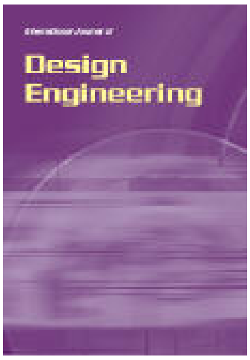


International Journal of Design Engineering

ISSN online: 1751-5882 - ISSN print: 1751-5874
https://www.inderscience.com/ijde

A genetic algorithm-based structural topology optimisation

S.L. Gavali, Y.P. Reddy, K.N. Vijayakumar

DOI: 10.1504/IJDE.2022.10049133

Article History:

Received:	01 February 2022
Accepted:	19 June 2022
Published online:	21 November 2022

A genetic algorithm-based structural topology optimisation

S.L. Gavali* and Y.P. Reddy

Department of Mechanical Engineering,
Sinhgad College of Engineering – Research Center,
S.P. Pune University,
Pune, India
Email: gavali.suresh@gmail.com
Email: ypreddy.scoe@sinhgad.edu
*Corresponding author

K.N. Vijayakumar

Department of Mechanical Engineering,
Dwarkadas J. Sanghvi College of Engineering,
Mumbai, India
Email: kotturvijaykumar@gmail.com

Abstract: In recent times advancements in 3D printing technology, primarily pertaining to biocompatible materials, have paved the way for customisable biomedical implants. However, the costs of these implants are very high mostly due to expensive materials like Ti64 and associated printing technologies. One method to mitigate this cost challenge is by optimising the material content in the implant structure while keeping its robustness intact. Various methods of structural topology optimisation have been explored by researchers in this field to overcome this challenge. In this paper, a novel genetic algorithm (GA)-based topology optimisation procedure is compared with methods like structural, lattice topology optimisation and Ad joint method. Comparisons are made with respect to mass reduction with minimum deformation. The procedure is implementing a MATLAB code to obtain structurally optimised topologies for various canonical structures. For experimental validation of optimisation procedure, a cantilever beam structure made of Ti64 material was printed as a test coupon and compared with MATLAB simulation. The obtained optimised topologies were found to be in agreement with topologies obtained using different optimisation techniques with similar boundary conditions.

Keywords: topology optimisation; stiffness matrix; bone implant; genetic algorithms; lattice.

Reference to this paper should be made as follows: Gavali, S.L., Reddy, Y.P. and Vijayakumar, K.N. (2022) ‘A genetic algorithm-based structural topology optimisation’, *Int. J. Design Engineering*, Vol. 11, No. 1, pp.27–46.

Biographical notes: S.L. Gavali is a Lecturer at Cusrow Wadia Institute of Technology, Pune, Maharashtra, India. He obtained MTech in Mechanical Engineering with Design Engineering as specialisation from College of Engineering Pune, Savitribai Phule Pune University. His research areas are design, simulation and experimentation in Orthopedic Implant. He has six years of industrial and 14 years of teaching experience in various courses of mechanical engineering. He has two publications in various national and international conference proceedings.

Y.P. Reddy is currently Vice Principal and Professor in Mechanical Engineering at the Sinhgad College of Engineering, Pune (India). He received his PhD from Jawaharlal Nehru Technological University, Hyderabad and MTech. from IIT, Kharagpur. He served as member of senate, board of studies and also as subject expert on the research and recognition committee for the subject of production and industrial engineering under the Faculty of Science and Technology at Savitribai Phule Pune University. The author's primary research interests include manufacturing processes, manufacturing systems modelling and simulation.

K.N. Vijayakumar is a Professor of Mechanical Engineering Department at Dwarkadas J. Sanghvi College of Engineering, Mumbai University, India. He obtained PhD degree in Mechanical Engineering from Mumbai University. His specialisation includes mechanical design, total quality management, WCM. He has 28 years of teaching experience in various subjects of mechanical engineering. He has more than nine national and five international research publications along with eight national and international conferences to his credit.

1 Introduction

With remarkable development in medical, bio-medical engineering and material science fields, replacement of impaired bone joints have increased. The utilisation of hip implants is to grow by a compound annual growth rate (CAGR) of 1.2%, leading to an increase from 1.8 million per year in 2015 to 2.8 (2.6–2.9) in 2050 (Pabinger et al., 2018). Typically, subtractive machining techniques have been used for manufacturing bone implants. However, for better customisation and lesser lead times, 3D printing has been lately explored extensively (Ma et al., 2017). Although, with the use of additive manufacturing very high degree of customisation is possible, the conventionally used bio-compatible materials like Ti-6Al-4V powders used for manufacturing are still very expensive. Moreover, manufacturing lead times also augment the cost. Hence, optimisation to reduce material as well as manufacturing time has been researched extensively in this field (Ma et al., 2017; Chate and Deshpande, 2017). One such technique, namely structural topology optimisation has been reported to be very effective and thus has been a standard optimisation option (Zhu et al., 2021; Kumar and Rakshit, 2020). However, structural topology optimisation does not guarantee globally optimal solutions at times (Cai et al., 2020). Another technique, namely lattice optimisation allows to generate a lattice structure of different configurations within the region of interest (Sigmund, 2001). It includes varying thickness of the lattice members as part of the optimisation. Lattice structures can be highly beneficial because weight can be substantially reduced compared to solid parts made using traditional manufacturing methods. Furthermore, recent advances in additive manufacturing enable the creation of lattice structures in ways that were not possible with traditional manufacturing (Meneses et al., 2018). The background for lattice optimisation is analogous to the discrete optimisation process used for 2D trusses (Cheng et al., 2017). The lattice configurations can be visualised to be formed of truss members of different cross-section areas. As a result of the optimisation procedure, a sensitivity map is obtained. It is utilised to decide the cell densities which in turn are dependent on the lattice member areas.

Although, lattice optimisation procedure is highly compatible with 3D printing, it does not guarantee a globally optimal structure. Genetic algorithms (GAs) for topology optimisation have been explored extensively by researchers worldwide (Cheng et al., 2017). A GA-based structural topology optimisation technique was investigated using bit-array representation (Cheng et al., 2017). Similar approach was explored for sizing optimisation of truss structures (Yang and Tai, 2005; Okwu and Tartibu, 2021). A constraint handling strategy was proposed for bit-array representative GA technique (Šešok and Belevičius, 2007). Since GA-based optimisation is known to be computationally expensive, an approach to improve performance of topological optimisation tool by introducing dynamic variation of the population size of children during the process of optimisation was investigated (Wang and Tai, 2003). Basically GA evolves as a set of individual termed population. According to Darwinian survival of fittest principle, the fittest individual that is near to optimal point of the function will appear. The steps involved are selection, crossover and mutation (Du et al., 2018). The classical 1 point, 2 point and uniform crossovers, was the first choice for operators and divides in two different operators, diagonal crossover and block crossover. The mutation proceeds by flipping randomly selected bits and are classified as Boundary mutations and epistatic mutation (Du et al., 2018).

