Optimization of cascade stilling basins using GA and PSO approaches

R. Bakhtyar and D. A. Barry

ABSTRACT

In high head dams, the kinetic energy at the spillway toe is very high and the tail-water depth available for energy dissipation is relatively small. Cascade stilling basins are energy dissipation systems for high head dams, the design of which is based on a trial-and-error procedure. Although such an approach yields feasible designs in which hydraulic and topographic considerations are met, there may exist many cost-effective designs. Therefore, optimization tools can help find the least construction cost while keeping hydraulic and topographic considerations satisfied. Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) were used to determine the optimal design of cascade stilling basins in terms of the height of falls and length of stilling basins. The approach was evaluated by application to the design of an energy dissipation system for the Tehri Dam on the Bhagirathi River. Comparison of the proposed methods with dynamic programming and an alternative approach not utilizing an optimization tool revealed that GA and PSO lead to significant savings in the construction cost with less computational effort.

Key words | construction cost, evolutionary algorithms, high head dams, sensitivity, spillway, tail-water

R. Bakhtyar (corresponding author)
D. A. Barry
Institut des Sciences et Technologies de l'Environnement,
Ecole Polytechnique Fédérale de Lausanne,
CH-1015 Lausanne,
Switzerland
Tel.: +41 21 693 8086
Fax: +41 21 693 5670
E-mail: *roham.bakhtyar@gmail.com*

INTRODUCTION

When spillway flows fall from the level of the reservoir to the downstream river, the static head is converted to kinetic energy. This energy manifests itself in the form of high velocities that, if obstructed, result in large pressures. Means of returning the flow to the river without serious scour, erosion of the toe of the dam or damage to adjacent structures must usually be provided. This can be accomplished by the use of an energy-dissipation device such as a hydraulic jump basin, a roller bucket, a sill block apron or a basin incorporating impact baffles and walls (USBR 1958).

Cascade stilling basins have also been used in cases of high head dams to reduce the spillway cost. The other advantage, besides the construction savings, is the higher energy dissipation along the spillway in comparison to a conventional spillway, leading to a reduction of the stilling basin dimensions at the end of the spillway. For high flow rates in a conventional spillway, possible damage to the structure may be caused by the flow-induced dynamic loads. For cascade stilling basins, however, the flow velocity along the spillway would be much less and, consequently, phenomena such as cavitation, vibration and abrasion would not feature in design considerations.

A cascade stilling basin system consists of a series of falls, with a stilling basin at each fall. The flow energy is broken into several parts, each dissipated by a set of falls and stilling basins. The design rules applicable to cascade stilling basins are the same as those of normal stilling basins. Vittal & Porey (1987) proposed a stage-by-stage procedure for the design of a system of cascade stilling basins. In their approach, however, meeting hydraulic rules and criteria were the main concern and no attempt was made to achieve an optimally cost-effective design.

doi: 10.2166/hydro.2009.046

Vittal and Porey's method (VP) is based on a series of hand calculations which evaluates a limited number of alternatives and selects the best one. First, the number of preceding falls, specification of the falls, height and length of falls and stilling basins are determined. This is followed by graphical optimization of the longitudinal profile of the spillway by estimating the excavation volumes. By changing the longitudinal coordinate of each fall, one may find an alternative with the least excavation volume. Bakhtyar *et al.* (2007) have developed a dynamic programming (DP) approach to optimal design of cascade stilling basins based on this procedure. DP methods, although theoretically capable of finding the global optimum solution, suffer from the so-called curse of dimensionality and therefore may not be applicable to large scale real-world cascade stilling basin systems.

In recent years, algorithms motivated by natural processes, such as the Genetic Algorithm (GA) and Particle Swarm Optimization (PSO), have been used for the solution of optimization problems. GA is a combinatorial search technique based on the mechanics of natural selection and genetics (Holland 1975; Goldberg 1989). It is widely used in optimizing the design of water distribution systems (e.g. Savic & Walters 1997), reservoir operations (e.g. Wardlaw & Sharif 1999), flood forecasting (e.g. Wu & Chau 2006) and rainfall–runoff modeling (e.g. Cheng *et al.* 2005, 2006). PSO is an evolutionary optimization technique that simulates the social behavior of bird flocking (Kennedy & Eberhart 1995). The key idea is that, in the flock, any agent of the group can profit from the discoveries and previous experiences of all members of the flock during a food search. This advantage can become decisive, outweighing the inconvenience of competition for food items whenever the resource is unpredictably distributed in patches. Information sharing gives an evolutionary advantage. The main algorithmic idea is to generate particles randomly and assign to them a motion law using the notion of a velocity. The PSO algorithm has been applied successfully to solve several optimization problems (e.g. Kennedy *et al.* 2001; Chau 2006, 2007*a*,*b*). GA and PSO have no special restriction for the problem's linearity, continuity and convexity, and provide a higher probability of obtaining the global optimal solution in situations with many local solutions. However, these approaches have yet to be fully exploited in the design of hydraulic structures.

In this study two general nonlinear optimization models, based on Vittal and Porey's approach, are developed. Both a GA and a PSO are employed for the design of cascade stilling basins, in which both the height of falls and length of stilling basins are defined as decision variables. Two types of constraints are considered, i.e. topographic constraints and hydraulic design criteria. The objective of the optimization problem is to minimize the total cost of construction while satisfying the constraints. The rest of this paper is as follows. First, evolutionary computation algorithms are concisely presented. Then, the applications of PSO and GA to the design of cascade stilling basins are described, and the results compared with those given by the VP and DP schemes. Finally, conclusions are presented and discussed.

OPTIMIZATION METHODS

In this study, GA and PSO are employed to optimize the cascade stilling basins. The utility of these methods and their comparative performance will be examined. First, however, we provide a brief description of each approach.

Genetic algorithms

Genetic Algorithms (GAs) are adaptive heuristic search algorithms premised on the evolutionary process of natural selection and genetic. GA represents an intelligent exploitation of a random search within a defined search space to solve an optimization problem. The method is modeled loosely on improving fitness via natural selection, i.e. employing a population of individuals that undergo selection in the presence of variation-inducing operators such as recombination (crossover) and mutation. Crossover specifies how GA combines two individuals, or parents, to form a child, or children, for the next generation. Mutation indicates how a GA makes small random changes in individuals in the population to create mutated children. Mutation provides genetic diversity and enables the genetic algorithm to search a broader space, reducing the probability of convergence to a local optimum. In order to preserve and use previously found best solutions in subsequent generations, an elite-preserving operator is often recommended. In an elitist GA, the best-population solutions cannot degrade with generations. A fitness function is used to evaluate individuals, and reproductive success varies with fitness. The fitness function evaluates the quality of individuals (chromosomes) as solutions to a particular problem. The basic steps of a standard GA can be summarized as follows:

(i) Randomly generate an initial population $I(0)$and set $t = 0$;
(ii) Compute the fitness $f(m)$ of each individual m in the current population $I(t)$
(iii) Calculate selection probabilities $p(m)$ for each individual m in $I(t)$ so that $p(m)$ is proportional to $f(m)$;
(iv) Generate $I(t+1)$ by probabilistically selecting individuals from $I(t)$ to produce offspring via genetic operators;
(v) Repeat steps (ii)–(iv) until convergence is obtained.

The convergence of the GA is dependent on the choice of the population size, probabilities of crossover and mutation, and number of generations without improvement.

