Advances in Computational Design, *Vol. 9, No. 2 (2024) 115-136*
https://doi.org/10.12989/acd.2024.9.2.115

An enhanced simulated annealing algorithm for topology optimization of steel double-layer grid structures

Mostafa Mashayekhi* and Hamzeh Ghasemi[a]

Department of Civil Engineering, Vali-e-Asr University of Rafsanjan, Rafsanjan, Iran

(Received January 8, 2024, Revised March 24, 2024, Accepted May 16, 2024)

Abstract. Stochastic optimization methods have been extensively studied for structural optimization in recent decades. In this study, a novel algorithm named the CA-SA method, is proposed for topology optimization of steel double-layer grid structures. The CA-SA method is a hybridized algorithm combining the Simulated Annealing (SA) algorithm and the Cellular Automata (CA) method. In the CA-SA method, during the initial iterations of the SA algorithm, some of the preliminary designs obtained by SA are placed in the cells of the CA. In each successive iteration, a cell is randomly chosen from the CA. Then, the "local leader" (LL) is determined by selecting the best design from the chosen cell and its neighboring ones. This LL then serves as the leader for modifying the SA algorithm. To evaluate the performance of the proposed CA-SA algorithm, two square-on-square steel double-layer grid structures are considered, with discrete cross-sectional areas. These numerical examples demonstrate the superiority of the CA-SA method over SA, and other meta-heuristic algorithms reported in the literature in the topology optimization of large-scale skeletal structures.

Keywords: cellular automata method; simulated annealing algorithm; steel double-layer grids; stochastic optimization methods; topology optimization

1. Introduction

Currently, due to the competitive development of economies and technologies worldwide, optimization problems have become increasingly complex, with larger scales, more variables, and constraints (Cui *et al*. 2017). Recently, meta-heuristic optimization techniques have been developed to tackle complex optimization problems that other algorithms have failed to solve effectively or efficiently. Although some algorithms perform better than others on specific design problems, there is no single algorithm that performs optimally on all optimization problems. However, the practical advantage of meta-heuristics lies in their effectiveness and general applicability, as evidenced by previous studies (Cho and Kang 2021, Babaei *et al*. 2022, Ghasemi *et al*. 2022, Mohammadnejad and Kazemi 2022, Mashayekhi *et al*. 2023, Amiri *et al*. 2023).

The importance of structural optimization is growing due to several factors, such as limited material resources, environmental impact, and intense technological competition. Lightweight,

*Corresponding author, Assistant Professor, E-mail: m.mashayekhi@vru.ac.ir
[a] M.Sc., E-mail: ghasemi.hamze69@gmail.com

http://www.techno-press.org/?journal=acd&subpage=7 ISSN: 2383-8477 (Print), 2466-0523 (Online)

low-cost, and high-performance structures are in demand to address these issues (Faramarzi and Afshar 2012). In structural optimization, the structural weight is typically the objective function, while the constraints depend on the specific optimization problem and can include joint displacement, stress and slenderness ratio of elements, structural reliability, failure probability, and frequency constraints (Kaveh *et al*. 2023, Kaveh and Zaerreza 2022). The optimization of structures has long been a significant area of research due to its widespread application. The design of structures can be categorized into four optimization problems, which include material, sizing, topology (configuration), and geometry (shape) optimization. Material optimization involves the optimal design of elastic structures, where the design variables are material properties (Faramarzi and Afshar 2012). Sizing optimization involves finding the optimal cross-sectional area of members in skeletal structures or the thickness of continuous structures, assuming that the nodal coordinates of the structure are fixed. Topology optimization deals with the connectivity of elements between nodes, specifically the presence or absence of such elements. On the other hand, shape optimization is concerned with determining the optimal nodal coordinates, given that the topology is fixed (Faramarzi and Afshar 2012). Employing meta-heuristic search methods for optimal structural design is a relatively novel area that demands further research, as evidenced by the works of several recent studies (Hasançebi 2008, Sonmez 2011, Dogan and Saka 2012, Kaveh and Zaerreza 2022, Azad *et al*. 2013, Kaveh *et al*. 2023, Nabil *et al*. 2023).

The structural topology optimization technique aims to attain the optimal performance of a structure subject to various constraints. Compared to sizing and shape optimizations, topology optimization offers greater flexibility and empowers the designer to create innovative and highly efficient conceptual designs for structures (Mozafari *et al*. 2012). Nowadays, numerous hybridized meta-heuristic topology optimization techniques have been developed in recent years, particularly for optimizing the topology of large-scale skeletal structures with discrete cross-sectional areas (Mashayekhi *et al*. 2011, 2012, 2015, 2016, Dehghani *et al*. 2016).

The Simulated Annealing (SA) algorithm is a meta-heuristic approach that seeks to approximate the global optimum of a given function in a vast search space. SA uses iterative movement based on a variable temperature parameter that mimics the annealing process in metallurgy, as described in the works of Kirkpatrick *et al*. (1983) and Cerny (1985). Furthermore, several optimization techniques based on SA have been proposed to tackle structural design optimization problems with a discrete set of design variables (Liu *et al*. 2023, Vasile *et al*. 2022, Rettl *et al*. 2023, Hasançebi *et al*. 2009).

Credit for developing cellular automata is typically given to Ulam (1952) and Neumann (1966), who introduced the concept in the late 1940s as a practical model for simulating the behavior of complex systems. Since the promising beginnings of cellular automata, numerous theoretical and practical applications have been developed, and computational models based on classical cellular automata have been successful in defining and solving various models of natural phenomena. Some of these applications are discussed in (Hoekstra *et al*. 2010). In the area of structural optimization, several families of combined structural optimization techniques have been proposed that utilize cellular automata (CA) (Zhang *et al*. 2023, Duan *et al*. 2023, Bouzouiki *et al*. 2021, Canyurt and Hajela 2005, Mashayekhi and Yousefi 2021, Dehghani *et al*. 2021).

This study introduces a new and improved algorithm, called CA-SA, to optimize the topology of steel double-layer grid (SDLG) structures. The CA-SA enhances the efficiency of the simulated annealing (SA) method by incorporating a cellular automata (CA) algorithm. The CA algorithm is implemented by distributing cells in a square lattice structure that includes N_{CA} cells in each direction, resulting in a total number of N_{CA}^2 cells. Moore's neighborhood is also considered for

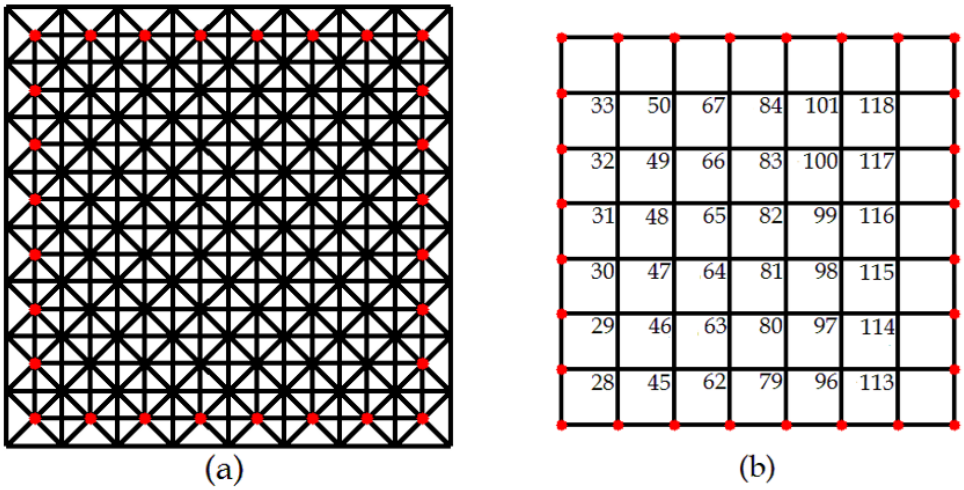

Fig. 1 Steel double layer grid ground structure (a), and joint numbers of bottom layer (b) (Mashayekhi *et al.* 2012)

these cells (Rajasekaran 2001). During the initial iterations of the CA-SA algorithm, new N_{CA}^2 designs are generated and randomly assigned to the CA cells. In subsequent iterations, a cell is selected at random from the CA square lattice structure, and the best design from the cell and its neighborhood is chosen as the "*local leader*" (LL). The LL design is employed to enhance the search for the optimal design in the SA algorithm. Updating the CA cell state involves replacing the worst design in the CA with the newly obtained design for each iteration, provided that the objective function has improved. Numerical examples demonstrate that the CA-SA algorithm produces superior results in the topology optimization of large-scale skeletal structures, compared to those obtained using the SA algorithm and several other meta-heuristic algorithms described in previous studies.

2. Steel double-layer grids topology optimization

In topology optimization of steel double-layer grids (SDLG), the geometry, support positions, and node coordinates remain fixed, while the design variables consist of the presence/absence of joints and cross-sectional areas. To reduce the design space, the structure's symmetry properties are utilized for joint tabulation, resulting in groups of 8, 4, or 1 joint (Mashayekhi *et al*. 2012). For example, in the SDLG shown in Fig. 1, number of joints with the same geometric positions is presented in Table 1.

