A COMBINATION OF PARTICLE SWARM OPTIMIZATION AND MULTI-CRITERION DECISION-MAKING FOR OPTIMUM DESIGN OF REINFORCED CONCRETE FRAMES

M.J. Esfandiary[*, †], S. Sheikholarefin and H.A. Rahimi Bondarabadi
Department of Civil Engineering, Yazd University, Yazd, Iran

ABSTRACT

Structural design optimization usually deals with multiple conflicting objectives to obtain the minimum construction cost, minimum weight, and maximum safety of the final design. Therefore, finding the optimum design is hard and time-consuming for such problems. In this paper, we borrow the basic concept of multi-criterion decision-making and combine it with Particle Swarm Optimization (PSO) to develop an algorithm for accelerating convergence toward the optimum solution in structural multi-objective optimization scenarios. The effectiveness of the proposed algorithm was illustrated in some benchmark reinforced concrete (RC) optimization problems. The main goal was to minimize the cost or weight of structures while satisfying all design requirements imposed by design codes. The results confirm the ability of the proposed algorithm to efficiently find optimal solutions for structural optimization problems.

1. INTRODUCTION

In the last few decades, a number of optimization techniques have been developed and used in structural optimization problems where the main objectives are to evaluate the merit of a design such as minimum construction cost, minimum life-cycle cost, and minimum weight, as well as maximum stiffness [1]. Because these are usually conflicting goals, multi-objective optimization techniques are a highly valuable tool for handling structural optimization problems. Furthermore, the RC frames optimum design is known as benchmark

---

[*]Corresponding author: Department Of Civil Engineering, Yazd University, Yazd, Iran
[†]E-mail address: mohesfandiari@yahoo.com (M.J. Esfandiary)

examples in the field of difficult and time-consuming optimization problems due to the presence of many design variables, large size of the search space, and many constraints [2]. More precisely, a reinforced concrete member can be designed with a semi-infinite set of member dimensions and different arrangements of reinforcing bars. Besides, in optimization of reinforce concrete structures the cost of different materials (i.e. steel, concrete, and formwork), which are closely tied together, should be considered.

Two different research fields exist for solving multi-objective optimization problems (MOP): Multiple Criteria Decision Making (MCDM) [3–5] and Evolutionary Multi-objective Optimization (EMO) [6,7]. Although they address similar problems as emphasized, they have different research goals [8]. MCDM supports a human Decision Maker (DM) in identifying the most preferred solution. In other words, a DM checks the results in every iteration and indicates what kind of changes in the design variables would lead to a more preferred solution. In addition, the DM can ignore unfeasible designs or ones which obviously cannot result in better optimal solutions before the calculation process. Consequently, the computational resources available are not wasted since only such Pareto optimal solutions which are interesting to the DM are generated. This method has a strong disadvantage in that a human DM must be available and willing to actively participate in the solution process and direct it according to the preferences.

On the other hand, EMO works with a population of individuals and attempts to find a set of non-dominated solutions near the Pareto optimal front. Typically, EMO algorithms explore the design space thoroughly and do not involve any preference information [9]. Many researchers have shown that EMOs perform well for global searching due to their capability of exploring and finding promising regions in the search space, but they take a relatively long time to converge to an optimum solution [6,7,10,11]. However, EMOs have been widely utilized to solve structural optimization problems. Rajeev and Krishnamoorthy [12] used a simple Genetic-Algorithm (GA) to optimize a 2D RC frame. Aspects such as detailing and placing of reinforcements in beams and columns and other issues related to construction were considered in their study. Govindaraj and Ramasmy [13,14] studied optimization of continuous beams and also 2D and 3D RC frames using Genetic-Algorithm (GA). The detailing of reinforcements in the beam members was carried out as a sub-level optimization problem. This reduced the size of the optimization problem and saved computational time. Kwak and Kim [15,16] developed an improved optimum design method for reinforced concrete (RC) frames using an integrated GA with a direct search method. The method proposed in their research used a predetermined section database (DB) when determining trial sections for the next iteration. Kaveh and Sabzi [17] applied heuristic big bang-big crunch (HBB-BC) and a heuristic particle swarm ant colony optimization (HPSACO) to minimize the cost of 2D RC frames while Kaveh and Behnam [18] tried the charged system search (CSS) algorithm for the optimization of 3D RC frames. Gholizadeh and Aligholizadeh [19] employed the bat algorithm (BA) to optimize 2D RC frames and compared the results of BA with those of other meta-heuristics. Gheyratmand, Gholizadeh, and Vababzadeh [20] proposed an improved artificial bee colony algorithm to optimize RC frames. They also compared the result with other algorithms.

In this article, an algorithm combining the PSO algorithm with the basic concept of multi-criterion decision-making, referred to as DMPSO, is proposed for the optimization of engineering structures. Enhancements are also proposed for the algorithm for constrained

structural optimization with a new implementation of a constraint handling tool. Also the modular sizes of members, standard reinforcement bar diameters, spacing requirements of reinforcing bars, architectural requirements and other practical requirements in addition to relevant provisions are considered to obtain directly constructible designs without any further modifications. Furthermore, three 2D RC benchmark design examples were tested using DMPSO. The numerical results show that the proposed DMPSO algorithm performed very well in terms of convergence speed and final results achieved.

2. FORMULATION OF THE STRUCTURAL OPTIMIZATION PROBLEM FOR REINFORCED CONCRETE FRAMES

A general structural optimization problem can be mathematically stated as follows:

$$\begin{cases} \min\limits_{x \in \mathbb{R}^n} f(x) & x = [x_1, \dots, x_n]^T \\ g_k(x) \leq 0, & k = 1, \dots, m \\ \quad x^L \leq x \leq x^U \end{cases} \tag{1}$$

where x is a vector of n design variables, $f(x)\colon \mathbb{R}^n \to \mathbb{R}$ is the objective function which returns a scalar value to be minimized (usually the cost or the weight of the structure), the vector function $g(x)\colon \mathbb{R}^n \to \mathbb{R}^m$ returns a vector of length m containing the values of the inequality constraints evaluated at x, and x^L, x^U are two vectors of length n containing the lower and upper bounds of the design variables, respectively. The above mathematical formulation contains only inequality constraints, as equality constraints are usually not found in the case of structural optimization.

A typical constraint k in structural optimization has the form:

$$g_k(x) = |q_k(x)| - q_{allow,k} \tag{2}$$

where $q_k(x)$ is a response measure for design x and $q_{allow,k}$ is its maximum allowable absolute value.

2.1 Objective function

In structural optimization problems, the objective function is generally described as the weight or total cost of structure. For the optimum design of reinforced concrete (RC) structures, the cost of structure is more convenient as an objective function, because concrete structures involve different materials and in reality, the minimum weight design may not be the minimum cost design for especially concrete structures. Considering the fact that the unit costs of different materials involved in concrete construction influence the total cost of the concrete structures, it is better to formulate the problem in terms of the total cost, which includes the costs of concrete, steel and formwork [21]. However, as C_c, C_f and C_s are user-defined variables, when the optimum weight is needed, the objective function can be easily modified by considering different values for these variables.

Considering the cost of the structure, the objective function becomes

$$f_{cost} = Cost_c + Cost_s + Cost_f \tag{3}$$

where $Cost_c$, $Cost_s$, and $Cost_f$ are the cost of concrete, the cost of reinforcing bars, and the cost of formwork (includes labor and placement), respectively.

The costs of components, $Cost_c$, $Cost_s$, and $Cost_f$, are calculated by the following equations:

$$Cost_c = C_C \left(\sum_{i=1}^{N_{col}} b_i . d_i . L_{n \; column \; ,i} + \sum_{j=1}^{N_{beam}} b_{w \, j} . h_j . L_{beam \; j} \right) \tag{4}$$

$$Cost_s = C_s . \gamma_s . \left(\sum_{i=1}^{N_{col}} \sum_{j=1}^{N_{bar \, ,i}} A_{St \, j} . L_{bar \; j} + \sum_{i=1}^{N_{col}} \sum_{k=1}^{N_{tie,i}} A_{Sh \; k} . L_{tie \; k} + \sum_{m=1}^{N_{beam}} \sum_{l=1}^{N_{bar \, ,m}} A_{st} . L_{bar \; l} \right.$$
$$\left. + \sum_{m=1}^{N_{col}} \sum_{n=1}^{N_{tie \, ,m}} A_{Sh \; n} . L_{tie \; n} \right) \tag{5}$$

$$Cost_f = C_f \left(\sum_{i=1}^{N_{col}} \left[2(b_i + d_i) . L_{n \; column \; ,i} \right] + \sum_{j=1}^{N_{beam}} \left[\left(b_{w \, j} + 2h_j \right) . L_{beam \; j} \right] - \sum_{k=1}^{N_{col}} b_k . d_k \right) \tag{6}$$

where N_{col}, N_{beam}, b, d, b_w, h, L, and L_n are the number of column members, the number of beam members, the width of column section, the depth of column section, the width of beam section, the height of beam section, the length of the members, and the length of clear span measured face-to-face to the supports, respectively; C_c, C_f and C_s are unit cost of the concrete, the formwork and the steel, respectively; A_{St}, L_{bar}, and N_{bar} are the Area, the length and the number of longitudinal reinforcement bars placed in the member while A_{Sh}, L_{tie}, and N_{tie} are the area, the length and the number of shear reinforcement bars (ties) used in the member respectively. γ_s is the density of steel reinforcements (kg/m3).

