OPTIMUM DESIGN OF STEEL MOMENT FRAMES USING AN ARTIFICIAL BEE COLONY ALGORITHM (ABC-FB)

H. Forouzani¹ and M. Obeydi²

ABSTRACT

A building design in today’s competitive world, with its limited material resources, should be more economical yet efficient, with adequate seismic resistance. To achieve this goal, different methods for optimizing the weight of structures are presented in order to select the lightest sections from a list of available profiles that satisfy the existing provisions in design codes and specifications. This article uses the proposed Karaboga’s modified artificial bee colony algorithm to solve numerical problems while discretely optimizing steel frames under gravity and lateral loads. The fly-back mechanism technique is used for constraint handling. Ultimately using a basic practical example that has been used in similar studies, the performance of this algorithm in solving problems is evaluated and compared to other nature-inspired algorithms such as the ant colony algorithm and the particle swarm algorithm. The results indicate that the quicker convergence rate and the lighter weight are comparable to other algorithms.

¹Graduate Researcher, Dept. of Civil Engineering and Applied Mechanics, California State University, Northridge
²Ph.D. Student, Dept. of Civil Engineering, University of Isfahan, Isfahan, Iran, (email: m_obeydi83@yahoo.com)

Optimum Design of Steel Moment Frames Using an Artificial Bee Colony Algorithm (ABC-FB)

H. Forouzani\textsuperscript{1} and M. Obeydi\textsuperscript{2}

ABSTRACT

A building design in today’s competitive world, with its limited material resources, should be more economical yet efficient, with adequate seismic resistance. To achieve this goal, different methods for optimizing the weight of structures are presented in order to select the lightest sections from a list of available profiles that satisfy the existing provisions in design codes and specifications. This article uses the proposed Karaboga’s modified artificial bee colony algorithm to solve numerical problems while discretely optimizing steel frames under gravity and lateral loads. The fly-back mechanism technique is used for constraint handling. Ultimately using a basic practical example that has been used in similar studies, the performance of this algorithm in solving problems is evaluated and compared to other nature-inspired algorithms such as the ant colony algorithm and the particle swarm algorithm. The results indicate that the quicker convergence rate and the lighter weight are comparable to other algorithms.

Introduction

Minimizing the structural weight by considering existing provisions in design codes and specification charts is the main design goal. Using optimization methods allows the designer to obtain better results while reducing cost and time. Over time, evolutionary algorithms based on nature have demonstrated good results in comparison with classical optimization methods. Additionally, classical methods are applicable only for differentiable functions, and in each problem, they must adhere to a specific procedure, whereas the modern algorithms, with slight modifications, are applicable to all kinds of problems. Optimization problems can be classified using various methods, such as constrained and unconstrained as well as continuous and discrete methods. Structural problems are constrained, and due to the selection of variables from a list of available profiles, they will become discrete.

\textsuperscript{1}Graduate Researcher, Dept. of Civil Engineering and Applied Mechanics, California State University, Northridge
\textsuperscript{2}Ph.D. Student, Dept. of Civil Engineering, University of Isfahan, Isfahan, Iran, (email: m_obeydi83@yahoo.com)

Thus far, many studies on sizing optimization of steel frames have been conducted. Kaveh and Shojaee [1] employed the ant colony optimization (ACO) algorithm for the optimal design of skeletal structures, such as frames, under gravity and lateral loads, and they compared their results with other optimization algorithms. Memory, which improves future searches by storing past search data, is the main characteristic of ACO. Another ACO capability is the probability function obtained by the pheromone on the variables in selecting a new design. In the same year, Kaveh and Talatahari [2] presented a discrete version of particle swarm ant colony optimization (DPSACO) algorithm for the weight optimization of frame structures. They also defined a new formula for particle velocity to improve the exploration in the proposed algorithm, and they compared their results with other algorithms. In 2010, they used the imperialist competitive algorithm (ICA) as a new metaheuristic algorithm to optimize skeletal structures such as steel frames, and they demonstrated the effectiveness of the proposed method [3]. In addition, Kaveh et al. [4] used ACO for performance-based seismic design of steel frames and showed the superiority of the ACO for this type of optimization problem. Doğan and Saka [5] applied the PSO algorithm to the discrete optimization of the weight of steel frames and handled constraints using the fly-back mechanism which was introduced by He et al. [6]. The results showed that the method was applicable to the design of steel frames. Gholizadeh [7] Gholizadeh and Poorhoseini [8, 9] implemented a modified firefly algorithm using a new neural network model, bat algorithm and an improved dolphin echolocation algorithm to find performance-based optimum seismic design of steel moment frames and steel dual braced frames at the performance levels. The results illustrated that these algorithms can be reliably used for performance-based seismic design of steel structures. Kaveh and Ghazzan [10] improved a new enhanced whale optimization algorithm (EWOA) in sizing optimization of steel frames and trusses which had more efficiency than its original version. Kaveh et al. [11] implemented the enhanced colliding bodies (ECBO) algorithm and the PSO algorithm to optimize steel frames considering element sections and the connection types under the earthquake lateral loads and gravity loads. The results showed that the ECBO algorithm is much better than the PSO algorithm for this type of problems. They also revealed more efficiency of simultaneous optimization of connections and frames sections with respect to the frame section optimization. Maheri et al. [12] improved a new enhanced honey bee mating optimization (EHBMO) algorithm for design of side sway steel frames. They compared the results with the other algorithms and the original HBMO and found the lightest design in all investigated problems.

