Karbala International Journal of Modern Science

Volume 5 | Issue 4 Article 7

Design of Reinforced Concrete Deep Beams using Particle Swarm Optimization Technique

Qais F. Hasan
Northern Technical University, dr.qaishasan@gmail.com

Dler A. Al-Mamany
Northern Technical University, Dlieromer@yahoo.com

Omer K. Fayadh
Al-Nahrain University, omerkhalid79@yahoo.com

Recommended Citation
Hasan, Qais F.; Al-Mamany, Dler A.; and Fayadh, Omer K. (2019) "Design of Reinforced Concrete Deep Beams using Particle Swarm Optimization Technique," *Karbala International Journal of Modern Science*: Vol. 5 : Iss. 4 , Article 7.
Available at: https://doi.org/10.33640/2405-609X.1180

Design of Reinforced Concrete Deep Beams using Particle Swarm Optimization Technique

Abstract

Researches available in literature interrelating neural networks to civil engineering design problems, especially for beep beams, are very rare. Therefore, an optimization algorithm is developed and verified in this study and coded using MATLAB functions to determine the optimum cost design of reinforced concrete deep beams. ACI 318-14 code method is used benefiting from iterative particle swarm optimization technique due to its efficiency and reliability. Minimizing total cost is used as the objective function in terms of four decision variables. Self-adaptive penalty function technique is used to handle constraints for each of the 300 randomly selected particles, and in each of the 50 total iterations followed for each one of four suggested deep beam design case studies. Performing all iterations is used as a stopping criteria for the developed algorithm. Comparative studies are made to show the effect of concrete compressive strength, live load scheme, and length of deep beam, on the optimum total cost and the corresponding decision variables. Results presented in the form of graphs and tables show that the loading condition has a significant effect on the total cost of deep beams. The cost increase is accompanied by deep beam length increase, height increase, longitudinal reinforcement area increase and vertical shear reinforcement area decrease. The calculated optimum cost is noticed for beam DB1, which is 1255 US$, with 1.29 m beam height, 0.01445 m^2 vertical shear reinforcement, 0.00914 m^2 horizontal shear reinforcement and 0.00238 m^2 main longitudinal reinforcement. The results show a relatively less difference in total cost between all the four beams at 4 m length compared to 8 m length. Also, a relatively mild increase in total cost is happened for beams DB3 and DB4 as the height increases, especially above 1.7 m height. As the main longitudinal reinforcement increases, cost of DB4 is affected more significantly than others, and as the vertical shear reinforcement increases, DB4 curve shows a relatively low degradation in cost.

Keywords

Cost optimization, Reinforced concrete, Deep beam, Particle swarm, Penalty function.

Cover Page Footnote

I approve the manuscript PDF file sent to me with no corrections..Thanks

1. Introduction

The traditional design procedure of deep beams is an iterative process, and the designer's experience, sense, and skill play important roles in getting a compatible, rigid, and safe design [1]. This process is continued then terminated, only, when a design state that satisfies accepted criteria is reached. In most design cases, cross-sectional dimensions and material grades are first adopted based on common experience or practice, with the remaining challenge to design efficient and cost-effective structures without losing system integrity [2]. Therefore, design via optimization is another and more systematic method [3], which proceeds by analyzing a trial design to determine if it is the "best" or not. For the case of deep beams, best means the most efficient, reliable, durable, and cost-effective design state [4]. The designer needs to formulate the optimized design problem by defining a set of decision (design) variables, an objective function and constraint functions [5]. Particle Swarm Optimization (PSO) is a heuristic global neural network optimization method put forward originally in 1995 [6]. PSO can be applied easily and efficiently to solve complex design problems involving various objective functions [7−11].

The behavior of the biological bacterial colony is studied using the PSO algorithm [11]. Variation in the component of the bacteria algorithm is compared using the hybridization of an evolutionary algorithm. Applications of the bacteria algorithm are analyzed in various domains to show its response reality. The PSO algorithm was proposed [12] to solve problems of safety and real-time responses of intelligent vehicles. The objective function is defined concerning the intelligent vehicles driving characteristics, the distance between intelligent vehicles and obstacles and distance of intelligent vehicles and targets. The simultaneous results showed that the PSO method improved the perturbations of the vehicle planning path and real-time and reliability. An investigation using PSO algorithm is made for the tuning problem of digital proportional-integral-derivative (PID) parameters for a DC motor controlled via the controller area network (CAN) [13]. A sufficient condition is adopted to guarantee the stability of the time-varying delay system. An optimization method based on the particle swarm optimization (PSO) algorithm and the linear-quadratic-regulator (LQR) technique is proposed and verified. An optimal privacy preserving clustering on homomorphically encrypted data using K-means clustering and PSO algorithm

List of symbols and abbreviations

A_s	Main longitudinal reinforcement area (m^2)
A_v	Vertical shear reinforcement area (m^2)
A_{vh}	Horizontal shear reinforcement area (m^2)
b	Deep beam width (m)
c	Depth of neutral axis (m)
C_c	Cost of concrete ($)
C_f	Cost of framework ($)
C_s	Cost of steel reinforcement ($)
C_T	Total deep beam cost ($)
C_c^1	Plain concrete cost per unit volume ($/$m^3$)
C_f^1	Framework cost per unit area ($/$m^2$)
C_s^1	Longitudinal steel bar cost per unit mass ($/kg)
C_s^2	Shear stirrups cost per unit mass ($/kg)
d	Depth of bottom reinforcement (m)
$d(x)$	Distance value for a particle in the swarm
E_c	Concrete elastic modulus (MPa)
E_s	Steel elastic modulus (MPa)
F	Objective function