It should be noted that the unit costs are different from time to time and also from country to country. So, they cannot be fixed and need to be updated for every design problems. However, when C_c, C_f and C_s are selected equal to concrete density, one, and zero, respectively, the above equations will consider the minimum weight of the structure as the objective function.

2.2 Design variables

A semi-infinite set of member sizes and steel reinforcement arrangements can be considered for RC structure elements. In that case, as the dimensions of the design space are very large, the computational burden of the optimization process increases [17]. Consequently, in the studies available in the literature, a countable number of cross-sections have been employed in order to reduce the dimensions of design space as well as the computational cost. However, while the DMPSO performs well in terms of convergence speed and does not waste the computational resources, no restrictions were applied to the size of sections and arrangement of steel reinforcements. More precisely, the algorithm can use all of the possible sections in the common practical range to find the best optimum solution for the

problem. The only constraints considered for the sections were those derived from the design provisions of ACI 318-11 [22]. Eighteen design variables were defined for each span of a beam which can enumerate the width and height of a cross section, number and diameter of top and bottom continuous bars, number and diameter of top and bottom additional bars at left support, number and diameter of top and bottom additional bars at midspan, number and diameter of top and bottom additional bars at right support. The design variables for the column section were selected as the dimensions of columns in x and y directions, the diameter of reinforcement bars at the cross-section of column and numbers of reinforcement bars in both sides of the column.

It is important to emphasize that all the rebars were not considered continuously the entire length of the beams. The algorithm calculated the needed bars for each part and placed it in the right position. So, the development length of reinforcement, location and length of lap splices, and length for bars were calculated exactly according to ACI 318-11 [22]. According to the code splices should, if possible, be located away from points of maximum tensile stress; so, in the top of the beams the splices are located at the middle of the beam's length and the bottom bars are lapped at a third of the beam's lengths. A typical beam and a typical column with the considered splices details are represented in Fig. 1.

Some of other considerations, according to the ACI 318-11 [22] code and other construction requirements, are enumerated as follows:
- The lower bound, upper bound, and increments of cross sectional dimensions were considered as 200, 1000, and 50 millimeters, respectively.
- At least 4 bars are used in the four corners of the cross sections.
- The minimum cover of concrete is taken as tc = 40cm.
- Minimum diameter of transversal steel is considered as $\phi 10$.
- A symmetrical pattern for bars is considered on opposite sides of the columns sections.

Figure 1. Typical beam and column details
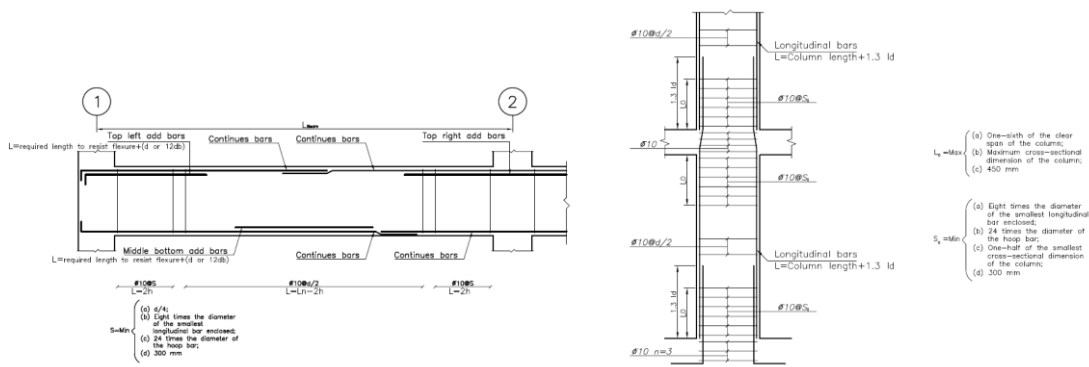
2.3 Constraints

Constraints derived from design provisions ACI 318-11 [22] for intermediate moment frames are strength, serviceability, ductility and other side constraints. More precisely, the constraints considered for the RC members can be categorized into two main types. The first type comprises constraints on the load-carrying capacities of the sections, clear spacing

limits between reinforcing bars, and the minimum and the maximum allowed reinforcement of the members. The second type consists of those constraints defining architectural requirements, constructible designs, and detailing practices. The minimum and the maximum dimensions of the member's section, the maximum aspect ratio of the section, maximum number of reinforcing bars and other reinforcement requirements. The constraints, which can be imposed to column groups, beam groups or joint regions, are explained and expressed in a normalized form as given below.

2.3.1 Constraints for beam groups

To avoid repetition, it is necessary to note that in this sub-section i represent the number of the beam (ith $beam$); j is the load combination type; while N_{beam} and $N_{load-combination}$ are the total number of the beams and total number of the combination loads, respectively. b_w and d indicate the width and the distance from extreme compression fiber to centroid of longitudinal tension reinforcement of the beam section, respectively; f'_c and f_y denote the compressive strength of concrete and the yielding strength of reinforcing steel, respectively. Other notations were declared where needed.

At every section of a flexural member where tensile reinforcement is required by analysis, the tension area of longitudinal steel reinforcement, A_s, should satisfy the minimum and the maximum requirements permitted by design specification. It should be noted that a minimum and maximum amount of tensile reinforcement is required for both positive and negative moment regions;

$$G_{b1}(x) = \frac{A_{s,min,i}}{A_{s,i}} - 1 \leq 0 \qquad i = 1, \ldots, N_{beam} \tag{7}$$

$$G_{b2}(x) = \frac{A_{s,i}}{A_{s,max,i}} - 1 \leq 0 \qquad i = 1, \ldots, N_{beam} \tag{8}$$

$$As_{min} = max \begin{cases} \dfrac{0.25\sqrt{f'_c}}{f_y} b_w.d \\[3mm] \dfrac{1.4}{f_y} b.d \end{cases} \tag{9}$$

Although the American code specified the maximum percentage of steel as 75 percent of balanced reinforcement ratio in the earlier versions, in the most recent version of the code the maximum amount of reinforcements is calculated according to the fact that the net tensile strain in the extreme tension steel, ε_t, should be equal to or greater than 0.005 when the concrete in compression reaches its assumed strain limit of 0.003.

A minimum area of shear reinforcement, $A_{v,min}$, shall be provided in all reinforced concrete flexural members where V_u exceeds $0.5\emptyset V_c$;

$$G_{b3}(x) = \frac{A_{v,min,i}}{A_{v,i}} - 1 \leq 0 \qquad i = 1, \ldots, N_{beam} \tag{10}$$

$$A_{v,min} = max \begin{cases} 0.062 \dfrac{\sqrt{f'_c}}{f_y} b_w . s \\[4mm] 0.35 \dfrac{b_w . s}{f_y} \end{cases} \tag{11}$$

where s is the spacing between stirrups (ties).

The minimum clear spacing between parallel bars in a layer should be greater than the maximum of the nominal diameter of bars Φ_l and 25 mm. However, for constructional requirements, the minimum free distance between the longitudinal bars was considered as $S_b = 40mm$. In the case that parallel reinforcement was placed in two layers, bars in the upper layers were placed directly above bars in the bottom layer with a clear distance between layers equal to 25 mm.

$$G_{b4}(x) = \frac{S_{b,min}}{S_{b,i}} - 1 \leq 0 \qquad i = 1, ..., N_{beam} \tag{12}$$

The spacing between stirrups, S_v, should satisfy the maximum requirements permitted by design specification;

$$G_{b5}(x) = \frac{S_{v,i}}{S_{v,max,i}} - 1 \leq 0 \qquad i = 1, ..., N_{beam} \tag{13}$$

$$S_{v,max}$$

$$\rightarrow \begin{cases} min \begin{cases} d/2 \\ 600mm \end{cases} & \textit{Along the span of beam members} \\[4mm] min \begin{cases} d/4 \\ 300mm \end{cases} & \textit{At the location of beam members where } v_s \textit{ exceeds } 0.33\sqrt{f'_c}b_w d \\[4mm] min \begin{cases} d/4 \\ 300mm \\ 8\Phi_l \\ 24\Phi_v \end{cases} & \textit{At the left and right ends of beam members} \end{cases} \tag{14}$$

where Φ_l denotes the diameter of longitudinal reinforcing bars, and Φ_v= the diameter of shear reinforcing bars (ties).