The artificial bee colony (ABC) algorithm is one of the most efficient metaheuristic algorithms and belongs to the subcategory of swarm intelligence methods. First presented by Karaboga [13], this algorithm simulates the behavior of honeybees in nature as they forage the food, and it was modified and improved to be applicable to different scientific fields [14-18]. Recent applications of the ABC algorithm in structural design have been used as a powerful technique for the optimization of trusses, space frames and placement of braces [19-23]. The purpose of this study is to apply a combination of ABC algorithms offered by Karaboga and Akay [18] and Sonmez [22, 23] based on the fly-back mechanism constraint-handling technique [6] to determine the optimum weight of frame structures, the most common structures in the civil engineering, under gravity and lateral loads, thus satisfying design constraints.

**Structural Optimization Problem**

In general, each optimization problem is composed of three parts: the objective function, the
constraints and the optimum solution. With respect to the structural problem, the objective function could be the structural weight. Whereas the constraints are related to the stress and displacement, the optimum solution minimizes the objective function as follows:

$$\text{To minimize } W(\{X\}) = \sum_{i=1}^{mn} y_i x_i L_i$$

where $W(\{X\})$, $X$, $x_i$, $y_i$, $L_i$ and $mn$ are the structural weight, the vector of design variables, the $i$-th design variable, the unit weight of the $i$-th member, the length of the $i$-th member and the number of structural members, respectively. In this study, the optimal solution is the lightest section from a list of available profiles.

**Modeling Behavior of Real Bees vs. Artificial Bees**

Honeybees are social insects that live in a hive. Members of the hive include the queen, the worker bees and the male bees. The workers are responsible for gathering the nectar. As they do so, they scatter in flower farms around the hive in different directions and often fly long distances to explore and exploit food sources. The process of collecting food requires that the worker bees explore the areas surrounding the hive and taste the nectar of the flowers to assess their quality. In an artificial bee colony algorithm, the first diploid batch of bees represents the scout bees. They will become employed bees after finding their desired flowers. They inform other bees in the hive of the location and quality of the food sources. This information is communicated by performing a “waggle dance” [13]. The dance contains information such as the direction and the distance of the flowers and the quality of the nectar. The onlooker bees, after observing the dance, choose the higher quality flowers equivalent to the probability selection of the desired algorithm. The onlooker bees then fly to the location and choose the best nectar. Nectar quality is represented as fitness in an artificial bee colony algorithm. This process is repeated until the best flowers on the farm are selected. Some bees may not change their selected flowers, though they leave the initial flowers and search the trough of other flowers to avoid missing the best flower. These bees then become scout bees.

**The Artificial Bee Colony Algorithm With A Fly-Back Mechanism (ABC-FB)**

The first step is to determine the number of bees in a colony (total employed bees and onlooker bees, NP), the maximum number of searching cycles as a stopping criterion (MNC) and the number of specific iterations when there is no improvement in the solution (LIMIT). Using the following equation, which shows the matrix with NP rows and D columns, an initial population is randomly generated:

$$A_{ij} = \text{Fix}(A_{ij}^{\min} + \lambda_{ij}(A_{ij}^{\max} - A_{ij}^{\min}))$$

where Fix(x) is a function in Eq. 2 that fixes the value of members of x vector to the nearest permissible discrete value and $A_{ij}$, $\lambda_{ij}$, $A_{ij}^{\max}$, $A_{ij}^{\min}$, NP and D are, respectively, the $j$-th variable of the $i$-th desired solution, a random number between 0 and 1, the upper bound of the $j$-th variable, the lower bound of the $j$-th variable, the number of desired solutions and the number of design variables. Kaveh and Talatahahi [2] presented this discretization method. The structure is then analyzed for each existing solution and structural response under the obtained loads. With the
stress and displacement response, the existing solution in the population (food source) is confirmed to see whether it satisfies the constraints. If it does not, new solutions must be produced. After all the solutions satisfy the constraints, Eqs. 1 and 3 are solved to find the structural weight and fitness for each solution, respectively.