The general framework of a cantilever plate is that it is fixed on vertical part of its boundary and a single force is applied on the middle. According to Chapman (1994) and Jana et al. (1992) the quality of solution greatly depends on α (angle of deflection of cantilever beam during loading). A medium value of α is considered and then it is increased by a factor of 10 to satisfy the constraints and iterated solutions of GAs are demonstrated. For the cantilever plate the height, loading, displacements are provided so that more optimal solutions can exists and GA method is then able to find such multiple solutions. Finally, the compliance fitness is used in order to compare the results with those of homogenisation method to check its feasibility (Jouve, 1993; Melanie, 1998).

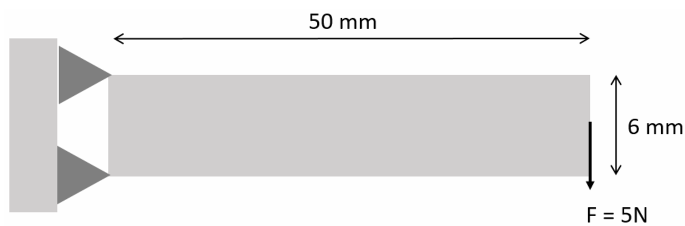
In literature, various methods of structural topology optimisation and machine learning techniques are available. Prominent among them are GAs, synthetic annealing, and convoluted neural networks (CNNs) (Jensen, 1992). GAs are either used directly or in combination with conventional topology optimisation methods. GA-based algorithms are robust optimisers, and hence can handle problems with multiple local minimum. However, these methods are more time consuming. Synthetic annealing method is inspired from metallurgy; it is a regenerative topology optimisation technique similar to GAs. This method is a robust optimisation method unlike gradient-based methods and can effectively deal with multiple extrimum. Very unique topology features can be ascertained using this technique. CNN is a machine learning technique mostly employed in image processing, object detection and computational fluid dynamics. However, this method has now found application in structural topology optimisation. A set of different classes of structures with associated boundary conditions and stress strain fields are used as training data for the CNN model. Once the CNN model is trained, any input domain with boundary conditions and be input in the model and the output topology is ascertained (Meneses et al., 2018; Klarbring and Christensen, 2009; Yang and Tai, 2005).

In this work, a GA-based optimisation methodology is explored for material optimisation of hip joint bone implants. It is carried out by taking a cantilever beam as an example for topology optimisation. The objective is to optimise the material content in the structure so as to reduce the cost of cantilever beam by use of different lattice structures to augment the lattice optimisation process. The procedure is extended to the hip joint bone implant to reduce the material content.

2 Topology optimisation problem

A canonical structural configuration as depicted in Figure 1 is used for topology optimisation with different boundary conditions. A cantilever beam structure of size 50mm×6mm×6mm made of Ti64 material is 3D printed modelled as a test coupon. The boundary conditions used are, fixed at one end and single force applied at other end as 5 N.

Figure 1 Cantilever beam depicting dimensions and boundary conditions used for analysis



In general, a structural optimisation problem takes the following form:

$$(SO)\begin{cases} \quad\quad\quad min: f(x, y) \\ Subject\ to\!: \begin{cases} Behavioural\ constraints\ on\ y \\ Design\ constraints\ on\ x \\ Equilibrium\ constraints \end{cases} \end{cases} \tag{1}$$

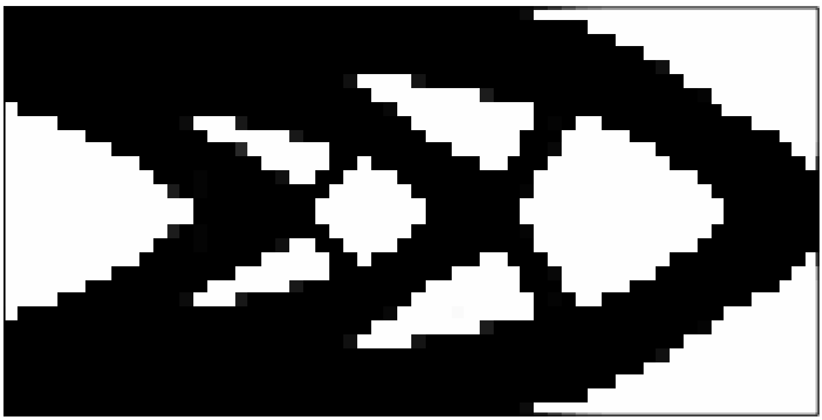
The structural optimisation problem can be divided into three classes, namely sizing optimisation: this is when x is some type of structural thickness, i.e., cross-sectional areas of truss members, shape optimisation where x represents the form or contour of some part of the boundary of the structural domain and topology optimisation, which is the most general form of structural optimisation. In a discrete case, such as for a truss, it is achieved by taking cross-sectional areas of truss members as design variables.

A topology optimisation problem based on the power law approach, where the objective is to minimise compliance can be written as (Zhu et al., 2021):

$$\begin{aligned} &\min_x : c(x) = U^T KU = \sum_{e=1}^{N} (x_e) u_e^T k_0 u_e \\ &\text{subject to:} \quad \frac{V(x)}{V_0} = f \\ &\qquad\qquad\quad KU = F \\ &\qquad\qquad\quad 0 < x_{\min} \le x \le 1 \end{aligned} \tag{2}$$

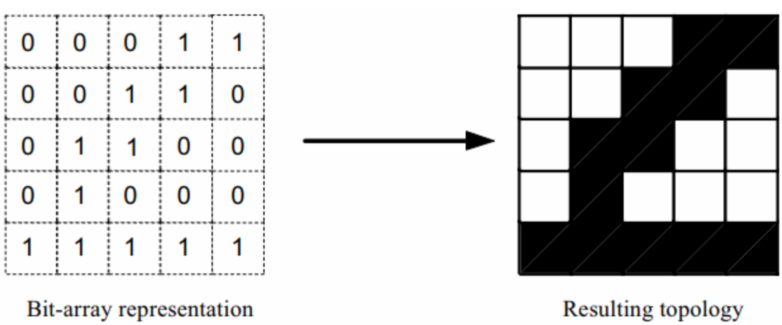
where U and F are the global displacement and force vectors, respectively, K is the global stiffness matrix, u_e and k_e are the element displacement vector and stiffness, $V(x)$ and $V(0)$ are volume and design domain volume, respectively and f is the prescribed volume fraction, and x is the vector of design variables. The power law approach is applied to a 2D cantilever beam using MATLAB for obtaining the output topology. Figure 2 shows the output topology of the beam obtained through structural topology optimisation using power law approach.

