Optimum Design of 3D Reinforced Concrete Building Frames with the Flower Pollination Algorithm

Panagiotis E. Mergos [a,*]

[a] Department of Civil Engineering, City, University of London, London EC1V 0HB, UK

Abstract.

The flower pollination algorithm (FPA) is a highly efficient metaheuristic optimization algorithm that is inspired by the pollination process of flowering species. FPA is characterised by simplicity in its formulation as well as high computational performance and it has been found to outperform other well-established algorithms in a range of diverse optimization problems. The present study applies, for first time, the FPA to the computationally challenging optimum design of real-world 3D reinforced concrete (RC) building frame structures after a set of appropriate modifications to its original formulation. To serve this goal, a new computationally efficient framework for the optimum design of 3D RC frames is developed that is interacting with the well-known software SAP2000 for the purposes of structural analysis and design. The framework is then applied to the minimum material cost design of a 4-storey and a 12-storey RC building in accordance with Eurocode regulations. It is found that the FPA exhibits better or similar computational performance than other well-established algorithms in these optimization tasks. Furthermore, parameter tuning analysis reveals the FPA parameter values that maximize its computational performance in the optimum structural design of 3D RC building frames.

Keywords: Structural optimization; Flower pollination algorithm; Reinforced concrete; Building frames; Parameter tuning

1 Introduction

In many complex, multi-modal design problems in industry and engineering, tracking of global optimum solutions remains a highly challenging task. Often, conventional optimization methodologies do not perform adequately in this class of problems as they may get trapped in local optima. In these cases, the use of nature-inspired metaheuristic algorithms is recommended (Yang 2008). There is a high number of efficient metaheuristic optimisation

---

[*] Corresponding author. Panagiotis E. Mergos, Senior Lecturer in Structural Engineering, Research Centre for Civil Engineering Structures, City, University of London, London, EC1V 0HB, UK.

E-mail address: panagiotis.mergos.1@city.ac.uk, Tel. 0044 (0) 207040 8417

algorithms in literature, including the Genetic Algorithm (GA) (Holland 1975), Simulated Annealing (SA) (Kirkpatrick et al. 1983), Cuckoo Search (CS) (Yang and Deb 2009), Firefly Algorithm (Yang 2010), Particle Swarm Optimization (PSO) (Kennedy 2011) and many others.

Recently, the Flower Pollination Algorithm (FPA) was developed (Yang 2012), which is a population-based metaheuristic optimization algorithm inspired by the evolution process of flowering plants. FPA is characterised by simplicity and flexibility in its formulation as well as high efficiency in its computational performance (Yang 2012, Abdel-Basset and Shawky 2018, Alyasseri et al. 2018). Furthermore, many studies show that it can outperform other well-established metaheuristic optimization algorithms (e.g. Yang 2012, Bekdas et al. 2015, Mergos and Mantoglou 2020). A simple explanation of the efficiency of FPA is based on the fact that it is imitating the reproduction process of flowering plants. The latter has been so successful that flower species dominate the landscape of earth (Walker 2009). As a result, FPA has been adopted by many optimization studies and it has been applied successfully to numerous optimization problems in diverse scientific fields, including electrical and power systems (e.g. Abdelaziz et al. 2016a, Abdelaziz et al. 2016b, Singh and Salgotra 2018), computer gaming (e.g. Abdel-Raouf et al. 2014), meteorology (e.g. Heng et al. 2016), image science (Zhou et al. 2016) and many others (Abdel-Basset and Shawky 2018, Alyasseri et al. 2018).

Nevertheless, the applications of FPA to structural engineering problems are limited. Bekdas et al. (2015) used FPA to minimize the weight of 2D and 3D steel truss structures. They found that the FPA is competitive with other state-of-the-art metaheuristic algorithms. Nigdeli et al. (2016a) examined the application of FPA to several basic structural engineering optimization problems. They found that FPA is effective to find the optimum solutions of these problems. Furthermore, Nigdeli et al. (2016b) used the FPA for the optimum tuning of mass dampers for earthquake-resistant structures concluding that FPA is efficient in tracking precise optimal values for this design task. Bekdas (2018) applied FPA to the cost-optimal design of post-tensioned axially symmetric cylindrical walls finding that FPA is one of the most robust algorithms for this optimization problem. Mergos and Mantoglou (2020) applied FPA to the optimum design of reinforced concrete retaining walls and compared its performance with other well-established optimization algorithms. It was found that FPA outperforms GA and PSO in this optimization problem. Ulusoy et al. (2020) used the FPA to actively control multi-story structures considering soil-structure interaction. Toklu et al. (2020) employed FPA to the structural analysis of plates for plane-stress conditions with nonlinear stress-strain equations.

Finally, Kayabekir et al. (2020) examined the effects of soil geotechnical properties on the optimal dimensions of restricted reinforced concrete retaining walls by employing the FPA. Extensive research has been conducted over the past decades on the optimum structural design of reinforced concrete structures. However, most of the previous research efforts focus either on single structural members (i.e. beams, columns and others) (e.g. Yeo and Gabai 2011, Medeiros and Kripka 2014, Mergos 2018b) or planar concrete frames (e.g. Paya et al. 2008, Akin and Saka 2015, Kaveh et al. 2020a, 2020b). Indeed, there is limited amount of research work on the optimum design of real-world 3D RC frames mainly due to the complexity and significant computational cost involved in the structural design of these systems (Sarma and Adeli 1997).

Fadaee and Grierson (1996) presented the minimum cost design of a one-bay, one-story 3D RC frame using the optimality criteria approach in accordance with the ACI-318 code. Balling and Yao (1997) conducted a comparative study of the optimum design of space RC frames following the ACI-318 code by examining one-, two- and four-story building frames subjected to vertical and lateral loads with the aid of sequential quadratic programming or a gradient-based method. Sahab et al. (2005) investigated the cost-optimization of RC flat slab buildings, according to the British Code of Practice BS 8110 (BSI 1997), using a multi-level optimization procedure combining exhaustive search and a hybrid genetic algorithm. Govindaraj and Ramasany (2007) examined the optimum detailed design of a 4-story space RC frame based on Indian Standards specifications and using a genetic algorithm. Sharafi et al. (2012) applied a heuristic approach for the cost optimization of the preliminary layout design of 3D RC frames using the ant colony optimization algorithm and adopting an alternative objective function that simplifies the optimization problem. Three RC flat slab buildings with different structural features and number of storeys were examined. Kaveh and Behnam (2013) examined the optimal design of 3D multi-story RC buildings using the Charged System Search (CSS) and the enhanced charged system search (ECSS) algorithms. The designs are based on the ACI-318 design code. Three-story and 7-story RC buildings are considered. The objective function is taken as the weight of the structure. Lagaros (2014) developed a general-purpose real-world structural design optimization computing platform employing eight different metaheuristic optimization algorithms but not the FPA. The platform was applied to the minimum cost design of four 3D structures, including two RC buildings, leading to cost savings in the order of 20-30% with respect to conventional structural designs. Esfandiari et al. (2018) applied the DMPSO optimization algorithm, which combines multi-criteria decision making and the PSO algorithm, to minimize the construction cost of 3D RC frames while satisfying the limitations

and specifications of the ACI-318 (ACI Committee 318, 2019) design code. Dehnavipour et al. (2019) used the PSO optimization algorithm to minimize the cost of a 3D six-story RC frame building according to the ACI-318 specifications. Martins et al. (2020) designed optimally a 3D building RC frame of five storeys and three bays in each horizontal direction for minimum economic cost according to Eurocode 2 (CEN 2000) by using a gradient-based algorithm in combination with a multi-start strategy approach.

It can be concluded from the previous literature review that the FPA algorithm has not yet been applied to the computationally challenging task of optimizing the structural design of real-world 3D RC building frames. This is despite the fact that FPA has been proven highly efficient in many other optimization problems and despite the urgent need for optimizing these structural systems that are associated with massive economic and environmental impacts on a global scale (Lagaros 2018).

In this study, a new computationally efficient framework for the optimum structural design of real-world 3D RC building frame structures is developed. Furthermore, the developed framework employs, for first time, the FPA to this demanding optimization problem after a set of appropriate modifications. The performance of the FPA is compared with other well-established optimization algorithms and conclusions are made with respect to its computational efficiency. Finally, a parameter tuning study is conducted to identify the FPA parameter values that maximize its computational performance in this optimization problem.

2 Framework for the optimum structural design of RC building frames

2.1 Optimization framework description

The structural design of RC building frames can be stated in the form of a single-objective optimization problem with discrete design variables as follows:

Minimize: $\qquad\qquad\qquad f(\boldsymbol{x})$

Subject to: $\qquad\qquad\qquad g_j(\boldsymbol{x}) \leq 0, \; j = 1 \; to \; m$ $\qquad\qquad\qquad$ (1)

Where:

$$\boldsymbol{x} = (x_1, x_2, \dots, x_d)$$

$$x_i \in \boldsymbol{D}_i = \left(d_{i1}, d_{i2}, \dots, d_{ik_i}\right), i = 1 \; to \; d$$

In this formulation, $f(x)$ is the objective function and x is the design solution vector that comprises of d independent design variables x_i ($i = 1$ to d). The design variables x_i take values from discrete values sets $D_i = (d_{i1}, d_{i2}, \ldots, d_{iki})$, where d_{it} ($t = 1$ to k_i) is the t-th possible discrete value of design variable x_i and k_i is the total number of possible discrete values of x_i. Furthermore, the solution should be subject to m number of constraints $g_j(x) \leq 0$ ($j = 1$ to m). The specification of the optimization problem components in the case of structural design of RC building frames examined in this study is described in the following.