https://doi.org/10.33640/2405-609X.1180

f_y Yield strength of flexural bars (MPa)
f_{ys} Yield strength of shear bars (MPa)
f'_c Concrete compressive strength at 28 days (MPa)
gbest Global particle best position
h Deep beam height (m)
i Current particle
iw Individuality constant
I_{cr} Cracked moment of inertia (m^4)
I_e Effective moment of inertia (m^4)
I_g Gross moment of inertia (m^4)
jd Lever arm (m)
lbest Particle best position in the neighbors
L Overall length of deep beam (m)
L_n Clear span of deep beam (m)
M_{cr} Cracking moment (kN.m)
M_u Ultimate moment (kN.m)
n Modular ratio
pbest Particle's best position visited ever
p(x) Penalty value for a particle in the swarm
PSO Particle Swarm Optimization
P_u Ultimate concentrated load (kN)
s_h Spacing of horizontal shear reinforcement (m)
s_v Spacing of vertical stirrups (m)
sw Sociality weight
t Iteration or time step
U Random number between [0,1]
V_c Concrete shear capacity (kN)
V_s Stirrup shear capacity (kN)
V_u Ultimate shear capacity (kN)
V_{ij} Particles velocity in the swarm
w Inertia weight
w_{DL} Dead load (kN/m)
w_{LL} Live load (kN/m)
w_s Unit weight of longitudinal steel bar (kg/m^3)
w_v Unit weight of shear stirrups (kg/m^3)
w_{SDL} Superimposed load (kN/m)
w_u Ultimate load (kN/m)
x,y Decision variables
Δ_i Immediate mid-span deflection (m)
$(\Delta_i)_{DL}$ Immediate dead load deflection (m)
$(\Delta_i)_{LL}$ Immediate live load deflection (m)
Δ_{LT} Long term deflection (m)
ρ Longitudinal steel ratio
Φ Material reduction factor

is proposed [14]. Experimental analysis carried out on standard datasets show reduced quantization error while preserving the privacy of the original dataset.

The PSO algorithm was, also, used to optimize the cost of one-way concrete slabs [15] according to ACI 318-08 code. The objective was to minimize the total cost of four different slabs while satisfying all the design requirements using a multi-stage dynamic penalty implementation to solve the constrained design problem. A procedure was presented for the section design of FRP-reinforced concrete beams following the recommendations of ACI 440.1 R-06 using the PSO algorithm [16]. A rectangular cross-section beam was considered with two design variables, which are the width and height of the beam, as well as the number and diameter of reinforcing bars. A single objective cost optimization algorithm was presented in Ref. [17] for pre-stressed beams using a special differential evolutionary technique following the European building code. A hybrid PSO and Genetic algorithm were developed in Ref. [18] to solve force method-based simultaneous analysis and design problems for frame structures. In that work, comparisons were made for some design problems to demonstrate the algorithm's efficiency and superiority, especially for structures with a large number of redundant forces.

The conventional design of reinforced concrete deep beams leads to safe designs, but the economy of the design is very much linked to experience of the structural designer. Scarcity and the need for efficiency in today's competitive world have forced engineers to evince greater interest in economical and optimized designs by recommending the use of nowadays techniques and algorithms. In contrast, up to the knowledge of the present authors, researches available in the literature interrelating neural network optimizations algorithms to civil engineering design problems, especially for beep beams, are very rare. Therefore, it seems to be important to develop an algorithm like PSO which can efficiently and optimally solve various reinforced concrete deep beam design problems. Such an algorithm should produce the most cost-efficient design state that satisfies all the intended serviceability and safety conditions.

2. Design of deep beam sections

A procedure is developed, based on the ACI code provisions [19], and followed here to design deep beams for shear under uniformly distributed and concentrated loads with simply supported beam conditions. Traditionally, when a reinforced concrete deep beam philosophy is followed, the cross-sectional dimensions are initially estimated according to a standard ratio between overall depth of the beam, h, and the clear span between its supports' faces L_n, where $L_n/h \leq 4$. The uniformly distributed dead load w_{DL} can be computed as the superimposed dead load w_{SDL} plus the self-weight of the beam. The factored (ultimate) uniformly distributed load w_u can then be computed as the combination of the uniformly distributed dead and live loads. Also, the ultimate moment, M_u, due to both the distributed, w_u, and concentrated, P_u, ultimate loads can be calculated according to the boundary conditions for the simply supported deep beam and the distance between supports L (center-to-center). The critical section recommended to calculate the ultimate shear V_u is at a distance of 0.5h from the face of support, where h is the overall depth of the beam. The ultimate shear force due to w_u and P_u is:

$$V_u = \frac{w_u L}{2} + \frac{P_u}{2} - w_u(0.5\,h) \tag{1}$$

The ACI code recommends a shear limit which imposes a dimensional restriction to control cracking under service loads and to guard against diagonal compression failure in deep beams. If this limit is exceeded, an enlargement of the section is needed. In the current algorithm, the width of deep beam b is given and the enlargement is available only in beam depth h. Because of the fact that the ACI code [19] does not specify a simplified procedure for flexural analysis and design for deep beams, which follows rigorous nonlinear approach, therefore, simplified provisions are presented depending on the recommendations of CEB “Comité Européen de Béton” [20] and the flexural reinforcement area A_s can be calculated from applied ultimate moment M_u, yield strength of the flexural bars f_y, moment lever arm jd and a material reduction factor Φ;

$$A_s = \frac{M_u}{\Phi f_y\, jd} \tag{2}$$

knowing that A_s must not be less than the minimum flexural reinforcement area specified in CEB [20]. The least of the following expressions is recommended to calculate shear strength capacity V_c of the concrete section for deep beams from longitudinal reinforcement steel ratio ρ, concrete compressive strength at 28 days $f'c$ and effective depth of bottom reinforcement d [19];