Also, the minimum spacing between stirrups, S_v, is limited to 50mm because of constructional requirements;

$$G_{b6}(x) = \frac{S_{v,min}}{S_{v,i}} - 1 \leq 0 \qquad i = 1, ..., N_{beam} \tag{15}$$

The height of the beams, h, should be greater than the allowable minimum height for beams, h_{min}, and less than the maximum height limit value given for beams, h_{max};

$$G_{b7}(x) = \frac{h_{min,i}}{h_i} - 1 \leq 0 \qquad i = 1, ..., N_{beam} \tag{16}$$

$$G_{b8} = \frac{h_i}{h_{max,i}} - 1 \leq 0 \qquad i = 1, ..., N_{beam} \tag{17}$$

$$h_{min} \rightarrow \begin{cases} \dfrac{L_n}{21} & for\ both\ end\ continuous \\[2ex] \dfrac{L_n}{18.5} & for\ one\ end\ continuous \end{cases} \tag{18}$$

where L_n *is* the length of clear span measured face-to-face to the supports. The maximum height limit value should be defined for each problem separately according to its architectural requirements.

For each section of the beams, the negative and the positive reduced moment strength of the section $\emptyset M_n$, should be greater than the factored moment force at the section, M_u. Besides, the positive moment strength at any end (support) of the beams, $M_{n,end}{}^+$, should be not less than one-third the negative moment strength, $M_{n,end}{}^-$. Furthermore, neither the negative nor the positive moment strength at any section along the length of the beam, $M_n{}^+, M_n{}^-$, should be less than one-fifth the maximum moment strength at the ends of the beams, $M_{n,end}$;

$$G_{b9} = \frac{|M_{u(i,j,k)}|}{|\emptyset M_{n,i}|} - 1 \le 0 \qquad i = 1, \dots, N_{beam} \qquad j = 1, \dots, N_{load-combination} \qquad k = 1,2 \tag{19}$$

$$G_{b10} = \frac{\left|\frac{1}{3} M_{n,end,i}{}^-\right|}{|M_{n,i}{}^+|} - 1 \le 0 \qquad i = 1, \dots, N_{beam} \tag{20}$$

$$G_{b11} = \frac{\left|\frac{1}{5} M_{n,end,i}\right|}{|M_{n,i}{}^+|} - 1 \le 0 \qquad i = 1, \dots, N_{beam} \tag{21}$$

$$G_{b12} = \frac{\left|\frac{1}{5} M_{n,end,i}\right|}{|M_{n,i}{}^-|} - 1 \le 0 \qquad i = 1, \dots, N_{beam} \tag{22}$$

where *k* represents the negative moment and positive moment situations.

For each section of the beams, the shear force capacity, $\emptyset V_n$, should be greater than the factored shear force at the section, V_u. In addition, the factored shear force at the section, V_u, should be less than allowed maximum shear force capacity, V_{max}. Furthermore, $\emptyset V_n$ of beams resisting earthquake effect, E, should be greater than the allowed minimum shear force capacity, V_{min} ;

$$G_{b13} = \frac{|V_{u(i,j)}|}{|\emptyset V_{n,i}|} - 1 \le 0 \qquad i = 1, \dots, N_{beam} \qquad j = 1, \dots, N_{load-combination} \tag{23}$$

$$G_{b14} = \frac{|V_{u,i,j}|}{|\emptyset V_{max}|} - 1 \le 0 \qquad i = 1, \dots, N_{beam} \qquad j = 1, \dots, N_{load-combination} \tag{24}$$

$$G_{b15} = \frac{|V_{min}|}{|\emptyset V_{n,i}|} - 1 \le 0 \qquad i = 1, \dots, N_{beam} \tag{25}$$

2.3.2 Constraints for column groups

To avoid repetition, it is necessary to note that in this sub-section i represent the number of the column ($ith\ column$) ; j is the load combination type; while N_{column} and

$N_{load-combination}$ are the total number of the columns and total number of the combination loads, respectively. b and d indicate the width and height of the column section, respectively; f'_c and f_y denote the compressive strength of concrete and the yielding strength of reinforcing steel, respectively. Other notations were declared where needed.

The width, b, and the height, h, of a column section should not be less than the minimum dimensions limit value given for columns;

$$G_{c1}(x) = \frac{b_{min,i}}{b_i} - 1 \leq 0 \qquad i = 1, ..., N_{column} \tag{26}$$

$$G_{c2}(x) = \frac{h_{min,i}}{h_i} - 1 \leq 0 \qquad i = 1, ..., N_{column} \tag{27}$$

The minimum width, b_{min}, and height, h_{min}, should be defined for each problem separately according to its architectural requirements.

Area of longitudinal reinforcement, A_{st}, in a column section should be between the minimum and maximum limits, $A_{st,min}$, permitted by design specification;

$$G_{c3}(x) = \frac{A_{st,min,i}}{A_{st,i}} - 1 \leq 0 \qquad i = 1, ..., N_{Column} \tag{28}$$

$$G_{c4}(x) = \frac{A_{st,i}}{A_{st,max,i}} - 1 \leq 0 \qquad i = 1, ..., N_{Column} \tag{29}$$

$$A_{st,min} = 0.01 \, A_g \tag{30}$$

$$A_{st,max} = 0.08 \, A_g \tag{31}$$

where A_g is the gross area of concrete section.

The spacing between longitude steel reinforcements, S_c, should satisfy the minimum and maximum requirements permitted by design specification, $S_{c,min}$ and $S_{c,max}$, as well as constructional limits ($S_{c,min} = 40mm$);

$$G_{c5}(x) = \frac{S_{c,min}}{S_{c,i}} - 1 \leq 0 \qquad i = 1, ..., N_{column} \tag{32}$$

$$G_{c6}(x) = \frac{S_{c,i}}{S_{c,max}} - 1 \leq 0 \qquad i = 1, ..., N_{column} \tag{33}$$

A minimum area of shear reinforcement, $A_{sh,min}$, should be provided in all reinforced concrete flexural members where factored shear force at section, V_u, exceeds $0.5\emptyset V_c$

$$G_{c7}(x) = \frac{A_{sh,min,i}}{A_{sh,i}} - 1 \leq 0 \qquad i = 1, ..., N_{column} \tag{34}$$

where V_c denotes nominal shear strength provided by the concrete and A_{sh} is the area of the shear reinforcements.

The spacing between stirrups, S_{cv}, should satisfy the maximum requirements permitted by design specification, $S_{cv,max}$. Also, the minimum spacing between stirrups of the columns, $S_{cv,min}$ is limited to 50mm because of constructional requirements;

$$G_{c8}(x) = \frac{S_{cv,i}}{S_{cv,max,i}} - 1 \leq 0 \qquad i = 1, \dots, N_{column} \tag{35}$$

$$G_{c9}(x) = \frac{S_{cv,min}}{S_{cv,i}} - 1 \leq 0 \qquad i = 1, \dots, N_{column} \tag{36}$$

$$S_{cv,max} \rightarrow \begin{cases} min \begin{cases} (min(b,d))/2 \\ 600mm \\ 16\Phi_l \\ 48\Phi_v \end{cases} & \text{Along the span of column members} \\[2em] min \begin{cases} (min(b,d))/4 \\ 300mm \\ 16\Phi_l \\ 48\Phi_v \end{cases} & \text{At the location of column members where } v_s \text{ exceeds } 0.33\sqrt{f'_c}b_w d \\[2em] min \begin{cases} (min(b,d))/4 \\ 300mm \\ 8\Phi_l \\ 24\Phi_v \end{cases} & \text{At the both ends of column members over the length of } l_0 \end{cases} \tag{37}$$

$$l_0 \rightarrow \begin{cases} L_{n,column}/6 \\ max(b,d) \\ 450\ mm \end{cases} \tag{38}$$

where Φ_l and Φ_v are the diameter of longitudinal reinforcing bars and the diameter of stirrups (ties). $L_{n,column}$ is the length of clear span of the column measured face-to-face to the supports.

For each section of the columns, the strength capacity of the section, σ_n, should be greater than the stress of the applied force, σ_u.

$$G_{c10}(x) = \frac{|\sigma_{u(i,j)}|}{|\sigma_{n,i}|} - 1 \leq 0 \qquad i = 1, \dots, N_{column} \qquad j = 1, \dots, N_{load-comb\ ination} \tag{39}$$

The strength of the columns should be obtained by the P-M interaction curve. For each section of the column, the P-M point from the applied force should lies within the interaction curve. In this study, to approximate the curves with acceptable accuracy, 11 interaction points for each curve was considered. A typical interaction diagram is shown. Fig. 1. Furthermore, according to the ACI code [22], for compression members not braced sideways, the slenderness effects can be neglected, when $kl_u/r < 22$. And when slenderness effects are not neglected as permitted above, the design of compression members, restraining beams, and other supporting members shall be based on the factored forces and moments from a second-order analysis. To satisfy this obligation, the Second-order effects are considered along the length of compression members in this study.