In optimization problems, the input data are divided into design parameters that are assumed fixed and design variables that change values during the solution process. In this study, geometry, boundary conditions, material properties, concrete cover and loading of RC frames are treated as design parameters. Therefore, the present study examines solely sizing optimization of RC building frames, where only RC members cross-sectional characteristics need to be determined including both the concrete cross-sectional dimensions and the configuration of the steel reinforcement. More specifically, in this study, concrete cross-sections are treated as independent design variables and then steel reinforcement is calculated, for a given arrangement of cross-sections in the RC building frame, following standard structural design procedures in accordance with Eurocode 2 (EC2) (CEN 2000) and Eurocode 8 (EC8) (CEN 2004) for low ductility class (DCL) design guidelines. This approach greatly reduces the number of independent design variables and thereby the size of the search space allowing the optimization algorithms to track faster and more robustly optimum solutions of complex concrete frames (Mergos 2018a). Furthermore, it is most suitable for the design methodologies prescribed in codes of practice, such as EC2 (CEN 2000) and EC8 (CEN 2004), that employ elastic structural analysis procedures based exclusively on the geometric dimensions of concrete sections and not the placed steel reinforcement (Mergos 2018a, 2018c). The concrete sections are taken from discrete cross-section lists pre-specified by the designers in accordance with standard construction practices. For simplicity, square cross-sections are assumed herein for concrete columns and rectangular cross-sections for concrete beams with steel reinforcement configurations, as shown in Fig. 1.

Consistently with the previous assumptions, each design variable x_i ($i = 1$ to d), in the optimization problem of Eq. (1), represents a different concrete cross-section given to one or more concrete members belonging to the same group of members. Hence, the number of design variables d (i.e. number of problem dimensions) coincides with the number of different cross-sections in the concrete frame. The design variables x_i assume only integer values representing the position indices of the sections in the corresponding lists of cross-sections set by the

designers. Therefore, it holds $x_i \in \boldsymbol{D}_i = (1, 2, \ldots, k_i)$, where k_i is the total number of sections in the list of sections attributed to section i.

Fig. 1: Assumed concrete cross-sections and steel reinforcement configurations; a) column sections; b) beam sections

The objective function $f(\boldsymbol{x})$ in the optimum design of RC building frames considered herein is the sum of concrete and reinforcing steel material costs. These materials costs are taken as the sums of the individual costs of all structural members (i.e. beams and columns) comprising the RC frames. Therefore, $f(\boldsymbol{x})$ can be expressed by Eq. (2), where V_c (m³) represents the total volume of concrete and m_s (kg) the total mass of steel reinforcement of all structural members including both longitudinal and transversal reinforcing bars. In Eq. (2), f_{co} and f_{so} are the prices of concrete per unit volume and reinforcing steel per unit mass, respectively. Clearly, more comprehensive objective functions can be used to represent the construction cost or even better the life-cycle economic cost and/or environmental impact of RC structures (Mergos 2018b, Mergos 2018c, Yucel *et al.* 2021). Nevertheless, the adopted objective function is deemed adequate for the purposes of this study focusing mostly on the efficiency of optimization algorithms in the structural design of RC building frames.

$$f(\boldsymbol{x}) = V_c(\boldsymbol{x}) \cdot f_{co} + m_s(\boldsymbol{x}) \cdot f_{so} \tag{2}$$

The design constraints $g_j(\boldsymbol{x}) \leq 0$ ($j = 1$ to m) herein reflect the provisions for the structural design of concrete building frames of EC2 (CEN 2000) and EC8 (CEN 2004) for ductility class low (DCL). This includes structural detailing rules and safety verifications for the serviceability (SLS) and ultimate (ULS) limit state in terms of either displacements or forces.

2.2 Software application STROLAB

For the goals of the present research, a new MATLAB (MathWorks 2020a) application, namely STROLAB (STRuctural Optimization LABoratory), was developed. STROLAB is interacting, for the purposes of structural analysis and design, with the well-established structural analysis and design software SAP2000 (CSI 2020) via its Application Programming Interface (API). STROLAB can be applied to the optimum design of a range of structural systems, including 3D reinforced concrete building frames, by employing different optimization algorithms.

The use of STROLAB is straightforward. The user generates a structural model in SAP2000 employing standard procedures to specify materials, cross-sections, geometry, boundary conditions and loads. The user should also provide input related to the structural design of the model such as the design code and the design load combinations. Next, the user determines in STROLAB the lists of possible cross-sections attributed to each cross-section of the SAP2000 structural model. Additional information related to structural optimization, such as unit materials costs and optimization algorithm preferences, is also provided in STROLAB.

The application then sets the optimization problem and calls an optimizer to select sections from the sections lists. Next, STROLAB calls SAP2000 to execute structural analysis and design of the structural model using the sections selected by the optimizer. All other parameters of the structural model are kept the same as in the original SAP2000 file created by the user. Then, STROLAB reads the results of the structural design, checks whether the design constraints are fulfilled in SAP2000 and calculates the cost of the objective function. The afore-described procedure is repeated until terminated by the optimizer either because it converges to an optimal solution or because it reaches the maximum number of iterations specified by the user.

2.3 Optimization framework implementation

In this section, the procedures followed herein to address the optimization framework of §2.1 are discussed in more detail. Following the development of the structural model of the RC building frame in SAP2000 by the user, STROLAB sets the optimization problem and calls the optimizer to select column and beam sections, in the form of Fig. 1, from lists of sections pre-specified by the user. Next, STROLAB calls SAP2000 to conduct structural analysis for the various load cases (e.g. dead, live, wind, earthquake loads) specified by the user. Then, SAP2000 calculates the design action effects (i.e. internal moments, shears and axial forces)

using the design load combinations specified by Eurocode 0 (EC0) (CEN 2002) Eq. (6.10) for persistent and transient design situations and Eq. (6.12) for seismic design situations.

Next, STROLAB calls SAP2000 to conduct structural design of the concrete members for the ULS based on EC2 (CEN 2000) guidelines. For each design load combination, RC frame members are designed at a number of equally spaced cross-sections along their lengths. In the present study, eleven cross-sections are designed for each beam member and three for each column member.

Concrete beams are designed for major direction bending, shear and torsion (CSI 2016). Beam flexural reinforcement (i.e. $A_{s,top}$ and $A_{s,bot}$ in Fig. 1) is calculated using the simplified stress block of EC2 §3.1.7(3). Tension reinforcement is typically calculated, whereas compression reinforcement is only added when required to obtain a balanced failure type. No moment redistribution is assumed. The calculated tension steel reinforcement should be between a minimum and a maximum value specified by the code. For beam shear design, the design shear resistance of the concrete member without shear reinforcement $V_{Rd,c}$ is first calculated as well as the maximum design shear force that the beam can sustain without crushing of the diagonal concrete compression struts $V_{Rd,max}$. The design shear force V_{Ed} should be lesser than $V_{Rd,max}$. If V_{Ed} is between $V_{Rd,c}$ and $V_{Rd,max}$ then the required shear reinforcement (i.e. A_{sw} / s in Fig. 1) is calculated using the variable strut inclination method of EC2. In any case, the provided shear reinforcement should be greater than the minimum required by EC2. For torsion, the need for additional torsional reinforcement is first examined. If additional torsional reinforcement is required, then this is provided in the form of additional stirrups and longitudinal reinforcement. Furthermore, an upper limit for the combination of the design shear force and torsional moment is examined to avoid crushing of the compressive concrete struts (CSI 2016).

For column members, the longitudinal reinforcement $A_{s,tot}$ (Fig. 1) is designed for combined biaxial bending moments and axial load effects (i.e. $M_x + M_y + N$). To serve this goal, 3D axial force-biaxial bending moment interaction surfaces are generated for a range of reinforcement ratios between the minimum and maximum permissible values (CSI 2016). The interaction surfaces are generated assuming concrete and reinforcing steel material laws and mechanical properties prescribed in §3.1 and §3.2 of EC2. To calculate the design action effects, the first order bending moments and axial forces are first established for each design load combination. The design bending moments are then increased for geometric imperfections based on EC2 §3.2. For slender columns, the resulting moments are further increased to account for second-order effects by using the nominal curvature method (EC2, §5.8.8). The longitudinal steel

reinforcement is then determined so that the design action points (M_x, M_y, N) of all design load combinations are within the interaction surface and that the maximum utilization factor is close to unity. The design of columns for shear is very similar to the design of beams except for the effect of the axial force on the shear capacity of the concrete member without shear reinforcement. Furthermore, the shear design for column members takes place in both horizontal directions to calculate the corresponding required shear reinforcement areas (i.e. A_{swx} / s_x and A_{swy} / s_y in Fig. 1).

For the serviceability limit state, SAP2000 does not conduct a check for beam deflections (CSI 2016). To address this design requirement, the limiting span-to-depth ratio approach is adopted in this study as an additional check to SAP2000 (Moss and Brooker 2006). This method conservatively ensures that span deflections do not exceed the span length divided by 250.

To establish the objective function $f(\boldsymbol{x})$ of Eq. (2) for RC building frames, the mass of reinforcing steel must be determined in addition to the volume of the concrete members. To serve this goal, STROLAB reads the required flexural and shear steel reinforcement areas of the column and beam concrete members as calculated by SAP2000 at different design sections along the member lengths, following the design procedures described previously. Then, for simplicity, it assumes that the steel areas of the design sections are extended till the neighbouring design sections of the same member with lower reinforcement demands. Clearly, more elaborated approaches can be used to imitate real-life construction practices. However, this is not judged as crucial for the purposes of this study, which focusses on the efficiency of different optimization algorithms.

The design constraints $g_j(\boldsymbol{x})$ (j = 1 to m), in the present study, are also closely related to the EC2 structural design procedures described above. A design constraint is assumed not to be satisfied when the corresponding design checks cannot be fulfilled by any permissible amount of steel reinforcement in the concrete sections. This is the case because only the concrete sections are treated as independent variables herein. More particularly, a design constraint for the ULS of a column or a beam design section is assumed not to be satisfied when the required steel reinforcement in the section exceeds the maximum permissible by EC2. Furthermore, a design constraint for the ULS of a column or a beam design section is assumed not to be satisfied when the design shear forces and torsional moments at the section exceed the maximum capacity of the concrete member compressive struts according to EC2. In addition, a beam section is assumed not to fulfil the SLS design constraints when the check for deflections of the corresponding beam members cannot be satisfied.