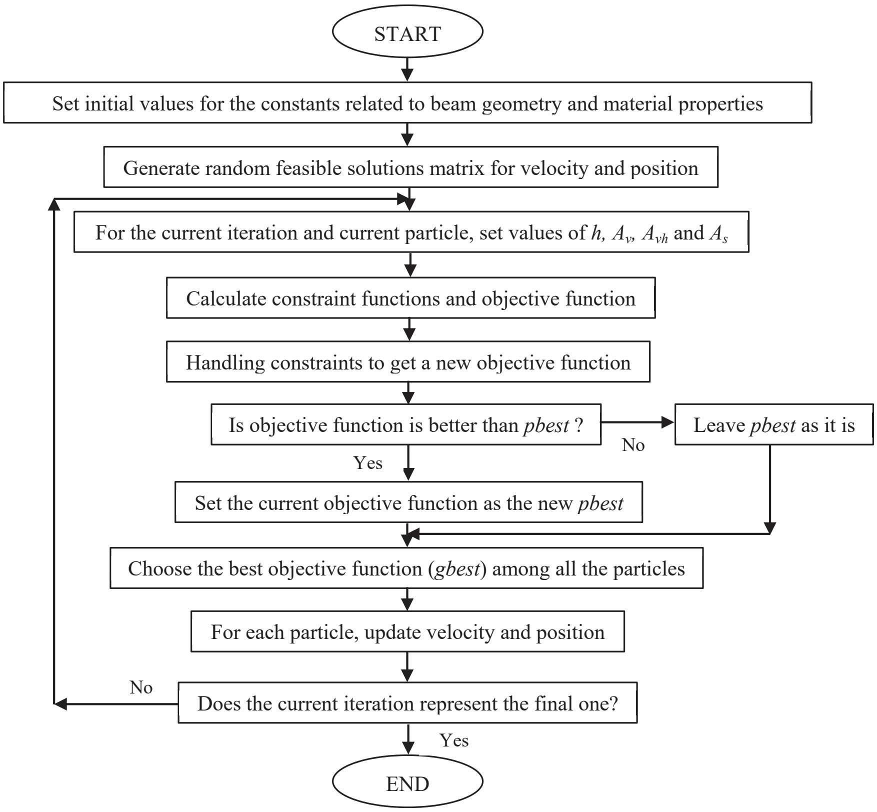


Fig. 1. Flowchart of the optimization process.

$$V_c = \left[0.16\sqrt{f'_c} + 17\rho \frac{V_u\, d}{M_u}\right] b\, d \tag{3}$$

$$V_c = \left(0.16\sqrt{f'_c} + 17\rho\right) b\, d \tag{4}$$

$$V_c = 2\sqrt{f'_c}\; b\, d \tag{5}$$

The requirements for deep beams stay that no shear reinforcement is needed if the ultimate shear force (V_u) is less than $0.5\Phi V_c$ [19], where Φ is a material reduction factor taken as 0.85. However, for practical reasons, minimum horizontal and vertical shear reinforcements must be provided. If the ultimate shear force is more than $0.5\Phi V_c$, then a shear reinforcement is required for the deep beam, and the shear carried by the stirrups, V_s, can be calculated. The ACI code methodology for calculating the required vertical and horizontal stirrups is followed by assuming initial values for the shear steel areas in both directions and increasing either one or both of them until the following basic design inequality is satisfied:

$$V_s \le \frac{A_v f_{ys}\, d}{s_v} + \frac{A_{vh} f_{ys}\, d}{s_h} \tag{6}$$

where f_{ys} is the yield strength of shear bars, A_v and A_{vh} are vertical and horizontal shear reinforcement areas, respectively, and s_v, and s_h are spacing values of the distributed vertical and horizontal shear reinforcements, respectively. Assuming that the applied moment exceeds the cracking moment and the occurred cracks will cause a reduction in stiffness, the cracked section moment of inertia I_{cr} is:

$$I_{cr} = \frac{bc^3}{3} + n A_s (d-c)^2 + n A_{vh}\left(\frac{h}{2} - c\right)^2 \tag{7}$$

where n is the modular ratio, and c represents the depth from the top beam face to the neutral axis, which is found by solving a quadratic formula representing moments of areas about the neutral axis;

$$n A_s (d-c) + n A_{vh}\left(\frac{h}{2} - c\right) = b\, c\, \frac{c}{2} \tag{8}$$

The effective moment of inertia, I_e, depends on the magnitude of the applied ultimate moment, M_u, and the cracking moment, M_{cr}, according to the following equation:

$$I_e = \left(\frac{M_{cr}}{M_u}\right)^3 I_g + \left[1 - \left(\frac{M_{cr}}{M_u}\right)^3\right] I_{cr} \tag{9}$$

where I_g is gross moment of inertia of the section. The immediate mid-span deflection due to dead and live loads is computed using the concrete elastic modulus, E_c, as in the following elastic formula:

$$\Delta_i = \frac{5\, w_u\, L^4}{384\, E_c\, I_e} + \frac{P_u\, L^3}{48\, E_c\, I_e} \tag{10}$$

According to ACI code limitations, the immediate deflection should not exceeds $L/360$. For long term deflections, separation should be made between immediate dead load deflection $(\Delta_i)_{DL}$ and immediate live load deflection $(\Delta_i)_{LL}$ caused by the dead, w_{DL}, and the live, w_{LL}, loads using a weighted average formula, as follows:

$$(\Delta_i)_{DL} = \frac{W_{DL}}{w_{DL} + w_{LL}}.\Delta_i \tag{11}$$

$$(\Delta_i)_{LL} = \frac{W_{LL}}{w_{DL} + w_{LL}}.\Delta_i \tag{12}$$

Finally, the long term deflection is calculated as follows:

$$\Delta_{LT} = (\Delta_i)_{LL} + 0.6\,\xi\,[(\Delta_i)_{DL} + (\%)(\Delta_i)_{LL}] \tag{13}$$

where (%) represents the percentage of the live load that can be considered sustained (typically set to 20%), and $\xi = 2$ for more than 5 years noting that ACI code recommends a value not exceeding $L/480$ for the long term deflection.