For each section of the column, the shear force capacity, $\emptyset V_n$, should be greater than the factored shear force at the section, V_u. In addition, the factored shear force at the section, V_u, should be less than allowed maximum shear force capacity, V_{max}. Furthermore, $\emptyset V_n$ of columns resisting earthquake effect, E, should be greater than the allowed minimum shear force capacity, V_{min} ;

$$G_{b11} = \frac{|V_{u(i,j)}|}{|\emptyset V_{n,i}|} - 1 \leq 0 \qquad i = 1, \dots, N_{columns} \qquad j = 1, \dots, N_{load-combination} \tag{40}$$

$$G_{b12} = \frac{|V_{u,i,j}|}{|\emptyset V_{max}|} - 1 \leq 0 \qquad i = 1, ..., N_{column} \qquad j = 1, ..., N_{load-combination} \qquad (41)$$

$$G_{b13} = \frac{|V_{min}|}{|\emptyset V_{n,i}|} - 1 \leq 0 \qquad i = 1, ..., N_{column} \qquad (42)$$
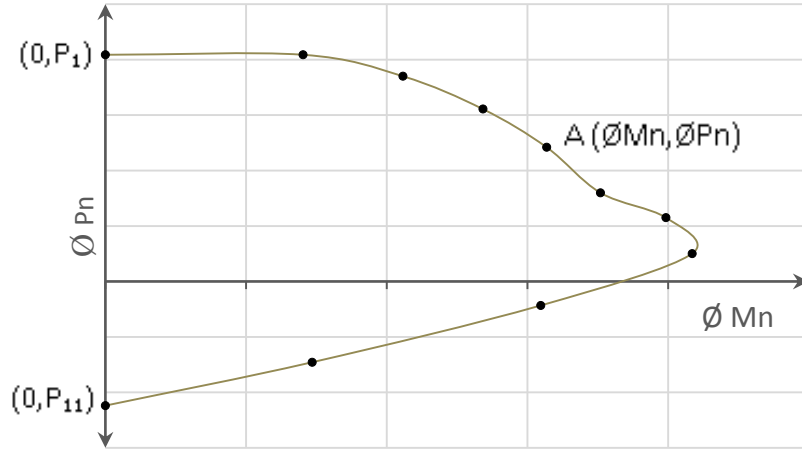
Figure 1. A Typical φPn, φMn interaction curve

2.3.3 Constraints for joints

At frame joints, the width of beams, b_w, should be smaller than the corresponding dimensions of intersected columns, d_c. Furthermore, at the joints with two columns, the cross-sectional dimension of bottom column, b_b and d_b, should be equal or greater than the corresponding cross-sectional dimension of the top column, b_t and d_t.

$$G_{j1}(x) = \frac{b_{w,i}}{d_{c,i}} - 1 \leq 0 \qquad i = 1, ..., N_{joint} \qquad (43)$$

$$G_{j2}(x) = \frac{b_{t,i}}{b_{b,i}} - 1 \leq 0 \qquad i = 1, ..., N_{joint} \qquad (44)$$

$$G_{j3}(x) = \frac{d_{t,i}}{d_{b,i}} - 1 \leq 0 \qquad i = 1, ..., N_{joint} \qquad (45)$$

where N_{joint} is the total number of joints of the frame.

The design story drift, Δ, should not exceed the allowable story drift, Δ_a.

$$G_{j4}(x) = \frac{\Delta_i}{\Delta_{a,i}} - 1 \leq 0 \qquad i = 1, ..., N_{story} \qquad (46)$$

where N_{story} is the total number of stories in the frame.

In this study, the elastic drifts were determined using seismic design forces based on the

exact computed fundamental period of the structure without any upper limit according to ASCE 7-10 [23].

3. A BRIEF DESCRIPTION THE PROPOSED ALGORITHM

In a PSO formulation, multiple candidate solutions, called particles, fly through the problem search space looking for the optimal position. Each particle has a position and a velocity in the multidimensional design space. Additionally, it has a fitness value which is evaluated by the objective function at its current location. A 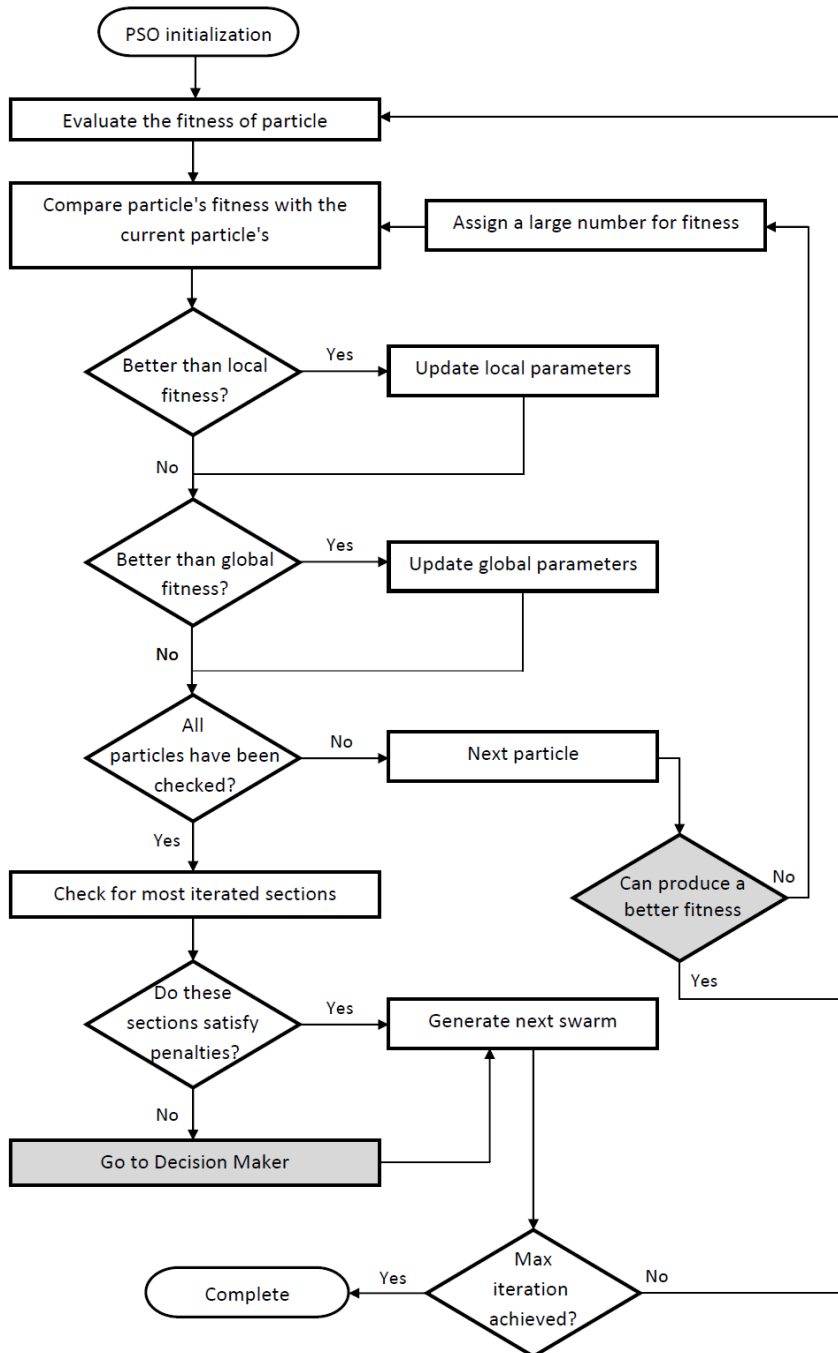particle by itself has almost no power to solve any problem; progress occurs only when the particles interact. Consequently, each particle communicates with other particles to determine its movement through the search space and adjusts its velocity and position according to the best solution it has achieved so far as well as the best point found by any member of its neighborhood. The next iteration takes place after all particles have been moved. This is expected to move the swarm toward the best solutions. However, metaheuristics such as PSO do not guarantee an optimal solution is ever found. On the other hand, RC frames optimum problems are well known for their time-consuming process due to the presence of many design variables, large size of the search space, and various constraints which must be satisfied simultaneously. Consequently, in this study, a supplementary algorithm was developed based on the concept of multi-criterion decision-making and was combined by the PSO algorithm to accelerate convergence toward the optimum solution in structural multi-objective optimization problems.