Figure 2 Output configuration of a cantilever beam obtained through structural topology optimisation using power law approach



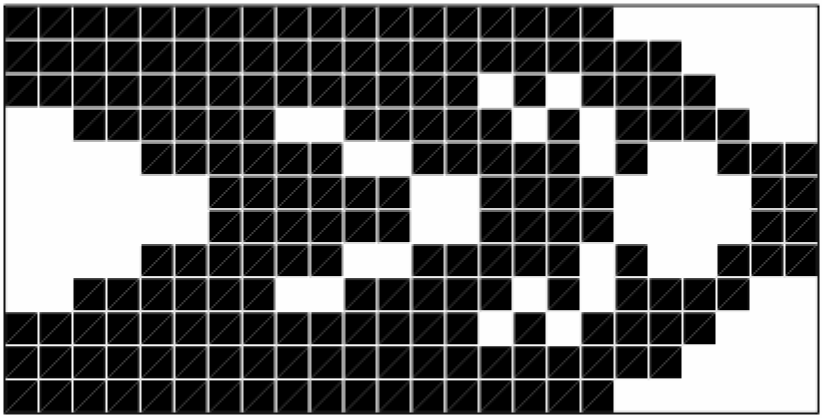
In case of GA-based topology optimisation, parent structures have to be represented in a manner which is suitable for operations like crossover and mutation. Several methods have been explored for topology representation. Figure 3 describes the bit-array representation and resulting topology commonly used in GA-based topology optimisation. The optimal topology obtained by the GA bit-array representation method is shown in Figure 4.

Figure 3 Decoding step of the bit-array representation



Source: Yang and Tai (2005)

Figure 4 Optimal topology for a cantilever beam obtained by the GA with bit-array representation



The GA presented in this paper represents the structure using finite truss elements, where the areas of cross-section of these truss elements are randomly chosen and subsequently represented by binary code.

3 GA-based topology optimisation in MATLAB

The MATLAB code is written for optimisation of cantilever beam with the boundary conditions shown in Figure 1. MATLAB implementation involves creation of a geometric model function, 2D finite element function, boundary condition function, genetic cross-over function, genetic mutation function and post processing function (Annexure). The topology optimisation problem based on the power law approach stated in equation (2) can be solved by using different approaches such as optimality criteria (OC), sequential linear programming (SLP) or method of moving asymptotes OC (Cai et al., 2020). For simplicity, OC method is used in the present work. Functions are developed in MATLAB for different optimality criterions. The GA used in this work randomly initiates a sample set of initial parent population. In this case parents are truss structures with variable member cross-section areas which are randomly specified. The fitness of each parent configuration is evaluated using the finite element code as per the fitness criterion. Once the parent generation is ranked, mating step is initiated and a new generation is produced using an elitist strategy (Denies, 2012).

3.1 Main program

In the main script, first step is to initialise the geometry by setting up the domain coordinates (Annexure). The geometric model function then automatically generates the nodes and elements and stores this information in a node list and an element list vector. Once the mesh in the form of truss elements has been created, the code manually takes the input boundary conditions using the boundary condition function. In the GA part of the code, a set of parent configurations is generated initially with random truss element areas. These initial parent generations as shown in Figure 5 undergo various permutation and combinations to finally generate the optimum structure with the help of GA.

3.2 Finite element code

The finite element code written for this work is based on 2D truss elements (Annexure). Each element has only two nodes and each node has 2 degrees of freedom. The element stiffness matrix is defined as:

$$K_e = \frac{EA}{L}\begin{bmatrix} C^2 & CS & -C^2 & -CS \\ CS & S^2 & -CS & -S^2 \\ -C^2 & -CS & C^2 & CS \\ -CS & -S^2 & CS & S^2 \end{bmatrix} \tag{3}$$

where E = modulus of elasticity of the element, A = cross section Area of the element, L = Length of element and $C = \cos\theta$, $S = \sin\theta$ and θ is the angle made by the element w.r.t the basis.

3.3 Objective function

The objective function is the important quantity in any optimisation procedure. In case of structural topology optimisation problem shown in Figure 1, traditionally the mean compliance is chosen as the objective function to be minimised. Compliance is defined as:

$$C(x) = F^T u \tag{4}$$

where F is the external force and u is the nodal displacement.

3.4 Genetic algorithm

The first step involved in GA is the conversion of a structural configuration into genetic code (Wang and Tai, 2003). Each specific member has a genetic code which specifies the member cross section area. Hence, the area A_n is unique to a specific element. As a rule, no element shall have zero area of cross section in the beginning. First get a set of random parents, find their fitness, and perform a mating ritual as per rule. Randomly selected parent configuration set for optimisation process is as depicted in Figure 5.

Crossover is performed for all the elements as per the rule described in Figure 6. After a number of generations have passed, mutation is performed as per the rule as shown in Figure 7. After a decided number of iterations suitable configuration is arrived, which can further be optimised using gradient decent and sensitivity analysis. The complete algorithm flow is described in Figure 8. Once the parent generation is analysed for fitness using the FEA function, the set of parent generation is sorted and ranked as per their fitness level. A mating pool is then specified where the most fit configurations undergo the process of crossover. To achieve this, the element areas of a particular configuration are first converted to a binary code and subsequently rendered to a crossover operation. The new generations formed after this iteration is further sorted and this process is repeated in a loop until the change in the objective function is less than a set magnitude.

Figure 5 Randomly selected parent configuration set for optimisation process (see online version for colours)