3. Particle swarm optimization algorithm

PSO is an iterative technique, where the potential solution called particles fly through the problem space by following the current optimum particles. Each particle keeps track of its coordinates in the problem space which are associated with the best solution achieved so far [6]. To remember the previous experience, each particle has a separate area of memory to store the best position visited in the search space. This value is called as *pbest*. Another best value that is tracked by the particle swarm optimizer is the best value obtained so far by any particle in the neighborhood of the particle, which is called *lbest*. When a particle takes all the population as its topological neighbors, the best value is a global best and is called as *gbest*. The particle swarm optimization concept consists of, at each time step, changing the velocity of each particle toward its *pbest* and *gbest*. After finding the best values, each particle updates its velocity and position according to the following equations:

$$\begin{aligned} V_{ij}^{(t)} &= w^{(t)} * V_{ij}^{(t-1)} + iw^{(t)} * U_{(0,1)} * (pbest_{ij}^{(t-1)} - X_{ij}^{(t-1)}) \\ &\quad + sw^{(t)} * U_{(0,1)} * (gbest_j^{(t-1)} - X_{ij}^{(t-1)}) \end{aligned} \tag{14}$$

$$X_{ij}^{(t)} = X_{ij}^{(t-1)} + V_{ij}^{(t)} \tag{15}$$

where;

$X_{ij}^{(t)}$: the *j*th component of the position of particle *i* at iteration or time step *t*,

$V_{ij}^{(t)}$: the *j*th component of the velocity of particle *i* at time step *t*,

$w^{(t)}$, $iw^{(t)}$, $sw^{(t)}$: inertia, individuality and sociality weights, respectively, at time step *t*,

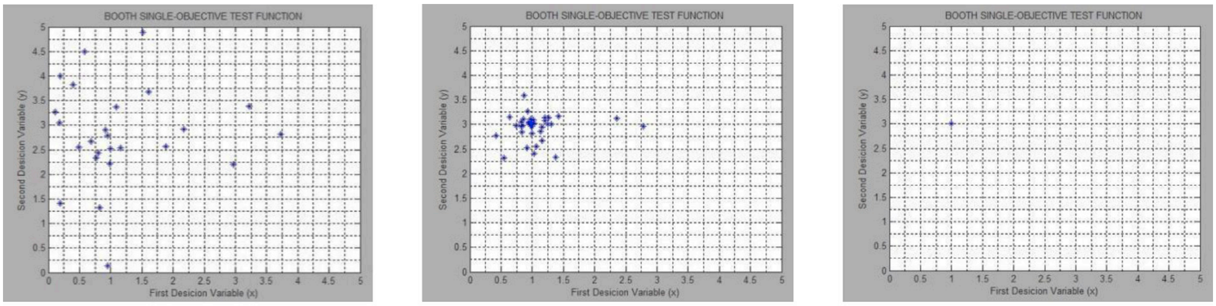
$U_{(0,1)}$: random number generated from a uniform distribution in the range [0,1],

$pbest_{ij}^{(t-1)}$, $gbest_j^{(t-1)}$: coordinate *j* of the best position found by particle *i* and by the whole swarm, respectively, up to time step *t-1*.

An annealing algorithm is used [21] to determine the value of $w^{(t)}$. A stopping criterion, represented by the maximum number of iterations, is used as a condition for the termination of the search process. The overall optimization PSO process followed in this study is abbreviated in Fig. 1.

4. Constraints handling technique

In realistic optimization problems, there are certain constraints imposed on the decision variables, this type is called a *constrained optimization problem* [21]. An excellent survey for the different constraints handling techniques is presented in Ref. [22]. In most of these techniques, the analyst resorts to converting the main objective function with its constraints to one or more unconstrained optimization problems [23]. In any constrained optimization problem, the search space can be divided into feasible and infeasible points or regions. At feasible points, all the constraints are met, while at infeasible points, at least one constraint is violated. The most common approach in this field is the use of the so-called “Penalty Function” approach which was originally proposed in 1943 by Courant [24]. The idea of this method is to transform a constrained optimization problem into unconstrained penalized one by adding or a certain value to the original objective function or subtracting another value from it, based on the amount of constraint



a) At the 20th iteration b) At the 60th iteration c) At the 100th iteration

Fig. 2. Particles' search for the optimum solution at different iterations for the Booth test function.

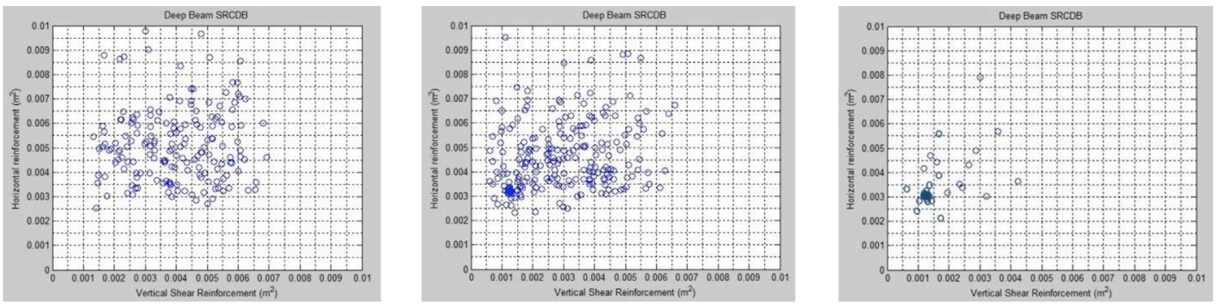
Table 1
Specifications and calculated results for verification the beam SRCDB.

f'_c	f_y	f_{ys}	E_s	h	L	L_n	P_u	Result in Ref. [31]		Calculated Result		Difference	
MPa	MPa	MPa	GPa	m	M	m	kN	A_v (m2)	A_h (m2)	A_v (m2)	A_h (m2)	A_v	A_h
55	500	500	200	0.4	1.8	1.6	450	0.00157	0.00262	0.00135	0.00310	14%	18%

violation present in a certain solution. In the literature, results obtained using dynamic penalty functions are always superior when compared to those obtained using static penalty functions [7,25]. One of the recently introduced dynamic penalty functions is the "Self-Adaptive Penalty Function" [26,27], where a new penalized objective function for the infeasible individuals can be calculated according to the following equation:

$$F(\vec{x}) = d(\vec{x}) + p(\vec{x}) \tag{16}$$

where $d(\vec{x})$ and $p(\vec{x})$ are distance and penalty values for each individual, respectively. The first part $(d(\vec{x}))$. $(d(\vec{x}))$ can be calculated in terms of the normalized constraint violation and the normalized objective function, while the second part represents a two-part penalty function added to the distance function. In this algorithm, besides the exact definition and formulation



a) At the 20th iteration b) At the 60th iteration c) At the 100th iteration

Fig. 3. Particles' search for the optimum solution at different iterations for verification beam SRCDB.