Each joint group's presence or absence is determined by a binary topology variable, which can only be assigned the values 0 or 1. Hence, this structure requires six topology variables to represent the variability of the joint groups. A topology variable with a value of 0 signifies the removal of the corresponding joint group from the ground structure, leading to the elimination of all elements connected to that node.

In a SDLG topology optimization problem, the total number of design variables (NDV) is the sum of the number of topology variables (NTV) and the count of compressive and tensile element types. For instance, in the topology optimization process of the ground structure illustrated in Fig.

Table 1 Symmetrical joint group number (Mashayekhi *et al*. 2012)

Group No.	Joints in each group
1	28, 33, 118, 113
2	29, 32, 50, 101, 117, 114, 96, 45
3	30, 31, 67, 84, 116, 115, 79, 62
4	46, 49, 100, 97
5	47, 48, 66, 83, 99, 98, 80, 63
6	64, 65, 82, 81

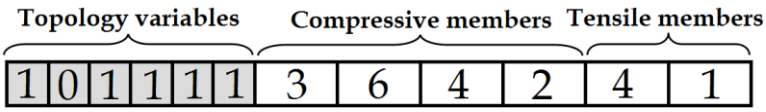

Fig. 2 The specified design variables (Mashayekhi *et al*. 2012)

1, the design variable vector is depicted in Fig. 2. Assuming there are 4 compressive and 2 tensile element types considered (Mashayekhi *et al*. 2012).

Given that the second topology variable is assigned a value of zero, all the joints belonging to the second group in Table 1 are eliminated from the ground structure. The resulting topology is depicted in Fig. 3 (Mashayekhi *et al*. 2012).

The 6 remaining design parameters are utilized to allocate the cross-sectional area to members within any specific group by consulting the profile table (Mashayekhi *et al*. 2012). Discrete variables specify the appropriate cross-sectional area of structural members, chosen from pipe sections with specified thickness, outer diameter, and number of profiles. The existence of nodes in the top grid is not considered a variable to achieve a practical structure, resulting in unchanged load-bearing areas of top layer joints (Mashayekhi *et al*. 2012).

In the SDLG topology optimization problem, the design variables are determined from a discrete set of values to minimize the structural weight (W), while constraints on stress (g_σ), slenderness ratio ($g_{\sigma\lambda}$), and displacement (g_δ) are satisfied:

$$\begin{aligned} &Minimize: W(X) = \rho \sum_{k=1}^{NMG} a_k \sum_{i=1}^{N_k} l_i \\ &Subject to:\ g_\sigma, g_\lambda, g_\delta \le 0 \\ &\qquad X = [J_1, J_2, ..., J_{NTV}, a_1, a_2, ..., a_{NMG}]^T \\ &\qquad J_i \in (0,1),\ \ i = 1,2,..., NTV \\ &\qquad a_k \in \tilde{\mathbf{A}},\ \ k = 1,2,..., NMG \end{aligned} \tag{1}$$

This formula involves various variables, including NMG, which represents the number of member groups. Additionally, N_k indicates the number of members in the kth member group, while a_k is the discrete cross-sectional area of the kth member group, chosen from a list of steel pipe sections with NP profiles denoted by $\tilde{\mathbf{A}}$. Other variables include, ρ the material density, li the length of the ith element, as well as stress, slenderness ratio, and displacement constraints, which must be satisfied.

The topology optimization problem is solved using both SA and CA-SA methods. To solve

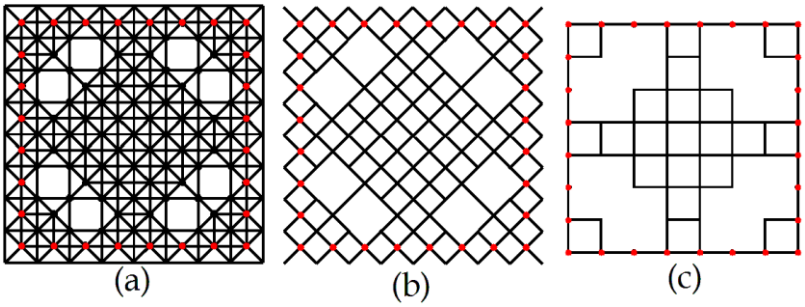

Fig. 3 The outcome structure illustrated in Fig. 2: (a) steel double layer grid, (b) diagonal layer, and (c) bottom layer (Mashayekhi *et al*. 2012)

a constrained optimization problem, the objective function (W) is modified to convert the constrained optimization problem into an unconstrained one. This modification results in a modified objective function (Ψ), defined as follows (Mashayekhi *et al*. 2012):

$$\Psi = W(1+C)^2 \tag{2}$$

in which

$$C = \sum_{i=1}^{ne}(g_{\sigma,i} + g_{\lambda,i}) + \sum_{j=1}^{nj} g_{\delta,j} \tag{3}$$

where C is the penalty function, which depends on the number of elements (ne) and joints (nj) in the structure.

The stress and the slenderness ratio constraints are defined as follows (Mashayekhi *et al*. 2012):

$$g_\sigma = \sum_k \max(|\sigma_k|/\bar{\sigma}_k - 1, 0) \tag{4}$$

$$g_\lambda = \sum_k \max(\lambda_k/\bar{\lambda}_k - 1, 0) \tag{5}$$

where σ_k is, $\bar{\sigma}_k$, λ_k and $\bar{\lambda}_k$ are the member stress, allowable stress, member slenderness ratio and its upper limit for the kth member of the steel double layer grids, respectively. In this study, the AISC code provisions are applied for the stress limits and local buckling criteria (AISC Manual 2005), as follows:

$$\begin{aligned} \bar{\sigma}_k &= 0.6F_y \quad (\textit{for tensile members}) \\ \bar{\sigma}_k &= \bar{\sigma}_c \quad (\textit{for compression members}) \end{aligned} \tag{6}$$

in which

$$\bar{\sigma}_c = \begin{cases} \left[\left(1 - \dfrac{\lambda_k^2}{2C_c^2}\right)F_y\right] \Big/ \left(\dfrac{5}{3} + \dfrac{3\lambda_k}{8C_c} - \dfrac{\lambda_k^3}{8C_c^3}\right) & \text{for } \lambda_k < C_c \\ \dfrac{12\pi^2 E}{23\lambda_k^2} & \text{for } \lambda_k \geq C_c \end{cases} \tag{7}$$

where the yield stress, modulus of elasticity, and C_c, denoted as F_y, E, and $\sqrt{2\pi^2 E/F_y}$, respectively.

The maximum slenderness ratio is restricted to 300 for tension members and 240 for compression members. Consequently, the design constraints related to slenderness ratio can be expressed as follows (Mashayekhi *et al*. 2012):

$$\begin{aligned} \lambda_k &= K_k l_k / r_k \le 300 \qquad (\textit{for tensile members}) \\ \lambda_k &= K_k l_k / r_k \le 240 \qquad (\textit{for compression members}) \end{aligned} \tag{9}$$

where δ_j is the displacement of the jth node.

3. Simulated annealing (SA) algorithm

Simulated annealing is named after the physical process of annealing with solids, in which a crystalline solid is heated and allowed to cool slowly until it reaches its minimum energy state. This results in a crystal lattice configuration that is free of defects and has superior structural integrity. By simulating this thermodynamic behavior, simulated annealing provides an algorithmic approach for searching for global minima in discrete optimization problems. This approach establishes a connection between the thermodynamic behavior and optimization problems, and provides a means for leveraging this connection. The concept of simulating the annealing process to solve optimization problems was first independently introduced by Kirkpatrick *et al*. (1983) and Cerny (1985). They established an analogy between minimizing the energy level of a physical system and reducing the cost of an objective function. This analogy forms the basis for applying simulated annealing algorithms to a wide range of optimization problems. The SA algorithm used in this study is based on the work of Hasançebi *et al*. (2009). Their algorithm was designed for sizing optimization of truss structures, and the basic steps involved in their approach are outlined below (Canyurt and Hajela 2005). However, some modifications are made to adapt the algorithm for topology optimization of steel double-layer grid structures.

Step 1: Setting up an appropriate cooling schedule

The cooling process is defined by three key parameters: T_S, T_f, and η. T_S denotes the starting temperature, T_f represents the temperature at the end of the cooling cycle, while η indicates the temperature decrement factor during the cooling process. Determining these parameters requires selecting suitable values for the initial acceptance probability (P_S), final acceptance probability (P_f), and the number of cooling cycles (N_C), as follows:

$$T_S = -\frac{1}{\ln(P_S)},\ T_f = -\frac{1}{\ln(P_f)},\ \eta = \left[\frac{\ln(P_S)}{\ln(P_f)}\right]^{1/N_C - 1} \tag{10}$$

The optimization procedure commences with T being set equal to T_S, and subsequently decreasing during the operation. The initial and final temperatures of the system, as well as the rate at which T decreases, are critical factors influencing the efficacy and efficiency of the metal annealing method.

Step 2: Creating an initial design

The initial design is randomly selected from a list of available values for the design variables,

and considered current design (X_{Cu}). This design is assigned as the current design of the optimization problem, and its objective function is calculated.