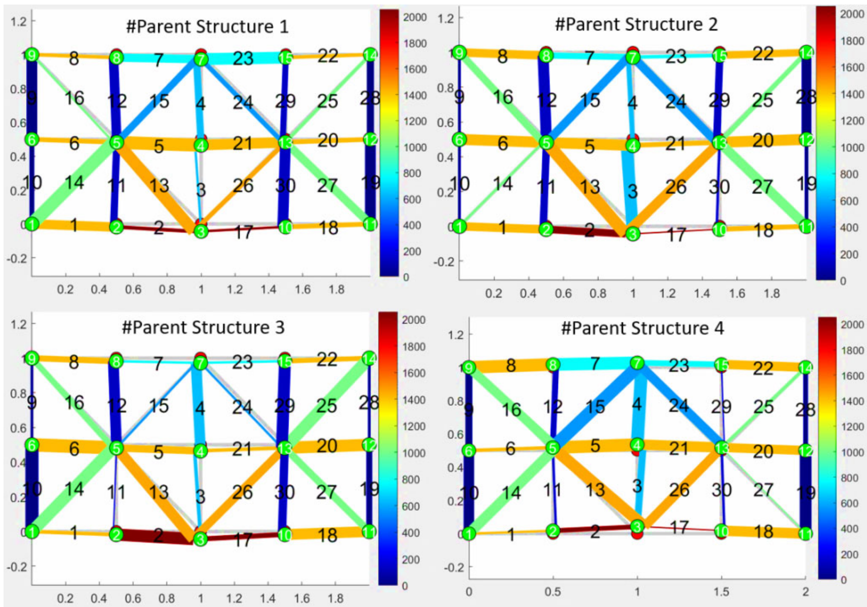


Figure 6 Pictorial description of crossover in genetic algorithms

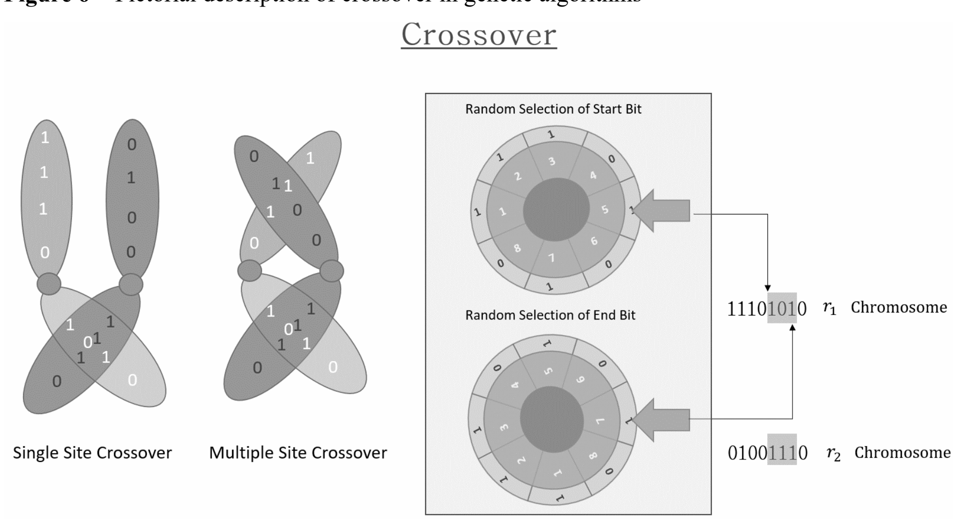
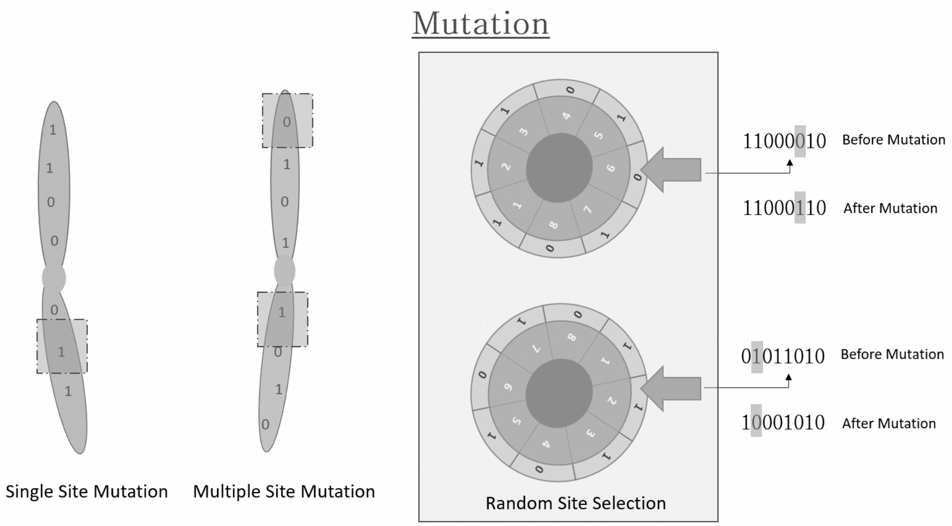
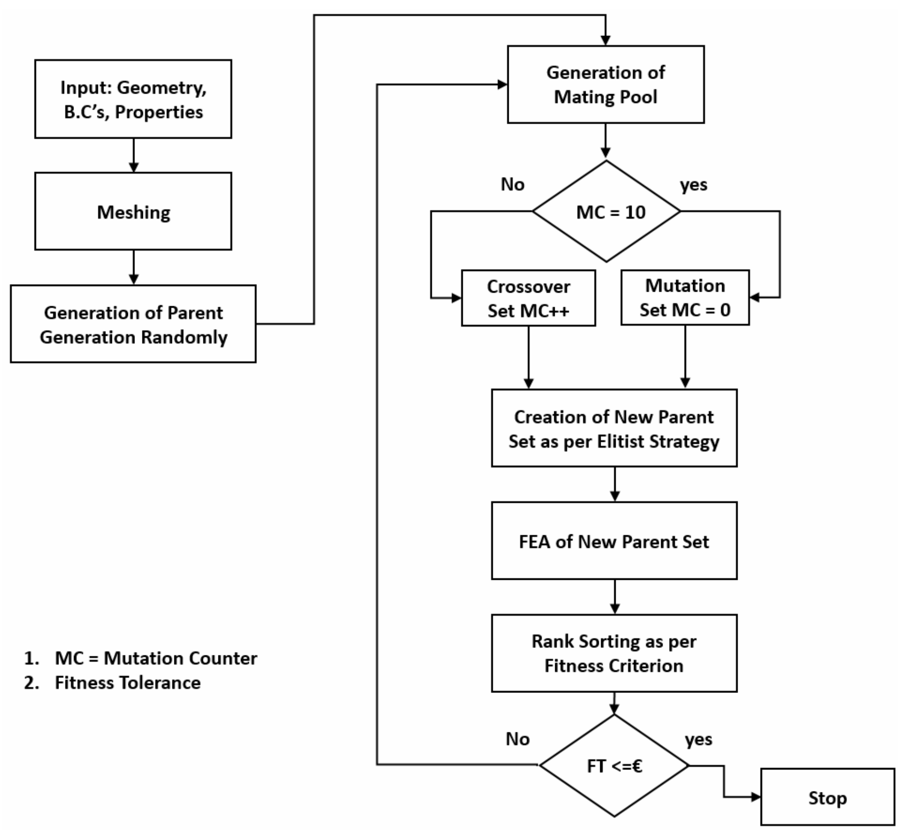


Figure 7 Pictorial description of mutation in GAs



Source: Jouve (1993)

Figure 8 Algorithm flow diagram for genetic topology optimisation



The mutation operation is performed after a gap of every few iterations to obtain the global extremum. Without mutation operation the algorithm has a tendency to converge to a local extremum. Moreover, an elitist strategy has been used while sorting the population so that a good parent configuration is retained. The flow diagram for genetic topology optimisation is shown in Figure 8.