Particle swarm optimization algorithms

PSO (Kennedy & Eberhart 1995) is a stochastic population-based optimization approach that mimics the social dynamics of bird flocking. A large number of birds flock synchronously, change direction suddenly, scatter and regroup iteratively, and finally perch on a target. This form of social intelligence not only increases the success rate for food foraging but also expedites the process. The PSO algorithm facilitates simple rules simulating bird flocking and serves as an optimizer for nonlinear functions. It is a population-based algorithm, formed by a set of particles, representing potential solutions for a given problem.

The basic elements of PSO technique are briefly stated and defined as follows (Yin 2006):

Particle: A candidate solution represented by a d-dimensional vector, where d is the number of parameters to optimize.

Population: Set of particles at each iteration. Each particle moves through a d-dimensional search space, with associated position and velocity vectors for the current evolutionary iteration.

Particle representation. The particle is a candidate solution to the underlying problem and moves iteratively through the solution space. The particle is represented as a real-valued vector containing an instance of all parameters that characterize the optimization problem. We denote the ith particle by $P_i = (P_{i1}, \ldots, P_{id})^{\mathrm{T}} \in R^d$ where d is the number of parameters.

Swarm. The PSO explores the solution space by evolving a number of particles, called a swarm. The initial swarm is generated at random and the size of swarm is usually kept constant through iterations. At each iteration, the swarm searches for an optimal solution by referring to previous experiences.

Personal best experience and swarm's best experience. The PSO enriches the swarm intelligence by storing the best positions visited so far by every particle. In particular, particle i remembers the best position among those it has visited, referred to as $pbest_i$, and the best position of its neighbors. There are two versions for keeping the neighbor's best position, namely *lbest* and *gbest*. In the local version, each particle keeps track of the best position *lbest* attained by its local neighboring particles. For the global version, the best position *gbest* is determined by any particle in the entire swarm. Hence, the *gbest* model is a special case of the *lbest* model. It has been shown that the local version is often better, particularly the one using a random topology neighborhood where each particle generates L links randomly after each iteration if there has been no improvement, i.e. if the best solution seen so far by the swarm is still the same.

Particle movement. The PSO is an iterative algorithm in which a swarm of particles flies about the solution space until the stopping criterion is satisfied. At each iteration, particle i adjusts its velocity v_{ij} and position p_{ij} through each dimension j by referring to, with random multipliers, the personal best position ($pbest_{ij}$) and the swarm's best position ($lbest_{ij}$, if the local version is adopted) using equations as follows (the updated values appear on the left):

$$v_{ij} = v_{ij} + c_1 r_1 (pbest_{ij} - p_{ij}) + c_2 r_2 (lbest_{ij} - p_{ij}) \quad (1)$$

and

$$p_{ij} = p_{ij} + v_{ij} \quad (2)$$

where c_1 and c_2 are the acceleration coefficients and r_1 and r_2 are random real numbers drawn from $U(0,1)$. The three terms on the right-hand side of Equation (1) affect the velocity of a particle as follows:

- v_{ij}, serves as a momentum term to prevent excessive oscillations in the search direction;
- $c_1r_1(pbest_{ij} - p_{ij})$, referred to as the cognitive component. This component represents the distance of a particle from the best solution, $pbest_{ij}$, found by itself. The cognitive component represents the natural tendency of individuals to return to environments where they experienced their best performance;
- $c_2r_2(lbest_{ij} - p_{ij})$, referred to as the *social component*. This component represents the distance that a particle is from the best position found in its neighborhood. It represents the tendency of individuals to follow the success of other individuals.

Thus, the particle tends to fly toward *pbest* and *lbest* while still exploring new areas by the stochastic mechanism (i.e. r_1 and r_2) to escape from local optima.

Initial PSO studies used $c_1 = c_2 = 2$. Although good results were obtained, it was observed that velocities quickly exploded to large values, especially for particles far from their global best and personal best positions. Consequently, these particles have large position updates and leave the boundaries of the search space. To control the increase in velocity, velocities are clamped. While velocity clamping does not prevent a particle from leaving the boundaries of its search space, it does limit the particle step sizes, thereby restricting divergent behavior.

The *inertia weight* was introduced by Shi & Eberhart (1998*a*,*b*) to eliminate the need for velocity clamping, but still to restrict divergent behavior. The inertia weight, w, controls the momentum of the particle by weighing the contribution of the previous velocity, thereby controlling how much memory of the previous flight direction will influence the new velocity. The velocity equation changes to

$$v_{ij} = wv_{ij} + c_1r_1(pbest_{ij} - p_{ij}) + c_2r_2(lbest_{ij} - p_{ij}) \tag{3}$$

where $c_1 + c_2 \leq 4$ (Carlisle & Dozier 2001).

Initial empirical studies of PSO with inertia have shown that the value of w is extremely important to ensure convergent behavior (Shi & Eberhart 1998*a*,*b*). For $w > 1$, velocities increase over time, causing divergent behavior. Particles fail to change direction in order to move back towards promising areas. For $w < 1$, particles decelerate until their velocities reach zero.

- *Stopping criterion.* The PSO algorithm is terminated upon reaching a maximal number of iterations or if the best particle position of the entire swarm cannot be improved further after a sufficiently large number of iterations.

Mathematical formulation

The aim is to calculate decision variables in the search space using both GA and PSO. To achieve this, one has to solve a constrained optimization problem:

$$\begin{aligned}&\text{Minimize}\, f(x)\\&\text{subject to}\, g_j(x) \geq 0, \quad j = 1, 2, \ldots, J\end{aligned} \tag{4}$$

where J is the total number of constraints and g_j is the jth constraint. The solution of this problem is obtained using the penalty function approach. For each solution $x^{(i)}$, the penalty, d_j, calculated for the violation of the jth constraint is

$$d_j(x^{(i)}) = \begin{cases} |g_jx^{(i)}| \text{ or } g_j^2(x^{(i)}) \text{ if } g_j(x^{(i)}) < 0 \\ \quad 0 \quad \text{otherwise} \end{cases} \tag{5}$$

Thereafter, all constraint violations are added together to get the overall constraint violation. It is then multiplied with a penalty parameter *Pen* (strictly positive) and the product is added to each of the objective function values:

$$\text{Minimize}\, F(x^{(i)}) = f(x^{(i)}) + Pen\sum_{j=1}^{J} d_j(x^{(i)}). \tag{6}$$

The choice of the penalty coefficient *Pen* in Equation (6) is crucial for the convergence of the search toward the solution of Equation (4). Generally, *Pen* should be sufficiently large that each of the two terms on the right-hand side of Equation (6) is of the same order of magnitude. Therefore, to determine this coefficient, a sequence of Pen_{iter} values is used, with $Pen_{\text{iter}+1} = 10\ Pen_{\text{iter}}$, and

$Pen_0 = 1$, where iter is the iteration number. Generally, convergence is achieved quickly.