\[
\text{fitness}_i = \frac{1}{w_i(A_j)} \tag{3}
\]

In the ABC algorithm, half of the population is allocated to the employed bees and the other half to the onlooker bees. Using Sonmez’s suggestion [22, 23], half of the lightest weights are considered to be employed bees (\(SN = NP/2\)). The best (lightest) weight among existing weights is then selected. The steps for each of the solutions (\(i = 1,2, ..., SN\)) in the employed bees’ phase are repeated. For each solution, a new solution is obtained by changing one or more variables using Eq. 4. If no variables (due to a smaller value of the modification rate control parameter (MR) [18] than a random number between 0 and 1 for each variable \(j = 1,2, ..., D\)) are changed, one of them will be randomly chosen and replaced with the value in the neighborhood using Eq. 5:

\[
A_{ij}^{\text{new}} = \begin{cases} 
\text{Fix}\left(A_{ij} + \phi_{ij}(A_{ij} - A_{kj})\right), & \text{if } R_j < \text{MR} \\
\text{Fix}(A_{ij}), & \text{Otherwise}
\end{cases}
\]

\[
A_{ij}^{\text{new}} = \text{Fix}\left(A_{ij} + \phi_{ij}(A_{ij} - A_{kj})\right) \tag{4}
\]

\[
i = 1,2, ..., NP, \quad j = 1,2, ..., D, \quad k = 1,2, ..., NP
\]

where \(\phi_{ij}, R_j, k\) and \(\text{MR}\) are a random number between 1 and -1, a random number between 0 and 1, the index that is randomly selected and should be different from \(i\) and the modification rate control parameter, respectively. Because the variables are selected from a list of available profiles, the obtained values should be rounded to the nearest permissible value. This concept is evident in Eqs. 4 and 5. The structural weight and fitness are then calculated using Eqs. 1 and 3. If the fitness level of the new solution is better than that of the previous solution and it satisfies the constraints, then it will replace the previous solution. If the new solution does not satisfy the constraints, even if the fitness is better, then it will return to the previous solution. The concept of the fly-back mechanism technique maintains the feasible space. Using Eq. 6, the probability of the onlooker bees choosing each of the employed bees \(i = 1,2, ..., SN\) is calculated:

\[
P_i = \frac{\text{fitness}_i}{\sum_{i=1}^{NP} \text{fitness}_i} \tag{6}
\]

After the completion of this process, the onlooker bee phase is initiated. If the probability of choosing the founded food source (the solution) by each employed bee was more than a random value between 0 and 1, then an onlooker bee would select the food source. Otherwise, the onlooker bee chooses the found food source of another employed bee. Hence, a new solution is obtained by changing one or more variables in the previous solution using Eq. 4. If no variables are changed, one of them will be randomly chosen and replaced with the value in the
neighborhood using Eq. 5. The solutions are fixed to the nearest permissible discrete value in the profile list. The structural weight and fitness are then calculated using Eqs. 1 and 3. The best solution is chosen through greedy selection from among the new and previous solutions, and if it satisfies the constraints, then it will replace the previous solution. The phase of scout bees is then initiated. One randomly selected solution among the solutions, which shows no improvement in the certain number of iterations (LIMIT) and is not the best solution, is replaced with a new feasible solution using Eq. 2. This process will not stop until it reaches the stopping criterion. The artificial bee colony’s flowchart is shown in Fig. 1.

Figure 1. Flowchart of the artificial bee colony with fly-back mechanism.

**Numerical Example**

In this section, the ABC-FB algorithm results are compared with other algorithms using a three-bay ten-story steel frame (also used in other articles), shown in Fig. 2. The applied loads, frame
member dimensions and grouping are represented in the figure. Saka and Kameshki [24] first analyzed this frame based on displacement and the AISC combined strength constraints. The structure is analyzed using the displacement method. The optimization algorithm and structural analysis were programmed using MATLAB. The size of the bee colony \( NP = 50 \), \( MNC = 1000 \) and \( \text{LIMIT} = MNC/3 \) are used. The 70 members of the structure have been categorized into nine groups. The materials’ elastic modulus is denoted by \( E = 200 \text{GPa} \) (29000 ksi), and the yield stress is denoted by \( F_y = 248.2 \text{MPa} \) (36 ksi).

The three-bay ten-story frame [24].