Step 3: Creating the candidate design

To create the candidate design (X_{Ca}), first, the current design of the optimization problem is considered. Then, a small change is made to one of its variables (the *i*th variable), which is randomly selected between 1 and ($NTV+NMG$), as follows:

$$\begin{aligned} &X_{Ca} = X_{Cu} \\ &if\ (i \le NTV)\ then \\ &\quad xi = 1 - X_{Cu}(i) \\ &elseif\ (NTV < i \le NTV + NMG)\ then \\ &\quad xi = X_{Cu}(i) \pm dz, dz \in [1, 2, ..., dz_{\max}] \\ &end\,if \\ &X_{Ca}(i) = xi \end{aligned} \tag{11}$$

In the above equation, dz_{max} represents the maximum allowable change in size variables, and dz is the change applied to the *i*th variable, which is randomly selected from natural numbers between 1 and dz_{max}. It should be noted that this change should not violate the upper or lower limit of the selected variable. In this study, since the value of each size variable corresponds to the profile number used, the upper limit of the size variables is equal to the number of steel sections, and their lower limit is equal to one. Therefore, the candidate design of the problem is obtained by considering the change made to the selected design variable. Therefore, it can be said that the candidate design differs from the current design only in terms of one design variable. It should be noted that each design variable is selected only once during a single cooling cycle to create a candidate design. In other words, the number of candidate designs created during a single cooling cycle is equals the number of design variables.

Step 4: Evaluating the candidate design

Each time a candidate design is created, its objective function is also calculated. If the objective function of the candidate design (Ψ_{Ca}) is less than the objective function of the current design (Ψ_{Cu}), the candidate design is accepted without any conditions and replaces the current design of the problem:

$$\begin{aligned} &\Delta\Psi = \Psi_{Ca} - \Psi_{Cu} \\ &if\ \Delta\Psi \le 0 \rightarrow X_{Cu} = X_{Ca},\ \Psi_{Cu} = \Psi_{Ca} \end{aligned} \tag{12}$$

Otherwise, if the objective function of the candidate design is greater than the objective function of the current design ($\Delta\Psi > 0$), the acceptance of the candidate design and its placement as the current design of the problem will be conditional. Conditional acceptance of this weak candidate design is used to escape local minima. The probability of accepting this weak candidate design is high at high temperatures, but decreases as the temperature decreases.

To determine the acceptance or rejection of the weak candidate design, first, the Boltzmann parameter is obtained using the following equation:

$$K = \Delta\Psi_{ave} \tag{13}$$

where $\Delta\Psi_{ave}$ is the average value of $\Delta\Psi$ obtained from the beginning of the corresponding cooling cycle until the current moment in the above equation.

In general, a design can be infinitely weak to the extent that sampling from such designs can drive the Boltzmann parameter towards worthless values and hinder the appropriate search of the algorithm in the design space. To address this issue, first, the value of $\Delta\Psi$ is obtained within a range of zero to one using Eq. (14), despite having relatively large values, and then the Boltzmann parameter is determined using Eq. (15):

$$\Delta\Psi_{tra} = \tanh\left(0.35 * \frac{\Delta\Psi}{K}\right) \tag{14}$$

$$K = (\Delta\Psi_{tra})_{ave} \tag{15}$$

where $(\Delta\Psi_{tra})_{ave}$ represents the average value of $\Delta\Psi_{tra}$ obtained from the beginning of the relevant cooling cycle until the current moment. Finally, the probability of accepting a weak design (P) is calculated as follows:

$$\begin{aligned} P_p &= \exp\left(-\frac{\Delta\Psi}{K * T}\right) \\ P_t &= \exp\left(-\frac{1}{T}\right) \\ \varphi &= \left(\sqrt[3]{\overline{P}_t / \overline{P}_p}\right)^{(k-1)}, 0.9 \le \varphi \le 1.1 \\ P &= \varphi * \exp\left(-\frac{\Delta\Psi}{K * T}\right) \end{aligned} \tag{16}$$

In Eq. (16), $(\overline{P}_t)^{(k-1)}$ and $(\overline{P}_p)^{(k-1)}$ represent the theoretical and practical average acceptance probabilities in the (k-1)th cooling cycle, respectively.

Step 5: Repeating a single cooling cycle

A single cooling cycle performs Steps 3 and 4 for all problem variables. Generally, a single cooling cycle should be repeated several times to ensure that the objective function has decreased to a logical value proportional to the cooling cycle temperature. For this purpose, the number of repetitions of a single cooling cycle should be determined at the beginning (i_s) and end (i_f) of the cooling process. Finally, the number of repetitions of a single cooling cycle at a specific temperature (i_c) is calculated using the following equation:

$$i_c = i_f + (i_f - i_s)\left(\frac{T - T_f}{T_f - T_s}\right) \tag{17}$$

Step 6: Decreasing the temperature

After completing k cooling cycle repetitions, the system temperature must be reduced. The temperature of the system in the (k+1)th cooling cycle is reduced using the ratio of the cooling factor η as follows:

$$T^{k+1} = T^k * \eta \tag{18}$$

Step 7: Optimization termination criterion

Steps 3 to 6 must be repeated until all cooling cycles have been performed.

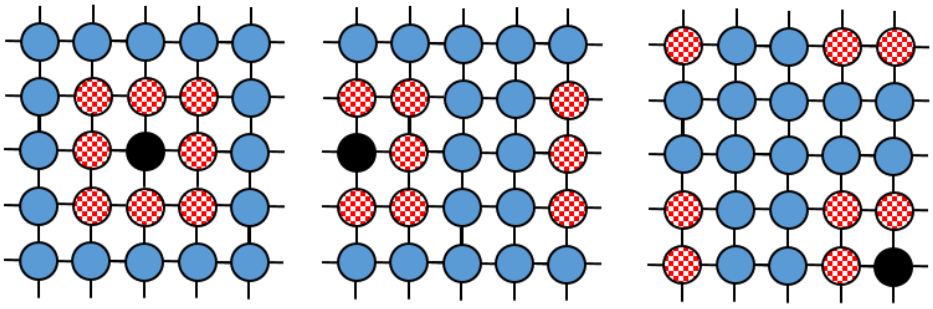

Fig. 4 A visual depiction of Moore's neighbourhood structure for a middle, edge, and corner black cells in a 5×5 grid (Mashayekhi and Yousefi 2021)

4. Cellular automata (CA) method

Cellular automata were first introduced by Ulam (1952) and Neumann (1966) as a pragmatic model for complex systems. Since their inception, cellular automata have found numerous theoretical and practical applications, including the modeling of natural phenomena. Various computational models have been developed using classical cellular automata, and some examples are presented in (Hoekstra *et al*. 2010).

Cellular automata divide the physical domain of the problem into a regular lattice of cells. Each cell is assigned a finite set of values known as the cell state, updated over time using an updating rule. The updating rule modifies the state of each cell by using only information from the cell and its neighboring cells. For this study, we employed Moore's neighborhood structure of cellular automata, where every cell has the same neighborhood structure (Rajasekaran 2001). We considered a square grid of $N_{CA}\times N_{CA}$ cells, where N_{CA} is the number of cells in each direction. The cells at the boundary of the physical domain have neighboring cells outside the border, and are assumed to be connected to the cells on the opposite boundary. Specifically, Moore's neighborhood structure of a middle, edge, and corner black cells in a 5×5 grid, with N_{CA} =5, is shown by the hatched red cells in Fig. 4 (Mashayekhi and Yousefi 2021).

5. An improved simulated annealing algorithm (CA-SA)

The SA algorithm's approach to escaping local minima involves conditionally accepting weak designs. However, repeated acceptance of weak designs may lead to deviation of the algorithm from the optimal solution space. Additionally, admitting mediocre designs in the initial iterations of the optimization process may not be beneficial, especially when the quality of all the solutions is poor. Furthermore, as the number and diversity of variables in the optimization problem increase, the convergence rate of the SA method may decrease, particularly when a candidate design differs from the current one in terms of a single design variable. To address this issue, the present study modifies the SA method using the CA approach to enhance the quality of the new generated points. Specifically, the role of inadequate designs is decreased, and the use of better ones in generating new points is increased. To achieve this, a new enhanced algorithm called CA-SA is introduced for the topology optimization of steel double-layer grid structures. In CA-