The DM algorithm plays three major roles in the DMPSO algorithm. First, it checks the results in each iteration and indicates what kind of changes in the design variables would lead to a more preferred solution. Therefore, the DMPSO adjusts its velocity and position according its own experience, the experience of neighboring particles, and the preference of the decision maker. Furthermore, the DM algorithm can ignore a solution at any point of the calculation process where perceives that it cannot produce a better fitness. Consequently, the computational resources available are not wasted since only such Pareto optimal solutions which are interesting to the DM are generated. Finally, the basic PSO is easily trapped into a local minimum. Considering that, the DM maker imposes its preference to intelligently escape from the local minimum.

The flowchart of the optimization by DMPSO algorithm is shown schematically in Fig. 2.
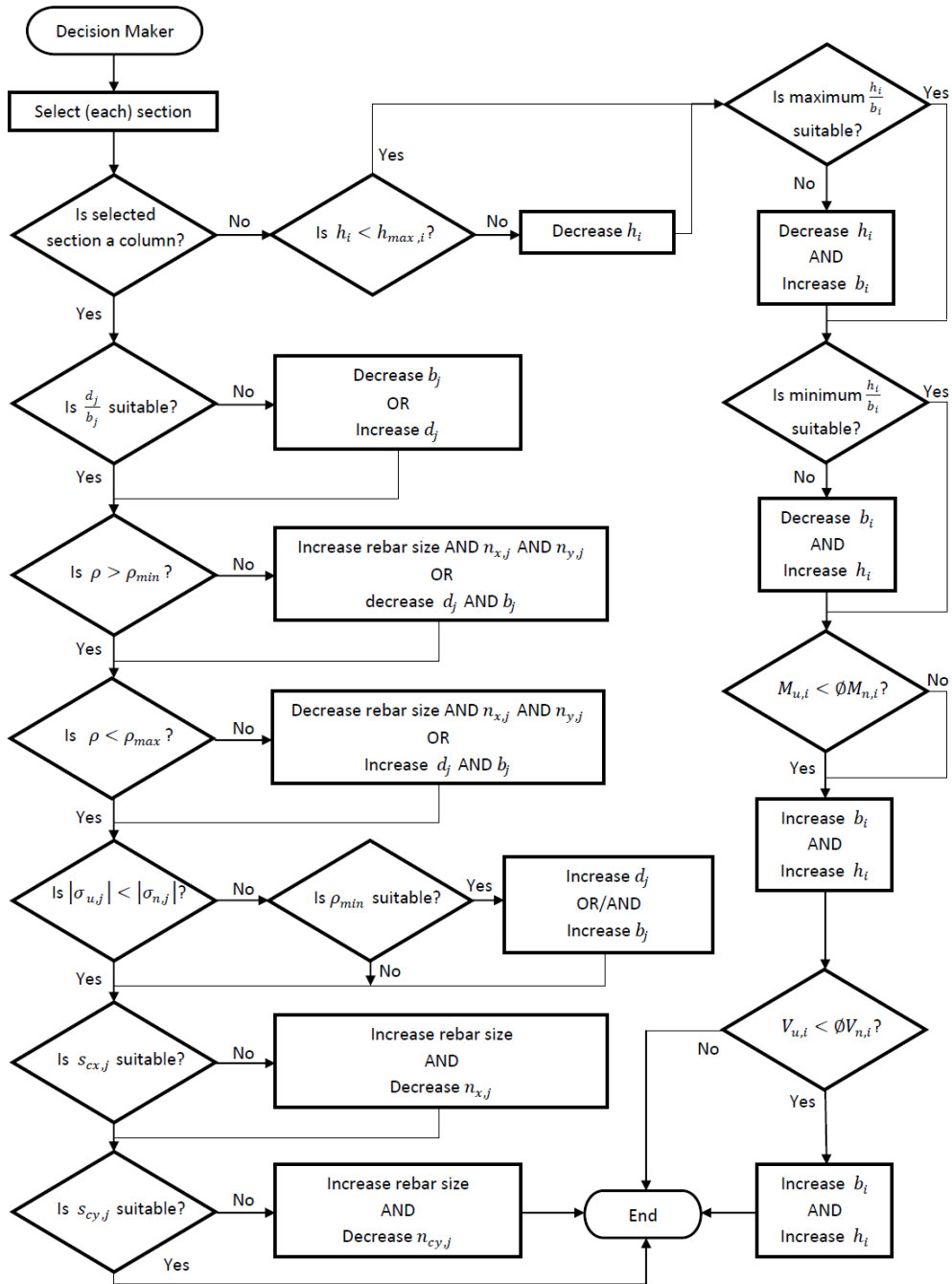
```
                    ┌──────────────────┐
                    │ PSO initialization │
                    └────────┬─────────┘
                             │
           ┌─────────────────▼─────────────────┐◄──────────────────────────────┐
           │  Evaluate the fitness of particle  │                               │
           └─────────────────┬─────────────────┘                               │
                             │                                                 │
  ┌────────────────────────────────────────┐    ┌──────────────────────────┐  │
  │ Compare particle's fitness with the     │◄───│ Assign a large number    │◄─┤
  │ current particle's                      │    │ for fitness              │  │
  └──────────────────┬─────────────────────┘    └──────────────────────────┘  │
                     │                                                         │
              ╱──────▼──────╲        Yes    ┌──────────────────────────┐       │
             ╱ Better than    ╲─────────────►│ Update local parameters  │       │
             ╲ local fitness? ╱             └─────────────┬────────────┘       │
              ╲──────┬──────╱                             │                    │
                  No │◄───────────────────────────────────┘                    │
              ╱──────▼──────╲        Yes    ┌──────────────────────────┐       │
             ╱ Better than    ╲─────────────►│ Update global parameters │       │
             ╲ global fitness?╱             └─────────────┬────────────┘       │
              ╲──────┬──────╱                             │                    │
                  No │◄───────────────────────────────────┘                    │
              ╱──────▼──────╲        No     ┌──────────────┐                   │
             ╱ All particles  ╲─────────────►│ Next particle│                   │
             ╲ have been       ╱            └──────┬───────┘                   │
             ╲ checked?       ╱                    │                           │
              ╲──────┬──────╱                      │                           │
                 Yes │                      ╱──────▼──────╲   No                │
                     │                     ╱ Can produce a ╲────────────────────┘
  ┌──────────────────▼──────────────┐      ╲ better fitness ╱
  │ Check for most iterated sections│      ╲──────┬──────╱
  └──────────────────┬──────────────┘         Yes │
                     │                             │
              ╱──────▼──────╲        Yes    ┌──────▼─────────────┐
             ╱ Do these      ╲─────────────►│ Generate next swarm│
             ╲ sections satisfy╱            └──────┬─────────────┘
             ╲ penalties?     ╱                    │
              ╲──────┬──────╱                      │
                  No │                             │
  ┌──────────────────▼──────────────┐              │
  │ Go to Decision Maker            │              │
  └─────────────────────────────────┘       ╱──────▼──────╲
                                            ╱ Max           ╲   No
                ┌──────────┐       Yes     ╱ iteration       ╲──────►
                │ Complete │◄──────────────╲ achieved?       ╱
                └──────────┘                ╲──────────────╱
```

Figure 2. Flowchart of the optimization by DMPSO algorithm

3.1 Mathematical formulation of DMPSO

The velocity and position of the particle are updated in a stochastic way as follows:

$$v^j(t+1) = wv^j(t) + c_1 r_1 \circ \left(x^{Pb,j} - x^j(t)\right) + c_2 r_2 \circ \left(x^{Gb,j} - x^j(t)\right) + c_3 r_3 \circ \left(x^{DM,j} - x^j(t)\right) \quad (47)$$

$$x^j(t+1) = x^j(t) + v^j(t+1) \quad (48)$$

where $v^j(t)$ and $x^j(t)$ represent the velocity and the position vectors of particle j at time t, respectively. The term w is the inertia weight, a scaling factor employed to control the exploration abilities of the swarm. Vector $x^{Pb,j}$ denotes the personal best position which is recorded by particle j, vector $x^{Gb,j}$ is the global best position obtained by the entire swarm up to the current iteration, and vector $x^{DM,j}$ indicates the position of preference of the decision maker in the search space. The acceleration coefficients c_1 and c_2, and c_3 are coefficients which control the impact of the particle's own experiences, the other particles' experiences and the decision maker's preference on the trajectory of each particle, respectively. r_1, r_2, r_3 are three random vectors with numbers uniformly distributed in the interval [0, 1]. The symbol "∘" is the element-wise product of two vectors.

Particles' velocities in each dimension i ($i = 1,\dots,n$) are restricted to a maximum velocity v_i^{max}. The vector v_i^{max} determines the maximum change each dimension can undergo in its positional coordinates during an iteration. It is more appropriate to use a vector rather than a scalar, as in the general case different velocity restrictions can be applied for different dimensions of the particle [24]. However, providing that the particle moves outside the bounds for a dimension i after the position update, $x_i \leq x^L \ or \ x^U \geq x_i$, the design variable x_i limits the closest bound, $x_i = x^L \ or \ x_i = x^U$.