The constraints used in this example are the strength constraints (Eqs. 7 and 8) and the displacement constraints (Eqs. 9 and 10) according to LRFD-AISC provisions [25]:

\[
\frac{P_u}{2\phi_c P_n} + \left( \frac{M_{ux}}{\phi_b M_{nx}} + \frac{M_{uy}}{\phi_b M_{ny}} \right) \leq 1.0, \text{ for } \frac{P_u}{\phi_c} < 0.2 \\
\frac{P_u}{\phi_c P_n} + \frac{8}{9} \left( \frac{M_{ux}}{\phi_b M_{nx}} + \frac{M_{uy}}{\phi_b M_{ny}} \right) \leq 1.0, \text{ for } \frac{P_u}{\phi_c} \geq 0.2
\]

where \( P \), \( P_n \), \( \phi_c \), \( M_{ux}, M_{uy}, M_{nx}, M_{ny} \) and \( \phi_b \) are, respectively, the required axial strength (tension or compression), the nominal axial strength (tension or compression), the resistance factor (\( \phi_c = 0.9 \) for tension, \( \phi_c = 0.85 \) for compression), the required flexural strength in the x direction, the required flexural strength in the y direction, the nominal flexural strength in the x direction, the nominal flexural strength in the y direction (for two-dimensional structures; \( M_{ny} = 0 \)) and the flexural resistance reduction factor (\( \phi_c = 0.9 \)).

\[
\frac{\Delta T}{H} \leq R, \quad \text{Maximum lateral displacement constraint}
\]
\[ \frac{\Delta_j - \Delta_{j-1}}{h_j} \leq R_I, \quad j = 1, 2, \ldots, ns, \]  
Inter-story displacement constraint \ (10) 

where \( \sigma_i, \sigma^a_i, \Delta_T, H, R, \Delta_j, h_j, R_I \) and \( ns \) are, respectively, the stress in the \( i \)-th member, the allowable stress in the \( i \)-th member, the maximum lateral displacement, the height of the structure, the maximum drift index (1/300), the lateral displacement of the \( j \)-th story, the height of the \( j \)-th storey, the inter-story drift index permitted by the code of the practice (1/300) and the number of stories.

As the results in Table 1 indicate, the ABC-FB algorithm was able to reach an optimal design of 178.05 kN (40.03 kips), which is lighter than other algorithm results. The algorithm identified the optimal solution in 20 design runs, and there was no difference between the best and the worst solution. In addition to a lighter design, the algorithm found the optimal solution in 284 iterations in the best case, which is shown in Fig. 3. The ABC-FB algorithm requires an average of 343 iterations to converge the best solution. The DPSACO algorithm [2] found an optimal solution of 211.41 kN (47.53 kips) in 692 iterations.

Table 1. Optimal design results for the three-bay ten-story frame

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1 A_1</td>
<td>W30 × 99</td>
<td></td>
<td>W14 × 74</td>
<td>W30 × 99</td>
<td>W12 × 72</td>
<td>W21 × 57</td>
</tr>
<tr>
<td>2 A_2</td>
<td>W12 × 96</td>
<td>W14 × 90</td>
<td>W12 × 87</td>
<td>W21 × 101</td>
<td>W30 × 99</td>
<td></td>
</tr>
<tr>
<td>3 A_3</td>
<td>W14 × 82</td>
<td>W10 × 49</td>
<td>W14 × 43</td>
<td>W21 × 48</td>
<td>W18 × 40</td>
<td></td>
</tr>
<tr>
<td>4 A_4</td>
<td>W18 × 76</td>
<td>W12 × 72</td>
<td>W21 × 83</td>
<td>W24 × 68</td>
<td>W24 × 62</td>
<td></td>
</tr>
<tr>
<td>5 A_5</td>
<td>W8 × 31</td>
<td>W14 × 43</td>
<td>W10 × 39</td>
<td>W14 × 43</td>
<td>W16 × 26</td>
<td></td>
</tr>
<tr>
<td>6 A_6</td>
<td>W14 × 53</td>
<td>W18 × 65</td>
<td>W14 × 53</td>
<td>W18 × 65</td>
<td>W14 × 38</td>
<td></td>
</tr>
<tr>
<td>7 A_7</td>
<td>W8 × 28</td>
<td>W16 × 36</td>
<td>W8 × 28</td>
<td>W8 × 28</td>
<td>W12 × 16</td>
<td></td>
</tr>
<tr>
<td>8 A_8</td>
<td>W8 × 24</td>
<td>W12 × 30</td>
<td>W8 × 24</td>
<td>W8 × 24</td>
<td>W10 × 22</td>
<td></td>
</tr>
<tr>
<td>9 A_9</td>
<td>W21 × 44</td>
<td>W21 × 44</td>
<td>W18 × 40</td>
<td>W18 × 40</td>
<td>W18 × 35</td>
<td></td>
</tr>
<tr>
<td>Best. kN</td>
<td>235.08</td>
<td></td>
<td>221.69</td>
<td>215.19</td>
<td>211.41</td>
<td>178.05</td>
</tr>
<tr>
<td>(kips)</td>
<td>(52.85)</td>
<td></td>
<td>(49.84)</td>
<td>(48.38)</td>
<td>(47.53)</td>
<td>(40.03)</td>
</tr>
<tr>
<td>Average. kN</td>
<td>N/A</td>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>178.05</td>
</tr>
<tr>
<td>(kips)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(40.03)</td>
</tr>
<tr>
<td>Worst. kN</td>
<td>N/A</td>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>178.05</td>
</tr>
<tr>
<td>(kips)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(40.03)</td>
</tr>
</tbody>
</table>