SA, the efficiency of the SA method is improved using the CA algorithm. To achieve this aim, a set of cells are distributed in a square lattice structure, including N_{CA} cells in any direction. Therefore, the total number of cells will be N_{CA}^2. Also, Moore's neighborhood is considered for its cells (Rajasekaran 2001). In the initial iterations of the CA-SA algorithm, new N_{CA}^2 designs are produced and randomly located in the CA cells. After the dedication of N_{CA}^2 different designs to all CA cells, in the following iterations, a cell is first selected randomly from the CA square lattice structure. The best design is chosen from the selected cell and its neighborhood and considered as a "local leader" (LL). The LL has been used to improve the search for the optimal design of Eq. (1). In the topology optimization problem of SDLG structures, two critical questions must be answered: 1. Which members of the structure should be removed, and which ones should remain? 2. What should be the cross-sectional area of the remaining members? The first question is more important than the second, so even for an optimal topology, assigning inappropriate values to the cross-sectional area can make its objective function unsuitable. Therefore, in this article, to allocate a more appropriate cross-sectional area to the members of each new topology (X_{Ca}), a new value for some cross-sectional variables is randomly selected from LL, using the following equation:

$$\begin{aligned} &if\ (r1 < P_{LL})\,then \\ &\quad for\,(i = NTV+1 : NTV + NMG): \\ &\quad\quad if\,(r2 < P_{UseLLi})\,then \\ &\quad\quad\quad X_{Ca}(i) = LL(i) \\ &\quad\quad endif \\ &\quad end\ for \\ &endif \end{aligned} \tag{19}$$

where *r1* and *r2* are random numbers that are generated in the intervals 0 and 1 with the normal distribution, P_{UseLLi} is the probability of using the *i*th variable of LL, and P_{LL} is the probability of using LL to produce X_{Ca}, which is calculated as follows:

$$P_{LL} = P_{(LL)S} + \left(\left(P_{(LL)F} - P_{(LL)S}\right) * \frac{k}{N_C} \right) \tag{20}$$

where $P_{(LL)S}$ and $P_{(LL)F}$ are the initial and final probability of using LL, respectively. In fact, in the early stages of optimization, where the quality of solutions and LL are not very suitable, the likelihood of using LL is low, but gradually, with the improvement of solution quality and, consequently, the progress of LL quality, the likelihood of using LL increases.

In this article, to update the CA cell state, if the objective function of new obtained design in each iteration is improved, the worst design of the CA is replaced with it. In addition, the step-by-step process of topology optimization of SDLG structures using the CA-SA optimization method will be presented.

Step 1: In addition to the optimization problem specification, the parameters required for the CA-SA algorithm ($P_{(LL)S}$, $P_{(LL)F}$, P_s, P_f, N_C, dz_{max}, N_{CA}, i_s, i_f, and η) are set, and a square lattice including N_{CA}^2 cells is produced.

Step 2: An initial current design (X_{Cu}) is randomly generated, and its modified objective function (Ψ_{Cu}) is calculated using Eq. (2).

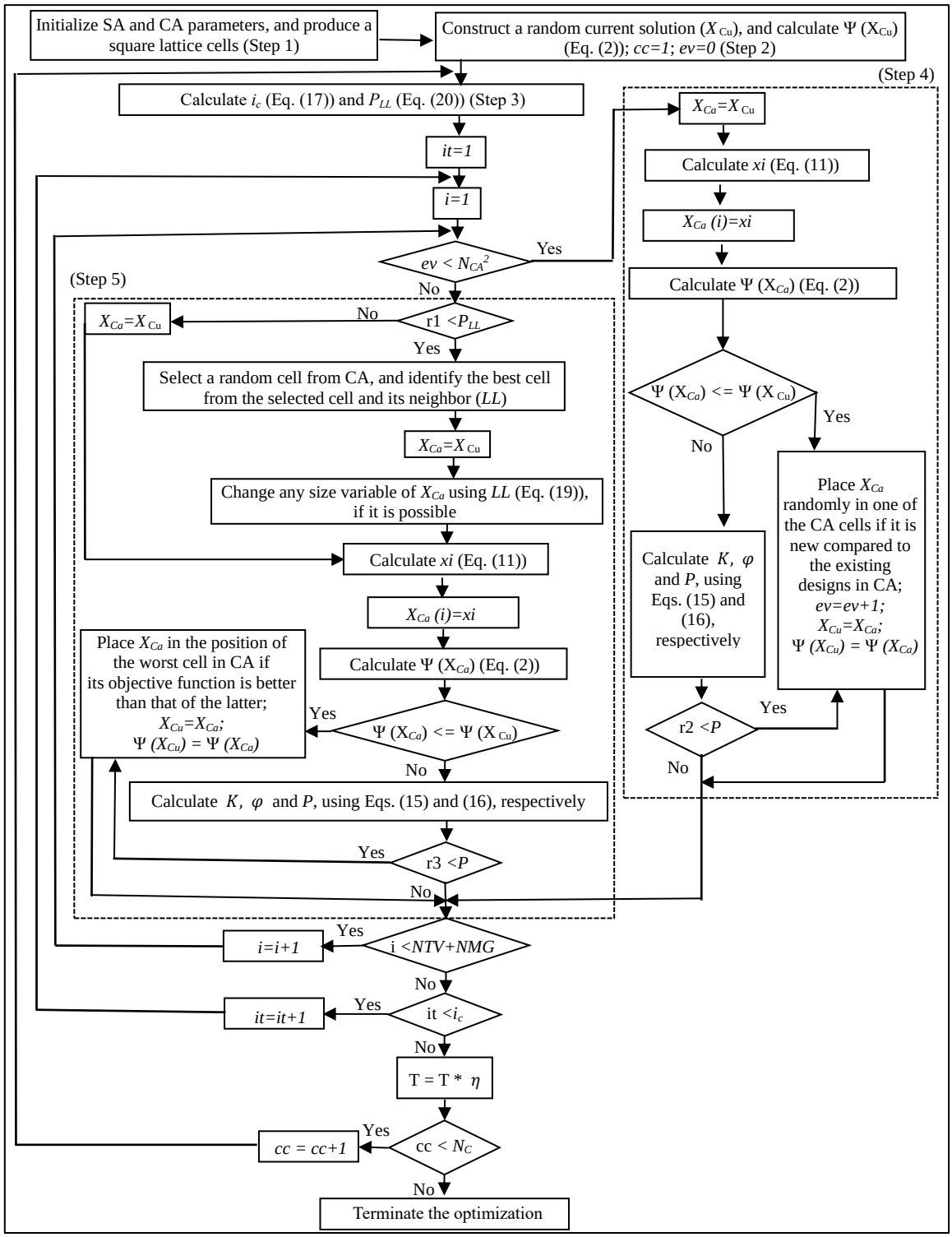

Fig. 5 The topology optimization process employing the CA-SA method

Table 2 Available pipe profiles (Mashayekhi *et al*. 2012)

No.	*OD* (cm)	*TH* (cm)	No.	*OD* (cm)	*TH* (cm)	No.	*OD* (cm)	*TH* (cm)	No.	*OD* (cm)	*TH* (cm)
1	4.83	0.26	6	10.80	0.36	11	16.86	0.45	16	32.39	0.71
2	6.03	0.29	7	11.43	0.36	12	19.37	0.45	17	35.56	0.80
3	7.61	0.29	8	13.30	0.40	13	21.91	0.45	18	40.64	0.88
4	8.89	0.32	9	13.97	0.40	14	24.45	0.63	19	45.72	1.00
5	10.16	0.36	10	15.90	0.45	15	27.30	0.63			

Table 3 Constant count and probability parameters of the SA and CA algorithms

	Constant count parameters					Constant probability parameters				
Parameter	i_s	i_f	dz_{max}	N_C	N_{CA}	P_S	P_f	$P_{(LL)F}$	$P_{(LL)S}$	P_{UseLLi}
Value	1	3	5	200	4	0.5	0.001	0.9	0.1	1/*NMG*

Step 3: The number of repetitions of a single cooling cycle at a specific temperature (i_c) and the probability of using LL (P_{LL}) are determined using Eqs. (17) and (20), respectively.

Step 4 (There is still an empty cell in the CA): The candidate design (X_{Ca}) is created using Eq. (11), and its objective function (Ψ_{Ca}) is determined using Eq. (2). If the objective function of the candidate design is better than the objective function of the current design, the candidate design is randomly placed in one of the empty CA cells and replaces the current design. If it is not better, K, φ, and P (Eq. (16)) are calculated, and the candidate design is placed randomly in one of the empty CA cells with probability P and replaces the current design.

Step 5 (All the CA cells are complete): In this case, either with probability P_{LL}, LL is used to determine X_{Ca}. To do this, a cell from CA is randomly selected, and LL is determined. Then, using LL, X_{Ca} is created using Eq. (19). Alternatively, with probability (1- P_{LL}), X_{Ca} is created without determining and using LL using Eq. (11). After determining X_{Ca} using either of the two previous methods, its objective function (Ψ_{Ca}) is calculated. If the objective function of the candidate design is better than the objective function of the current design, the candidate design replaces the current design and the worst CA cell. But if it is not better, K, φ, and P are calculated (Eq. (16)), and the candidate design is placed with probability P in addition to replacing the current design. It also replaces the worst CA cell. It should be noted that replacing the candidate design with the worst CA cell is conditional on the candidate design being better than it.

Step 6: Steps 4 and 5 are repeated for all design variables.

Step 7: Steps 4 to 6 are repeated i_c times.

Step 8 (Termination): The temperature is gradually reduced, and Steps 3 to 7 are iterated until a predefined maximum number of cycles, denoted by N_C, is reached.

The topology optimization process with CA-SA is illustrated in Fig. 5, which depicts the algorithm's flowchart.