APPLICATION OF EVOLUTIONARY APPROACHES TO THE DESIGN OF CASCADE STILLING BASINS

The object of the optimization model is to minimize the construction costs via changing the geometric and design variables–height of falls and length of stilling basins–while fulfilling the hydraulic and topographic criteria used in the VP approach. In this approach, the height of the lowest fall is determined by the available river tail-water depth at the design discharge, whereas the number of preceding equal-height falls is determined by the available distance for the spillway sections and stilling basins. The steps of the design procedure for the preceding falls are, briefly:

(i) The number of falls is assumed and the height of the crest above the stilling-basin floor is determined by trial and error;

(ii) Having determined the height of the crest, one can compute conjugate depth, pre-jump Froude number and pre-jump depth of flow. The base widths of spillway sections and lengths of stilling basins are determined afterwards. Subsequently, the total length, required for accommodation of all spillways and stilling basins, is determined;

(iii) The above procedure is repeated until a number of falls and basins are obtained for which the total required length is equal to or less than the available length at the site;

(iv) The corresponding heights and rise of crests are chosen. Thereafter, the spillway profiles are computed for the falls and the appurtenances of the stilling basins are proportioned based on the recommendations of USBR (1958).

The design principles used in the optimization model are those of the VP scheme with the difference that, in VP's method, the total horizontal length of the system remains the same during the trial and error process explained above. In other words, the position of the toe of the last fall is fixed. In the optimization model, the total length of the system is considered as a decision variable of the model to be determined. A reach with a specified length is taken as the range in which the toe of the terminal stilling basin could be placed.

Considering the design criteria and equations presented in VP as the constraints of a nonlinear mathematical program, one may formulate the resulting optimization model as follows:

$$\text{Minimize TotCost} = \sum_{k=1}^{N} [\text{ConCost}(P_k, \ell_k) + \text{ExCost}(P_k, \ell_k)] \qquad (7)$$

where TotCost is the total construction cost, ConCost (P_k, ℓ_k) and ExCost (P_k, ℓ_k) are the concrete and excavation costs of the kth cascade, respectively; P_k is the height of the kth fall, ℓ_k is the length of the kth fall and N is the total number of cascades (Figure 1). The constraints that have to be satisfied to optimize Equation (7) are discussed below.

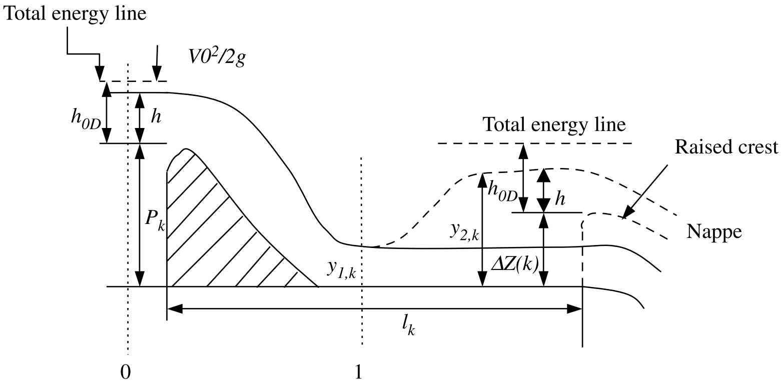

Figure 1 | Schematic definition of the kth fall and the related stilling basin (Bakhtyar *et al.* 2007).

Hydraulic constraints

Minimum and maximum pre-jump Froude number

$$P_{\min} \le P_k \le P_{\max} \tag{8}$$

$$\Delta Z(k) = 1.671\frac{q_D^{0.5}P_k}{g^{1/4}} - \left(\frac{q_D}{C\sqrt{2g}}\right)^{2/3} + 0.179\frac{q_D}{g^{1/2}P_k^{1/2}}, \qquad \Delta Z(N) = 0 \tag{9}$$

where q_D is the unit design discharge, g is the magnitude of gravitational acceleration, C is the discharge coefficient with a conservative value of 0.47 and $\Delta Z(k)$ is the height of the crest for the kth fall or stilling basin. To force the jump in stilling basins, a control or crest, preferably with an ogee profile, is placed at the end of the basin. The required height of crest $\Delta Z(k)$ for jump formation and for the kth fall at the design discharge is given by Equation (9) (Vittal & Porey 1987). $P_{\max}$ and $P_{\min}$ are the maximum and minimum height of the falls calculated using the maximum and minimum pre-jump Froude numbers $Fr_{1_{\max}}$ and $Fr_{1_{\min}}$ of the flow in the corresponding stilling basin, respectively, as follows:

$$P_{\max} = \frac{q_D^{2/3}}{g^{1/3}}\left(\frac{1}{2}Fr_{1_{\max}}^{4/3} + Fr_{1_{\max}}^{(-2/3)} - \frac{1}{2^{1/3}C^{2/3}}\right) \tag{10}$$

$$P_{\min} = \frac{q_D^{2/3}}{g^{1/3}}\left(\frac{1}{2}Fr_{1_{\min}}^{4/3} + Fr_{1_{\min}}^{(-2/3)} - \frac{1}{2^{1/3}C^{2/3}}\right) \tag{11}$$

$$Fr_{1_{\min}} \le Fr_{1k} \le Fr_{1_{\min}} \tag{12}$$

The minimum and maximum Froude numbers $Fr_{1_{\min}}$ and $Fr_{1_{\max}}$ of the flow are, respectively, 4.5 and 9. Within this range a completely developed hydraulic jump can form (USBR 1958). This range specified for the Froude number is given by constraint (12). The minimum and maximum heights of the kth fall, $P_{k\min}$ and $P_{k\max}$, presented in the VP approach are considered in the optimization model using Equations (10) and (11). Therefore, the model is forced to satisfy that specified range for the height of falls by constraint (8).

Minimum length of the stilling basins

$$\ell_k \ge \ell_{k\min} \tag{13}$$

$$\ell_{k\min} = 1.455\, h_{0D}\left(\frac{P_k}{h_{0D}}\right)^{1/1.85} + 6(y_{2,k} - y_{1,k}) \tag{14}$$

$$h_{0D} = \left(\frac{q_D}{C\sqrt{2g}}\right)^{2/3} \tag{15}$$

$$y_{1,k} = \left(\frac{q_D}{g^{1/2}Fr_{1_k}}\right)^{2/3} \tag{16}$$

$$y_{2,k} = \frac{y_{1,k}}{2}\left(\sqrt{1 + 8\,Fr_{1k}^2} - 1\right) \tag{17}$$

$$y_{\mathrm{tD}} = \left(\frac{7.81 q_D^2 P_N}{g}\right)^{1/4} \tag{18}$$

where y_{tD} is the tail-water depth at the design discharge, $\ell_{k\min}$ is the minimum allowable length of the falls calculated in terms of the total head over the crest of spillway, h_{0D}, the pre-jump and conjugate depth of flow, $y_{1,k}$ and $y_{2,k}$, and the pre-jump Froude number, Fr_{1k}, in the kth stilling basin (Poggi 1956).

Geometrical constraints

Maximum height and length available for the system

$$\sum_{k=1}^{N}[P_k - \Delta Z(k)] + \Delta\bar{Z}(N) \le H_{\mathrm{T}} \tag{19}$$

$$\sum_{k=1}^{N} l_k \le L_{\mathrm{T}} \tag{20}$$

where H_{T} is the total height available, which is the difference between the main (first) fall crest's elevation and the terminal point of the last fall, and L_{T} is the total length available, which is the horizontal distance between the center point of the main (first) fall and the terminal point of the last basin.

The deficiency or excess of tail-water at partial discharge can be found by comparing the free-jump-height curve (FJHC) for the terminal fall with the tail-water rating curve (TWRC) of the river. In the event of a tail-water excess, the stilling basin need not be depressed whereas, in the event of a deficiency, the floor will be lowered by $\Delta\bar{Z}(N)$ equal to the maximum difference in the ordinates of FJHC and TWRC at partial discharge. With the drop in the floor level, $P_{(N)}$ is replaced by $P_{(N)} + \Delta\bar{Z}(N)$.