Table 2
Common parameters for the optimized deep beams.

Parameter	C_s^1	C_s^2	C_f^1	w_s	w_v	L	L_n	b	f_y	f_{ys}	w_{SDL}
	\$/kg	\$/kg	\$/m^2	kg/m^3	kg/m^3	m	m	m	MPa	MPa	kN/m
All the four beams	1.0	1.0	50	7850	7850	5	4.6	0.4	480	380	60

Table 3
Different parameters for the optimized deep beams.

Beam	w_{LL}	P_u	f'_c	C_c^l
	kN/m	kN	MPa	\$/m^3
DB1	0	0	25	100
DB2	0	0	50	140
DB3	50	0	25	100
DB4	0	80	25	100

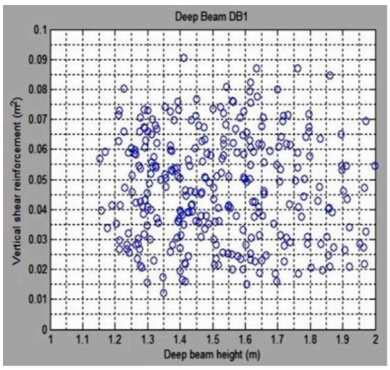
Table 4
Optimum cost and the corresponding decision variables for the optimized beams.

Beam	Optimum cost	h	A_v	A_{vh}	A_s
	US\$	m	m^2	m^2	m^2
DB1	1255	1.29	0.01445	0.00914	0.00238
DB2	1380	1.25	0.01358	0.00782	0.00377
DB3	1641	1.49	0.01972	0.01089	0.00394
DB4	1602	1.45	0.01633	0.00819	0.00527

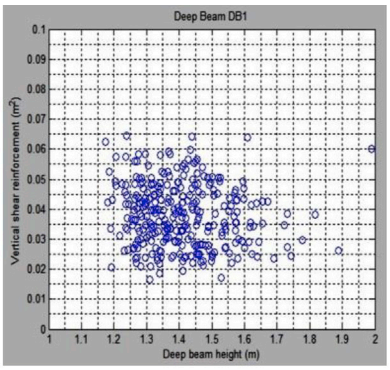
of the deep beam design problem, the compatibility and agreement of constraints with the real problem are critical since otherwise, the solution might be infeasible. Thus, the problem needs to be reduced to a function of the design variables which has to be optimized, plus a number of constraints that limit the feasible region of the search space. In summary, a well-posed optimization problem requires a good definition of the objective function, the search space and the constraints that define the feasible part of the search space.

5. Optimum cost design of deep beams

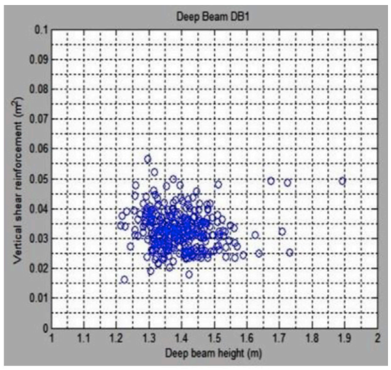
Design guidelines for the different concrete structures including deep beams, available in all the codes and most of the literature recommendations, are concerned with the satisfaction of many strength and serviceability constraints. Among the infinite design cases which may meet those requirements, it is worthwhile seeking the designs that result in the minimal cost of the structure [28,29]. It is always possible to construct a computer program to design reinforced concrete deep beams according to the standard procedure detailed in the literature so that numerous design states can be



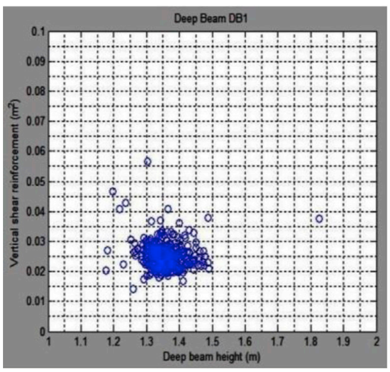
a) at 5th iteration



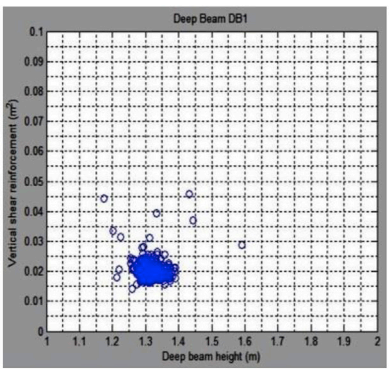
b) at 10th iteration



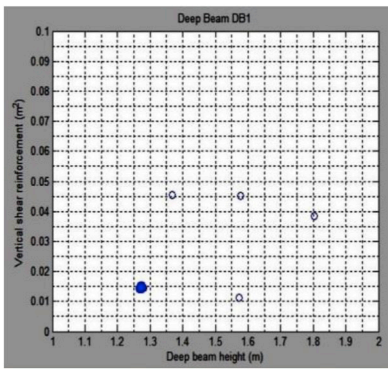
c) at 15th iteration



d) at 20th iteration



e) at 25th iteration



f) at 35th iteration

Fig. 4. Particles' search for the optimum solution at different iterations for beam DB1.

