4.0$, the initial values $X_0 = \{0.1, 0.9\}$ have the same convergence properties. For all the above three benchmark problems (TLN, BRN, and GYN), the CDGA-I model with $a = 3.94$, 3.98, and 4.0 are efficient in converging to an optimal solution with fewer function evaluations. Among $a = 3.94$, 3.98, and 4.0, the Logistic sequence with $a = 3.98$ excels in reducing the computational effort by converging faster to the optimal solution. With $a = 3.94$, it is efficient in attaining an optimal cost at almost every parameter combination of the algorithm considered in the study. The problem faced with the Logistic sequence with $a = 4.0$ is that the sequence soon converges to a similar value at the initial values of 0.25, 0.5, and 0.75. In addition, for larger iteration sizes, for any other initial values except 0.25, 0.5, and 0.75, the Logistic sequence with $a = 4.0$ converges to zero.

Of the two approaches introduced in the present study for simulating the chaotic force in GA evolutionary process, i.e., NSA and SA, in the majority of the cases (CDGA-I model for BRN and GYN; CDGA-II model for TLN and GYN; and CDGA-III model for TLN), the NSA outperforms the SA in terms of computational efficiency. When the CDGA-II model formulated using the Henon map is considered, for TLN and GYN, the non-sequential $y$ excels in locating the optimal solutions with less computational effort. However, for BRN, the sequential $x$ sequence outperforms the other approaches (see Table 2). Further, while using the CDGA-III model with the Lorenz map, only the $S$ sequence efficiently converges to the optimal solution. Essentially, the main drawback observed with the CDGA-III model is the normalization technique used. With the Lorenz sequence, the difference between the subsequent numbers is very small. As a result, while normalizing the data in the range (0, 1), the magnitude of subsequent values observed is almost the same for a long stretch. Due to these repetitive values in a stretch, the CDGA-III model with both the approaches does not aid in the diversification or exploration of the search space (selection of an appropriate normalization technique needs to be explored further and will be done in a future study).

Thus, the CDGA-III model requires more computational effort to converge to an optimal solution, as is the case for TLN and BRN. In addition, the lack of diversification in the search also leads to scenarios of sub-optimum convergence, as is the case with GYN. The same phenomenon is encountered when another higher-dimensional chaotic map, the Mackey–Glass equation is used (results not presented here). Further, one more interesting feature observed with the CDGA-III model due to the repetitive values is that the algorithm requires more population for the effective search. When large $P_s$ values are allowed, the algorithm converges faster at the early generations and, thus, requires less computational time, as reported in earlier sections.

4.2 Computational results of GA and CDGA models

As discussed in the previous section, overall, for the three benchmark problems TLN, BRN, and GYN, of the CDGA models with different chaotic maps, the CDGA-I model is the best. Further, of the two approaches introduced, the NSA is superior to SA. Thus, the scalability of the best model, the CDGA-I with the NSA, is evaluated considering a large 454-dimensional problem, the Balerma irrigation network (BIN).

4.2.1 Scalability of CDGA-I model

Initially, the sensitivity analysis is performed, and the optimal parameter set ascertained for BIN is $P_s = 700$, $G_s = 3000$, $P_m = 0.01, PM = 1,000,000, elitism percentage = 20$, and $P_c = 0.9$. With this combination of parameters, the CDGA-I model with NSA at $a = 4.0$ results in an improved design cost of €2,145,627 compared to €2,302,423 reported by Reca and Martinez (2006), using the GA model. The MFEs to locate €2,145,627 is 1,944,600, far less than the 10 million required for locating €2,302,423. When the $G_s$ is increased to 10,000, an optimal cost of €2,091,547 is obtained, requiring 4,565,400 MFEs,
still less than 10 million reported by Reca and Martinez (2006). Further, by increasing $P_s$ to 2400, an optimal cost of €2,047,814 is obtained, which is less by €254,609 from €2,302,423. Moreover, the MFEs required to locate €2,047,814 are 7,024,800, less by 2,975,200 compared to 10 million. Finally, the optimal cost obtained using the CDGA-I model is far lesser than the design cost of €2.218 million, as Moosavian and Lence (2019) reported using the fittest individual referenced DE model. The convergence plot corresponding to the optimal convergence of €2,047,814 is shown in Fig. 6. From a design cost of €1.044 × 10^{12} at the first iteration, sooner by the 18th iteration, the order of design cost is decreased to €10^6 as shown in Fig. 6. From there, by 2927 iterations, an improved design cost of €2,047,814 is located.