3.2 Constraint handling

The PSO algorithm mainly focuses on finding an optimum solution in unconstrained problems. While most real-world applications have constraints, there has been relatively little work related to the incorporation of constraints into the PSO algorithm [26]. However, several methods have been previously proposed for handling constraint by EAs in general for optimization problems. They can be grouped into four categories: (i) methods based on preserving feasibility of the solutions; (ii) methods based on penalty functions; (iii) methods that search for feasibility; (iv) other hybrid methods [25].

One of the simplest methods is to generate feasible solutions in a random way. In this approach, the algorithm recalculates the velocity vector for an infeasible individual using new random numbers *r*1 and *r*2, until all of the constraints become satisfied [26, 27]. While it guarantees the feasibility of the final optimum design, it may have a very high computational cost in some cases. Besides, finite element analyses, which are needed for calculations of most constraint functions in structural engineering applications, require a huge computational cost. As a result, it is impractical to use this handling procedure for such constraints. Furthermore, in most structural optimization problems, because of numerous design variables and large size of the search space, even the generation of one million random points was insufficient to produce a single feasible solution. However, it can be used to handle a few of basic constraints in cooperation with other handling approaches.

Another approach is to eliminate unfeasible solutions from a population which is equivalent to applying a very severe penalty to every unfeasible design. This may work reasonably well when the feasible search space constitutes a reasonable part of the whole search space. More precisely, for structural optimization problems where the ratio between

the sizes of feasible search space and whole search space is small, an initial population may consist of no feasible individuals. It might be essential a search over the unfeasible region and improve unfeasible individuals rather than reject them [28].

The most common method to handle constraints in optimization problems is to use penalty functions, which penalize infeasible solutions by reducing their fitness values in proportion to their degrees of constraint violation [29–34]. The penalty function is an effective auxiliary tool to deal with constrained problems in general and has been a popular approach because of its simplicity and ease of implementation[24].

This study proposes a nonlinear penalty approach in which some levels of violation for each constraint are defined. If any of the constraints are violated, a penalty, whose value is related to the degree of violated constraint, is applied to the objective function;

$$\Phi = \sum_{k=1}^{N_{total}} \varphi_k(x) \tag{49}$$

$$\varphi_k(x) = c \times (G_k(x))^2 \tag{50}$$

$$c \rightarrow \begin{cases} 0 & if\ G_i(x) \leq 0 \\ 1 & if\ 0 < G_i(x) \leq 0.01 \\ 2 & if\ 0.01 < G_i(x) \leq 0.1 \\ 10 & if\ G_i(x) > 0.1 \end{cases} \tag{51}$$

where Φ is the total penalty value, N_{total} denotes the total number of members, φ_x represents the penalty function, $G_k(x)$ is the typical constraint k.

Considering the fact that much of the computing time belongs to the structural calculations, DMPSO firstly check the constraints which do not need finite element analysis for their calculation and simultaneously computes the penalty of the unsatisfied constraints. In each step, the DM checks the possibility of obtaining better fitness. Whenever it perceives that the penalized objective function cannot obtain a better value compared to the local optimum, Pb, found by the entire swarm until iteration t, it will ignore the rest of the calculation and will assign a value which is greater than the local best, Pb, and the global best, Gb, to the fitness. This assures that the fitness of this particle will not be taken into account in the calculation of Pb or Gb. As a consequence, a plethora of calculations for particles which do not satisfy the code's requirements or are not likely to have a better fitness are avoided. Otherwise, the corresponding objective function value is computed and a finite element analysis is performed for the constraints check. As long as no violation is detected, no penalty will be imposed on the objective function $f(x)$.

4. DESIGN EXAMPLES

To evaluate the performance of the proposed optimization algorithm, three benchmark test examples were examined. These are a three-bay four-story, a three-bay eight-story, and a three-bay twelve-story RC frames found in the literature [17,19,20]. Design examples with different design variable numbers were selected to show the efficiency of the DMPSO algorithm. The design examples were solved three times and among the optimum frames obtained for each set, the best one was taken as the optimum design.

The examples are three benchmarks with the following structural characteristics: service dead load of *D=22.3 kN/m*, uniform service live load of *L=10.7 kN/m*, compressive strength of concrete *f'c =23.5* MPa, yield strength of steel reinforcements *f_y =392 MPa, C_c=105 $/m³, C_s=0.9 $/kg, C_f=92 $/m²* and *$\gamma_s$=7850 kg/m³*. Furthermore, six different factored load combinations are considered as suggested in ACI 318-11 [22] design code [16] as follows:

$$U_1 = 1.2D + 1.6L \tag{52}$$
$$U_2 = 1.2D + 1.0L \pm 1.4E \tag{53}$$
$$U_3 = 0.9D \pm 1.4E \tag{54}$$

where *D*, *L* and *E* are the assumed dead, live and lateral loads, respectively.

Furthermore, in order to consider the effect of cracking, the moment of inertia of the cross section for each member is calculated according to ACI 318-11 code [22] using the following relationships:

$$I_{Beam} = 0.35\, I_g \tag{55}$$
$$I_{Column} = 0.7\, I_g \tag{56}$$

where I_g is the gross moment of inertia of the section of the beam or column.

A population size of 100 is used for four-story and eight-story RC frames and a population size of 150 is used for the twelve-story RC frame to assure the best results for stochastic decline.

4.1 Example 1. *The three bay, four-story reinforced concrete frame*

The three-bay four-story reinforced concrete frame whose geometry, loading and grouping details are shown in Fig. 4 was designed by HBB-BC, HPSACO [17], bat [19], and improved artificial bee colony, IABC, algorithms, [20]. This frame has a total of 28 members, 12 beams and 16 columns, which are arranged into four groups; two groups for beams and two groups for columns. In this design example, there are 46 design variables, 36 of which are for beams (18 for each beam design group) and the remaining 10 is for columns (5 for each column design group). Beams and columns are grouped to satisfy the uniformity of members subject to close design forces and have similar behaviors according to their place in the frame and loading conditions.
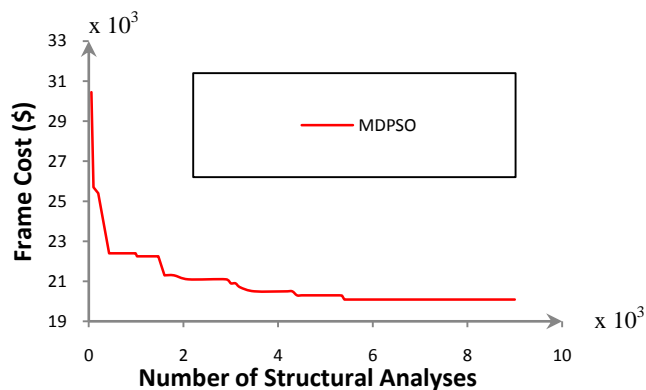

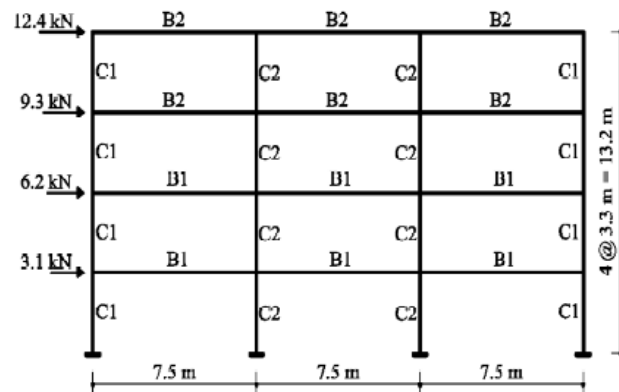Figure 3. Convergence rate for the three-bay four-story reinforced concrete frame

Figure 4. The three-bay, four-story reinforced concrete frame

The minimum cost of the optimum design was \$20,090 after 5,400 structural analyses. Another DMPSO feature to consider is its convergence rate. As it is obvious from Fig. 3, DMPSO found the best optimum solution after 5,400 structural analyses. The best design obtained by DMPSO presented in Table 1. Because no considerable limits were imposed to the design variables, the algorithm used the best section for the optimum solution. Table 2 demonstrates the maximum values of demand/capacity ratio under the critical loading case for member groups in the optimum solutions obtained by the algorithm.

4.2 Example 2. The three-bay eight-story reinforced concrete frame

The second test example was a three-bay eight-story reinforced concrete frame which comprised 56 elements, 24 beams and 32 columns. These elements were divided into three beam groups and four column groups. Consequently, 74 design variables exist in the problem, 54 for beams and 20 for columns. The structure geometry with its lateral loading and grouping details is depicted in Fig. 6.

