Journal of Aerospace Engineering & Technology
ISSN: 2231-038X (Online), ISSN: 2348-7887 (Print)
Volume 8, Issue 1
www.stmjournals.com

A Comparative Study on Evolutionary Algorithms to Perform Isogeometric Topology Optimization of Continuum Structures using Parallel Computing

N.S.S. Sahithi[1], K.N.V. Chandrasekhar[2,], T. Muralidhara Rao[3]*
[1–3]Department of Civil Engineering, CVR College of Engineering, Hyderabad, Telangana, India

Abstract
The optimization of computational performance along with the topology optimization of structures is the main focus of this study. The computational performance of the program is enhanced using two search engines in parallel. The search process is done using a least pop size which has reduced the number of computations required to optimize the objective function. The geometry can be exactly represented using isogeometric basis functions where the same basis is used to represent the geometry and calculate the response of the structure. Isogeometric analysis is used to conduct this study along with the metaheuristic swarm intelligence algorithms such as Firefly and Aqua search algorithms. The optimization is performed using evolutionary optimization process and metaheuristics have consistently been in use to optimize the distribution of material within the design domain. Few basic problems are optimized in this study and the results are compared. This study aims to make an attempt to reduce the computational effort and use newer ways to perform the topology optimization of continuum structures.

Keywords: *Parallel search engines, parallel computing, meta-heuristics, swarm intelligence, firefly, aqua search, isogeometric analysis, structural optimization, topology*

****Author for Correspondence*** E-mail: biml.koralla1@gmail.com

INTRODUCTION

The present study is focused on the application of parallel computing to perform isogeometric topology optimization of continuum structures using metaheuristic swarm intelligence algorithms. Isogeometric analysis was first introduced in the year 2005, who applied NURBS in the context of structural mechanics, where in the basic functions are complete with respect to the affine transformations which means that the rigid body motions and constant strain states are exactly represented [1]. Few numerical examples are solved to validate the new method of analysis of problems governed by partial differential equations. Adeli, in his paper on automated design of high rise buildings have used artificial intelligence based algorithms on a parallel machine [2]. The civil engineering structures as opposed to the other engineering models involve large number of degrees of freedom and are massive in nature. To automate such a design, newer computational models are created exploring new computing paradigms. A neural dynamics model for optimal design of structures by integrating the penalty method is used. Later, a non-linear neural dynamics model for optimization of large space structures is used. In another paper, Adeli reviews neural network articles on structural engineering, and construction engineering and management [3]. The articles on structural health monitoring, damage assessment, and structural control are reviewed. Articles written in other areas of civil engineering including environmental engineering, water resources engineering and transportation engineering are also reviewed. Sahithi used parallel search to optimize the arch truss structures using Evolutionary algorithms [4]. The weight of the structure is reduced when two engines were used instead of one engine. Aqua search swarm intelligence algorithm is used to conduct the present study [5]. Next part of the paper discusses the theoretical background to perform this study, the analysis and results, the conclusions and further study respectively.

Objectives of This Study

1. To use two search engines in parallel and perform the isogeometric topology optimization of continuum structures using swarm intelligence metaheuristic algorithms.
2. To minimize the weight of the structure, subject to the constraints of stress.

Scope of the Study

1. The study is limited to linear static elastic analysis only. The Hooke's law is valid.
2. The study does not include buckling analysis.

THEORETICAL BACKGROUND

This section provides the necessary formulation required to complete this study. This is discussed in the following sub-sections given below:

1. NURBS Formulation, and
2. Parallel Computing.

NURBS Formulation

Basis Functions *[6]*

$$N_{i,0}(\xi) = \begin{Bmatrix} 1 & if\ \xi_i \le \xi < \xi_{i+1} \\ 0 & otherwise \end{Bmatrix}$$

For p=1, 2, 3, …. They are defined by:

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi)$$

This is referred to as the Cox-de Boor recursion formula.