6. Numerical examples

This study evaluates the performance of the proposed CA-SA algorithm by optimizing two square-on-square SDLG structures. The structures are constructed from structural steel, with Young's modulus of E = 206 (GPa) and a material density of 7850 (kg/m^3). The structures are

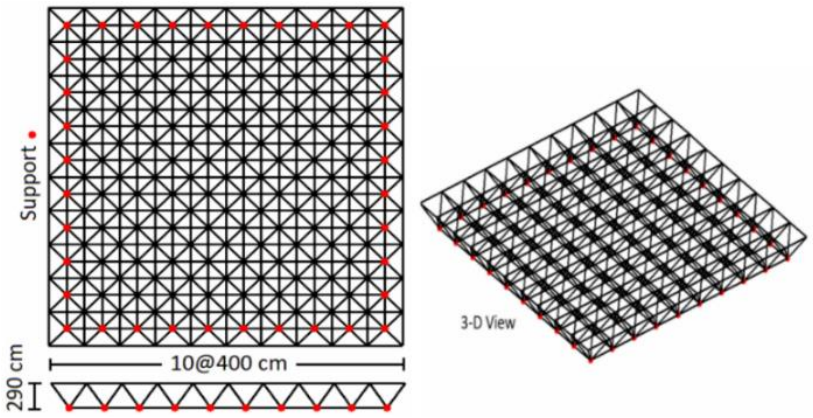

Fig. 6 The 10x10 steel double-layer grid layout (Dehghani *et al*. 2021)

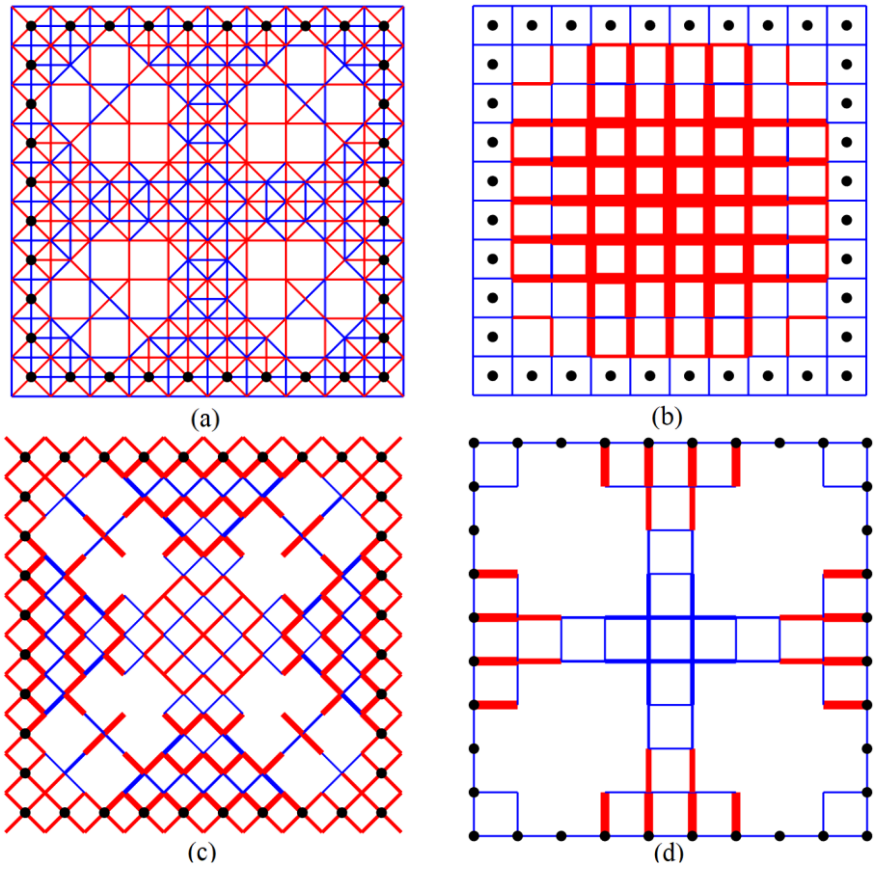

Fig. 7 Optimal topology achieved through SA in the topology optimization of the 10x10 SDLG structure

subjected to a distributed load of 180 (kg/m^2), applied to the nodes of the top grid in proportion to their load-bearing area, as described in (Mashayekhi *et al*. 2011, 2012). The ground structures are assumed to be supported by perimeter nodes of the bottom grids. The cross-sectional areas of the

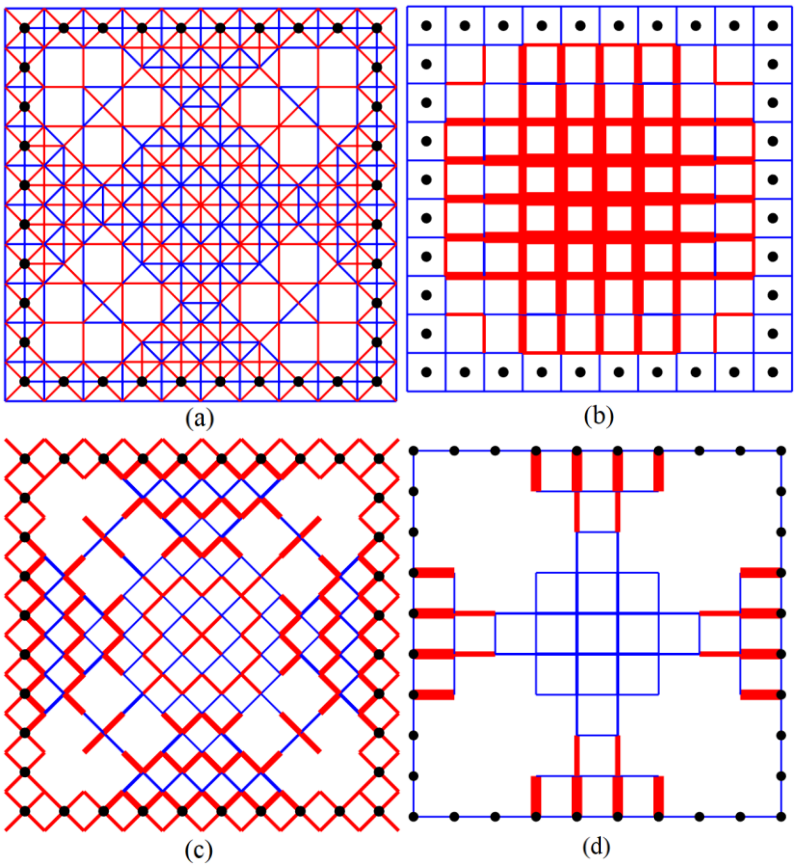

Fig. 8 Optimal topology achieved through CA-SA in the topology optimization of the 10x10 SDLG structure

elements can be chosen from a list of available pipe profiles, as presented in Table 2, where *OD* and *TH* denote the outer diameter and thickness (in cm) of the cross-sectional profile, respectively.

Two cases are examined in this study: (i) topology optimization with the SA, and (ii) topology optimization with the CA-SA. To account for the stochastic nature of SA and CA-SA, five separate optimization runs are conducted for each test problem, with different initial points for each run. It is worth noting that no clustering method is utilized to generate these initial populations. The member grouping approach developed by Mashayekhi *et al*. (2012) is consistently employed to optimize the structures. In all figures depicting the optimal designs, we use (a) to (d) to indicate the steel double-layer grid structure, top layer, diagonal layer, and bottom layer, respectively. Furthermore, the thickness of each element is directly proportional to its cross-sectional area, and the same scale is employed to represent element thicknesses for comparison. It is worth mentioning that, to display all of the elements in the optimal designs, Part (a) is presented using the same thickness for all members. Also, the members with internal compressive force are indicated in red color, and the members with internal zero or tensile force are shown in blue.

According to the computational outcomes, the constant count and probability parameters of the SA and the CA, as presented in Table 3, are considered suitable choices.