examined and the case corresponding to the minimum cost is chosen. However, every run of the program gives a different and independent design state, thus neglecting useful information gained from previous experiences. In addition, this procedure is bound to be tedious and time-consuming. Therefore, the importance of developing an algorithm to design a least-cost reinforced concrete deep beam, subject to strength and serviceability constraints, is higher than ever taking into consideration that such an algorithm is not yet available in the literature. The present authors undertake the work of developing this needed algorithm in this paper. The problem is represented by developing an objective function to compute the cost of a deep beam, which is, then minimized to get the optimal design including section dimensions and reinforcements represented by four decision variables (deep beam height h, vertical shear reinforcement area A_v, horizontal shear reinforcement area A_{vh} and main longitudinal reinforcement area A_s). In a reinforced concrete deep beam design, at least three different cost items should be considered in optimization: cost of the concrete, C_c, cost of the steel, C_s, and cost of the framework, C_f. The total cost function C_T, which implicitly includes the cost of materials, labor, and transportation, can be defined as:

$$C_T = C_c + C_s + C_f \tag{17}$$

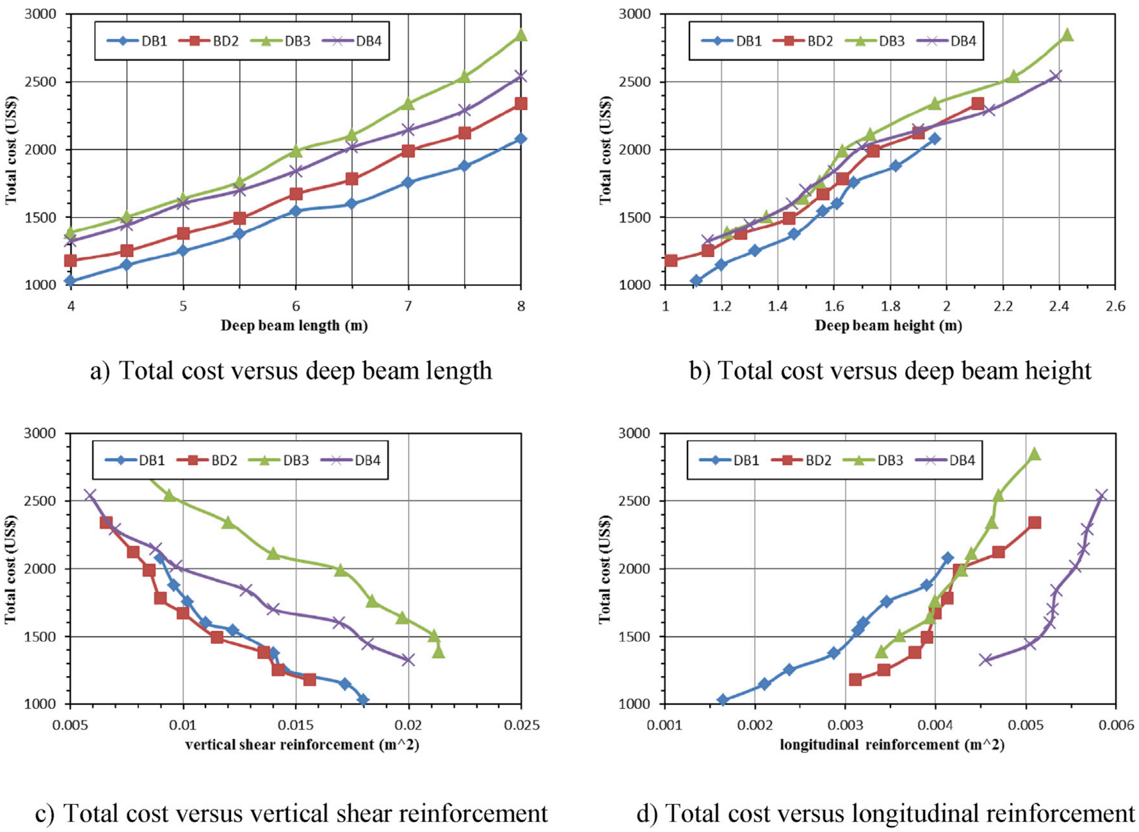
The concrete cost, C_c, per deep beam can be calculated by the equation:

$$C_c = L\,b\,h\,C_c^1 \tag{18}$$

where L, b, h, and C_c^1 are the length, the width, the height of the deep beam, and the plain concrete cost per unit volume, respectively. The steel reinforcement cost, C_s, per deep beam is computed from the sum of the costs of the main and shear reinforcements, as follows:

$$C_s = w_s\,L\,A_s\,C_s^1 + C_s^2\,w_v\,[A_v(2h+2b+0.1)+A_{vh}(2b+2L+0.1)] \tag{19}$$

where C_s^1, C_s^2, w_s, *and* w_v are costs of longitudinal steel bars per unit weight, cost of shear stirrups per unit weight, unit weight of longitudinal steel bars and



a) Total cost versus deep beam length

b) Total cost versus deep beam height

c) Total cost versus vertical shear reinforcement

d) Total cost versus longitudinal reinforcement

Fig. 5. Optimum total cost versus deep beam length, height, vertical shear reinforcement, and longitudinal reinforcement, for the four case studies.

unit weight of shear stirrups, respectively. The framework cost C_f per deep beam can be calculated according to the cost of framework per unit area C_f^1 by the equation:

$$C_f = C_f^1 \ L^*(b + 2h) \tag{20}$$

6. Results and discussion

6.1. Algorithm verification

In order to check the validity and efficiency of the proposed algorithm and the corresponding MATLAB code, the convergence aspect of the proposed algorithm is illustrated in a well-known optimization test problem named "Booth Test Function" [30]. This test problem uses the minimization of the objective function:

$$F(x,y) = (x + 2y - 7)^2 + (2x + y - 5)^2 \tag{21}$$

where; x and y are the decision variables. The standard answer for this test problems gives $x = 1$ and $y = 3$. The solution was calculated by the current algorithm, using an input data set of 40 particles operating through 100 iterations. The progressive convergence illustrated in Fig. 2 shows a perfect agreement with the standard answer available in the literature.