3.5 Post-processing

The post-processing function is used to visualise the results of the optimisation process. After the optimisation has been carried out, truss members with areas less than a set value have been eliminated. The function also calculates the node reactions and stresses.

4 ANSYS simulation

Topology optimisation of the test coupon geometry is carried out using ANSYS Workbench. The topology optimisation is carried out for the test coupon using different density (ρ – kg/m^3) cubic cellular lattices structures shown in Figure 9. Topology optimised structure is obtained after number of interactions iteration in ANSYS. Figure 10 shows the optimised topology obtained by ANSYS simulation. The lattice structure used is simple cubic lattice with relative density (ρ) – 0.1739 kg/m^3.

Figure 9 Cubic cellular structures with different relative densities used in lattice optimisation, (a) $\rho_r = 0.1739$ (b) $\rho_r = 0.2865$ (c) $\rho_r = 0.4123$ (d) $\rho_r = 0.5428$ (e) $\rho_r = 0.6695$

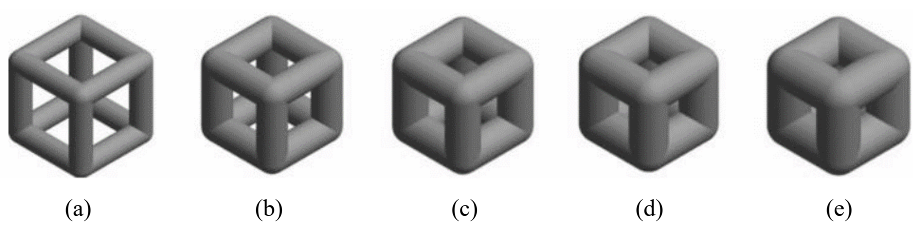


Figure 10 Topology optimisation of cantilever beam structure in ANSYS (see online version for colours)

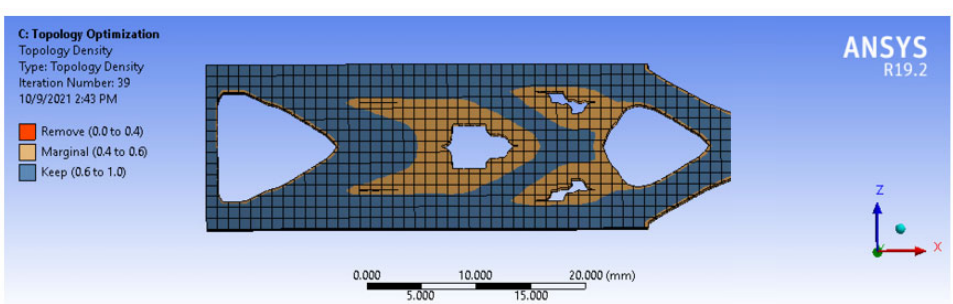


Figure 11 Lattice optimised structure with simple cubic lattice (see online version for colours)

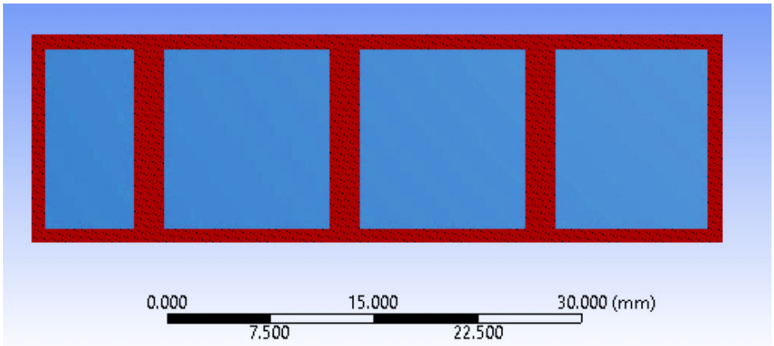
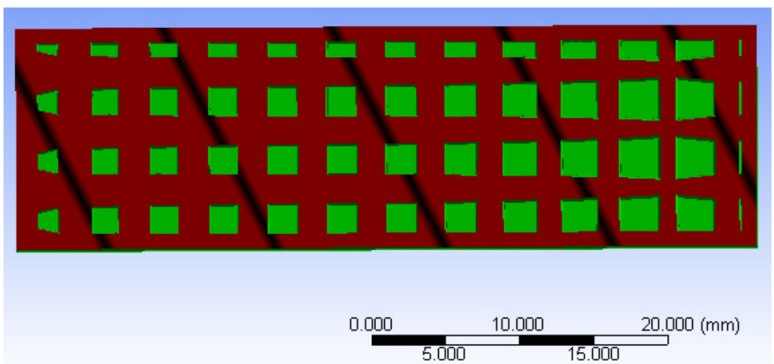


Figure 12 Lattice optimised structure with sensitivity map (see online version for colours)



As a result of the optimisation procedure using ANSYS simulation, a sensitivity map is obtained. Figure 12 shows the sensitivity map, it is utilised to decide the cell densities which in turn are dependent on the lattice member areas.

5 Result and discussions

Cantilever beam geometry simulated using ANSYS topology optimisation was taken to analyse the efficacy of the developed genetic topology optimisation code. Figure 13 shows the first four random parent configurations of initial generation.

Figure 14 shows the convergence curve for the cantilever beam with fixed boundary conditions. For every iteration random parents of previous generation is selected. Structure becomes better in every iteration and converges to the applied boundary condition. Figure 14 shows the convergence curve for genetic topology optimisation. Objective function 1 is to minimise the given cost function, i.e., *Umin* (minimum Deflection). The final optimised topology can be seen in Figure 15 for these boundary conditions.

Figure 13 First four random parents of initial generation (see online version for colours)

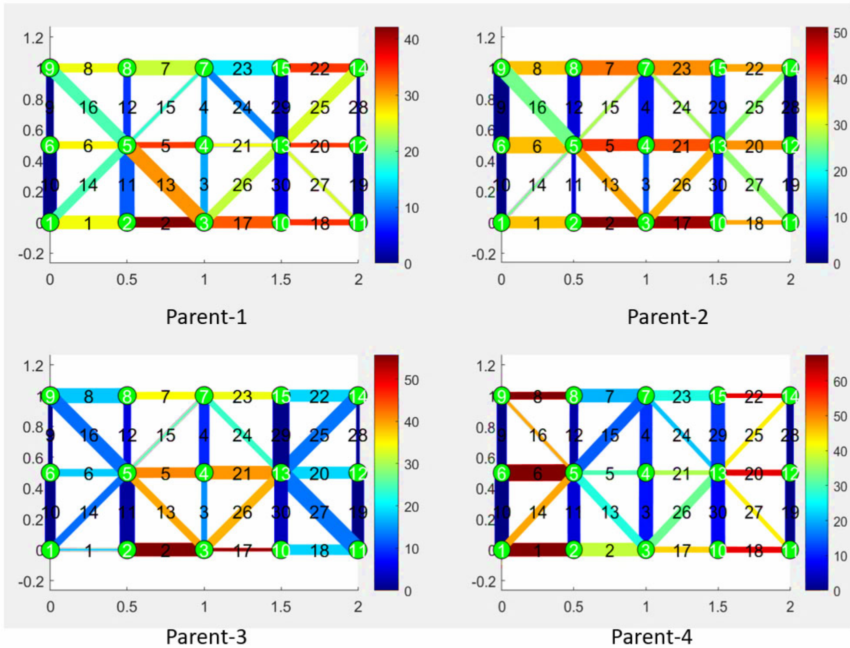


Figure 14 Convergence curve for genetic topology optimisation (see online version for colours)