Thus, from the results, it is clear that from small to large size networks, the design cost saved is significant. In addition, of all the CDGA models formulated in this study, the CDGA-I model is outstanding with its excellent performance in converging to the optimal design cost (for the respective benchmark problems) with a reduced computational effort compared to certain deterministic and metaheuristic optimization techniques, as presented in Table 3.

### 5 Conclusions

This paper presented the chaos-directed genetic algorithm (CDGA) to replace the random phenomenon of GA, considering two approaches: the non-sequential approach (NSA) and the sequential approach (SA). Three chaotic maps of different complexities and dimensions, namely the Logistic map, the Henon map, and the Lorenz map, were considered, and their effects on the computational effort of GA were studied by validating them on three benchmark problems: Two-loop network (TLN), Bakryan network (BRN), and Goyang network (GYN). The chaos-directed models CDGA-I and CDGA-II (with the Logistic and Henon map, respectively) outperformed GA with better search efficacy in reaching the optimum solution with less computational effort. The main advantage of the CDGA-II and CDGA-III models is that, with the optimal parameter values of GA calibrated through a sensitivity analysis, the algorithm excels in converging to the optimal solution at every trial. Since the parameters of the algorithm fall in a certain range, the effort in performing sensitivity analysis is the same except the time of computation that varies with the increase in the dimension of the design problem. Further, the scalability of the best of all the models was successfully evaluated considering a large-dimensional problem, the Balerma irrigation network (BIN). The results revealed that the GA and CDAG-I models are more efficient with reduced computational effort than other widely studied algorithms reported in the literature.
The analysis and results in this study suggest that the combination of chaos ergodicity with GA heuristics has great potential and can be extended for practical applications. The study also suggests combining other bio-inspired or swarm-based metaheuristic algorithms with chaos ergodicity to improve their search efficacy to deal with large-scale nondeterministic polynomial-time hard (NP-hard) problems.

### Funding
No funding was received for conducting this study.

### Data availability
The benchmark problems considered in the present study are taken from the Centre for Water Systems, Benchmark Problems, University of Exeter (https://emps.exeter.ac.uk/engineering/research/cws/resources/benchmarks/pareto/).

### Code availability
All the models or codes that support the findings of this study are available from the corresponding author (written in the MATLAB software and are compiled with the simulation software EPANET using MATLAB-EPANET toolkit).

### Declarations

#### Conflict of interest
The authors declare that they have no conflict of interest.

---

### Table 3 Performance comparison of GA and CDGA with other optimization models

| Authors                                      | Algorithm                                           | Functional evaluations |
|----------------------------------------------|-----------------------------------------------------|------------------------|
| **Two-loop network**                         |                                                     |                        |
| Alperovits and Shamir (1977); Krapivka and Ostfeld (2009) | Linear programming (LP); GA-LP sub-optimal solutions |                        |
| Savic and Walters (1997); Wu and Simpson (2001); Reca and Martinez (2006); Poojitha et al. (2020) | GA                                                | 65,000; 7467; 10,000; 3600 |
| Cunha and Sousa (1999)                       |                                                     |                        |
| Eusuff and Lansey (2003)                     |                                                     |                        |
| Geem (2006)                                  |                                                     |                        |
| Suribabu and Neelakantan (2006); Sedki and Ouazar (2012); Ezzeldin et al. (2014) | Particle swarm optimization (PSO)                  | 1875; 3120; 549         |
| Poojitha et al. (2020); Suribabu (2010); Moosavian and Lence (2019) | Differential Evolution (DE); DE; FDE               | 7600; 1320; 1116        |
| Sedki and Ouazar (2012); Babu and Vijayalakshmi (2013); Jain and Khare (2021) | PSO-DE; PSO-GA; DDA-Rao (i) GA (ii) CDGA-I (i) GA (ii) CDGA-I | 3080; 1300; 3000 (i) 3000 (ii) 1600 |
| Present study                                |                                                     |                        |
| **Bakryan network**                          |                                                     |                        |
| Lee and Lee (2001); Geem (2006)              | GA; HS                                              | 15,000; 5000            |
| Poojitha et al. (2020)                       | GA; DE                                              | 2600                   |
| Present study                                | (i) GA (ii) CDGA-I (i) GA (ii) CDGA-I               | (i) 2200 (ii) 728       |
| **Goyang network**                           |                                                     |                        |
| Kim et al. (1994); Geem (2006); Mora-Melia et al. (2015); Sheikholeslami and Talatahari (2016); Jain and Khare (2021) | GA; HS; Developed swarm optimizer; DDA-Rao         | Sub-optimal solutions   |
| Present study                                | GA; DE                                              | 7800; 9550             |
| Poojitha et al. (2020)                       | (i) GA (ii) CDGA-I (i) GA (ii) CDGA-I               | (i) 10,088 (ii) 7280    |

---

### Supplementary Information
The online version contains supplementary material available at https://doi.org/10.1007/s00477-022-02200-7.