3 Flower Pollination Algorithm

3.1 Original FPA

FPA imitates the process of reproduction of flowering plants. Like other biological systems, the ultimate goal of flower species is reproduction that is achieved by pollination. Flower pollination, that is mainly related to pollen transfer, can be either abiotic or biotic (Glover 2007, Yang 2012). In the former type of pollination, pollen is transferred via water and/or wind diffusion. Perhaps the most well-known example of abiotic pollination is the grass (Glover 2007, Yang 2012). Typically, abiotic pollination takes place in short distances. Therefore, it can be considered as a local optimization mechanism (Yang 2012). In biotic pollination, pollen transfer occurs by insects and animals (e.g. bees, birds, bats, butterflies) that are called pollinators. Pollinators are able to fly rather long distances. Hence, biotic pollination can be assumed as a global optimization scheme (Yang 2012). It is also worth noting that the flight behaviour of pollinators shares similar characteristics to Lévy flights (Pavlyukevich 2007, Yang 2012). An additional feature of flower pollination is the so-called flower constancy. More specifically, some pollinators have the tendency to select specific flower species and bypass others (Yang 2012). In this manner, pollinators transfer more pollen to the same species ensuring guaranteed nectar intake and avoiding the risks related to exploring new flower species. The afore-described characteristics of flower pollination process have been idealized in the following basic rules of FPA:

1. Biotic pollination is treated as a global optimization mechanism with pollinators performing Lévy flights.
2. Abiotic pollination is treated as a local optimization mechanism.
3. Flower constancy is considered by assuming that the reproduction probability is proportional to the similarity of flowers involved.
4. The type of pollination mechanism (biotic or abiotic) is governed by a random switching probability p in [0, 1].

In the following, for simplicity, it is assumed that each plant develops one flower, which produces only one pollen gamete (Yang 2012). Following this assumption, there exists no need to differentiate between pollen gametes, flowers and plants.

In FPA, a solution vector $\boldsymbol{x_i}$ is represented by a flower i. The algorithm applies two different search procedures: the global and local pollination. Following the first and third rules of FPA, the global pollination procedure could be represented mathematically by the following equation:

$$x_i^{t+1} = x_i^t + \gamma \cdot L(\lambda) \cdot (g^* - x_i^t), \tag{3}$$

where $\boldsymbol{x_i^t}$ represents flower i at iteration t, $\boldsymbol{g}^*$ is the best flower of all the population of flowers at iteration t, λ is a constant, γ is a scaling factor to control the step size, $L(\lambda) > 0$ is the Lévy flight step size that represents the strength of the pollination and is drawn from the Lévy distribution given below, where $\Gamma(\lambda)$ is the standard gamma function and s > 0.

$$L \sim \frac{\lambda \Gamma(\lambda) \sin\left(\frac{\pi\lambda}{2}\right)}{\pi} \cdot \frac{1}{s^{1+\lambda}}, \quad (s > 0), \tag{4}$$

Set objective min $f(\boldsymbol{x})$, $\boldsymbol{x} = (x_1, x_2, \ldots, x_d)$
Initialize a population of n flowers with random procedures.
Evaluate objective function values of the initial population.
Determine the best solution $\boldsymbol{g}^*$ of the initial population.
Determine the value of switch probability $p \in [0, 1]$
while ($t < MaxIteration$)
    for $i = 1 : n$ (for all flowers of the population)
        if rand $< p$
            Draw a d-dimensional Lévy distribution step vector L
            Do global pollination by $x_i^{t+1} = x_i^t + \gamma \cdot L(\lambda) \cdot (g^* - x_i^t)$
        else
            Draw ε from a uniform distribution in [0, 1]
            Select randomly j and k among all flowers of the population.
            Do local pollination by $x_i^{t+1} = x_i^t + \varepsilon \cdot (x_j^t - x_k^t)$
        end if
        Evaluate objective function values of new solutions.
        When better than previous, update new solutions in the population.
    end for
    Determine the best solution $\boldsymbol{g}^*$ of the new population.
end while

Fig. 2: Pseudo-code of FPA

On the other hand, the local pollination rule (second rule) and flower constancy (third rule) are represented by the following equation, where $\boldsymbol{x_j^t}$ and $\boldsymbol{x_k^t}$ are different flowers of the same population and ε is drawn from a uniform distribution in [0, 1].

$$x_i^{t+1} = x_i^t + \varepsilon \cdot \left(x_j^t - x_k^t \right). \tag{5}$$

Following the fourth rule, the type of flower pollination (local or global) is controlled by a switch probability p in [0, 1]. Summarizing the previous information, the pseudo code of FPA is shown in Fig. 2, where d represents the number of problem dimensions and n is the size of flowers population.

3.2 Modifications to the original FPA

The original FPA, shown in Fig. 2, has been formulated for unconstrained optimization problems. To address the design constraints of RC building frames in the optimization framework of §2.1, the penalty function approach is adopted herein. According to this approach, a large (penalty) value is added to the objective function when the design constraints are not satisfied. Moreover, the original FPA considers only continuous design variables. To account for the fact that the design variables x_i, in the optimization framework of the present study, assume only integer values, the continuous design variables of the original FPA are rounded to their nearest integer values.

The computational cost of the present optimization framework is increased due to the interface between SAP2000 and MATLAB and the requirement of conducting detailed structural designs of complex 3D RC building frames. The latter involves running multiple structural (FEA) analyses and calculating the required steel reinforcement at numerous locations in order to determine the objective function (see Eq. 2) and check the design constraints of each trial design. For example, the computational time required, on average, for a trial design of the 4-storey and 12-storey buildings examined later in this study are approximately 7s and 17s, respectively, on a personal computer using one core of an Intel i5–7500 processor with operating frequency 3.40 GHz. Therefore, it is important that the calls to SAP2000 and the subsequent structural design calculations are avoided when feasible.

In FPA, the position of each flower is updated only when it yields lesser costs than its previous location. To take advantage of this FPA provision, the cost of the new solution vector of each flower is first calculated assuming minimum steel reinforcement areas and prior to calling SAP2000 to conduct structural design. If this lower bound of the new cost is higher than the previous cost of the flower, then the position of the flower is not updated, and the corresponding structural analysis and design is avoided. Moreover, a memory matrix is used in the developed framework that stores all previous trial designs and their corresponding costs.

If the optimization algorithm returns to a previous design solution, then the cost of this design solution is revoked from the memory matrix instead of repeating the structural analysis and design calculations for the same design solution. To accommodate all previous considerations, a modified version of the FPA algorithm is employed in this study as presented in Fig. 3.

```
Set objective min f(x), x = (x₁, x₂, …, x_d)
Initialize a population of n flowers with random procedures.
Do structural design and evaluate RC frame costs including possible penalty terms of the initial population.
Determine the best solution g* of the initial population.
Determine the value of switch probability p ∈ [0, 1]
while (t < MaxIteration)
        for i = 1 : n (for all flowers of the population)
            if rand < p
                Draw a d-dimensional Lévy distribution step vector L
                Do global pollination by x_i^{t+1} = round(x_i^t + γ · L(λ) · (g* − x_i^t))
            else
                Draw ε from a uniform distribution in [0, 1]
                Select randomly j and k among all flowers of the population.
                Do local pollination by x_i^{t+1} = round(x_i^t + ε · (x_j^t − x_k^t))
            end if
            if x_i^{t+1} coincides with a previous design
                Get objective function value from memory matrix.
            else
                Evaluate cost of RC frame assuming minimum reinforcement ratios.
                if RC frame cost with minimum reinforcement ratios is lower than previous solution
                    Do structural design and evaluate RC frame cost including possible penalty terms.
                end if
            end if
            When better than previous, update new solutions in the population.
        end for
        Determine the best solution g* of the new population.
    end while
```
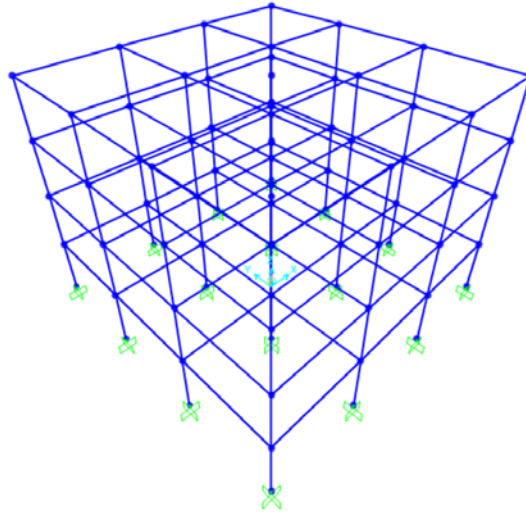
Fig. 3: Pseudo-code of modified FPA

4 Case studies

4.1 Four-storey RC building frame

The structure under examination in this section is a four-storey regular RC building 3D frame with storey height of three meters (Fig. 4). The building is doubly symmetric in plan and consists of 3 bays in each direction with an equal span length of 5m. Concrete C25/30 and reinforcing steel B500C materials are used following EC2 (CEN 2000) specifications. Concrete cover to longitudinal rebars centroid is assumed to be 50mm. For simplicity and

because of double symmetry, one rectangular section is used for all exterior beams of the first 3 storeys and one section for all interior beams of the same storeys. Furthermore, one additional rectangular beam section is applied to all exterior beams of the top storey as well as one additional section for all interior beams of the same storey. The beams of the top storey are considered different than the rest of the building to account for the increased dead loads of the roof as explained below. In addition to the beam sections, one section is assumed for all interior columns, one section for all corner columns and one section for all other perimeter columns. Therefore, 4 different beam sections and 3 different column sections are used in the RC frame leading to a total of 7 independent design variables in the optimization problem (i.e. $d = 7$). Concrete beams are assumed to have rectangular cross-sections and concrete columns square cross-sections. More particularly, beams assume sections from a list of 8 rectangular sections with a width of $b_b = 0.30$m and heights h_b increasing from 0.30m to 0.65m with a step of 0.05m (Fig. 1). Moreover, columns are assigned sections from a list of 8 possible square sections with sizes increasing from $h_c = 0.30$m to 0.65m with a step of 0.05m (Fig. 1). Therefore, the search space of this optimization problem is set to 8^7 potential design solutions.