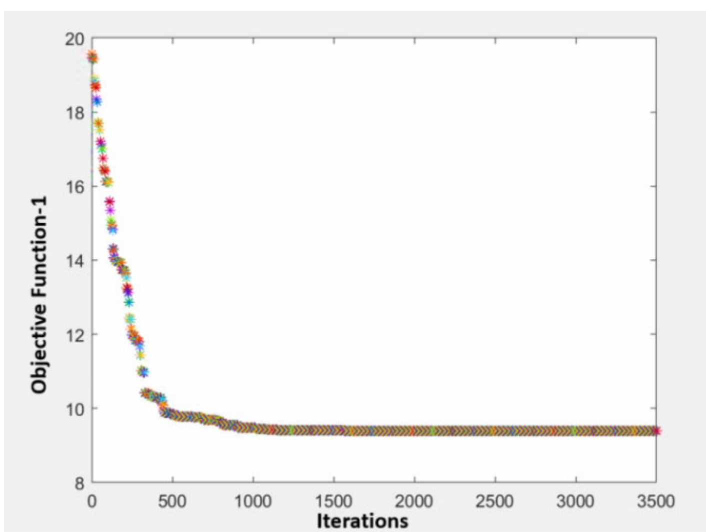


Figure 15 Optimised geometric configuration using GA-based optimiser (see online version for colours)

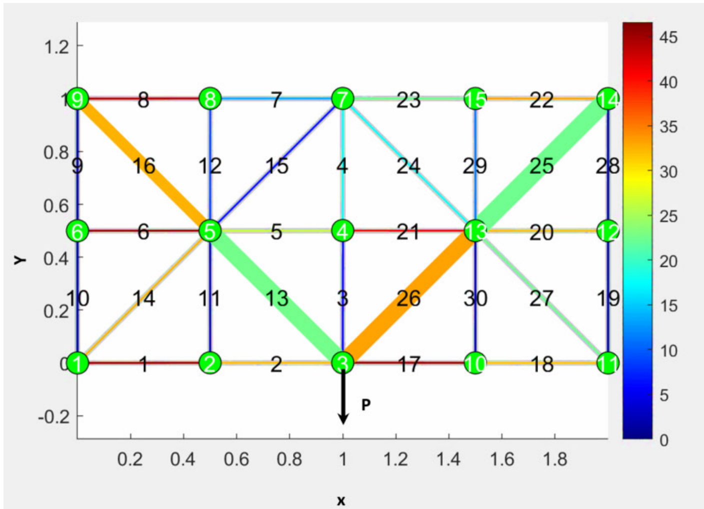
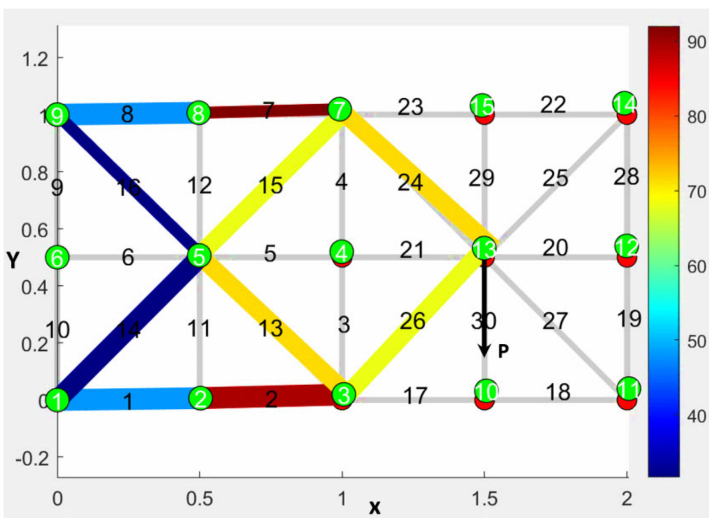
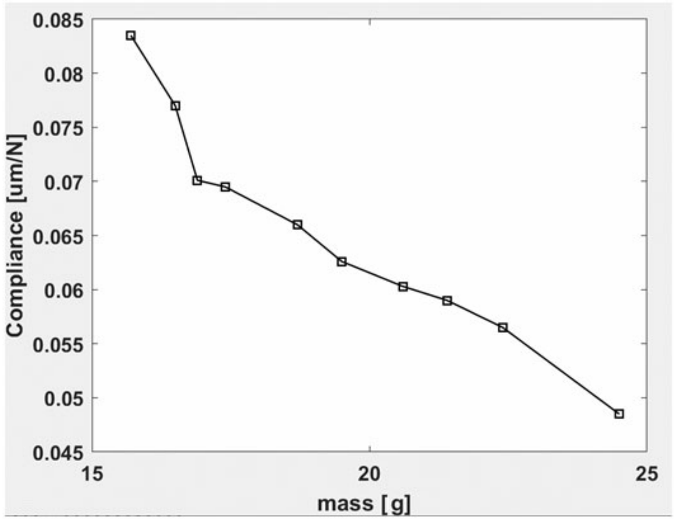


Figure 16 Optimised geometry for cantilever beam (see online version for colours)



The same geometry was analysed for cantilever beam boundary conditions and a force (P) of 5 N was specified at node 13 as shown in Figure 16. It can be inferred from the figure that the final configuration is similar in topology as obtained using ANSYS Topology optimisation simulation as shown in Figure 10. The trend of compliance w.r.t beam mass is depicted in Figure 17.

Figure 17 Variation of structure compliance w.r.t mass reduction



For the test coupon optimisation, Table 1 presents the achieved mass reduction using different optimisation techniques.