Another verification is made through testing a reinforced concrete deep beam test problem named SRCDB [31], which is within the scope of the present work and designed in the original work [31] using the automated finite elements cost optimization method with two decision variables (vertical and horizontal reinforcements). The beam is a simply supported one under four points loading designed to withstand a total static load, P_u, of 450 kN, as given in Table 1. The Convergence of the calculated results from 180 particles for 100 iterations, shown in Fig. 3 and presented in Table 1, indicates the reliability of the current algorithm in searching for the optimized design state that agrees well with that given in the literature, where the difference in the calculated vertical and horizontal reinforcement areas is ranging between 14% and 18%. This means that the present PSO algorithm and the MATLAB code give a realistic determination of the optimized design of reinforced concrete deep beams.

6.2. Deep beam case studies

In this work, a particle swarm optimization algorithm for the design of reinforced concrete deep beams is developed, coded using MATLAB functions, verified and used to work out the designs of four simply supported beam case studies. These optimized beams are divided into two groups. The first group consists of two deep beams DB1 and DB2, which have the same loading conditions and data given in Table 2. The comparative variable for these two beams is the concrete compressive strength *f 'c* and subsequently its cost, as given in Table 3. The beam DB1 has an *f 'c* of 25 MPa while DB2 has an *f 'c* of 50 MPa. The second group consists of two deep beams also, DB3 and DB4, which have the same *f 'c* of 25 MPa and the same data given in Table 2, but the difference is in the applied loading conditions. The beam DB3 has a uniformly distributed ultimate live load of 50 kN/m, while DB4 has a concentrated ultimate live load of 80 kN at the beam center, as shown in Table 3. Minimizing the total cost function for each of the four deep beams gives optimized values of the four design variables (h, A_v, A_{vh} and A_s), taking into consideration code specifications and limitations. A total swarm size of 300 particles with 50 total number of iterations, are used for each case study. Performing all iterations is used as a stopping criterion for the developed algorithm. The positions of those 300 particles at six selected iterations (5th, 10th, 15th, 20th, 25th, and 35th iterations) for DB1 is shown in Fig. 4 for the search space of two decision variables (h versus A_v). This figure verifies the developed algorithm by showing the swarm philosophy in the progressive searching for the optimum objective function represented in minimizing the total cost of the designed deep beam. The 35th iteration shows that most of the individuals reach a stable optimum solution, where the cost of the deep beam is about 1250 US$, giving h about 1.27 m and A_v about 0.014 m^2.

Summarized average results, for running the developed MATLAB code 10 times for each one of the four case studies, are shown in Table 4. Testing beam DB1, which is assumed to have no acting live loads and an *f 'c* of 25 MPa, as shown in Table 3, gives 1255 US$ optimum total cost. Testing beam DB2, which is assumed, also, to has no acting live loads but with an *f 'c* of 50 MPa, gives 1380 US$ optimum total cost which is about 10% more than that for DB1. This difference is due to the increment in concrete cost implied by the strength increment, while the h value for DB2 is 1.25 m which is about 3% less than that of the DB1 beam which was 1.29 m. Testing beam DB3, which is assumed to have 50 kN/m uniformly distributed ultimate live load gives a noticeably higher total cost of 1641 US$ compared to all the other cases including beam DB4 which is assumed to have an 80 kN concentrated ultimate acting live load at the mid-span corresponding to a total cost of 1602 US$.

Therefore, the loading condition seems to be a significant factor in the optimal cost calculations of deep beams.

The optimum total cost versus deep beam length curves for the four case studies are shown in Fig. 5a. These curves show that as the deep beam length increases the corresponding optimal total cost increases. It also shows a relatively less difference in total cost between the four case studies at 4 m beam length compared to that at 8 m beam length. Total optimal cost versus deep beam height curves, shown in Fig. 5b, also show a direct relationship for all the case studies. They show a relatively mild increase in total cost for both DB3 and DB4 as the beam height increases, especially above 1.7 m height. Total optimal cost versus vertical shear reinforcement curves, shown in Fig. 5c, show that as the vertical shear reinforcement decreases, which corresponds to the height increase, the total cost increases. This is due to the fact that as the height increases, the shear strength contribution by concrete section increases at the expense of vertical shear reinforcement. The DB4 curve shows a relatively low degradation in total cost, represented in its mild response, as the vertical shear reinforcement increases. The increments in optimal total cost as the main longitudinal reinforcement increases are shown by the curves in Fig. 5d. The cost corresponding to beam DB4 is affected more significantly than others, as represented by the steep curve, while the cost corresponding to beam DB1 is affected to the lesser extent as illustrated by the shallow curve.

7. Conclusions and recommendations

A PSO algorithm implemented through a MATLAB code is presented in the present work to design cost-effective reinforced concrete deep beams that satisfy all serviceability and safety conditions with four decision variables. Results gained from the verification and the four deep beam case studies indicate that the algorithm can perform well in optimal cost deep beam design. A comparative study is made for two different concrete compressive strengths and their corresponding costs. Another comparative study is made for two different live load conditions to show the effect of these parameters on the optimized total cost function. It is concluded that the loading condition has a significant effect on the cost of deep beams, and as the length increases, total cost, beam height and longitudinal reinforcement area also increase, but the vertical shear reinforcement area decreases. This decrease is due to the increase in concrete section contribution in resisting shear due to the increase in deep beam height. As future works, extending this study to a multi-objective PSO algorithm, optimizing deep beams using a strut and tie model and including modern strengthening techniques of deep beams, are suggested.

Consent for publication

I understand that the information will be published without my/my child or ward's/my relative's (circle as appropriate) name attached, but that full anonymity cannot be guaranteed. I understand that the text and any pictures published in the article will be freely available on the internet and may be seen by the general public. The pictures and text may also appear on other websites or in print, may be translated into other languages or used for commercial purposes. I have been offered the opportunity to read the article.