Derivatives of B-Spline Basis Functions

$$\frac{d}{dx} N_{i,p}(\xi) = \frac{p}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) - \frac{p}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi)$$

Generalize to Higher Order Derivatives *[5]*

$$\frac{d^k}{d^k\xi} N_{i,p}(\xi) = \frac{p}{\xi_{i+p} - \xi_i}\left(\frac{d^{k-1}}{d^{k-1}\xi} N_{i,p-1}(\xi)\right) - \frac{p}{\xi_{i+p+1} - \xi_{i+1}}\left(\frac{d^{k-1}}{d^{k-1}\xi} N_{i+1,p-1}(\xi)\right)$$

B-Spline Curves

$$C(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) B_i$$

B-Spline Surfaces:

$$S(\xi,\eta) = \sum_{i=1}^{n}\sum_{j=1}^{m} N_{i,p}(\xi) M_{j,q}(\eta)\, B_{i,j}$$

B-Spline Solids:

$$S(\xi,\eta,\zeta) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} N_{i,p}(\xi) M_{j,q}(\eta) L_{k,r}(\zeta)\, B_{i,j,k}$$

NURBS Basis Function

With a given projective B-spline curve and its associated projective control points in hand, the control points for the NURBS curve are obtained by using the following relations:

$$(B_i)_j = \frac{(B_i^w)_j}{w_i}\ j = 1,2 \ldots, d$$

$$w_i = (B_i^w)_{jd+1}$$

NURBS basis is given by:

For NURBS Curve:

$$R_i^p(\xi) = \frac{N_{i,p}(\xi) w_i}{\sum_{i=1}^{n} N_{i,p}(\xi) w_i}$$

$$C(\xi) = \sum_{i=1}^{n} R_i^p(\xi) B_i$$

This is identical to the B-Splines.

For NURBS Surfaces:

$$R_{i,j}^{p,q}(\xi,\eta) = \frac{N_{i,p}(\xi) M_{j,q}(\eta) w_{i,j}}{\sum_{i=1}^{n}\sum_{j=1}^{m} N_{i,p}(\xi) M_{j,q}(\eta) w_{i,j}}$$

For NURBS Solids:

$$R_{i,j,k}^{p,q,r}(\xi,\eta,\zeta) = \frac{N_{i,p}(\xi) M_{j,q}(\eta) L_{k,r}(\zeta) w_{i,j,k}}{\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} N_{i,p}(\xi) M_{j,q}(\eta) L_{k,r}(\zeta) w_{i,j,k}}$$

Derivatives of NURBS:

Apply the quotient rule,

$$\frac{d}{d\xi} R_i^p(\xi) = w_i \frac{W(\xi) N'_{i,p}(\xi) - W'(\xi) N_{i,p}(\xi)}{(W(\xi))^2}$$

$$where\ N'_{i,p}(\xi) = \frac{d}{d\xi} N_{i,p}(\xi)\ and\ W'(\xi) = \sum_{i=1}^{n} N'_{i,p}(\xi) w_i$$

For Higher Order Derivatives of NURBS Basis Functions [5]:

$$A_i^{(k)}(\xi) = w_i \frac{d^k}{d\xi^k} N_{i,p}(\xi)\,,(no\ sum\ on\ i)$$

We do not sum on the repeated index, and let,

$$W^{(k)}(\xi) = \frac{d^k}{d\xi^k} W(\xi)$$

Higher order derivatives can be expressed in terms of the lower order derivatives as:

$$\frac{d^k}{d\xi^k} R_i^p(\xi) = \frac{A_i^{(k)}(\xi) - \sum_{j=1}^{k} \binom{k}{j} W^{(j)}(\xi) \frac{d^{(k-j)}}{d\xi^{(k-j)}} R_i^p(\xi)}{W(\xi)}$$

$$where \binom{k}{j} = \frac{k!}{j!\,(k-j)!}$$

Parametric to Parent Mapping

$$\xi = \frac{1}{2}\left[(\xi_{i+1} - \xi_i)\hat{\xi} + (\xi_{i+1} - \xi_i)\right] \qquad \eta = \frac{1}{2}\left[(\eta_{i+1} - \eta_i)\hat{\eta} + (\eta_{i+1} - \eta_i)\right]$$

$$J_{\bar{\xi},\bar{\eta}} = \frac{1}{4}(\xi_{i+1} - \xi_i)(\eta_{i+1} - \eta_i)$$

Parametric Space to Physical Space [6]:

$X=N_1M_1X_1+N_2M_1X_2+N_2M_2X_3+N_1M_2X_4$ $Y=N_1M_1Y_1+N_2M_1Y_2+N_2M_2Y_3+N_1M_2Y_4$

$$\begin{bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \begin{bmatrix} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \end{bmatrix}$$

$$\frac{\partial x}{\partial \xi} = \frac{\partial}{\partial \xi}[N_1M_1N_2M_1N_2M_2N_1M_2][x_1x_2x_3x_4]^T$$

$$\frac{\partial x}{\partial \eta} = \frac{\partial}{\partial \eta}[N_1M_1N_2M_1N_2M_2N_1M_2][x_1x_2x_3x_4]^T$$

$$\frac{\partial y}{\partial \xi} = \frac{\partial}{\partial \xi}[N_1M_1N_2M_1N_2M_2N_1M_2][y_1y_2y_3y_4]^T$$

$$\frac{\partial y}{\partial \eta} = \frac{\partial}{\partial \eta}[N_1M_1N_2M_1N_2M_2N_1M_2][y_1y_2y_3y_4]^T$$

Strain Displacement Matrix

$$B = \begin{bmatrix} \frac{\partial N}{\partial x} & 0 \\ 0 & \frac{\partial N}{\partial y} \\ \frac{\partial N}{\partial y} & \frac{\partial N}{\partial x} \end{bmatrix} \qquad \epsilon = \begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{bmatrix}$$

$$\begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \end{bmatrix} = \frac{1}{|J|} \begin{bmatrix} J_{22} & -J_{12} \\ -J_{21} & J_{11} \end{bmatrix} \begin{bmatrix} \frac{\partial u}{\partial \xi} \\ \frac{\partial u}{\partial \eta} \end{bmatrix}$$

$$\epsilon = AG = \frac{1}{|J|} \begin{bmatrix} J_{22} & -J_{12} & 0 & 0 \\ 0 & 0 & -J_{21} & J_{11} \\ -J_{21} & J_{11} & J_{22} & -J_{12} \end{bmatrix} \begin{bmatrix} \frac{\partial u}{\partial \xi} \\ \frac{\partial u}{\partial \eta} \\ \frac{\partial v}{\partial \xi} \\ \frac{\partial v}{\partial \eta} \end{bmatrix}$$

For Element 1 [4]:

$$\begin{bmatrix} \frac{\partial u}{\partial \xi} \\ \frac{\partial u}{\partial \eta} \\ \frac{\partial v}{\partial \xi} \\ \frac{\partial v}{\partial \eta} \end{bmatrix}$$

$$= \begin{bmatrix} 4\eta - 2 & 0 & 2 - 4\eta & 0 4\eta & 0 & -4\eta & 0 \\ 4\xi - 2 & 0 & -4\xi & 0 4\xi & 0 & 2 - 4\xi & 0 \\ 0 & 4\eta - 2 & 0 & 2 - 4\eta 0 & 4\eta & 0 & -4\eta \\ 0 & 4\xi - 2 & 0 & -4\xi\ 0 & 4\xi & 0 & 2 - 4\xi \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \\ q_7 \\ q_8 \end{bmatrix}$$

Plane Stress:

$$D = \frac{E}{(1 - \upsilon^2)} \begin{bmatrix} 1 & \upsilon & 0 \\ \upsilon & 1 & 0 \\ 0 & 0 & (1 - \upsilon)/2 \end{bmatrix}$$

Algorithm to Perform the IGA Analysis

The algorithm to perform the isogeometric analysis of a two dimensional plate structure carrying in-plane loading:

1. Determine NURBS coordinates $(\xi_i, \xi_{i+}) * (\eta_j, \eta_{j+1})$ using elRangeU and elRangeV.
2. Store the connectivity of the element in an array names sctrB (of size nn).
3. Define strain displacement matrix B of size (1, 2*nn).
4. Set k_e=0.
5. Loop over Gauss points (GPs) $\{\xi_j', \omega_j'\}$ j=1, 2,…, n_{gp} where, n_{gp} is the number of gauss points.

a) Compute parametric coordinateξ corresponding to ξ_j'.

b) Compute $|J_{\xi'}|$ corresponding to the equations.

c) Compute the derivatives of the shape functions $R_{w\xi}^e$ and $R_{w\eta}^e$ at point ξ, η.

d) Compute J_ξ using control points (sctr(:,e)) $R_{w\xi}^e$ and $R_{w\eta}^e$.

e) Find J_ξ^{-1} and determinant $|J_\xi|$.
f) Compute the shape function derivatives $R_x = J_\xi^{-1}[R_{,\xi}^T R_{,\eta}^T]$.
g) Use Rx to build the strain displacement matrix B.
h) $k_e = k_e + B^T D B |J_{\xi'}||J_{\eta'}|\omega_j'$.