Table 1 Results of various optimisation techniques for cantilever structure shown in Figure 1

Sr. no.	*Structure*	*Mass (g)*	*Mass reduction (%)*	*Optimisation method*	*Max. deformation (10^{-3} mm)*
1	Cantilever	35.32	0	Un-optimised	3.84
2	Cantilever	14.82	58.2	Ad joint	9.25
3	Cantilever	18.4	47.8	Cubic lattice	8.36
4	Cantilever	23.7	32.8	Body diagonal lattice	5.18
5	Cantilever	27.6	21.8	Cross diagonal	4.22
6	Cantilever	19.8	43.9	Genetic algorithm	4.83

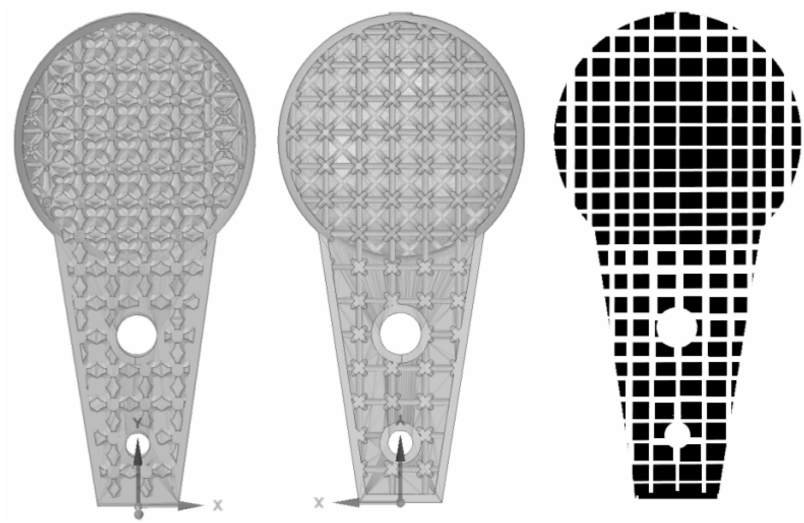
Table 1 shows the results of the various optimisation techniques for cantilever structure shown in Figure 1. GA optimisation shows mass reduction of 43.9% with maximum deformation of 4.83×10^{-3} mm.

Ti64 material is widely used in implant manufacturing due to good biocompatibility, but the costs of these implants are very high mostly due to expensive materials and associated printing technologies. One method to mitigate this challenge is by optimising the material content in the implant structure while keeping its robustness intact.

Optimised canonical structure GA technique printed by 3D printing technology with material as Ti64 and tested for deformation at applied load. The testing results are in line with the ANSYS simulation. It can be inferred from the figure that the final configuration is similar in topology as obtained using ANSYS topology optimisation simulation as shown in Figure 10. As a result ANSYS topology optimisation simulation extended for Femur bone implant and observed good reduction in the material content.

After validating the developed GA technique on a canonical structure, femur bone implant geometry was rendered using lattice optimisation in ANSYS and present GA technique in MATLAB. The resulted topologies are depicted in Figure 18.

Figure 18 (a, b) Bone implant geometry using lattice optimisation (c) Implant geometry using present GA



(a) (b) (c)

A model of the proposed implant geometry is shown in Figure 19. The implant is fabricated using 3D printing method. The material for the designed implant was chosen to be Ti-6Al-4V.

Figure 19 Cut section view of optimised bone implant with ports for infill material removal (see online version for colours)

5 Conclusions

A GA-based 2D topology optimisation code is developed to augment additive manufacturing technology. The initial parent pool comprised of parent structures having truss members with randomly selected cross-section areas. The final best fit configurations were compared with other optimised configurations using different optimisation methods. It was observed that the optimised structures derived using different methods had similar topologies and observed mass reduction by keeping structural compliance low. Although, 2D truss like genetic topology optimisation used in this work is similar to lattice topology optimisation, it can be used as a generative method having a higher degree of configuration space. The GA is a robust optimiser and achieved more accurate result of optimisation as compared with other methods. Mutation, crossover and ranking of parent set are performed on fitness criteria which lead to better structure after every iteration. An attenuation of 43.9% in mass of the structural topology of cantilever beam is achieved using GA and 58.2% is achieved using ad-joint method. But in ad-joint method the structure becomes low compliance to the applied boundary condition and increased deformation. However, a finer mesh in the domain will aid in further supporting the claim that this GA-based topology optimisation method is an effective alternative to conventional structural optimisation techniques.

References

Cai, L., Nauman, E.A., Pedersen, C.B.W. and Neu, C.P. (2020) 'Finite deformation elastography of articular cartilage and biomaterials based on imaging and topology optimization', *Sci. Rep.*, Vol. 10, No. 1, pp.1–12, doi: 10.1038/s41598-020-64723-9.

Chapman, C. (1994) *Structural Topology Optimization Via the Genetic Algorithm*, MS thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology,.

Chapman, C.D., Saitou, K. and Jakiela, M.J. (1994) 'Genetic algorithms as an approach to conguration and topology design', *Transactions of the ASME*, Vol. 116, No. 4, pp.1005–1012.

Chate, G. and Deshpande, A.S. (2017) 'Optimisation for geometrical dimension of a product using 3D printer based on fused deposition modelling', *International Journal of Additive and Subtractive Materials Manufacturing,* Vol. 1, No. 2, pp.133–144, doi:10.1504/IJASMM. 2017.088201.

Cheng, L., Zhang, P., Biyikli, E., Bai, J., Robbins, J. and To, A. (2017) 'Efficient design optimization of variable-density cellular structures for additive manufacturing: Theory and experimental validation', *Rapid Prototyp. J.*, Vol. 23, No. 4, pp.660–677, doi: 10.1108/RPJ-04-2016-0069.

Denies, J. (2012) 'Genetic algorithme-based topology optimization : performance improvement through dynamic evolution of the population size', *International Symposium on Power Electronics, Electrical Drives, Automation and Motion*, June, doi: 10.1109/SPEEDAM. 2012.6264469.

Du, H., Wang, Z., Zhan, W. and Guo, J. (2018) 'Elitism and distance strategy for selection of evolutionary algorithms', *IEEE Access*, Vol. 6, pp.44531–44541, doi: 10.1109/ACCESS.2018. 2861760.

Jensen, E. (1992) *Topological Structural Design using Genetic Algorithms*, Doctor of Philosophy thesis, November, Purdue University.

Jouve, F. (1993) *Modelisation mathematique de l'il en elasticite non-lineaire Recherches en Mathematiques Appliquees (RMA 26)*, Masson, Paris.

Klarbring, A. and Christensen, P.W. (2009) *An Introduction to Structural Optimization*, 1st ed., Springer International Publishing.

Kumar, K.E.S. and Rakshit, S. (2020) 'Topology optimization of the hip bone for a few activities of daily living', *Int. J. Adv. Eng. Sci. Appl. Math.*, Vol. 12, Nos. 3–4, pp.193–210, doi: 10.1007/s12572-020-00285-3.