Declaration of Competing Interest

The authors report no conflicts of interest. The authors alone are responsible for the content and writing of this article.

References

[1] ACI-ASCE Committee 445, Recent approaches to shear design of structural concrete, J. Struct. Eng. 124 (12) (1998) 1375—1417.
[2] M. Saka, O. Hasancebi, Z. Geem, Metaheuristics in structural optimization and discussions on harmony search algorithm, J. Swarm Evol. Comput. 28 (1) (2016) 88—97.
[3] N. Lagaros, A general purpose real-world structural design optimization computing platform, J. Struct Multidiscip Optim 49 (6) (2014) 1047—1066.
[4] K. Sarma, H. Adeli, Cost optimization of concrete structures, J. Struct. Eng. 124 (5) (1998) 570—578.
[5] J. Arora, Introduction to Optimum Design, fourth ed., Elsevier, 2017.
[6] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Proceedings of the Sixth International Symposium on Micro Machine and Human Science (MHS ’95), Ieee, 1995.
[7] E. Parsopoulos, N. Verahatis, Unified particle swarm optimization for solving constrained engineering optimization problems, in: Proceedings of First International Conference (ICNC 2005), Part III, 2005. Changsha, China.
[8] R. Poli, J. Kennedy, T. Blackwell, Particle swarm optimization: an overview, J. Swarm Intell. 1 (1) (2007) 33—57.
[9] O. Hasancebi, T. Dumlupinar, A neural network approach for approximate force response analyses of a bridge population, J. Neural Comput Appl 22 (1) (2011) 755—769.

[10] S. Azad, O. Hasancebi, Structural optimization using big bang-big crunch algorithm: a review, Int.J. Optim. Civil Eng. 6 (3) (2016) 433−445.
[11] B. Rani, C. Kumar, A comprehensive review on bacteria foraging optimization technique, in: S. Dehuri, A. Jagadev, M. Panda (Eds.), Multi-objective Swarm Intelligence, Theoretical Advances and Applications, Studies in Computational Intelligence, vol. 592, Springer, Berlin, Heidelberg, 2015.
[12] H. Graining, F. Weiping, W. Wen, The study of intelligent vehicle navigation path based on behavior coordination of particle swarm, J. Comput Intell. Neurosci 2016 (2016) 1−10.
[13] Z. Qi, Q. Shi, H. Zhang, Tuning of digital PID controllers using particle swarm optimization algorithm for a CAN-based DC motor subject to stochastic delays, IEEE Trans. Ind. Electron. 1 (1) (2019).
[14] R. Joannah, C. Kumar, A. Amir, An optimized privacy preserving K-means clustering using particle swarm optimization, J. Inf. Assur. Secur. 12 (5) (2017) 162−169.
[15] B. Nedushan, H. Varaee, Minimum cost design of concrete slabs using particle swarm optimization with time varying acceleration coefficients, J. World Appl. Sci. 13 (12) (2011) 2484−2494.
[16] S. Innocente, L. Torres, X. Cahis, G. Barbeta, A. Catalan, Optimal flexural design of FRP-reinforced concrete beams using a particle swarm optimizer, in: FRPRCS-8 Conference, University of Patras, Greece, 2007.
[17] G. Quaranta, A. Fiore, G. Marano, Optimal design of prestressed concrete beams using constrained differential evolution algorithm, J. Struct Multidiscip Optim 49 (3) (2014) 441−453.
[18] A. Kaveh, S. Malakouti Rad, Hybrid genetic algorithm and particle swarm optimization for the force method-based simultaneous analysis and design, Iran. J. Sci. Technol. Trans. B− Eng. 34 (B1) (2010) 15−34.
[19] American Concrete Institute Committee 318, Building Code Requirements for Structural Concrete (ACI318M.14) and Commentary, Farmington Hills, Michigan, USA, 2014.
[20] Comité Européen de Béton—Fédération Internationale de la Précontrainte, Model Code for Concrete Structures, English Edition, Cement and Concrete Association, London, 1987.
[21] Z. Michalewicz, M. Schoenauer, Evolutionary algorithms for constrained parameter optimization problems, J. Evol. Comput. 4 (1) (1996) 1−32.
[22] A. Coello Coello, Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art, J. Computer Methods Appl. Mech. Eng. 191 (1) (2002) 1245−1287.
[23] I. Saleh, U. Ozkaya, Q. Hasan, Improving multiobjective particle swarm optimization method, in: Proceedings of the 3rd International Conference, NTICT 2018, Baghdad, Iraq, 2018.
[24] R. Courant, Variational methods for the solution of problems of equilibrium and vibration, Bull. Am. Math. Soc. 49 (1943) 1−23.
[25] A. Joines, R. Houck, On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's, in: Proceedings of the First IEEE International Conference on Computational Intelligence, 1996. Washington, DC, USA.
[26] W. Berhe, Penalty function methods using matrix laboratory (MATLAB), Afr. J. Math. Comput. Sci. Res. 5 (13) (2012) 209−246.
[27] D. Wang, G. Du, A self-adaptive penalty function based genetic algorithm for value-bilevel programming problem, Int. J. Comput. Sci. Eng. 3 (2) (2014) 136−146.
[28] A. Kaveh, A. Abadi, Cost optimization of reinforced concrete one-way ribbed slabs using harmony search algorithm, Arabian J. Sci. Eng. 36 (7) (2011) 1179−1187.
[29] K. Alankar, S. Chaudhary, Cost optimization of composite beams using genetic algorithm and artificial neural network, in: International Conference on Computer Technology and Science (ICCTS 2012), 2012. New Delhi, India.
[30] M. Jamil, X. Yang, A literature survey of benchmark functions for global optimization problems, Int. J. Math. Model. Numer. Optim. 4 (2) (2013) 150−194.
[31] S. Shahidul Islam, A. Khennane, An Optimized Nonlinear Finite Element Design for Continuous RC Deep Beams, in: 7th International Structural Engineering and Construction Conference, 2013. Honolulu, Hawaii.