In Fig. 3, a graph of the mean weights for twenty runs is included. According to Figs. 4 and 5, changing the modification rate control parameter from MR = 0.4 to 1.0 does not affect the best weight and standard deviation. Therefore, according to Table 2, MR = 0.7 is chosen.

Conclusions

In this article, the discrete optimization of steel frames under gravity and lateral loads was
researched by combined use of Sonmez’s ABC algorithm [22, 23] and Karaboga’s constrained algorithm [18]. Using the fly-back technique for constraint-handling and a combination of these algorithms led to the most optimized design. In addition, the convergence rate and the number of iterations required to reach the optimum solution significantly decreased. The proposed optimal solution was 18.73% more accurate and had 349 less average iterations than the best solution obtained by the other algorithms.

Figure 3. Optimization history graph for the three-bay ten-story frame (1 lb = 0.45359 kg)

Figure 4. Mean weight results with a changing modification rate control parameter for the three-bay ten-story frame (1 kips = 4.45 kN)

Figure 5. Standard deviation results with a changing modification rate control parameter for the three-bay ten-story frame (1 kips = 4.45 kN)
In this study, the effect of changing the modification rate control parameter on the optimal weight and standard deviation was studied. The very high and very low values of the MR disturb the balance between exploitation and exploration. Based on these results, a suitable value for MR was chosen and the optimum solution was obtained. The results of this study demonstrate the high performance and accuracy of this algorithm in solving structural problems.

Table 2. Optimal design results with a changing modification rate control parameter (MR) for the three-bay ten-story frame

<table>
<thead>
<tr>
<th>MR</th>
<th>Best solution. kN (kips)</th>
<th>Worst solution. kN (kips)</th>
<th>Average solution of 10 runs. kN (kips)</th>
<th>Standard deviation (SD). kN (kips)</th>
<th>Number of runs to find the optimal solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>183.29 (41.21)</td>
<td>226.93 (51.02)</td>
<td>195.30 (43.91)</td>
<td>14.15 (3.1823)</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>178.05 (40.03)</td>
<td>193.92 (43.60)</td>
<td>182.09 (40.94)</td>
<td>6.07 (1.3663)</td>
<td>3</td>
</tr>
<tr>
<td>0.2</td>
<td>178.05 (40.03)</td>
<td>184.01 (41.37)</td>
<td>179.29 (40.31)</td>
<td>2.15 (0.4851)</td>
<td>6</td>
</tr>
<tr>
<td>0.3</td>
<td>178.05 (40.03)</td>
<td>178.58 (40.15)</td>
<td>178.13 (40.05)</td>
<td>0.20 (0.0463)</td>
<td>8</td>
</tr>
<tr>
<td>0.4</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>0.00 (0.0000)</td>
<td>10</td>
</tr>
<tr>
<td>0.5</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>0.00 (0.0000)</td>
<td>10</td>
</tr>
<tr>
<td>0.6</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>0.00 (0.0000)</td>
<td>10</td>
</tr>
<tr>
<td>0.7</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>0.00 (0.0000)</td>
<td>10</td>
</tr>
<tr>
<td>0.8</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>0.00 (0.0000)</td>
<td>10</td>
</tr>
<tr>
<td>0.9</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>0.00 (0.0000)</td>
<td>10</td>
</tr>
<tr>
<td>1.0</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>178.05 (40.03)</td>
<td>0.00 (0.0000)</td>
<td>10</td>
</tr>
</tbody>
</table>

References


8. Gholizadeh S, and Poorhoseini H. Performance-based optimum seismic design of steel dual braced frames by bat