Table 1: The results of optimum design form MDPSO

							PROPOSED ALGORITHM				
			Sectional dimensions			Reinforcements					
Member type	Type classification	Width (mm)	Depth (mm)	Top left	Top middle	Top right	Bot. left	Bot. middle	Bot. right
Beam	B1	300	350	3-D25 3-D18	2-D18	3-D25 3-D18	1-D22 2-D18	3-D22 2-D18	1-D22 2-D18
Beam	B2	300	350	3-D25 3-D18	2-D18	3-D25 3-D18	1-D22 2-D18	3-D22 2-D18	1-D22 2-D18
Column	C1	450	300		12-D18				
Column	C2	350	300		6-D22				
Frame cost					20090\$				
Number of Structural analysis					5400				
Max demand/capacity ratio for beams					0.988				
Max demand/capacity ratio for Columns					0.986				

Table 2: Maximum strength ratio for the member groups in the three-bay four-story reinforced concrete frame

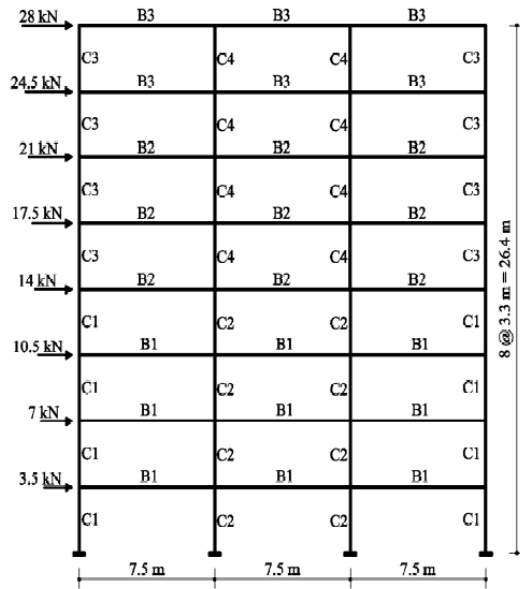| Beams			Columns		
Type classification	Strength ratio	Critical load combination	Type classification	Strength ratio	Critical load combination
B1	0.990	U_1	C1	0.965	U_1
B2	0.997	U_1	C2	0.985	U_1

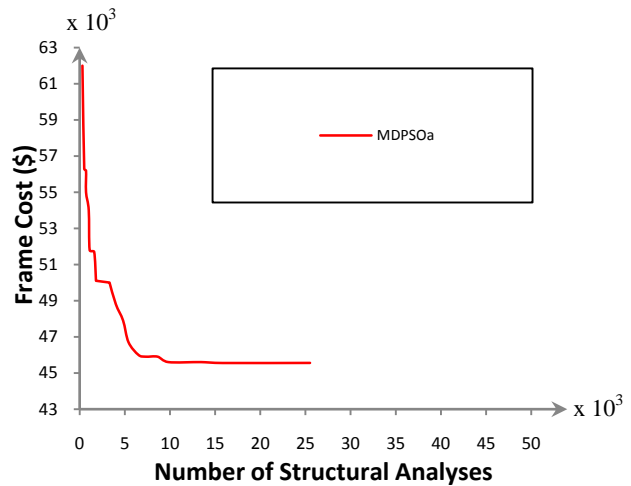Figure 6. The three-bay, eight-story reinforced concrete frame

Figure 7. Convergence history for DMPSO for the three-bay four-story reinforced concrete frame

For this problem, DMPSO needed 8,630 finite element structural analyses to converge to an optimum cost of $45,555. Figure depicts the convergence history for the DMPSO.

Table 3 shows the results of optimum design from the DMPSO algorithm. As well, Table 4 illustrates the maximum values of demand/capacity ratio under the critical loading case for member groups in the optimum solutions obtained by the DMPSO algorithm.

4.3 Example 3. The three-bay twelve-story reinforced concrete frame

The third test example is a three-bay twelve-story RC frame with 84 members, shown in Fig. 8. This is one of the largest planar benchmarks in the field of RC optimization, which has the most design variables. As it can be seen, the members are collected in three beam groups and six column groups. Consequently, this problem has 84 design variables, 54 of which are for beams and the rest for columns.
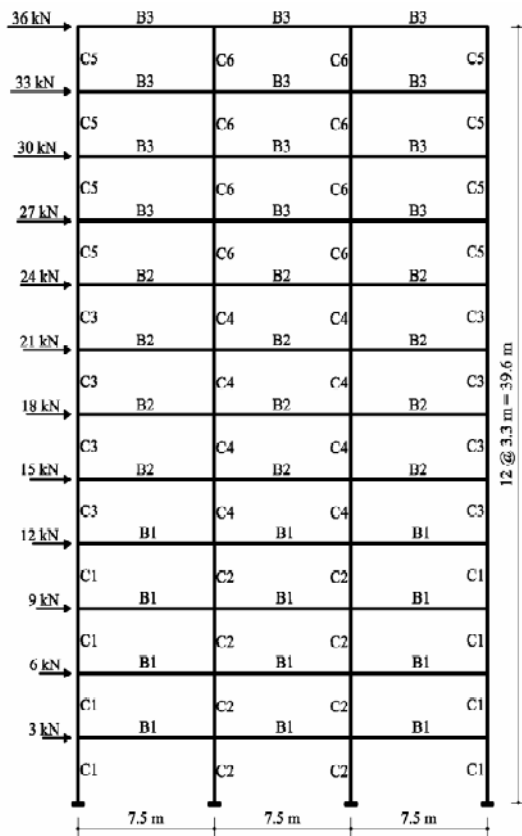
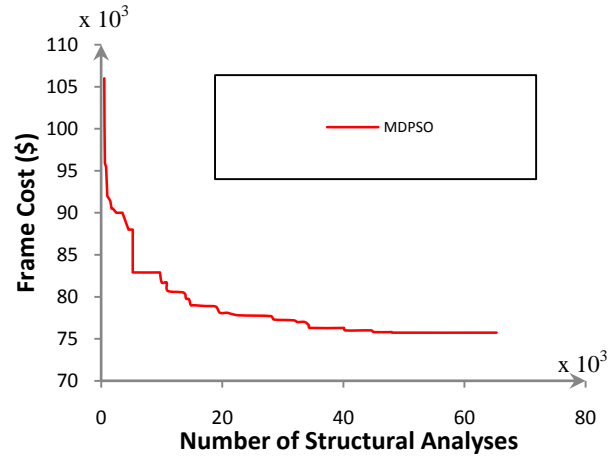Figure 8. Three-bay, twelve-story reinforced concrete frame

Figure 9. Convergence rate for DMPSO, HBB-BC, HPSACO, BA, and IABC for the three-bay Twelve-story reinforced concrete frame

The convergence history of the DMPSO is illustrated in Fig. 9. The DMPSO needed 40,300 structural analyses to reach an objective function value of $75,735.

Table 5 shows the results of optimum design from the DMPSO algorithm. Table 6 illustrates the maximum values of demand/capacity ratio under the critical loading case for member groups in the optimum solutions obtained by the DMPSO algorithm.

Table 3: The results of optimum design form MDPSO

				Optimization Results						
				Proposed Algorithm						
		Sectional dimensions				Reinforcements				
Member type	Type classification	Width (mm)	Depth (mm)	Top left	Top middle	Top right	Bot. left	Bot. middle	Bot. right	
Beam	B1	300	450	3-D25 3-D18	2-D18	3-D25 3-D18	3-D18	3-D18 1-D22	3-D18	

Member type	Type classification	Width	Depth	Top left	Top middle	Top right	Bot. left	Bot. middle	Bot. right
Beam	B2	300	400	3-D25 1-D22 2-D18	2-D18	3-D25 1-D22 2-D18	3-D18	4-D18 1-D22	3-D18
Beam	B3	300	350	3-D25 3-D18	2-D18	3-D25 3-D18	3-D18	3-D18 2-D25	3-D18
Column	C1	550	350			6-D25			
Column	C2	450	400			14-D25			
Column	C3	550	300			10-D18			
Column	C4	450	300			4-D22			
Frame cost						45555$			
Number of Structural analysis						8630			
Max demand/capacity ratio for beams						0.996			
Max demand/capacity ratio for columns						0.997			

Table 4: Maximum strength ratio for member groups in the three-bay eight-story reinforced concrete frame

Beams			Columns		
Type classification	Strength ratio	Critical load combination	Type classification	Strength ratio	Critical load combination
B1	0.988	U_2	C1	0.997	U_2
B2	0.991	U_2	C2	0.988	U_2
B3	0.996	U_1	C3	0.929	U_2
			C4	0.950	U_1

This excellent performance can be explained by the feature of DMPSO that considers the preference of DM. As a consequence, it tries to escape from the local minimum so it would not be easily trapped into a local value. Because the DMPSO has almost no restrictions for selecting sections, it used the most useful ones; so, the constraints were achieved when it was almost equal to the threshold values. Especially in beams, where the rebar was not continuous the entire length, the algorithm only used the needed adequate rebar.