The problem's constraints can be written in standard form as

$$g_1(P_k) = H_{\mathrm{T}} - \sum_{k=1}^{N} [P_k - \Delta Z(k)] - \Delta\bar{Z}(N) \geq 0 \tag{21}$$

$$g_2(\ell_k) = L_{\mathrm{T}} - \sum_{k=1}^{N} \ell_k \geq 0 \tag{22}$$

$$g_3(P_k) = P_{k\,\max} - P_k \geq 0 \tag{23}$$

$$g_4(P_k) = P_k - P_{k\,\min} \geq 0 \tag{24}$$

$$g_5(\ell_k) = l_k - \ell_{k\,\min} \geq 0 \tag{25}$$

with the constraint violations defined as

$$\begin{aligned} d_s &= \max\{g_s^2, 0\} \quad s = 1, 2 \\ d_s &= \max\{|g_s|, 0\} \quad s = 3, 4, 5 \end{aligned} \tag{26}$$

The premise for implementing $d_s = \max\{g_s^2, 0\}$ for the g_1 and g_2 constraints is to ensure that the total length and height of the cascades are contained within the length and height available at the site.

Applying fixed value penalty coefficients Pen_s for each constraint, the penalized objective function is now written as follows:

$$\begin{aligned} &\text{Minimize}\,\text{TotCost}(P, \ell) \\ &= \left\{ \left[\sum_{k=1}^{N} (\text{ConCost}(P_k, \ell_k) + \text{ExCost}(P_k, \ell_k)) \right] + \sum_{s=1}^{5} Pen_s d_s \right\} \end{aligned} \tag{27}$$

Here, contrary to the VP scheme, the position of the terminal fall is not considered to be known *a priori*. The end position is only assumed to lie within a specified range.

The GA procedure starts with the generation of the initial random population (height of falls and length of stilling basins as decision variables). Each member of the population is then evaluated based on its fitness value, TotCost. The members of the next generation are subsequently formed with the process of crossover that represents the exchange of genes of the parents to produce an offspring. The PSO optimization process starts with the generation of the random initial particles (height of falls and length of stilling basins as decision variables), with random initial velocities for each particle. Each particle in the initial population is then evaluated using the objective function, TotCost. PSO searches for the best value of the objective function, then sets the particle associated with the best objective function as the global best and sets the initial value of the inertia weight. A suitable value for the inertia weight w usually provides balance between global and local exploration abilities and consequently results in a reduction of the number of iterations required to locate the optimal solution. In the next iteration, PSO updates the inertia weight using the following Equation (Shi & Eberhart 1998*a*,*b*):

$$w(\text{iter}) = w_{\max} - (w_{\max} - w_{\min})\frac{\text{iter}}{\text{iter}_{\max}} \tag{28}$$

where $w(\text{iter})$ is the inertia weight at each iteration, $\text{iter}_{\max}$ is the maximum iteration number, and $w_{\max}$ and $w_{\min}$ are, respectively, the maximum and minimum inertia weights. With the global best and individual best of each particle, PSO updates the particle velocities. Based on the updated velocities, each particle changes its position. The next steps are evaluating the individual best updating and global best updating. This procedure is continued until the stopping criteria are satisfied.

CASE STUDY

The optimization of a cascade stilling basin system can be a difficult task since it possesses both highly nonlinear and complex aspects. The efficiency of the proposed methods can be assessed by applying them to a benchmark example, the Tehri Dam, as previously analyzed by Vittal & Porey (1987). The Tehri Dam is a rock-fill dam with a central clay core on the Bhagirathi River in India's Ganga Valley of the Central Himalayas. The spillway is located on the right abutment, supported by exposed layered rocks formed by weak quartzite. Tables 1 and 2 report the problem's input parameters, including basic characteristics of the dam and the topographic data of the spillway path.

The functionality and feasibility of all types of spillways were considered in the preliminary design. For example, a single spillway and stilling basin results in a 66 m s^{-1}

Table 1 | Basic characteristics of the Tehri Dam

Design flow	11,000 $\mathrm{m}^3\,\mathrm{s}^{-1}$
Spillway crest length	95 m
Downstream riverbed elevation	640 m
Total spillway height	220 m
River tail-water depth for the design flow	29.2 m
Cumulative horizontal length in the spillway	778 m
Main spillway crest elevation	860 m

velocity in the latter. Due to the inadequate depth of the tail-water in the basin, a 15 m excavation under the water table is needed. Because of the narrow valley at the dam site and the low strength of the rock formations that would bear the impact forces exerted by the projectile flow, the use of a chute spillway with a projectile flow at the end is not suitable. Clay seams exist in the rock formation and, if they become saturated from the projectile flow, there is a risk of landslides.

RESULTS AND DISCUSSION

Using the data in Tables 1 and 2, the design was carried out using VP, the DP model of Bakhtyar *et al.* (2007), PSO and GA schemes. The computational flow charts of the PSO and GA algorithms are shown in Figures 2(a, b), respectively. A GA with real-valued coding, a population size of 50, tournament selection, weighted averaging crossover, probability of crossover equal to 0.8, Gaussian mutation and the number of individuals guaranteed to survive at each generation (elitism) equal to 4, were used. A sensitivity analysis was also carried out to assess the effects of varying the crossover probability on the GA performance. GA runs were carried out with crossover probabilities ranging from 0.6–0.95. Figure 3 shows the sensitivity of the crossover probability on the GA. In this figure, the fitness results are scaled with those obtained with the crossover equal to 0.8. The best values were achieved for the crossover probability of 0.85. The influence of elitism on the GA was also evaluated. Figure 4 shows the sensitivity of the achieved fitness to the number of individuals guaranteed to survive at each generation. Here, fitness results are scaled with those obtained with the elitism equal to 4. Evidently, the best results were achieved with 4, 5 and 6 mutations.

The PSO had the following characteristics. The size of the population was 50 and the maximum and minimum of inertia weights were $w_{\max} = 0.9$ and $w_{\min} = 0.4$. The role of the inertia weight, w, is considered critical for the PSO's convergence behavior. The inertia weight controls the impact of the previous history of velocities on the current one. Accordingly, the parameter w regulates the trade-off between the global (wide-ranging) and local (nearby) exploration abilities of the swarm. A large inertia weight facilitates global exploration (searching new areas), while a small one tends to facilitate local exploration, i.e. fine-tuning the current search area. Equation (28) with an initial value around 0.9 and a gradual decline towards 0.4 was used for updating the inertia weight. Generally, more exploration should be carried out in the initial stages when the algorithm has very little knowledge about the search space. In contrast, more exploitation is needed in the later stages so that the algorithm is able to exploit the information it has gained so far. Figure 5 illustrates the fitness function during the evolution process for different maximum and minimum inertia weights. The acceleration constants in PSO scheme were $c_1 = 0.5$ and $c_2 = 0.5$. Other values of c_1 and c_2, like 1, 1.5 and 2, enabled algorithm convergence in an equivalent number of simulations. The choice of these constants is problem-dependent. If c_1 and $c_2$are too large with respect to the bounds of the search space, the particle will regularly leave the search space. The impact of these coefficients on the search is that, if they are not optimal, the number of iterations could be large. The success of PSO and GA depends on the proper choice of the penalty coefficients. The proper setting of the penalty parameters is very important in the solutions to the cascade optimization problems, as a low value of the penalty parameters could lead to a constraint-violating solution,