Fig. 4: Four-storey 3D RC building frame

The structure is designed for both static loads and wind loads. Slab loads are transferred manually to the beams since slabs do not represent part of the structural model and they are not considered in the optimization problem. The total dead load of the slabs, inclusive of self-weight and additional dead load, is 6 kN/m^2 for all stories except for the top storey, where it becomes 16 kN/m^2 due to the existence of a rooftop garden. Slabs live load is assumed to be 5 kN/m^2 for all storeys apart from the top storey, where it is taken as 2 kN/m^2. In addition to the vertical loads, a lateral uniform wind pressure of 1.5 kN/m^2 is assumed to act to the external
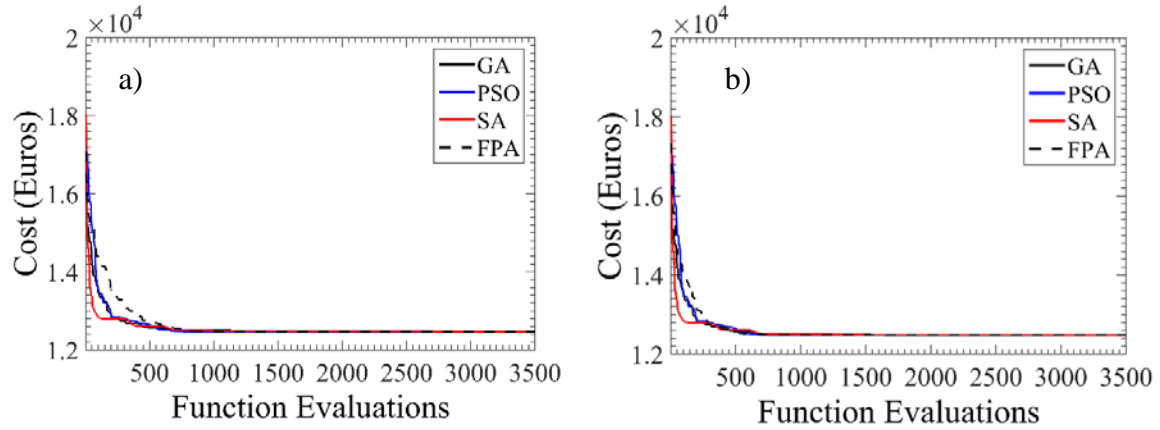
surfaces of the building. The structure is designed for static and wind loads according to the provisions of EC2 (CEN 2000). Material unit costs are taken as $f_{co} = 100 \ €/m^3$ and $f_{so} = 1 \ €/kg$ respectively.

Figure 5 shows comparisons of the mean optimization histories exhibited by the FPA and the GA, PSO and SA optimization algorithms after 3500 (= 500·d) objective function evaluations (i.e. trial structural design solutions). The GA (Holland 1975) is a metaheuristic optimization algorithm imitating Darwin's theory of evolution. GA gradually modifies populations (generations) of candidate solutions (individuals) until the improvement of next generations is below a pre-specified tolerance. Individuals of next generations (children) are formed from selected individuals of previous generations (parents) based on their objective function values. PSO (Kennedy 2011) is another metaheuristic population-based optimization algorithm inspired by the motion of bird flocks and schools of fish. Candidate solutions are represented by particles the movement of which is influenced by both their known local best positions and the global best-known position in the search-space. SA (Kirkpatrick et al. 1983) is inspired by the annealing process in metallurgy that involves heating a material and then gradually lowering the temperature to decrease defects, thus minimizing the energy of the system. To mimic the annealing process, SA iteratively moves a candidate solution according to a variable temperature parameter.

Five independent runs are applied for each optimization algorithm to account for the random procedures employed in each of these algorithms. For FPA, a population size of $n = 25$ flowers, a lambda value of $\lambda = 1.5$ and a scaling factor of $\gamma = 1$ are assumed in this comparison. Figure 5a shows the comparisons with FPA when the switch probability value is $p = 0.5$ and Fig. 5b when $p = 0.8$. For all other algorithms, default parameter values are used as specified in MATLAB R2020b – Global Optimization Toolbox (MathWorks 2020b).
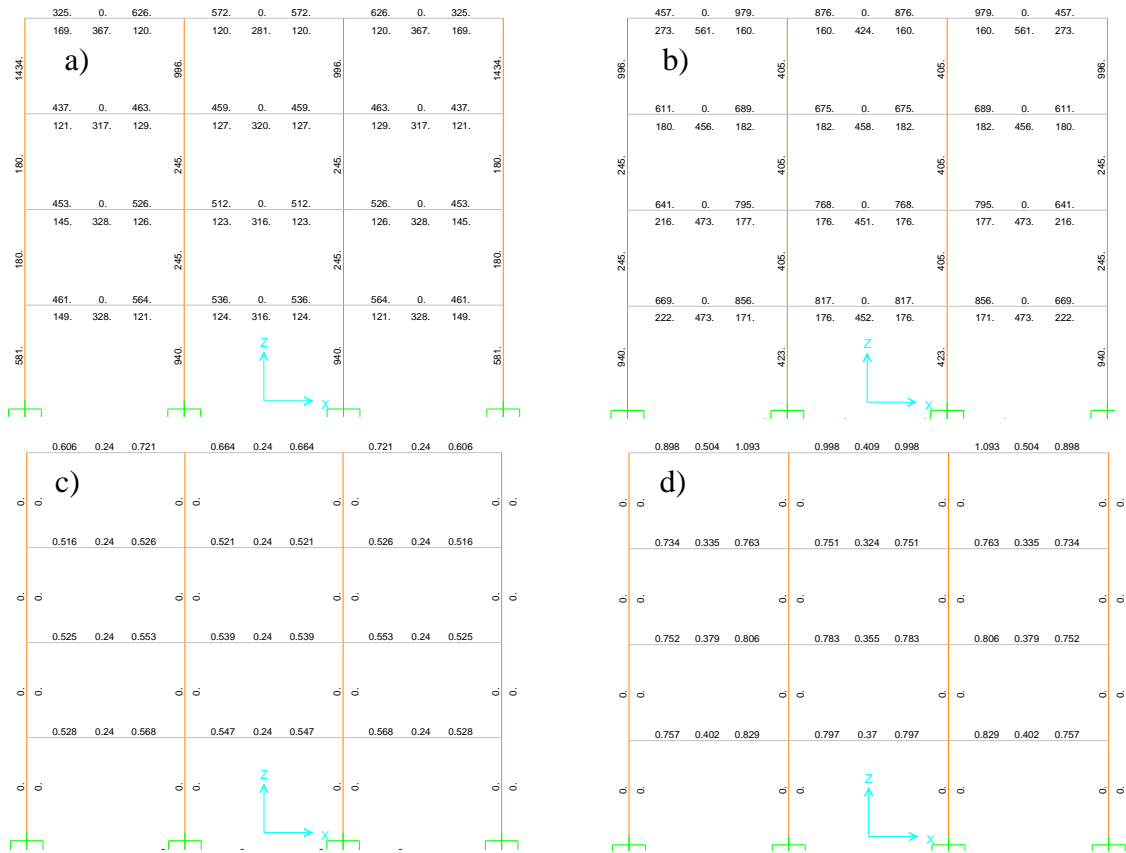
It is interesting to note in Fig. 5 that all mean optimization histories converge to almost the same minimum cost after approximately 700 function evaluations. The FPA with $p = 0.5$ seems to converge, on average, more slowly than the other algorithms but with $p = 0.8$ the FPA seems to be converging at a similar pace to the other algorithms. SA seems to be converging faster than all algorithms at the first steps of the analyses. It is also worth noting that all algorithms, after 5 independent runs, find the same optimum design solution with a minimum cost of approximately 12,477 Euros. Nevertheless, not all algorithm runs converge to the same optimum solution. For GA, SA and FPA with $p = 0.5$, the success rate is 100% meaning that all 5 independent runs of these algorithms converge to the best solution. For PSO and FPA with $p = 0.8$ the success rates are 80% and 60% respectively. When the latter algorithms did

not track the best solution, they converged to a slightly more expensive solution with only 0.1% higher cost.

Fig. 5: Comparison of FPA with other optimization algorithms with a) $p = 0.5$; b) $p = 0.8$

Table 1 presents the cross-sectional dimensions of the obtained optimum solution of the RC building frame. Furthermore, Fig. 6 shows the calculated flexural and shear steel reinforcement areas of the exterior and interior frames of the optimum solution of the concrete building as calculated by SAP2000. The zero shear reinforcement area requirements of the concrete columns mean that the minimum shear reinforcement areas must be placed in these members.