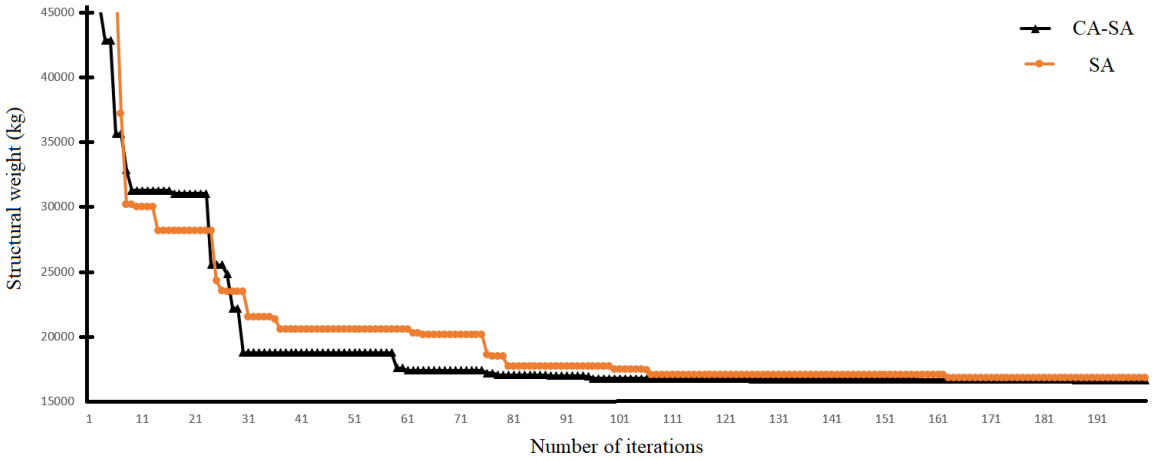

Fig. 9 Evaluation of convergence curves obtained from the top optimization runs of SA and CA-SA in the 10x10 SDLG topology optimization

Table 4 Comparison of statistical outcomes attained from SA, CA-SA, and other metaheuristic algorithms in the 10x10 SDLG topology optimization problem

Case No.	*Optimum Weight (kg)*						
	Sample 1	*Sample 2*	*Sample 3*	*Sample 4*	*Sample 5*	$\overline{W}\,(kg)$	*SD(kg)*
CA-SA	16688	16818	16781	16756	17939	16996	529
SA	17826	19146	17501	16853	19146	18094	1022
CA-ICEA (Dehghani *et al*. 2021)	16818	16688	17473	17311	18287	17315	634
ICA (Dehghani *et al*. 2021)	18632	19671	17906	19790	18985	18997	776
DE (Dehghani *et al*. 2021)	18736	18946	20690	18134	20714	19444	1187
ACO (Mashayekhi *et al*. 2012)	17913	18526	17583	17340	17256	17724	516
GPS (Dehghani *et al*. 2016)	17759	17505	16818	19277	17400	17752	920
ICA-GSA (Mashayekhi *et al*. 2015)	17127	17459	17402	18601	18973	17912	819
MGHSA (Mashayekhi *et al*. 2016)	16852	19943	18989	18683	18502	18594	1121

6.1 A 10x10 steel double-layer grid

To validate the proposed optimization method, a 10x10 square-on-square steel double-layer grid consisting of 221 nodes (joints) and 800 members is utilized, and the bottom joints are classified into 15 groups (NTV = 15) (Dehghani *et al*. 2021). The steel double-layer grid has a depth of 290 (cm) and a node spacing of 400 (cm) in both the top and bottom chords, as shown in Fig. 6.

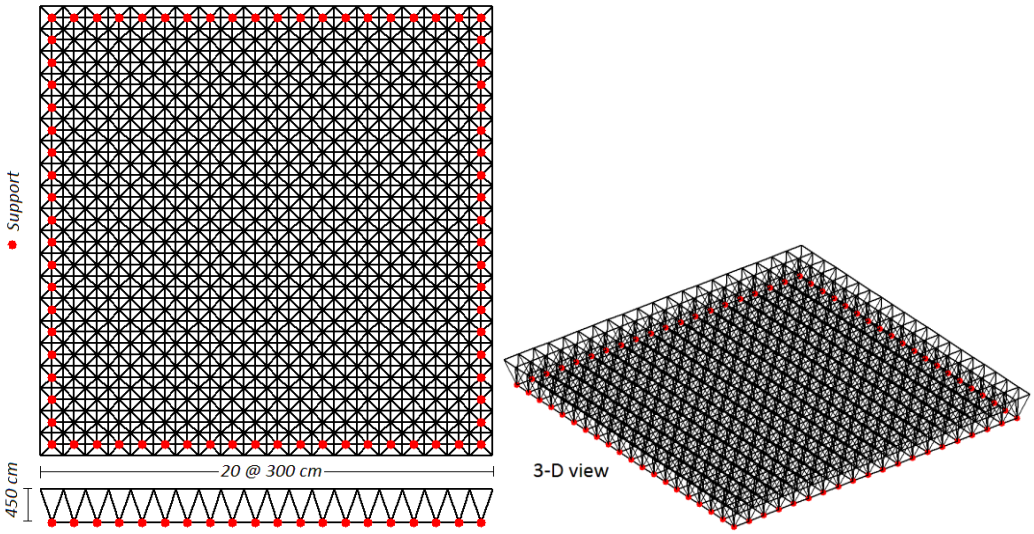

Fig. 10 The 20x20 steel double-layer grid layout (Mashayekhi *et al*. 2012)

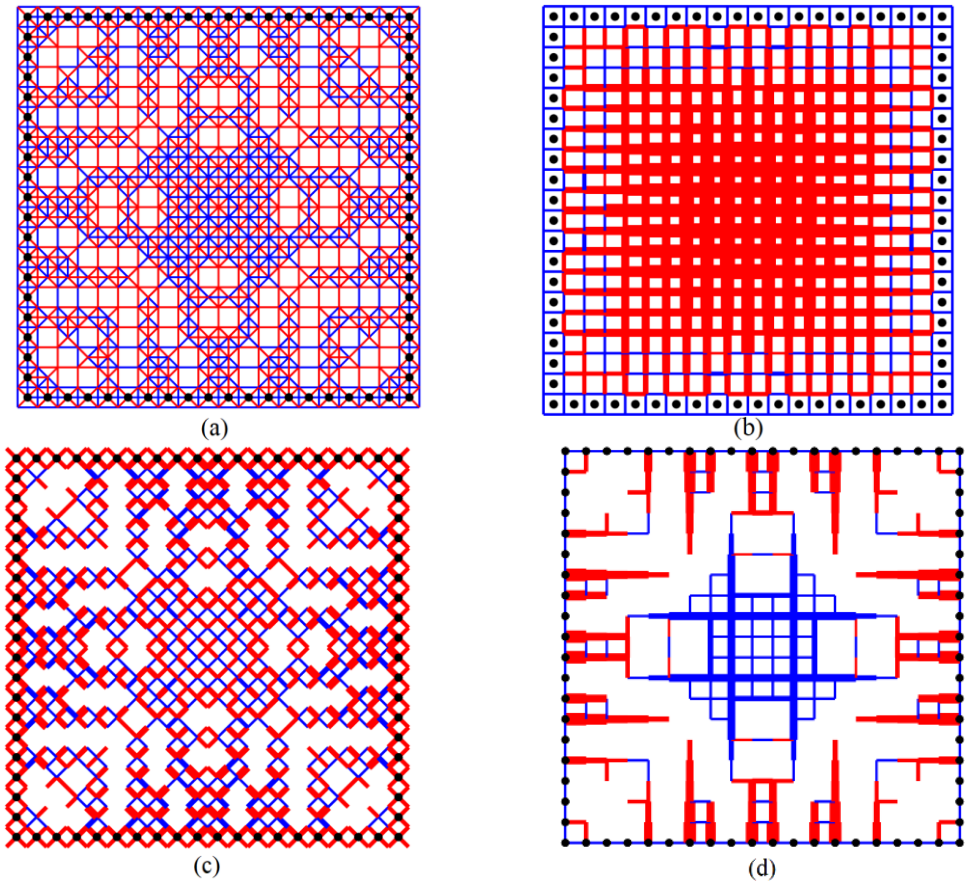

Fig. 11 Optimal topology achieved through SA in the topology optimization of the 20x20 SDLG structure

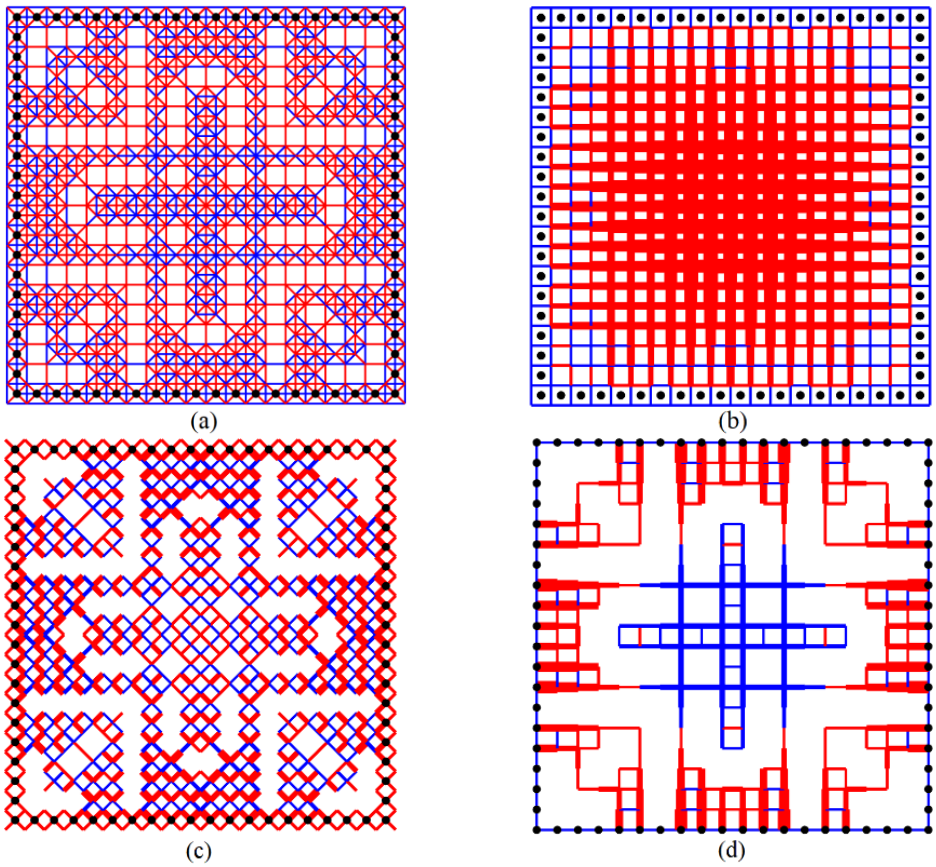

Fig. 12 Optimal topology achieved through CA-SA in the topology optimization of the 20x20 SDLG structure

For Cases 1 and 2, the optimal weights of the resulting structures are determined to be 16853 (kg) and 16688 (kg), respectively. The optimal structures for Cases 1 and 2 are presented in Figs. 7 and 8, respectively.