Table 2 | Topographic data (m) of the spillway path (Tehri Dam)

x	0	25	100	150	200	230	263	300	350	410	490	500	600	640	700	710	770	780
y	640	670	675	685	735	745	758	770	775	790	803	810	820	830	840	845	855	860

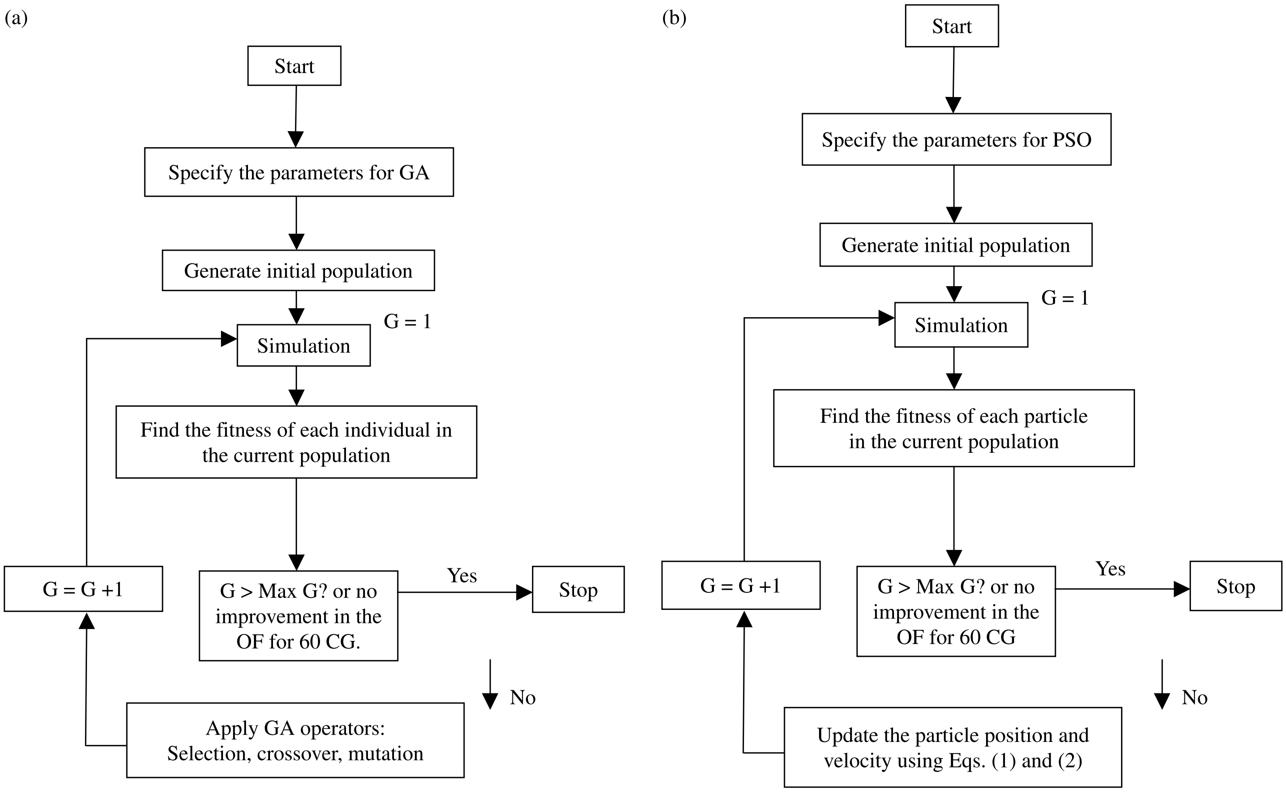
(a)
Start
Specify the parameters for GA
Generate initial population
G = 1
Simulation
Find the fitness of each individual in the current population
G = G +1
G > Max G? or no improvement in the OF for 60 CG.
Yes
Stop
No
Apply GA operators: Selection, crossover, mutation
(b)
Start
Specify the parameters for PSO
Generate initial population
G = 1
Simulation
Find the fitness of each particle in the current population
G = G +1
G > Max G? or no improvement in the OF for 60 CG
Yes
Stop
No
Update the particle position and velocity using Eqs. (1) and (2)

Figure 2 | Flowcharts of (a) genetic algorithm and (b) particle swarm optimization algorithm (G = generation, CG = consecutive generations, OF = objective function).

while a high value of the parameters would result in rejecting some of the constraint-violating solutions that nonetheless rely on useful information from the evolution process. The penalty coefficients were chosen using a trial-and-error procedure. The algorithm was stopped if there was no improvement in the objective function for 60 consecutive generations.

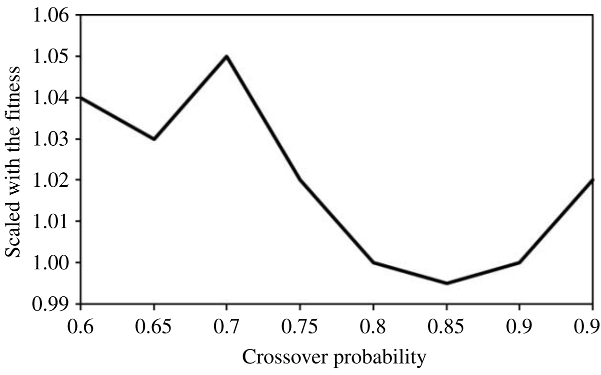

Figure 3 | Sensitivity of the crossover probability in GA (the fitness results are scaled with that obtained with the crossover equal to 0.8).

Figures 6 and 7 show the convergence history of the fitness function during the evolution process for different population sizes using the GA and PSO schemes. Clearly, the quality of the final solution improves with increasing population size as one would expect, although at the expense of increased computational effort and storage requirements. It is interesting to notice that the PSO runs converge to approximately the same solution. Figure 7

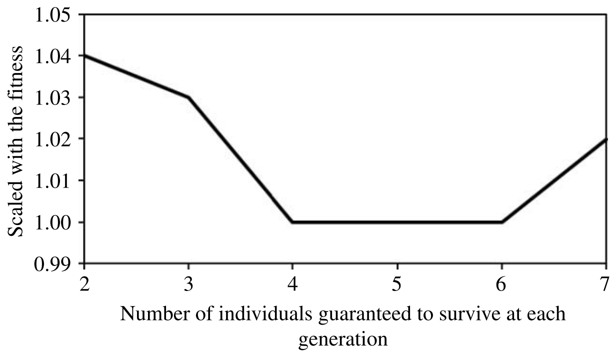

Figure 4 | Sensitivity of the mutation in GA.

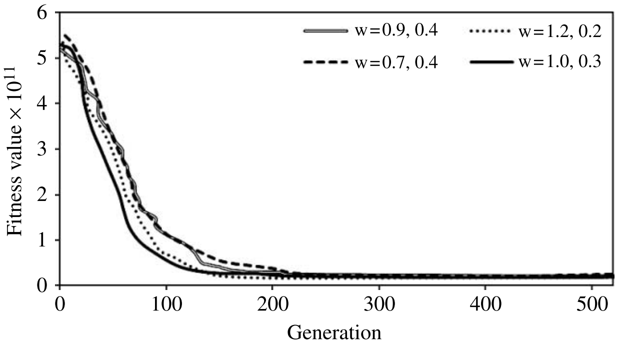

Figure 5 | Rate of convergence of the fitness function for different inertia weight values using PSO.

shows that the proposed PSO procedure considerably reduces the sensitivity of the model to the population size and improves the quality of the solution obtained with less computational effort.