Fig. 6: Steel reinforcement areas of the obtained optimum design solution: a) flexural reinforcement (mm^2) - exterior frames; b) flexural reinforcement – interior frames (mm^2); c) shear reinforcement – exterior frames (mm^2/mm); d) shear reinforcement – interior frames (mm^2/mm)

16

Table 1: Optimum design solution cross-sections

Members group	Optimal Cross section (m)
Exterior beams – storeys 1 - 3	0.30 X 0.30
Interior beams – storeys 1 - 3	0.40 X 0.30
Exterior beams – storey 4	0.35 X 0.30
Interior beams – storey 4	0.45 X 0.30
Interior columns	0.45 X 0.45
Perimeter columns	0.35 X 0.35
Corner columns	0.30 X 0.30

In addition to the previous comparisons, Table 2 and Fig. 7 present the results of a parametric study conducted to identify the optimal FPA parameter setting (n, p, γ, λ) to address this optimization problem. In this parametric study, a reference parameter set with $n = 25$ flowers, $\lambda = 1.5$, $\gamma = 1$ and $p = 0.5$ is initially employed and then each parameter takes different values, while keeping the other parameters fixed, to investigate how it affects the performance of the algorithm.
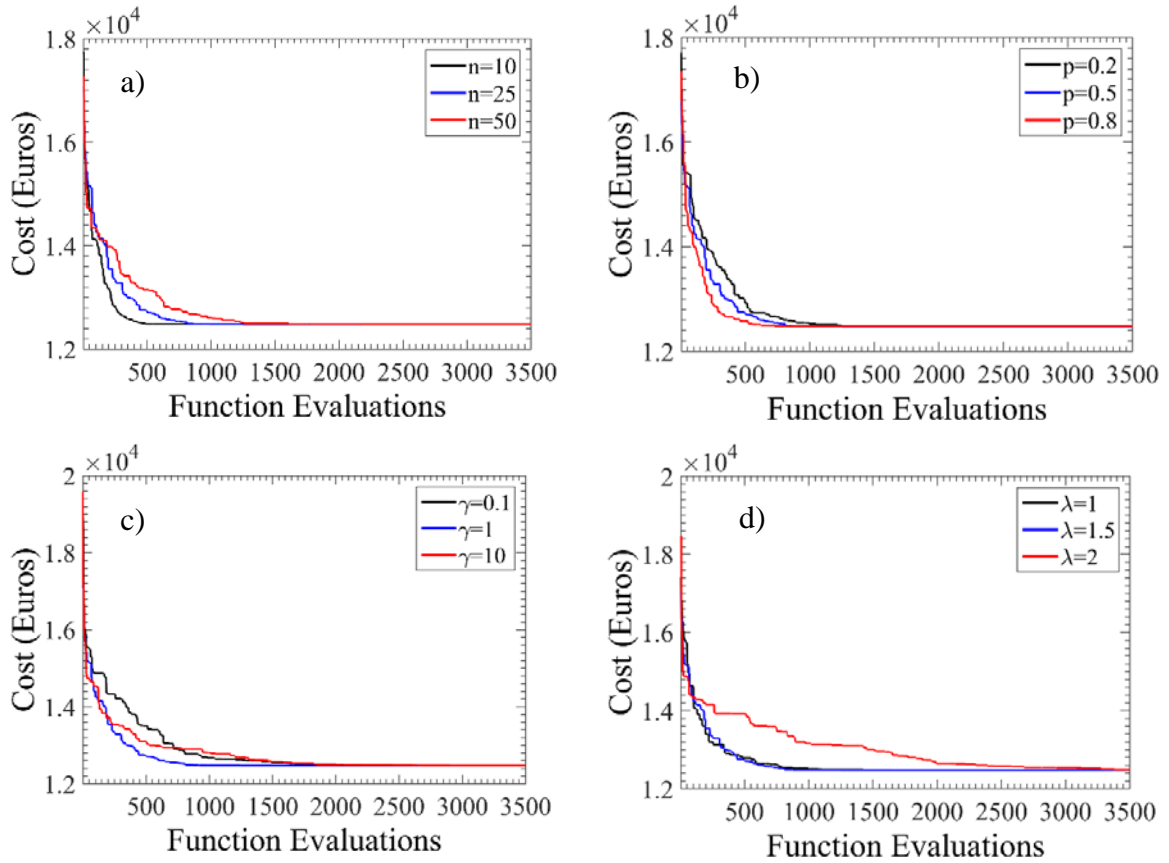
Figure 7a shows the effect of the population size on the convergence performance of the FPA with respect to function evaluations. It can be deducted that as the population sizes increases the rate of convergence decreases. Indeed, the population size of $n = 10$ demonstrates the highest rate of convergence. Table 2, however, shows that this population size demonstrates only 60% success rate in finding the best design solution as opposed to the larger population sizes with 100% success rates. Therefore, as n decreases the convergence rate increases but the reliability of the final solution decreases.

Furthermore, Fig. 7b presents the effect of the switch probability to the performance of the algorithm. It is seen that as the switch probability increases the speed of convergence of the algorithm is enhanced with the $p = 0.8$ value demonstrating best performance in the first iteration steps. However, the latter probability value achieves only 60% success rate when the other switch probabilities achieve 100%. Hence, increasing the switch probability seems to be improving the speed of the algorithm but it can also undermine the quality of the final solution. Moreover, Fig. 7c examines the influence of the scaling factor on the mean iteration histories of the FPA. It is noted that a scaling factor of $\gamma = 1.0$ offers the fastest convergence to the algorithm while maintaining 100% success rate in the final solutions. Therefore, this scaling factor value offers the best computational performance among the different γ values.

Table 2: FPA parametric analyses results

Parameters setting				Minimum cost	Mean cost	Success rate
n	p	γ	λ	(€)	(€)	
25	0.5	1	1.5	12477.0	12477.0	100%
25	0.8	1	1.5	12477.0	12480.6	60%
25	0.2	1	1.5	12477.0	12477.0	100%
50	0.5	1	1.5	12477.0	12477.0	100%
25	0.5	1	1.0	12477.0	12477.0	100%
25	0.5	1	2.0	12477.0	12487.4	20%
25	0.5	10	1.5	12477.0	12477.0	100%
25	0.5	0.1	1.5	12477.0	12477.0	100%
10	0.5	1	1.5	12477.0	12480.6	60%

Finally, Fig. 7d examines the role of λ values to the performance of the FPA. It is noted that lambda values of 1 and 1.5 offer very similar convergence performance that significantly exceeds the performance of $\lambda = 2$. The latter λ value has also only 20% success rate as opposed to 100% success rates for $\lambda = 1$ and 1.5. Hence, the λ values of 1 and 1.5 seem to be offering the best algorithm performance in this optimization problem.

Fig. 7: Mean optimization histories of FPA parametric analyses with respect to a) n; b) p; c) γ; d) λ
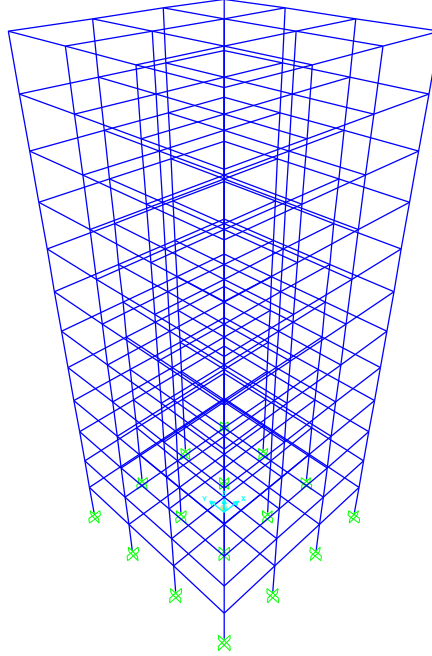
4.2 Twelve-storey RC building frame

The studied structure herein is a twelve-storey RC building 3D frame with storey height of three meters (Fig. 8). The building is doubly symmetric in plan and consists of 3 bays in each direction with an equal span length of five meters. Concrete C25/30 and reinforcing steel B500C materials are used in accordance with EC2 (CEN 2000) specifications. Concrete cover to longitudinal rebars centroid is assumed to be 50mm. For simplicity and because of double symmetry, one rectangular section is used for all exterior beams of two consecutive storeys and one section for all interior beams of two consecutive storeys. Furthermore, one section is assumed for all interior columns, one section for all corner columns and one section for all other perimeter columns. Therefore, twelve different beam sections and three different column sections are employed in this frame model leading to a total of 15 independent design variables in the optimization problem (i.e. $d = 15$).

Concrete beams are assumed to have rectangular and concrete columns square cross-sections. More particularly, beams assume sections from a list of 10 rectangular sections with a width of $b_b = 0.30$m and heights increasing from $h_b = 0.30$m to 1.20m with a step of 0.1m (Fig. 1). Moreover, columns are assigned sections from a list of 10 possible square sections with sizes increasing from $h_c = 0.30$m to 1.20m with a step of 0.10m (Fig. 1). Based on the previous, the search space of this optimization problem is set to 10^{15} potential design solutions.

The structure is designed for both static loads and seismic loads. Slab loads are transferred manually to the beams since slabs do not represent part of the structural model and they are not considered in the optimization problem. The total dead load of the slabs, inclusive of self-weight and additional dead load, is 6 kN/m^2 for all stories except for the top storey, where it becomes 16 kN/m^2 due to the existence of a rooftop garden. Slabs live load is assumed 2 kN/m^2 for all storeys.

The structure is designed for static loads according to the provisions of EC2 (CEN 2000). Furthermore, the building frame is designed for seismic loads in accordance with EC8 (CEN 2004) for ductility class low (DCL). The seismic action is represented by the Type 1 elastic response spectrum of EC8 (CEN 2004) for soil type D. The building is classified as importance class II and the corresponding importance factor amounts to one. The reference peak ground acceleration is equal to 0.36g that is representative of seismic zone III in Greece. A behaviour factor q of 1.5 is used for determining the design response spectrum. For the damage limitation requirement of EC8, it is assumed that inter-storey drifts are limited to 0.75% for the frequent design earthquake, which is required by EC8 for buildings having non-structural elements of

ductile materials attached to the structure (CEN 2004). Furthermore, to limit large lateral displacements the top displacement of the building is limited to 1% of the total height under the design earthquake.

Fig. 8: 12-storey 3D RC building frame

Figure 9 presents a comparison of the optimal materials costs predicted by the FPA and the GA, PSO and SA optimization algorithms after 7500 (= 500·d) function evaluations. Five independent runs are applied for each optimization algorithm to account for the random procedures employed in each of these algorithms. For FPA, a population size of $n = 25$ flowers, a switch probability of $p = 0.5$, a lambda value of $\lambda = 1.5$ and a scaling factor of $\gamma = 1$ are assumed in this comparison. For all other algorithms, default parameter values are used as specified in MATLAB R2020b – Global Optimization Toolbox (MathWorks 2020b).