### References

Ali MZ, Awad NH, Suganthan PN, Shatnawi AM (2018) An improved class of real-coded genetic algorithms for numerical optimization. Neurocomputing 275(2018):155–166

Alperovits E, Shamir U (1977) Design of optimal water distribution systems. Water Resour Res 13(6):885–900

Ammaruekarat P, Meesad P (2011) A chaos search for multi-objective memetic algorithm. In: 2011 international conference on information and electronics engineering IPCSIT, vol. 6. IACSIT Press, Singapore

Arunkumar R, Jothiparakash V (2013) Chaotic evolutionary algorithms for multi reservoir optimization. Water Resour Manag 27(15):5207–5222

Babu KSJ, Vijayalakshmi DP (2013) Self-adaptive PSO-GA hybrid model for combinatorial water distribution network design. J Pipeline Syst Eng Pract 4(1):57–67

Cheng C-T, Wang W-C, Xu D-M, Chau K-W (2008) Optimizing hydropower reservoir operation using hybrid genetic algorithm and chaos. Water Resour Manag 22(7):895–909

Cimorelli L, D’Aniello A, Cozzolino L (2020) Boosting genetic algorithm performance in pump scheduling problems with a
Lorenz EN (1963) Deterministic nonperiodic flow. J Atmos Sci

Krapivka A, Ostfeld A (2009) Coupled genetic algorithm-linear

Kim JH, Kim TG, Kim JH, Yoon YN (1994) A study on the pipe

Kadu MS, Gupta R, Bhave PR (2008) Optimal design of water

Grassberger P, Procaccia I (1983) Measuring the strangeness of

Lansey KE, Mays LW (1989) Optimization model for water

Kim JH, Kim TG, Kim JH, Yoon YN (1994) A study on the pipe

Henon M (1976) A two-dimensional mapping with a strange attractor. Commun Math Phys 50(1):69–77

Goldberg DE, Kuo CH (1989) Genetic algorithms in pipeline

Fallah H, Kisi O, Kim S, Rezaie-Balf M (2019) A new optimization approach for the least-cost design of water distribution networks: an improved crow search algorithm. Water Resour Manag 33:3595–3613

Geem ZW (2006) Optimal cost design of water distribution networks using harmony search. Eng Optim 38(3):259–277

Gessler J (1985) Pipe network optimization by enumeration. In: Proceedings of computer applications in water resources, ASCE, New York, pp 572–581

Goldberg DE, Kao CH (1989) Genetic algorithms in pipeline optimization. J Comput Civ Eng 1(2):128–141

Grassberger P, Procaccia I (1983) Measuring the strangeness of strange attractors. Physica D 9(1–2):189–208

Gupta I, Gupta A, Khanna P (1999) Genetic algorithm for optimization of water distribution systems. Environ Model Softw 14(5):437–446

Haghighi A, Samani HMV, Samani ZMV (2011) GA-ILP method for optimization of water distribution networks. J Pipeline Syst Eng Pract 5(1):04013013

Ingu T, Takagi H (1999) Accelerating a GA convergence by fitting a single-peak function. In: 1999 IEEE international fuzzy systems conference proceedings, Seoul, Korea, III-1415: III-1420

Jain P, Khare R (2021) Application of parameter-less Rao algorithm in optimization of water distribution networks through pressure in optimization. Water Resour Manag 35:4067–4084

Johns MB, Keedwell E, Savic D (2014) Adaptive locally constrained genetic algorithm for least-cost water distribution network design. J Hydroinf 16(2):288–301

Kadu MS, Gupta R, Bhave PR (2008) Optimal design of water networks using a modified genetic algorithm with reduction in search space. J Water Resour Plan Manag 134(2):147–160

Keedwell E, Khu ST (2005) A hybrid genetic algorithm for the design of water distribution networks. Eng Appl Artif Intell 18(4):461–472

Kim JH, Kim TG, Kim JH, Yoon YN (1994) A study on the pipe network system design using non-linear programming. J Korean Water Resour Assoc 27(4):59–67

Krapivka A, Ostfeld A (2009) Coupled genetic algorithm-linear programming scheme for least-cost pipe sizing of water-distribution systems. J Water Resour Plan Manage 135(4):298–302