Depending on the natural topography, foundation conditions and the excavation required, the spillway profile could consist of either a mildly sloping upstream section with a vertically sloped curve and a steep downstream section, or a steep section with a constant slope. The chute slope should normally not be very steep as they make construction difficult and more expensive, increase the potential for downhill creep to occur, and result in difficult access for inspection and maintenance. The chute profile should preferably follow the natural topography as closely as possible to minimize rebound due to deep excavations. By changing the longitudinal coordinate of each fall, Vittal and Porey attempted to find an alternative with the least excavation volumes. Thus, the first objective function of GA and PSO considered is to minimize the excavation volume. Previously (Bakhtyar *et al.* 2007), the design was carried out

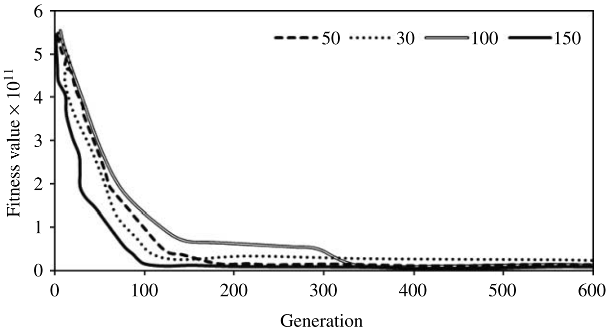

Figure 6 | Rate of convergence of fitness function for different population sizes (listed in the figure) using GA.

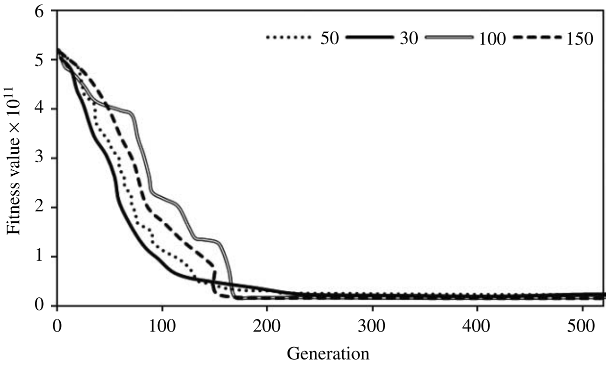

Figure 7 | Rate of convergence of fitness function for different population sizes (listed in the figure) using PSO.

using the VP and DP schemes. According to Tables 3 and 4 of that paper, the best configuration consists of four spillways and stilling basins. Further, it was found that, if the number of cascades is more than five, the hydraulic and topographic constraints used in the VP scheme cannot be satisfied. Therefore, here we compare the results of the GA and PSO schemes under the same conditions with the results of VP and DP for their best alternative (i.e. when $N = 4$). The total length of basins in the PSO, GA, DP and VP schemes are 771, 771.5, 772 and 781 m, respectively. The total net height of falls obtained by the PSO, GA, DP and VP schemes are 209.22, 210.33, 220 and 215 m, respectively. The excavation volumes in the PSO, GA, DP and VP schemes are 225×10^4, 228×10^4, 232×10^4 and $354 \times 10^4\,\mathrm{m}^3$, respectively. Compared with the VP scheme, the PSO, GA and DP results produce cost reductions of 36.4%, 35.5% and 34%, respectively. Figure 8 compares the longitudinal profile of the successive cascades suggested by VP, DP and GA. The concrete volumes in the PSO, GA, DP and VP schemes are 47.7×10^4, 48.4×10^4, 51.3×10^4 and $52.5 \times 10^4\,\mathrm{m}^3$, respectively.

Figures 9 and 10 show the rate of convergence of the fitness function from 10 independent runs to show the effect of the generation of random numbers in GA and PSO schemes. Figures 11–14 present the maximum, average and minimum rates of convergence of the height and length of stilling basins over 10 PSO runs. The results of the 10 different GA runs with their statistics are presented in Table 4. It is seen that the results obtained are similar.

To make the PSO and GA objective functions more realistic, the design was re-optimized using both the

Table 3 | Comparison of the results of the VP, GA and PSO schemes for the best alternative

	PSO				VP				GA				DP			
		Cascade horizontal length				Cascade horizontal length				Cascade horizontal length				Cascade horizontal length		
Number of sets of falls and basins	Net height of fall	Total length	Basin length	Horizontal length of fall	Net height of fall	Total length	Basin length	Horizontal length of fall	Net height of fall	Total length	Basin length	Horizontal length of fall	Net height of fall	Total length	Basin length	Horizontal length of fall
1st cascade	56.19	272.89	229.38	43.51	50	202	155	47	56.56	272.93	229.41	43.52	70	313	272	41
2nd cascade	40.89	179.19	142.92	36.27	50	202	155	47	41.12	179.24	142.83	36.41	30	133	102	31
3rd cascade	45.08	137.94	99.73	38.21	50	202	155	47	45.41	138.13	99.84	38.29	30	133	102	31
4th cascade	67.06	180.98	132.69	48.29	65	175	125	50	67.24	181.21	132.94	48.27	90	193	137	56

Table 4 | Statistical measures of decision variables for 10 runs using GA

Decision variables	Maximum	Average	Minimum
Basin length-1st cascade	229.41	229. 35	229.29
Basin length-2nd cascade	142.96	142.92	142.83
Basin length-3rd cascade	99.94	99.89	99.81
Basin length-4th cascade	132.96	132.88	132.84
Net height-1st cascade	56.61	56.49	56.46
Net height-2nd cascade	41.14	41.08	41.03
Net height-3rd cascade	45.50	45.44	45.37
Net height-4th cascade	67.31	67.29	67.22

concrete and excavation volumes as the cost. Table 5 compares the results of VP, DP, GA and PSO schemes for the two objective functions. For this case, the GA objective function and that of VP is reduced from 35.5% to 32.5% in the case for which the concrete-works cost is considered, while the difference between the PSO objective function and that of VP is reduced from 36.4% to 33.5%. We compared also the computational time needed to obtain the DP and the proposed evolutionary technique results with different objective functions. The evolutionary methods are more efficient. Similarly the excavation volume and concrete volume obtained from the GA and PSO methods are smaller compared to the excavation and concrete volumes obtained using DP. Also the proposed methods are free of the computational difficulties faced in the DP. Besides, DP methods suffer from the curse of dimensionality and therefore might not applicable to large scale real-world cascade stilling basin systems. Hence, it can be concluded

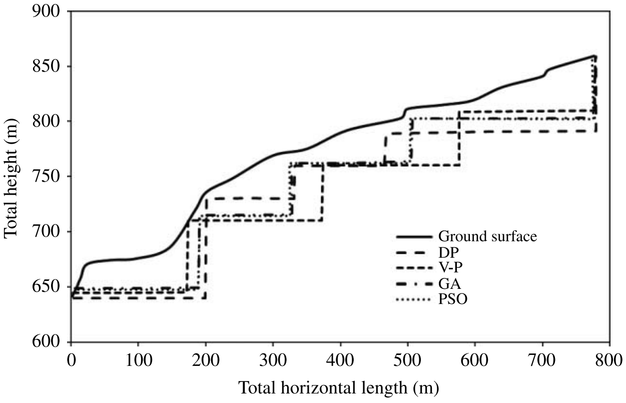

Figure 8 | Longitudinal profile of the successive cascades suggested by VP, DP and GA methods.