Figure 9a compares, in the form of box plots, the obtained costs at the end of the 5 independent runs for the various optimization algorithms. The box plots show the minimum, maximum and median (red line) material costs in Euros. Inside the boxes, the 25th to 75th percentiles are contained. It is clear that the FPA provides significantly better minimum costs than all other
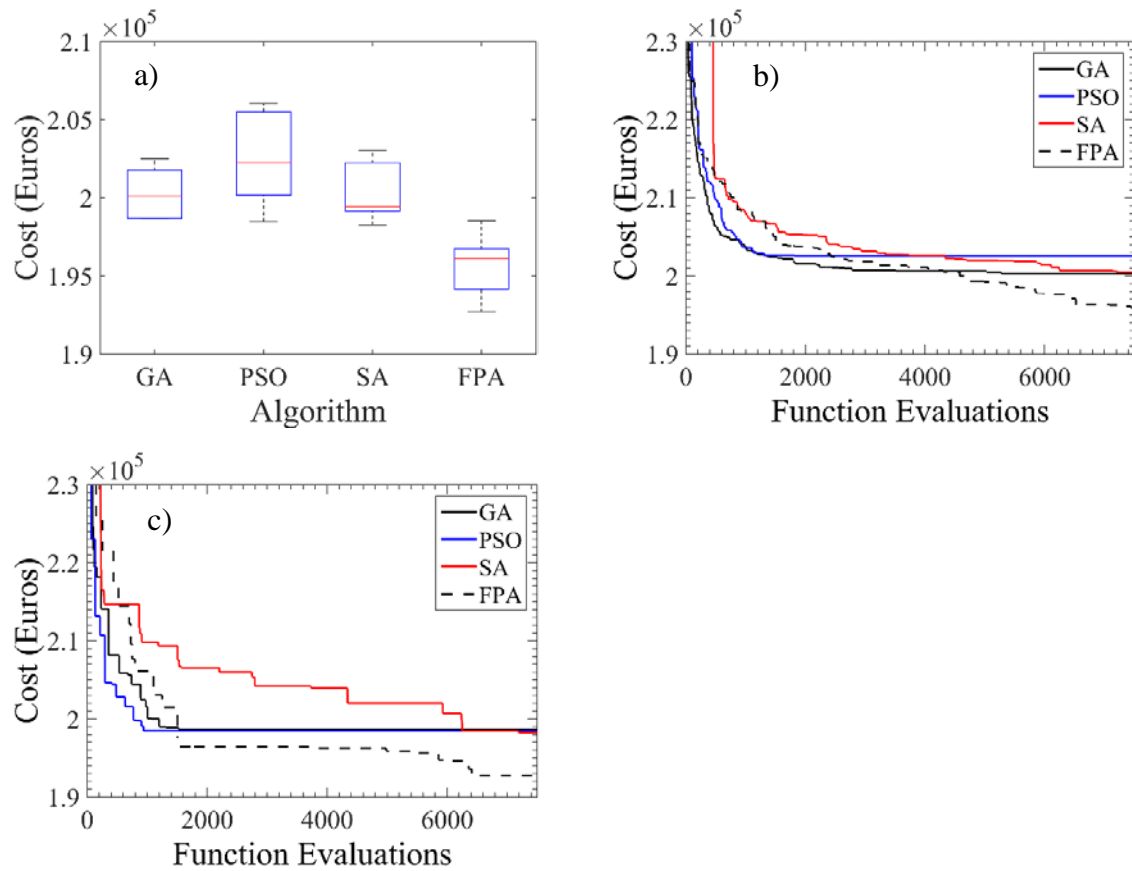
algorithms. As a matter of fact, it is found that almost all FPA runs offer better final costs than the best runs of all other algorithms.

Furthermore, Fig. 9b shows the mean convergence histories of the five independent runs of the different optimization algorithms. It is shown that the GA and PSO algorithms initially appear to converge faster to their optimal predictions. However, after a certain point (ranging between 1500 and 2000 function evaluations), it looks that they are trapped in local optima and they only marginally further improve their final costs. On the other hand, the FPA converges to the optimal solutions slower than the GA and PSO and at a similar pace to the SA algorithm in the first evaluation steps. Nevertheless, the FPA continues to gradually improve its predictions and it offers better predictions than all other algorithms after, approximately 4500 function evaluations on average. Interestingly, the FPA seems to be further and constantly improving its predictions even after this stage leading to importantly better average predictions at the end of the analyses. These results clearly demonstrate the high degree of diversification of this algorithm, which is able to track global optimum solutions in complex and large-scale problems, where other algorithms are trapped in local optima (Yang 2012).

Figure 9c shows the optimization histories of the runs of the different algorithms with the best final costs. The best solution found by FPA has a final cost of 192,694.3 Euros that is significantly lower than the best solutions of the other algorithms. The cross-sectional dimensions of this design solution are presented in Table 3. Furthermore, Fig. 10 shows the exterior and interior frames of the obtained optimum solution of the RC building with the corresponding cross-sections drawn to scale. As expected, section sizes are larger for the beams of the lower stories than the upper stories, due to the higher seismic demands at the lower stories. Furthermore, the sizes of the interior beams at the upper floors are higher than the exterior beams at the same floors since they undertake higher static loads. It is also interesting to observe that the perimeter columns have smaller sections than the interior columns and that the corner columns are smaller than the rest of the perimeter columns.

Figure 11 shows the lateral deflection response of the obtained optimum solution under the design earthquake. Similar response is observed for all frames in both directions due to the symmetry of the building. The top displacement is equal to 0.358 m (0.99%), which is marginally smaller than the 1% limit set by the design requirements. Furthermore, inter-storey drifts are less than 0.75% for the more-frequent earthquake. In addition, Table 4 shows the modal periods of vibration and the cumulative effective mass ratios for the modes contributing
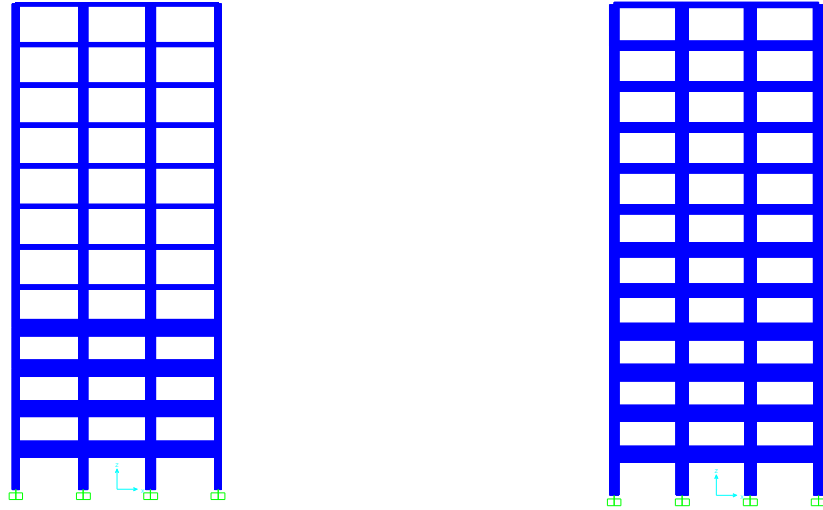
more than 90% of the total mass of the building. The fundamental period of vibration of the building is 1.099s. The base shear is equal to 8510.5 kN in both directions.

Fig. 9: Comparison of FPA with other optimization algorithms after 5 independent runs a) box plots of minimum costs; b) mean optimization histories; c) optimization histories of runs with best final costs

Table 3: Optimum design solution cross-sectional dimensions (in m)

Beams Groups		
Storeys	Exterior Beams	Interior Beams
1 - 2	1.2 X 0.3	1.2 X 0.3
3 – 4	1.2 X 0.3	1.2 X 0.3
5 - 6	0.3 X 0.3	1.0 X 0.3
7 - 8	0.3 X 0.3	0.7 X 0.3
9 - 10	0.3 X 0.3	0.7 X 0.3
11 - 12	0.3 X 0.3	0.4 X 0.3
Columns Groups		
Interior columns	0.9 X 0.9	
Perimeter columns	0.7 X 0.7	
Corner columns	0.5 X 0.5	

Fig. 10: Optimum design solution with cross-sections drawn to scale: a) exterior; b) interior frames

Fig. 11: Lateral deflection response of the optimum solution under the design earthquake (displacements in m)