Lansey KE, Mays LW (1989) Optimization model for water distribution system design. J Hydraul Eng 115(10):1401–1418

Lee SC, Lee SI (2001) Genetic algorithms for optimal augmentation of water distribution networks. J Korean Water Resour Assoc 34(5):567–575

Lorenz EN (1963) Deterministic nonperiodic flow. J Atmos Sci 20:130–141

May RM (1976) Simple mathematical models with very complicated dynamics. Nature 261:459–467

Michalewicz Z, Schoenauer M (1996) Evolutionary algorithms for constrained parameter optimization problems. Evol Comput 4(1):1–32

Montesinos P, Garcia-Guzman A, Ayuso JL (1999) Water distribution network optimization using a modified genetic algorithm. Water Resour Res 35(11):3467–3473

Moosavian N, Lence B (2019) Fittest individual referenced differential evolution algorithms for optimization of water distribution networks. J Comput Civ Eng 33(6):04019036

Mora-Melia D, Iglesias-Rey PL, Martinez-Solano F-M (2013) Design of water distribution networks using a pseudo-genetic algorithm and sensitivity of genetic operators. Water Resour Manag 27:4149–4162

Mora-Melia D, Iglesias-Rey PL, Martinez-Solano FJ, Ballesteros-Perez P (2015) Efficiency of evolutionary algorithms in water network pipe sizing. Water Resour Manage 29:4817–4831

Mozaffari A, Emami M, Fathi A (2018) A comprehensive investigation into the performance, robustness, scalability and convergence of chaos-enhanced evolutionary algorithms with boundary constraints. Artif Intell Rev 52(4):2319–2380

Murphy LJ, Simpson AR (1992) Pipe optimization using genetic algorithms. Research Report No. R93, University of Adelaide, Australia

Poojitha SN, Singh G, Jothiprakash V (2020) Improving the optimal solution of Goyang network – using genetic algorithm and differential evolution. Water Supply 20(1):95–102

Recal J, Martinez J (2006) Genetic algorithms for the design of looped irrigation water distribution networks. Water Resour Res 42(5):W05416

Rossmann LA (2000) EPANET 2 USERS MANUAL, U.S. Environmental Protection Agency, Washington, D. C.

Savic DA, Walters GA (1997) Genetic algorithms for least-cost design of water distribution networks. J Water Resour Plan Manag 123(2):67–77

Sedki A, Ouazar D (2012) Hybrid particle swarm optimization and differential evolution for optimal design of water distribution systems. Adv Eng Inform 26:582–591

Sheikhholeslami R, Talatahari S (2016) Developed swarm optimizer: a new method for sizing optimization of water distribution systems J Comput. Civ Eng 30(5):04016005

Simpson AR, Dandy GC, Murphy LJ (1994) Genetic algorithms compared to other techniques for pipe optimization. J Water Resour Plan Manag 120(4):423–443

Sivakumar B (2009) Nonlinear dynamics and chaos in hydrologic systems: latest developments and a look forward. Stoch Env Res Risk Assess 23:1027–1036

Sivakumar B (2017) Chaos in hydrology: bridging determinism and stochasticity. Springer, New York

Suribabu CR (2010) Differential evolution algorithm for optimal design of water distribution networks. J Hydroinf 12(1):66–82

Suribabu CR, Neelakantan TR (2006) Design of water distribution networks using particle swarm optimization. Urban Water J 3(2):111–120

Vairavamoorthy K, Ali M (2000) Optimal design of water distribution systems using genetic algorithms. Comput-Aided Civ Infrastructure Eng 15(5):374–382

Vairavamoorthy K, Ali M (2005) Pipe index vector: a method to improve genetic-algorithm-based pipe optimization. J Hydraul Eng 131(12):1117–1125

Wu ZY, Simpson AR (2001) Competent genetic-evolutionary optimization of water distribution systems. J Comput Civ Eng 15(2):89–101
Yuan X, Yuan Y, Zhang Y (2002) A hybrid chaotic genetic algorithm for short-term hydro system scheduling. Math Comput Simul 59:319–327
Zheng F, Simpson AR, Zecchin AC (2011) A combined NLP-differential evolution algorithm approach for the optimization of looped water distribution systems. Water Resour Res 47:1–18
Zheng F, Simpson AR, Zecchin AC (2014) Coupled binary linear programming-differential evolution algorithm approach for water distribution system optimization. J Water Resour Plan Manag 140(5):585–597
Zyl JEV, Savic DA, Walters GA (2004) Operational optimization of water distribution systems using a hybrid genetic algorithm. J Water Resour Plan Manag 130(2):160–170

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.