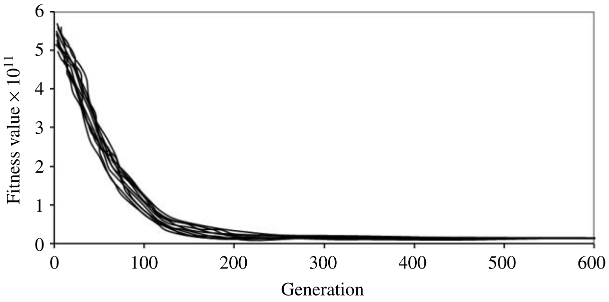

Figure 9 | Rate of convergence of the fitness function for 10 runs of the GA.

that the GA and PSO methods provide high accuracy and reliability for the design of cascade stilling basins.

Like the GA, PSO starts by initializing a population of random solutions and searches for optima by updating generations. However, PSO does not use any evolution operators. In PSO, each particle flies through the problem space following its own experience and the best experience attained by the swarm as a whole. In contrast to analytical or general heuristic methods, PSO is computationally efficient and has a great capability of escaping local optima. In addition, PSO has advantages over GA due to its easy implementation. It is difficult to say generally which approach performs better because each optimization method has its own characteristics with different operators. However, when the same population size and number of iterations are applied in the case study, the PSO tends to discover a better solution, with slower convergence than the GA approach. It was also noticed that, even though the cumulative computational time increases linearly with the number of generations for both PSO and GA, the computational time for GA is low compared to the PSO optimization algorithm. The higher computational time for

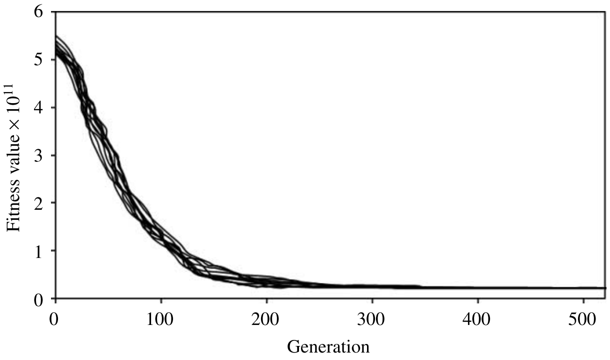

Figure 10 | Rate of convergence of the fitness function for 10 runs of the PSO.

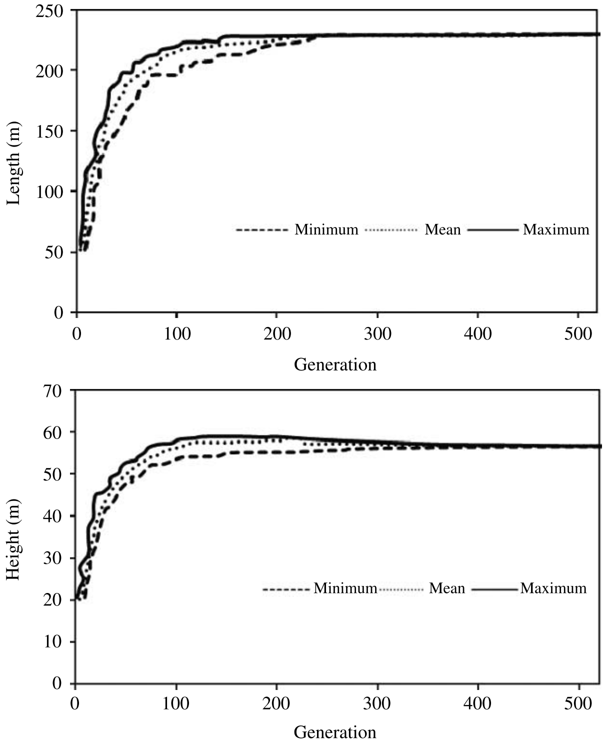

Figure 11 | Maximum, minimum and average values of first cascade's height and length over 10 runs.

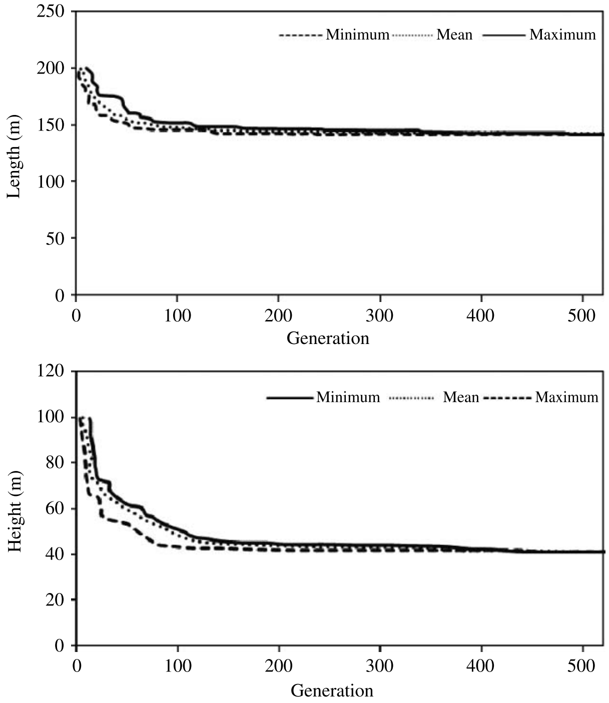

Figure 12 | Maximum, minimum and average values of second cascade's height and length over 10 runs.

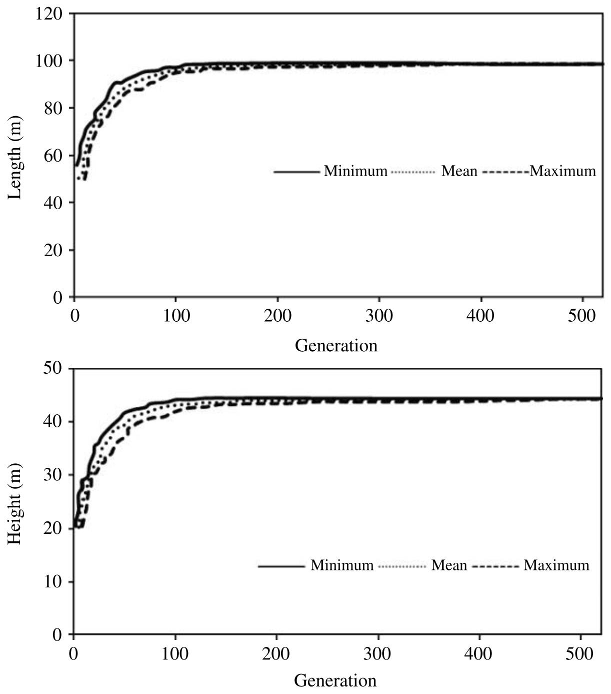

Figure 13 | Maximum, minimum and average values of third cascade's height and length over 10 runs.