The convergence rate of the best optimal solution found in Cases 1 and 2 is depicted in Fig. 9. The results demonstrate that the CA-SA algorithm has a higher convergence rate than the SA algorithm.

Table 4 provides a comparison of the structural weights and the associated statistical information (i.e., average weight ($\overline{W}$) and standard deviation (SD) of optimized weight) obtained from five independent optimization runs using SA and CA-SA in the 10x10 SDLG topology optimization problem. The table also compares with other meta-heuristic algorithms reported in the literature. The best designs are identified in the table.

Comparing the performance of CA-ICEA (Dehghani *et al*. 2021) and CA-SA algorithms based on statistical data, it is evident that although both achieved the same best weight of 16688 kg, CA-SA demonstrates superior performance. Specifically, the mean weight, worst weight, and standard deviation for CA-SA are 16996 kg, 17939 kg, and 529 kg, respectively. In contrast, these values for CA-ICEA (Dehghani *et al*. 2021) are 17315 kg, 18287 kg, and 634 kg, respectively, indicating

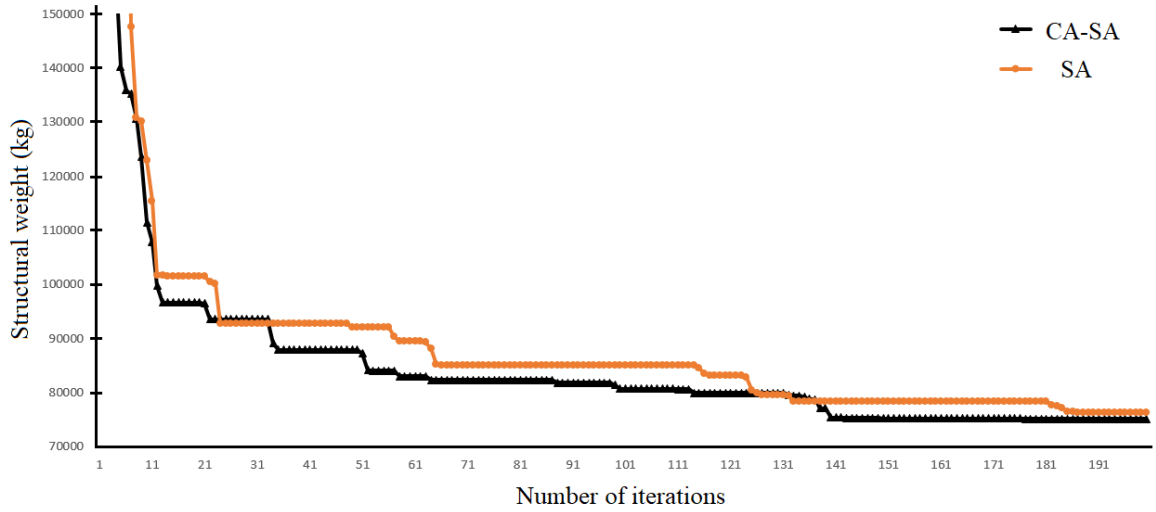

Fig. 13 Evaluation of convergence curves obtained from the top optimization runs of SA and CA-SA in the 20x20 SDLG topology optimization

the better performance of CA-SA. Additionally, the standard deviation of results from ACO (Mashayekhi *et al*. 2012) (516 kg) outperforms that of CA-SA (529 kg), yet CA-SA excels over ACO (Mashayekhi *et al*. 2012) in other statistical measures such as mean, worst, and best weight. The other statistical data presented in Table 4 indicate that the CA-SA method for the topology optimization of 10x10 SDLG outperforms both SA and other meta-heuristic algorithms reported in the literature. Hence, during the optimization process, CA-SA offers an improved balance between exploration and exploitation.

6.2 A 20x20 steel double-layer grid

As the second numerical example, a 20x20 square-on-square steel double-layer grid comprising 841 nodes (joints) and 3,200 members is presented. The bottom joints are classified into 55 groups ($NTV = 55$). The steel double-layer grid has a depth of 450 (cm) and a node spacing of 300 (cm) in both the top and bottom chord, as shown in Fig. 10 (Mashayekhi *et al*. 2012).

The optimal weights of the resulting structures are 76322 (kg) and 75103 (kg) for Cases 1 and 2, respectively. The optimal structures obtained for Cases 1 and 2 are depicted in Figs. 11 and 12, respectively.

The convergence rate of the best optimal solution achieved in Cases 1 and 2 is illustrated in Fig. 13. The results indicate that the CA-SA algorithm has a higher convergence rate than the SA algorithm.

Table 5 presents a comparison of the structural weights and their corresponding statistical information (i.e., average weight ($\overline{W}$) and standard deviation (SD) of optimized weight) obtained from five independent optimization runs using SA and CA-SA in the 20x20 SDLG topology optimization problem. The table also compares with other meta-heuristic algorithms reported in the literature, with the best designs highlighted.

Again, more statistical data presented in Table 5 indicate that the CA-SA method for

Table 5 Comparison of statistical outcomes attained from SA, CA-SA, and other metaheuristic algorithms in the 20x20 SDLG topology optimization problem

Case No.	*Optimum Weight (kg)*						
	Sample 1	*Sample 2*	*Sample 3*	*Sample 4*	*Sample 5*	$\overline{W}(kg)$	*SD(kg)*
CA-SA	78706	76325	81120	75103	82301	78711	3058
SA	77559	84545	80600	78763	76322	79558	3204
CA-ICEA (Dehghani *et al*. 2021)	86453	82557	76902	83874	77426	81442	3715
DE (Dehghani *et al*. 2021)	89054	81817	86455	92254	88471	87610	3850
ICA (Dehghani *et al*. 2021)	88504	90694	89877	89054	92254	90077	1473
MMA-ACO (Mashayekhi *et al*. 2012)	86152	85607	81927	87648	83160	84899	2317
ACO (Mashayekhi *et al*. 2012)	89652	89947	89910	85036	92542	89417	2717
ESO-GPS (Dehghani *et al*. 2016)	85895	82636	86323	86155	81312	84464	2325
GPS (Dehghani *et al*. 2016)	88571	87648	91013	87641	84396	87854	2374
ICA-GSA (Mashayekhi *et al*. 2015)	80601	81854	82589	83652	81973	82134	1114
MGHSA (Mashayekhi *et al*. 2016)	86679	76979	84079	88039	88304	84816	4690
GSA (Mashayekhi *et al*. 2016)	170503	162439	173214	165540	143362	163010	11758

the topology optimization of 20x20 SDLG outperforms both SA and all other meta-heuristic algorithms reported in the literature, in terms of the best, the worst, and the mean weight. Although the standard deviations of results obtained by ICA (1473 kg) (Dehghani *et al*. 2021), MMA-ACO, and ACO (2317 kg and 2717 kg, respectively) (Mashayekhi *et al*. 2012), ESO-GPS and GPS (2325 kg and 2374 kg, respectively) (Dehghani *et al*. 2016), and ICA-GSA (1114 kg) (Mashayekhi *et al*. 2015) are superior to that of CA-SA (3058 kg), these optimizers tend to get stuck in local optima and exhibit limited exploration capabilities. In contrast, CA-SA outperforms in terms of best, worst, and mean weight results in the topology optimization of SDLG. This is achieved by striking a balance between exploration and exploitation through leveraging past information within the CA cell to guide the search for new points in the design space using LL.

7. Conclusions

This paper proposes a hybridized stochastic optimization method, CA-SA, for topology optimization of steel double-layer grid structures. The CA-SA incorporates CA as an additional tool to enhance the SA search in the design space by implementing a local leader design (LL). The local leader design (LL) has been implemented in the current design to determine new values for multiple cross-sectional area variables simultaneously, instead of a single variable. To evaluate the

effectiveness of the proposed CA-SA method, two steel double-layer grid structures were examined. The efficiency of the proposed algorithm was demonstrated by comparing the numerical results of these 3D trusses obtained using the CA-SA method with those obtained using SA and other meta-heuristic optimization approaches. The two solved numerical examples show that using LL for simultaneous change of more than one cross-sectional area variable in the current design (X_{cu}) not only increases the convergence speed but also improves the quality of the final optimal solutions.

References

Amiri, H., Radfar, N., Arab Solghar, A. and Mashayekhi, M. (2023), "Two improved teaching-learning-based optimization algorithms for the solution of inverse boundary design problems", *Soft Comput.*, https://doi.org/10.1007/s00500-023-08415-2.