Table 4: Modal responses of the obtained optimum design solution
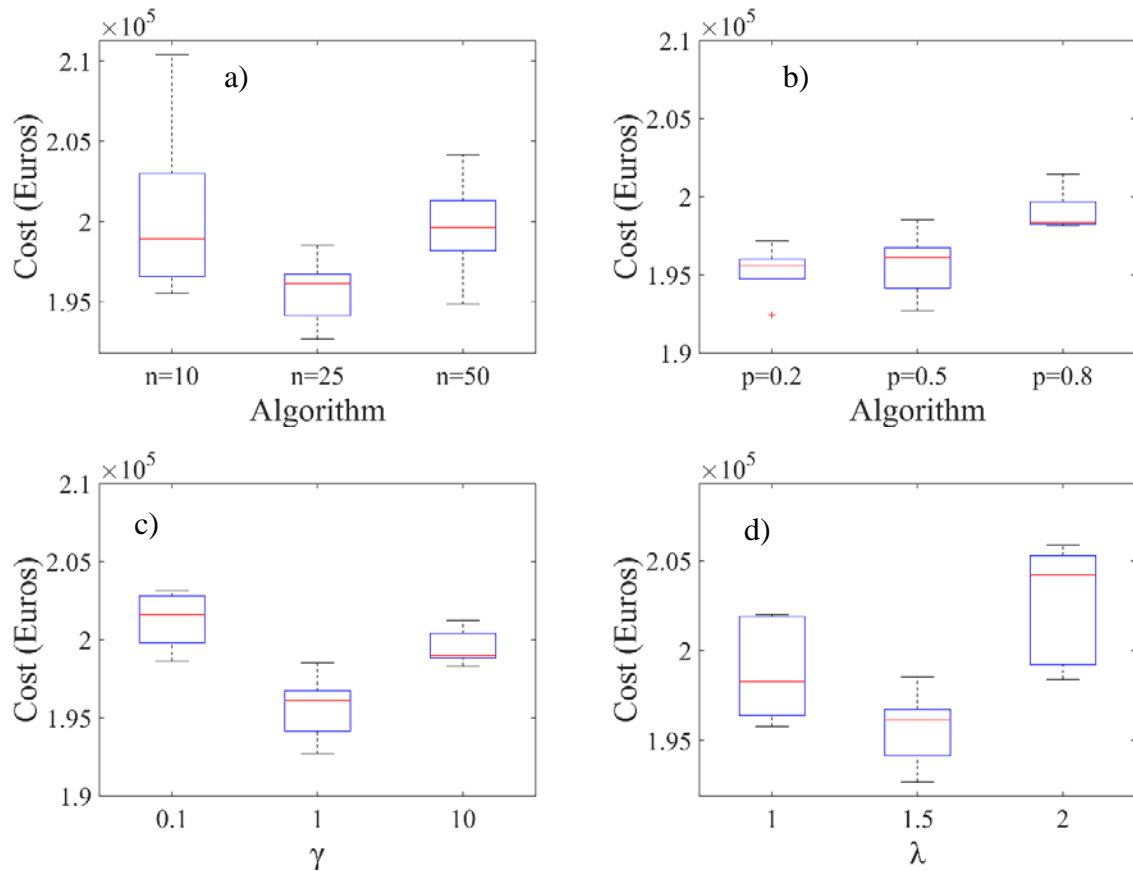
| Mode | Period (sec) | Cumulative effective mass ratios | |
		X-direction	Y-direction
1	1.099	0.302	0.349
2	1.099	0.651	0.651
3	0.361	0.687	0.811
4	0.361	0.847	0.847
5	0.356	0.847	0.847
6	0.205	0.898	0.869
7	0.205	0.921	0.921

In addition to the comparisons with other algorithms, Figs. 12 and 13 present the results of a parametric study conducted to identify the best FPA parameters settings for this optimization
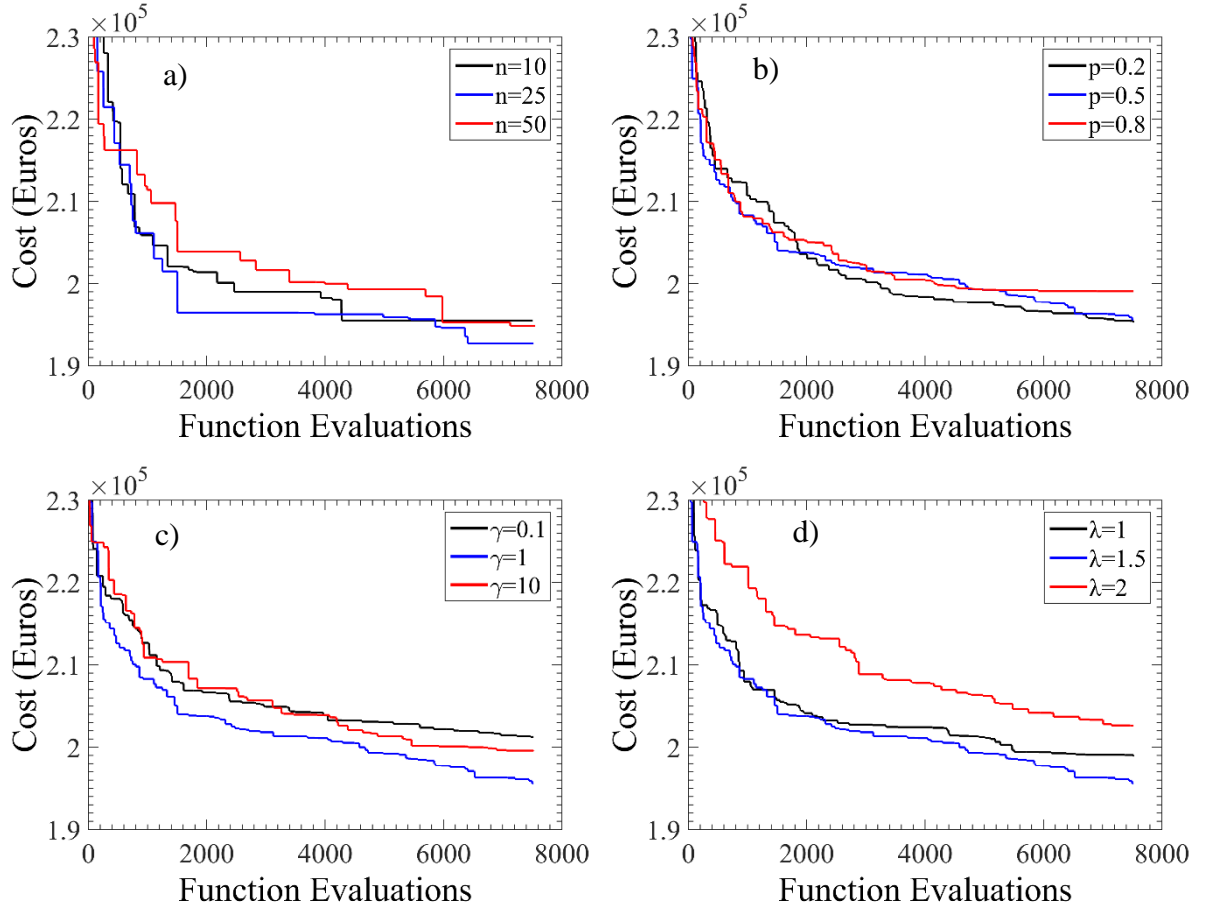
problem. Figure 12 presents box plots of the optimum costs obtained by FPA after 5 independent runs with 7500 maximum function evaluations and Fig. 13 the corresponding mean optimization histories. Similarly to the previous worked example, a reference parameter setting with $n = 25$ flowers, $\lambda = 1.5$, $\gamma = 1$ and $p = 0.5$ is initially used and then each parameter assumes different values to investigate its effect on the computational performance of the algorithm.

Figures 12a and 13a show the effect of the population size on the performance of the FPA. It can be seen that $n = 25$ offers the best performance in terms of final costs while the convergence rate is similar to $n = 10$ in the first iteration steps. The $n = 50$ population size seems to be providing the slowest convergence to the optimum solution.

Moreover, Figs. 12b and 13b illustrate the influence of the switch probability to the computational performance of FPA. It is observed that the $p = 0.2$ and $p = 0.5$ values offer rather similar performances of FPA with the $p = 0.2$ value to be providing less scatter in the final solutions but to be slightly slower in the first iteration steps. The $p = 0.8$ switch probability yields the worst final costs but it seems to be performing well in the first iteration steps.

Fig. 12: Box plots of FPA parametric analyses after 5 independent runs with respect to a) n; b) p; c) γ; d) λ

Fig. 13: Mean optimization histories of FPA after 5 independent runs with respect to a) n; b) p; c) γ; d) λ

Furthermore, Figs. 12c and 13c examine the influence of the scaling factor γ on the performance of FPA. It is found that the $\gamma = 1$ scaling factor value not only drives to significantly better final costs, but it also converges faster than the other scaling factors.

Finally, Figs. 12d and 13d investigate the role of λ values to the performance of the FPA. Similarly to the previous RC frame, the $\lambda = 2$ value yields the worst computational performance. However, in this example, the $\lambda = 1.5$ parameter value seems to be providing significantly better final costs than $\lambda = 1$ despite the fact that the two λ values converge at a similar rate in the first calculation steps.

5 Conclusions

The Flower Pollination Algorithm (FPA) is a recently developed population-based metaheuristic optimization algorithm imitating the evolution mechanisms of flowering plants. FPA is characterised by simplicity in its formulation as well as high computational performance and it has been found to outperform other established optimization algorithms in a range of diverse optimization problems. Furthermore, the optimum design of real-world 3D

reinforced concrete building frame structures is a computationally challenging optimization problem, associated with massive economic and environmental impacts, that has only been addressed by a very limited number of research studies and optimization algorithms in the past. Therefore, it is meaningful to investigate the computational performance of the FPA in this optimization problem.

For the purposes of the present study, a new computationally efficient framework, supported by the new MATLAB application STROLAB (i.e. Structural Optimization Laboratory), is developed for the optimum design of 3D reinforced concrete building frames. STROLAB is employing, for the goals of structural analysis and design, the well-established integrated software SAP2000 (CSI 2020) via its Application Programming Interface (API).

The developed framework uses, for first time in this study, the FPA to the optimum design of real-world 3D reinforced concrete building frames after a set of modifications to its original formulation to improve the speed of calculations and to account for the constraints set by design regulations as well as the discrete nature of the design variables in this optimization problem. More specifically, the structural designs of a four-storey and a twelve-storey 3D RC building frames for minimum material costs and in accordance with Eurocode 2 (CEN 2000) and Eurocode 8 (CEN 2004) for Low Ductility Class provisions are examined. The performance of the FPA is compared with other well-established optimization algorithms such as SA, GA and PSO and conclusions are made with respect to its solution efficiency. Furthermore, a parametric study is conducted to determine the FPA parameter values that maximize its computational performance in this optimization problem.

It is found that the FPA obtains similar results to the well-established optimization algorithms for the small-scale optimization problem of the four-storey RC frame and that it outperforms these algorithms for the larger-scale optimization problem of the twelve-storey building. This finding confirms the high degree of diversification of this algorithm that enables it to track global optima of complex and large-scale problems, where other algorithms are trapped in local optimum solutions (Yang 2012). It is also worth noting that FPA may be converging slower than the other algorithms at the first iteration steps, but this can be improved by selecting appropriate values for its controlling parameters.