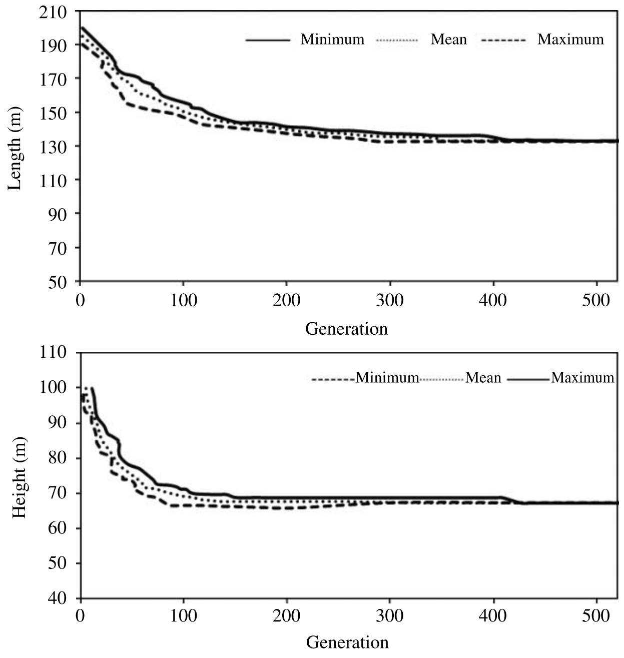

Figure 14 | Maximum, minimum and average values of fourth cascade's height and length over 10 runs.

Table 5 | Comparison of results for different objective functions and different schemes

Model	Objective function	CV (m^3)	EV (m^3)	Cost reduction compared with VP (%)
VP	–	525,000	3,540,000	–
DP	EV*	513,000	2,318,000	34.4
	EV + CV†	465,000	2,337,000	31.0
GA	EV	484,000	2,280,000	35.5
	EV + CV	440,400	2,301,000	32.5
PSO	EV	477,000	2,250,000	36.4
	EV + CV	429,300	2,272,000	33.5

*EV = Excavation Volume.
†CV = Concrete Volume.

PSO is due to the communication between the particles after each generation. However, PSO requires smaller population sizes and fewer generations to perform comparably with the GA.

CONCLUSIONS

In the case of high height dams, when the fall head is high, the flow velocity will increase and the Froude number at the chute toe will become very high so that the dimension of the stilling basin's walls will be large. The use of a series of spillways and stilling basins decreases the flow velocity, and thus the dimensions of the cascades and the overall costs of the dissipation system will be reduced. In this study, the GA and PSO formulations were developed to optimize the number and size of cascade spillways and basins. Both methods resulted in the same number of cascades whose specifications, i.e. the height of falls and the length of stilling basins, are, however, different. In this application, both approaches show similar evolution dynamics and optimal results.

The optimization results were compared with those of Vittal & Porey (1987) and showed that more significant cost savings in cascade stilling basin design could be obtained. Moreover, the results show that the proposed evolutionary techniques are computationally fast and quite amenable for optimal design of cascade stilling basins. Also, it was shown that GA and PSO are superior means of finding the optimal design than DP. The mathematical complexity in the DP approach is completely eliminated and the proposed models do not suffer from the problem of dimensionality. The major advantage of the GA and PSO models is that they can handle complex objective functions and constraints more efficiently. In addition, GA and PSO are very efficient to solve global optimization problems with continuous variables.

As an evolutionary optimization algorithm, the PSO requires a larger computation run time than a classical optimization technique, which still restricts its use in real-time applications. This work suggests some directions for future research. The application of GA and PSO to optimal design of cascade stilling basins, as any other constrained optimization problem, requires *a priori* defining a number of parameters such as population size, number of generations, crossover and mutation probabilities, and most important of all the constraint violation penalty parameter. While some general conceptual rules have been developed regarding the useful range of some of these parameters, limited research has been carried out concerning selection of the proper value of the penalty parameter, which could affect the quality of the final solution. The selection of this parameter is achieved at present by trial-and-error.

REFERENCES

Bakhtyar, R., Mousavi, S. J. & Afshar, A. 2007 Dynamic-programming approach to optimal design of cascade stilling basins. *J. Hydraul. Eng. ASCE* **133** (8), 949–954.

Carlisle, A. & Dozier, G. 2001 An off-the-shelf PSO. In: *Proceedings of the Particle Swarm Optimization Workshop, Indianapolis, Purdue school of engineering and technology*, IUPUI, IN. pp. 1–6.

Chau, K. W. 2006 Particle swarm optimization training algorithm for ANNs in stage prediction of Shing Mun River. *J. Hydrol.* **329** (3–4), 363–367.

Chau, K. W. 2007*a* Application of a PSO-based neural network in analysis of outcomes of construction claims. *Automat. Construct.* ***16*** *(5), 642–646.*

Chau, K. W. 2007*b* A split-step particle swarm optimization algorithm in river stage forecasting. *J. Hydrol.* ***346*** *(3–4), 131–135.*

Cheng, C. T., Wu, X. Y. & Chau, K. W. 2005 Multiple criteria rainfall-runoff model calibration using a parallel genetic algorithm in a cluster of computer. *Hydrol. Sci. J.* **50** (6), 1069–1087.

Cheng, C. T., Zhao, M. Y., Chau, K. W. & Wu, X. Y. 2006 Using genetic algorithm and TOPSIS for Xinanjiang model calibration with a single procedure. *J. Hydrol.* **316** (1–4), 129–140.

Goldberg, D. E. 1989 *Genetic Algorithms for Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, MA.

Holland, J. H. 1975 *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge, MA.

Kennedy, J. & Eberhart, R. C. 1995 Particle swarm optimization. In: *Proceedings of the IEEE International Joint Conference on Neural Networks (Perth, Australia)*, Vol. 4, IEEE Service Center, Piscataway, NJ. pp. 1942–1948.

Kennedy, J., Eberhart, R. C. & Shi, Y. 2001 *Swarm Intelligence*, Morgan Kaufmann, San Francisco. pp. 344–345.

Poggi, B. 1956 Lo scaricatore a scala di stramazzi: criteri di calcolo e rilievi sperimentali. *L'Energia electric* **33** (1), 33–40.

Savic, D. A. & Walters, G. A. 1997 Genetic algorithms for least-cost design of water distribution networks. *J. Water Res. Plan. Manage.* **123** (2), 67–77.

Shi, Y. & Eberhart, R. C. 1998*a* Parameter selection in particle swarm optimization. In: *Proceedings of the Seventh Annual Conference on Evolutionary Programming, New York, Lecture Notes in Computer Science 1447*. pp. 591–600. Springer-Verlag.

Shi, Y. & Eberhart, R. C. 1998*b* A modified particle swarm optimizer. In: *Proceedings of the IEEE Congress on Evolutionary Computation*, IEE Press, Piscataway, NJ. pp. 69–73.

USBR 1958 *Hydraulic Design of Stilling Basin and Bucket Energy Dissipaters, Engineering Monograph* No. 25. US Bureau of Reclamation, US Department of Interior, Bureau of Reclamation, Denver, CO.

Vittal, N. & Porey, P. D. 1987 Design of cascade stilling basins for high head dam spillways. *J. Hydraul. Div. ASCE* **113** (2), 225–237.

Wardlaw, R. & Sharif, M. 1999 Evaluation of genetic algorithms for optimal reservoir system operation. *J. Water Res. Plan. Manage.* **125** (1), 25–33.

Wu, C. L. & Chau, K. W. 2006 A flood forecasting neural network model with genetic algorithm. *Int. J. Environ. Pollut.* **28** (3–4), 261–273.

Yin, P.-Y. 2006 Particle swarm optimization for point pattern matching. *J. Visual Commun. Image Representation* **17** (1), 143–162.

First received 2 May 2008; accepted in revised form 28 August 2008.