Azad, S.K., Hasançebi, O. and Kazemzadeh Azad, S. (2013), "Upper bound strategy for metaheuristic based design optimization of steel frames", *Adv. Eng. Softw.*, **57**, 19-32. https://doi.org/10.1016/j.advengsoft.2012.11.016

Babaei, M., Atasoy, A., Hajirasouliha, I. and Mollaei, S. (2022), "Numerical solution of beam equation using neural networks and evolutionary optimization tools", *Adv. Comput. Des.*, **7**(1), 19-35. https://doi.org/10.12989/acd.2022.7.1.001.

Bouzouiki, M.E., Sedaghati, R. and Stiharu, I. (2021), "A non-uniform cellular automata framework for topology and sizing optimization of truss structures subjected to stress and displacement constraints", *Comput. Struct.*, **242**, 106394. https://doi.org/10.1016/j.compstruc.2020.106394.

Canyurt, O.E. and Hajela, P. (2005), "A cellular framework for structural analysis and optimization", *Comput. Methods Appl. Mech. Eng.*, **194**, 3516-3534. https://doi.org/10.1016/j.cma.2005.01.014

Cerny, V. (1985), "Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm", *J. Opt. Theory. Appl.*, **45**, 41-51. https://doi.org/10.1007/BF00940812

Cho, A. and Kang, T.H.K. (2021), "Computer-based design optimization of post-tensioned anchor for single-strand", *Adv. Comput. Des.*, **6**(4), 19-35. https://doi.org/10.12989/acd.2021.6.4.339.

Cui, Y., Geng, Z., Zhu, Q. and Han, Y. (2017), "Review: Multi-objective optimization methods and application in energy saving", *Energy*, **125**, 681-704. https://doi.org/10.1016/j.energy.2017.02.174

Dehghani, M., Mashayekhi, M. and Salajegheh, E. (2016), "Topology optimization of double- and triple-layer grids using a hybrid methodology", *Eng. Optim.*, **48**, 1333-1349. https://doi.org/10.1080/0305215X.2015.1105968

Dehghani, M., Mashayekhi, M. and Sharifi, M. (2021), "An efficient imperialist competitive algorithm with likelihood assimilation for topology, shape and sizing optimization of truss structures", *Appl. Math. Model.*, **93**, 1-27. https://doi.org/10.1016/j.apm.2020.11.044

Dogan, E. and Saka, M.P. (2012), "Optimum design of unbraced steel frames to LRFD-AISC using particle swarm optimization", *Adv. Eng. Softw.*, **46**, 27-34. https://doi.org/10.1016/j.advengsoft.2011.05.008

Duan, L., Xu, Z., Xu, W., Zhang, X., Du, Z., Liu, X. and Jiang, H. (2023), "Subdomain hybrid cellular automata method for material optimization of thin-walled frame structure under transverse impact", *Int. J. Impact Eng.*, **174**, 104524. https://doi.org/10.1016/j.ijimpeng.2023.104524.

Faramarzi, A. and Afshar, M.H. (2012), "Application of cellular automata to size and topology optimization of truss structures", *Sci. Iran.*, **19**, 373-380. https://doi.org/10.1016/j.scient.2012.04.009

Ghasemi, M.R., Ghasri, M. and Salarnia A. (2022), "Soccer league optimization-based championship algorithm (SLOCA): A fast novel meta-heuristic technique for optimization problems", *Adv. Comput. Des.*, **7**(4), 297-319. https://doi.org/10.12989/acd.2022.7.4.297.

Hasançebi, O. (2008), "Adaptive evolution strategies in structural optimization: Enhancing their computational performance with applications to large-scale structures", *Comput. Struct.*, **86**, 119-132.

https://doi.org/10.1016/j.compstruc.2007.05.012
Hasançebi, O., Çarbas, S., Dogan, E., Erdal, F. and Saka, M.P. (2009), "Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures", *Comput. Struct.*, **87**, 284-302. https://doi.org/10.1016/j.compstruc.2009.01.002
Hoekstra, A.G., Kroc, J. and Sloot, P.M.A. (2010), "*Simulating Complex Systems by Cellular Automata*", Springer, Berlin.
Kaveh, A., Eskandari, A. and Movasat, M. (2023), "Buckling resistance prediction of high-strength steel columns using metaheuristic-trained artificial neural networks", *Structures*, **56**, 104853. https://doi.org/10.1016/j.istruc.2023.07.043
Kaveh, A. and Zaerreza, A. (2022), "Reliability-based design optimization of the frame structures using the force method and SORA-DM framework", *Structures*, **45**, 814-827. https://doi.org/10.1016/j.istruc.2022.09.057
Kirkpatrick, S., Gerlatt, C.D. and Vecchi, M.P. (1983), "Optimization by simulated annealing", *Science.*, **220**, 671-680. https://doi.org/10.1126/science.220.4598.671
Liu, C., Zhang, F., Zhang, H., Shi, Z. and Zhu, H. (2023), "Optimization of assembly sequence of building components based on simulated annealing genetic algorithm", *Alexandria Eng. J.*, **62**, 257-268. https://doi.org/10.1016/j.aej.2022.07.025
Mashayekhi, M., Fadaee, M.J., Salajegheh, J. and Salajegheh, E. (2011), "Topology optimization of double-layer grids for earthquake loads using a two-stage ESO-ACO method", *Int. J. Optim. Civ. Eng.*, **1**, 211-232.
Mashayekhi, A., Mashayekhi, M. and Siciliano, B. (2023), "Identification and optimization of the operator's hand and a haptic device dynamic, using artificial intelligence methods", *Int. J. Dyn. Control*, https://doi.org/10.1007/s40435-023-01165-x.
Mashayekhi, M., Salajegheh, E., Salajegheh, J. and Fadaee, M.J. (2012), "Reliability-based topology optimization of double-layer grids using a two-stage optimization method", *Struct. Multidiscipl. Opt.*, **45**, 815-833. https://doi.org/10.1007/s00158-011-0744-6
Mashayekhi, M., Salajegheh, E. and Dehghani, M. (2015), "A new hybrid algorithm for topology optimization of double-layer grids", *Int. J. Optim. Civ. Eng.*, **1**(3), 353-374.
Mashayekhi, M., Salajegheh, E. and Dehghani, M. (2016), "Topology optimization of double and triple layer grid structures using a modified gravitational harmony search algorithm with efficient member grouping strategy", *Comput. Struct.*, **172**, 40-58. https://doi.org/10.1016/j.compstruc.2016.05.008
Mashayekhi, M. and Yousefi, R. (2021), "Topology and size optimization of truss structures using an improved crow search algorithm", *Struct. Eng. Mech.*, **77**(6), 779-795. https://doi.org/10.12989/sem.2021.77.6.779
Mohammadnejad, M. and Haji Kazemi, H. (2022), "Optimization of lateral resisting system of framed tubes combined with outrigger and belt truss", *Adv. Comput. Des.*, **7**(1), 19-35. https://doi.org/10.12989/acd.2022.7.1.019.
Mozafari, H., Ayob, A. and Kamali, F. (2012), "Optimization of functional graded plates for buckling load by using imperialist competitive algorithm", *Proced. Technol.*, **1**, 144-152. https://doi.org/10.1016/j.protcy.2012.02.028
Nabil, B., Khaled, B. and Mohamed, B. (2023), "Optimization of productivity in the rehabilitation of building linked to BIM", *Adv. Comput. Des.*, **8**(2), 179-190. https://doi.org/10.12989/acd.2023.8.2.179.
Neumann, J.V. (1966). "*Theory of self-reproducing automata*", University of Illinois Press.
Rajasekaran, S. (2001), "Optimization of large scale three dimensional reticulated structures using cellular genetics and neural networks", *Int. J. Space Struct.*, **16**, 315-324. https://doi.org/10.1260/026635101760832244
Rettl, M., Pletz, M. and Schuecker, C. (2023), "Evaluation of combinatorial algorithms for optimizing highly nonlinear structural problems", *Mater. Des.*, https://doi.org/10.1016/j.matdes.2023.111958. https://doi.org/10.1016/j.matdes.2023.111958
Sonmez, M. (2011), "Discrete optimum design of truss structures using artificial bee colony algorithm", *Struct. Multidiscipl. Opt.*, **43**, 85-97. https://doi.org/10.1007/s00158-010-0551-5

Ulam, S. (1952), "Random processes and transformations", *Proceedings of the International Congress of Mathematics*, **2**, 85-87.

Vasile, A., Coropetchi, I.C., Sorohan, S., Picu, C.R. and Constantinescu, D.M. (2022), "A simulated annealing algorithm for stiffness optimization", *Procedia Structural Integrity*, **37**, 857-864.

Zhang, X., Wang, D., Huang, B., Wang, S., Zhang, Z., Li, S., Xie, C., Kong, D. (2023), "A dynamic-static coupling topology optimization method based on hybrid cellular automata", *Structures*., **50**, 1573-1583. https://doi.org/10.1016/j.istruc.2023.02.120

TK