Ma, L. et al. (2017) '3D printed personalized titanium plates improve clinical outcome in microwave ablation of bone tumors around the knee', *Sci. Rep.*, Vol. 7, No. 1, pp.1–10, doi: 10.1038/s41598-017-07243-3.

Melanie, M. (1998) *An Introduction to Genetic Algorithms*, 5th ed., First MIT Press paperback edition; Cambridge, Massachusetts; London, England.

Meneses, G.A., Pereira, A. and Menezes, I.F.M. (2018) 'Lattice structures design by means of topology optimization', *Mecánica Comput.*, Vol. 36, No. 46, pp.2111–2120.

Okwu, M.O. and Tartibu, L.K. (2021) 'Sizing and topology optimization of trusses using genetic algorithm', *Stud. Comput. Intell.*, Vol. 927, pp.125–132, doi: 10.1007/978-3-030-61111-8_13.

Pabinger, C., Lothaller, H., Portner, N. and Geissler, A. (2018) 'Projections of hip arthroplasty in OECD countries up to 2050', *Hip Int.*, September, Vol. 28, No. 5, pp.498–506, Epub: 21 May 2018, PMID: 29783896, doi: 10.1177/1120700018757940.

Šešok, D. and Belevičius, R. (2007) 'Use of genetic algorithms in topology optimization of truss structures', *Mechanika*, Vol. 64, No. 2, pp.34–39, doi: 10.5755/j01.mech.64.2.14795.

Sigmund, O. (2001) 'A 99 line topology optimization code written in MATLAB', *Struct. Multidiscip. Optim.*, Vol. 21, No. 2, pp.120–127, doi: 10.1007/s001580050176.

Wang, S.Y. and Tai, K. (2003) 'A constraint handling strategy for bit-array representation GA in structural topology optimization', *2003 Congr. Evol. Comput. CEC 2003 - Proc.*, April, Vol. 1, pp.671–677, doi: 10.1109/CEC.2003.1299640.

Yang, S.Y. and Tai, K. (2005) 'Structural topology design optimization using genetic algorithms with a bit-array representation', *Comput. Methods Appl. Mech. Eng.*, Vol. 194, Nos. 36–38, pp.3749–3770.

Zhu, J., Zhou, H., Wang, C., Zhou, L., Yuan, S. and Zhang, W. (2021) 'A review of topology optimization for additive manufacturing: status and challenges', *Chinese J. Aeronaut.*, Vol. 34, No. 1, pp.91–110, doi: 10.1016/j.cja.2020.09.020.

Annexure

MATLAB code

```
%% Genetic Algorithm Code for Topology Optimization %%

%% Written by Suresh Gavali and Eshan Dhar (Gyrodrive Machineries (P) Ltd) %%

%% Genetic Algorithm Code for Topology Optimization %%

%% Function Description%%

%% Code takes parent vectors comprising of element areas as inputs %%

%% and returns new parents as per genetic Algorithm rules %%

clear all

format long

%% Step 1: Setup Configuration Geometry

[ENL,DOFs,DOCs,NL,EL,E] = Geometry_Setup();
```

```
%% Step 2: Generate First Parent Geometries
NoP = 4;%% Number of Parents
NoE = size(EL,1);%% Number of Elements
NoN = size(NL,1);%% Number of Nodes
PD = 2;%% Problem Dimension
a = 0.01;%% Lower Limit of Element Area
b = 1;%% Upper Limit of Element Area
ENL_Pool = zeros(NoN,PD*6,NoP);%% Pool of Extended Node List of all parents
A_Pool = zeros(NoE,NoP);
fori=1:NoP
A_Pool(:,i) = a + (b-a).*rand(NoE,1);%% Generate First Population Randomly between
[a,b]
end

%% Step 3: Perform FEA Analysis on All Parents
U = zeros(NoP,1);
fori=1:NoP
A = A_Pool(:,i);%% Generate First Population Randomly between [a,b]
[ENL] = FEA(ENL,EL,NL,E,A,DOFs,DOCs);

%% Post Processing %%
Node_flag = &#39;on&#39;;
Element_flag = &#39;on&#39;;
mag = 30;%% Scaling Factor
subplot(2,2,i);
post_process(NL,EL,ENL,E,Node_flag,Element_flag,mag,A);
ENL_Pool(:,:,i) = ENL;
u = ENL(:,10);
Ui,1) = abs(u(3,1)*sum(A));%% Optimality Criterion
end
%% Step 4: Rank All Parents as per Fitness Levels based on Optimality criterion %%
close all
```

```
[FF,I] = sort(U);%% Sort and Rank
A_Pool = A_Pool(:,I);%% Sort A_Pool according to rank [ Rank 1 is best ]
%% Now A_Pool is sorted and we will check that first %%
U = zeros(NoP,1);
fori=1:NoP
A = A_Pool(:,i);%% Generate First Population Randomly between [a,b]
[ENL] = FEA(ENL,EL,NL,E,A,DOFs,DOCs);
ENL_Pool(:,:,i) = ENL;
u = ENL(:,10);
U(i,1) = u(3,1)/sum(A);
End
%% Post Processing %%
Node_flag = &#39;on&#39;;
Element_flag = &#39;on&#39;;
mag = 300;%% Scaling Factor
subplot(2,2,i);
% post_process(NL,EL,ENL,E,Node_flag,Element_flag,mag,A);
%%Checked Works Fine %%%
close all
M = 0;%% Counter for Mutation
Opt_criterion = 100;
for j = 1:1500

%% Step 4: Mating and Creation of Offspring’s
[A_Pool] = Cross_Over(A_Pool,NoE);
NoP = size(A_Pool,2);
U = zeros(NoP,1);
if M==4
[A_Pool,M] = Mutation(A_Pool,NoE);
end
M = M+1;
fori=1:NoP
```

```
A = A_Pool(:,i);%% Generate First Population Randomly between [a,b]
[ENL] = FEA(ENL,EL,NL,E,A,DOFs,DOCs);
ENL_Pool(:,:,i) = ENL;
u = ENL(:,10);
U(i,1) = abs(u(13,1)*1e5+sum(A));
end
%% Check the Fitness and Sort %%
[FF,I] = sort(U);%% Sort and Rank
U_min = min(U)
A_Pool = A_Pool(:,I);%% Sort A_Pool according to rank [ Rank 1 is best ]
A_Pool(:,(NoP-2):NoP)=[];%% Reject the Weak Son
A = A_Pool(:,1);
plot(j,U_min,&#39;*&#39;);
hold on
end
pause(1);
hold off
close all
post_process(NL,EL,ENL,E,Node_flag,Element_flag,mag,A);
```