Moreover, parametric studies show that the FPA converges faster as the population size decreases. However, very small population sizes may not drive to the best designs at the end of the analyses. A population size of $n = 25$ found to offer a good compromise between speed of convergence and robustness in this study. The switch probability p is another parameter affecting the performance of FPA. It is found that as p increases the convergence rate of the

FPA may increase in the first iteration steps. Nevertheless, very high p values may undermine the quality of the final solutions. Herein, a switch probability value of $p = 0.5$ found to offer good combinations of convergence speed and quality of final solutions for the FPA algorithm. In terms of the scaling factor γ, it is observed that $\gamma = 1$ offers the best computational performance in terms of both convergence rate and quality of final solutions. The same holds for the λ parameter when this is set as $\lambda = 1.5$.

References

[1] Abdel-Basset M, Shawky LA (2018) Flower pollination algorithm: a comprehensive review. Artif Intell Rev 52:2533-2557

[2] Abdel-Raouf O, El-Henawy I, Abdel-Basset M (2014) A novel hybrid flower pollination algorithm with chaotic harmony search for solving sudoku puzzles. Int J Mod Educ Comput Sci 6:38

[3] Abdelaziz A, Ali E, Elazim SA (2016a) Combined economic and emission dispatch solution using flower pollination algorithm. Int J Electr Power Energy Syst 80:264–274

[4] Abdelaziz A, Ali E, Elazim SA (2016b) Implementation of flower pollination algorithm for solving economic load dispatch and combined economic emission dispatch problems in power systems, Energy 101:506–518

[5] ACI Committee 318 (2019) Building code requirements for structural concrete and commentary. Farmington Hills, MI:American Concrete Institute.

[6] Akin A, Saka MP (2015) Harmony search algorithm based optimum detailed design of reinforced concrete plane frames subject to ACI 318-05 provisions. Comput Struct 147:75-95

[7] Alyasseri ZAA, Khader AT, Al-Betar MA, Awadallah MA, Yang XS (2018) Variants of the flower pollination algorithm: A review. Studies in Computational Intelligence, Springer, 91-11

[8] Balling RJ, Yao X (1997) Optimization of reinforced concrete frames. J. Struct. Eng ASCE 123:193-202

[9] Bekdas G, Nigdeli SM, Yang XS (2015) Sizing optimization of truss structures using flower pollination algorithm. Appl Soft Comput 37:322-331

[10] Bekdas G (2018) New improved metaheuristic approaches for optimum design of posttensioned axially symmetric cylindrical reinforced concrete walls. Struct Des Tall Spec 27: e1461

[11] BSI (1997) BS 8110: Structural use of concrete, part 1, code of practice for design and construction. London

[12] CEN (2000) Eurocode 2: Design of concrete structures. Part 1-1: General rules and rules for buildings, Brussels: European Standard EN 1992-1-1

[13] CEN (2002) Eurocode 0: Basis of Structural Design. Brussels: European Standard EN 1992:2002

[14] CEN (2004) Eurocode 8: Design of structures for earthquake resistance. Part 1: General rules, seismic actions and rules for buildings, Brussels: European Standard EN 1990-1.

[15] CSI (2016) Concrete frame design manual: Eurocode 2-2004 with Eurocode 8-2004 for SAP2000. ISO SAP091415M29 Rev. 1, USA.

[16] CSI (2020) https://www.csiamerica.com/products/sap2000

[17] Dehnavipour H, Mehrabani M, Fakhriyat A, Jakubczyk-Galczynska A (2019) Optimization-based design of 3D reinforced concrete structures. Journal of soft computing in civil engineering 3:95-106

[18] Esfandiari MJ, Urgessa GS, Sheikholarefin S, Dehghan Manshadi SH (2018) Optimum design of 3D reinforced concrete frames using the DMPSO algorithm. Adv Eng Softw 115:149-160

[19] Fadaee MJ, Grierson DE (1996) Design optimization of 3D reinforced concrete structures. Struct. Optimization 12:127-134

[20] Glover BJ (2007) Understanding flowers and flowering: An integrated approach. Oxford University Press, UK

[21] Govindaraj V, Ramasany JV (2007) Optimum detailed design of reinforced concrete frames using genetic algorithms. Eng Optimiz 39:471-494

[22] Heng J, Wang C, Zhao X, Xiao L (2016) Research and application based on adaptive boosting strategy and modified CGFPA algorithm: a case study for wind speed forecasting. Sustainability 8:235

[23] Holland JH (1975) Adaptation in natural and artificial systems. An introductory analysis with application to biology, control and artificial intelligence. University of Michigan Press, Ann Arbor, MI

[24] Kaveh A, Behnam AF (2013) Design optimization of reinforced concrete 3D structures considering frequency constraints via a charged system search. Sci Ira Trans A 20: 387-396

[25] Kaveh A, Izadifard RA, Mottaghi I (2020a) Optimal design of planar RC frames considering CO_2 emissions using ECBO, EVPS and PSO metaheuristic algorithms. J Build Eng 28:101014

[26] Kaveh A, Izadifard RA, Mottaghi I (2020a) Cost optimization of RC frames using automated member grouping. International Journal of Optimization in Civil Engineering 10:91-100

[27] Kayabekir AE, Arama ZA, Bekdas G, Dalyan I (2020) Effects of soil geotechnical properties on the prediction of optimal dimensions of restricted reinforced concrete retaining walls. Hittite Journal of Science and Engineering 7:205-213

[28] Kennedy J (2011) Particle swarm optimization. Encyclopedia of Machine Learning, Springer, 760-766

[29] Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220: 671-680

[30] Lagaros ND (2014) A general purpose real-world structural design optimization computing platform. Struct Multidiscip O 49:1047-1066

[31] Lagaros ND (2018) The environmental and economic impact of structural optimization. Struct Multidiscip O 58:1751-1768

[32] Martins A, Simões L, Negrão J, Lopes A (2020) Sensitivity analysis and optimum design of reinforced concrete frames according to Eurocode 2. Eng Optimiz 52:2011-2032

[33] MathWorks (2020) https://uk.mathworks.com/products/matlab.html?s_tid=hp_products_matlab

[34] MathWorks (2020b) MATLAB R2020b – Global Optimization Toolbox. Natick, MA, USA

[35] Medeiros G, Kripka M (2014) Optimization of reinforced concrete columns according to different environmental impact assessment parameters. Eng Struct 59: 185-194

[36] Mergos PE (2018a) Efficient optimum seismic design of reinforced concrete frames with nonlinear structural analysis procedures. Struct Multidiscip O 58:2565-2581

[37] Mergos PE (2018b) Contribution to sustainable seismic design of reinforced concrete members through embodied CO2 emissions optimization. Struct Concrete 19:454–462

[38] Mergos PE (2018c) Seismic design of reinforced concrete frames for minimum embodied CO2 emissions. Energ Buildings 162:177–186

[39] Mergos PE, Mantoglou F (2020) Optimum design of reinforced concrete retaining walls with the flower pollination algorithm. Struct Multidiscip O 61:575-585

[40] Moss R, Brooker O (2006) How to design concrete structures using Eurocode 2: Beams. The Concrete Centre, Surrey, UK

[41] Nigdeli SM, Bekdaş G, Yang XS (2016a) Application of the flower pollination algorithm in structural engineering, Metaheuristics and Optimization in Civil Engineering, Springer, 25–42

[42] Nigdeli SM, Bekdaş G, Yang XS (2016b) Optimum tuning of mass dampers for seismic structures using flower pollination algorithm. International Journal of Theoretical and Applied Mechanics 1:264-268

[43] Paya-Zaforteza I, Yepes V, Gonzalez-Vidosa F, Hospitaler A (2008) Multiobjective optimization of concrete frames by simulated annealing. Comput-Aided Civ Inf 23:596-610

[44] Pavlyukevich I (2007) Lévy flights, non-local search and simulated annealing. J Comput Phys 226:1830-1844

[45] Sahab MG, Ashour AF, Toropov VV (2005) Cost optimization of reinforced concrete flat slab buildings. Eng Struct 27:313-322

[46] Sarma KC, Adeli H (1997) Cost optimization of concrete structures. J Struct Eng 124:570-578.

[47] Sharafi P, Muhammad NS, Hadi M (2012) Heuristic approach for optimum cost and layout design of 3D reinforced concrete frames. J Struct Eng 138:853-863

[48] Singh U, Salgotra R (2018) Synthesis of linear antenna array using flower pollination algorithm, Neural Comput Appl 29:435-445

[49] Toklu YC, Kayabekir AE, Bekdas G, Nigdeli SM, Yücel M (2020) Analysis of plane-stress systems via total potential optimization method considering nonlinear behaviour. J. Struct. Eng ASCE 146:04020249

[50] Ulusoy S, Bekdas G, Nigdeli SM (2020) Active structural control via metaheuristic algorithms considering soil-structure interaction. Structural Eng Mech 75:175-191

[51] Walker M (2009) How flowers conquered the world BBC Earth News. http://news.bbc.co.uk/earth/hi/earth_news/newsid_8143000/8143095.stm

[52] Yang XS (2008) Nature-inspired metaheuristic algorithms. Luniver Press, UK

[53] Yang XS, Deb S (2009) Cuckoo search via Lévy flights. Proc. of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), India

[54] Yang XS (2010) Firefly algorithm, stochastic test functions and design optimization. Int J Bio-inspir Com 2:78-84

[55] Yang XS (2012) Flower pollination algorithm for global optimization. Unconven Comput Nat Comput 7445:240-249

[56] Yeo D, Gabbai R (2011) Sustainable design of reinforced concrete structures through embodied energy optimization. Energ Buildings 43:2028-2033

[57] Yücel M, Kayabekir AE, Bekdaş G, Nigdeli SM, Kim S (2021) Adaptive-hybrid harmony search algorithm for multi-constrained optimum eco-design of reinforced concrete retaining walls. Sustainability 13:1639

[58] Zhou Y, Zhang S, Luo Q, Wen C (2016) Using flower pollination algorithm and atomic potential function for shape matching. Neural Comput Appl 29:21–40