Table 5: The results of optimum design form MDPSO

					Optimization Results				
					Proposed Algorithm				
		Sectional dimensions		Reinforcements					
Member type	Type classification	Width (mm)	Depth (mm)	Top left	Top middle	Top right	Bot. left	Bot. middle	Bot. right
Beam	B1	300	450	300	500	3-D25 4-D18	3-D18	3-D25 4-D18	3-D18

Beam	B2	300	400	300	500	3-D25 4-D18	3-D18	3-D25 4-D18	3-D18
Beam	B3	300	350	300	400	3-D25 4-D18	3-D18	3-D25 4-D18	3-D18
Column	C1	650	400			16-D18			
Column	C2	700	750			16-D22			
Column	C3	500	300			12-D22			
Column	C4	500	450			16-D18			
Column	C5	450	300			12-D18			
Column	C6	450	300			8-D22			
Frame cost						75735$			
Number of Structural analysis						40300			
Max demand/capacity ratio for beams						0.999			
Max demand/capacity ratio for columns						0.998			

* The best result among the mentioned studies from the literature is obtained by IABC [20].

Table 6: Strength ratio for member groups in the three-bay twelve-story reinforced concrete frame

Beams			Columns		
Type classification	Strength ratio	Critical load combination	Type classification	Strength ratio	Critical load combination
B1	0.999	U_2	C1	0.998	U_2
B2	0.997	U_2	C2	0.903	U_2
B3	0.996	U_1	C3	0.942	U_2
			C4	0.980	U_2
			C5	0.968	U_2
			C6	0.963	U_2

5. CONCLUSIONS

This article introduces an algorithm for optimization of reinforced concrete frames based on the multi-criterion Decision-Making and Particle Swarm Optimization algorithm. The main objective was to minimize the material and construction cost of reinforced concrete while satisfying the limitations and specifications of the ACI 318-11 [22] Code. The limitations and specifications are formulated as a series of constraints to the optimization problem and applied as penalties on the fitness function of the algorithm. Moreover, an efficient constraint handling technique, which demonstrated excellent performance, is proposed. It always led to feasible optimal designs, while also taking advantage of infeasible designs

during the optimization procedure. Furthermore, DMPSO not only considers the design code requirements but also constructional, architectural, and reinforcement detailing constraints. As a result, the optimum design obtained by DMPSO is ready for practical application without the need of any further process. The numerical results demonstrated the sound performance of the DMPSO algorithm, in terms of efficiency and the convergence rate, for finding optimum cost of RC structures. Based on the present work, it can be concluded that DMPSO, which accelerates and simplifies the process of the optimization process, provides an ideal technique to model practical design by considering the variations in cross-sectional dimensions of concrete frame members as well as detailing the combinations and placement of reinforcement bars to find realistic design solutions which are directly constructible.

REFERENCES

1. Sahab MG, Toropov VV, Gandomi AH. A review on traditional and modern structural optimization: problems and techniques, *Meta Appli Struct Inf* 2013, Oxford, Elsevier, 25-47.
2. Kaveh A, Talatahari S. Optimal design of skeletal structures via the charged system search algorithm, *Struct Multi Optim* 2010*; 41(6): 893-911.*
3. Vira C, Haimes YY. *Multiobjective Decision Making: Theory and Methodology*, North-Holland, 1983, No. 8.
4. Miettinen K. Nonlinear Multiobjective Optimization, volume 12 of International Series in Operations Research and Management Science, 1999.
5. Sawaragi Y, Nakayama H, Tanino T. *Theory of Multiobjective Optimization*, Acad, 1985.
6. Coello CC, Lamont GB, Van Veldhuizen DA. *Evolutionary Algorithms for Solving Multi-Objective Problems*, Springer Science & Business Media, 2007.
7. Justesen PD. Multi-objective Optimization using Evolutionary Algorithms, University of Aarhus, Department of Computer Science, Denmark, 2009.
8. Deb K, Miettinen K. *Multiobjective Optimization: Interactive and Evolutionary Approaches,* Springer Science & Business Media, Vol. 5252, 2008
9. Sindhya K, Miettinen K, Deb K. A hybrid framework for evolutionary multi-objective optimization, *IEEE Transactions on Evolutionary Computation* 2013; **17**(4): 495-511.
10. Michalewicz Z. *Genetic Algorithms+ Data Structures= Evolution Programs*, Springer Science & Business Media, 2013.
11. Coello CAC, Lamont GB. *Applications of Multi-Objective Evolutionary Algorithms*, Vol. 1, World Scientific, 2004.
12. Rajeev S, Krishnamoorthy CS. Genetic algorithm–based methodology for design optimization of reinforced concrete frames, *Comput-Aided Civ Inf Eng* 1998; **13**(1): 63-74.
13. Govindaraj V, Ramasamy JV. Optimum detailed design of reinforced concrete continuous beams using genetic algorithms, *Comput struct* 2005; **84**(1): 34-48.
14. Govindaraj V, Ramasamy JV. Optimum detailed design of reinforced concrete frames using genetic algorithms, *Eng Optim* 2007; **39**(4): 471-94.

15. Kwak HG, Kim J. Optimum design of reinforced concrete plane frames based on predetermined section database, *Comput-Aided Des* 2008; **40**(3): 396-408.
16. Kwak HG, Kim J. An integrated genetic algorithm complemented with direct search for optimum design of RC frames, *Comput-Aided Des* 2009; **41**(7): 490-500.
17. Kaveh A, Sabzi O. A comparative study of two meta-heuristic algorithms for optimum design of reinforced concrete frames, *Int J Civil Eng* 2011; **9**(3): 193-206.
18. Kaveh A, Behnam AF. Design optimization of reinforced concrete 3D structures considering frequency constraints via a charged system search, *Sci Iran* 2013; **20**(3): 387-96.
19. Gholizadeh S, Aligholizadeh V Optimum design of reinforced concrete frames using bat meta-heuristic algorithm, *Int J Optim Civil Eng* 2013l; **3**(3): 483-97.
20. Gheyratmand C, Gholizadeh S, Vababzadeh B. Optimization of RC frames by an improved artificial bee colony algorithm, *Int J Optim Civil Eng* 2015; **5**(2): 189-203.
21. Akin A, Saka MP. Harmony search algorithm based optimum detailed design of reinforced concrete plane frames subject to ACI 318-05 provisions, *Comput Struct* 2015; **147**: 79-95.
22. ACI Committee. Building code requirements for structural concrete (318-11) and commentary-(318R-11), *Detroit Mich Am Concr Inst* 2011.
23. ASCE. Minimum design loads for buildings and other structures, ASCE/SEI 7-05-2006, 2006.
24. Plevris V, Papadrakakis M. A hybrid particle swarm-gradient algorithm for global structural optimization, *Comput-Aided Civ Inf Eng* 2011; **26**(1): 48-68.
25. Koziel S, Michalewicz Z. Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization, *Evolut Comput* 1999; **7**(1): 19-44.
26. Hu X, Eberhart R. Multiobjective optimization using dynamic neighborhood particle swarm optimization, In Ieee *wcci* 2002; pp. 1677-1681.
27. Hu X, Eberhart RC, Shi, Y. Engineering optimization with particle swarm. In *Swarm Intelligence Symposium 2003. SIS'03. Proceedings of the 2003 IEEE*; pp. 53-57.
28. Michalewicz Z. A Survey of Constraint Handling Techniques in Evolutionary Computation Methods, *Evolut Prog* 1995; **4**: 135-55.
29. Michalewicz Z, Dasgupta D, Le Riche RG, Schoenauer M. Evolutionary algorithms for constrained engineering problems, *Comput Indus Eng* 1996; **30**(4): 851-70.
30. Michalewicz Z. Genetic algorithms, numerical optimization, and constraints, Morgan Kaufmann, San Mateo, CA In *Proceedings of the Sixth International Conference on Genetic Algorithms* 1995; **195**: pp. 151-158.
31. Powell D, Skolnick MM. Using genetic algorithms in engineering design optimization with non-linear constraints, Morgan Kaufmann Publishers Inc, In *Proceedings of the 5th International Conference on Genetic Algorithms* 1993; pp. 424-431.
32. Schoenauer M, Xanthakis S. Constrained GA Optimization, In *ICGA* 1993; pp. 573-580.
33. Smith AE, Tate DM. Genetic optimization using a penalty function, Morgan Kaufmann Publishers Inc, In *Proceedings of the 5th international conference on genetic algorithms* 1993; pp. 499-505.
34. Schoenauer M, Michalewicz Z. Evolutionary computation at the edge of feasibility, Springer Berlin Heidelberg, In *Parallel Problem Solving from Nature-PPSN IV* 1996; pp. 245-254.