6. End loop on gauss points.
7. Assemble k_e into global stiffness matrix K^G.
8. End loop over all the elements.

The flowchart to develop the code in C++ is as shown below in Figure 1.

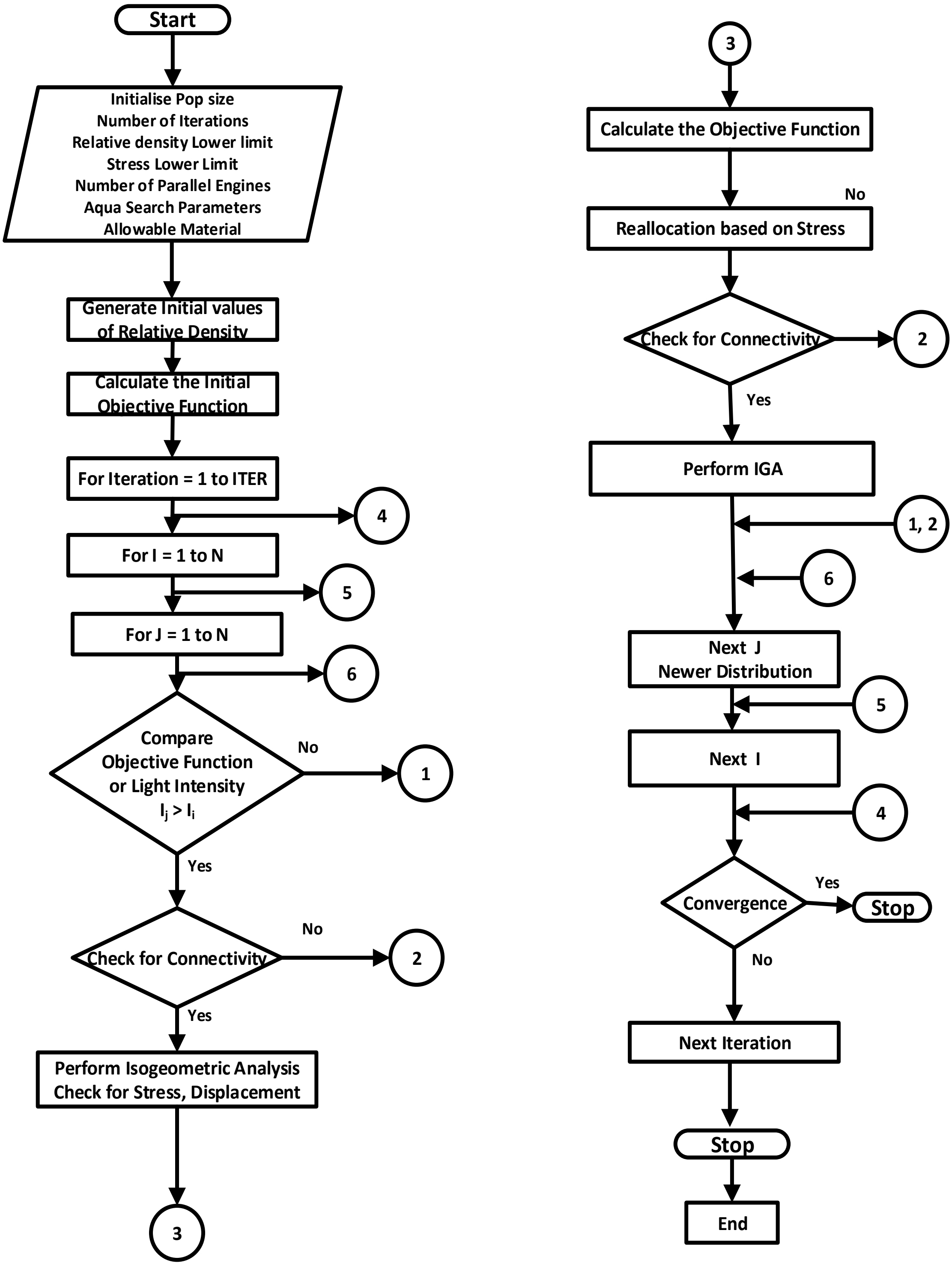


Fig. 1: *Showing the Flowchart to Use Parallel Search Engines and Perform Isogeometric Topology Optimization of Continuum Structures using Evolutionary Algorithms.*

ISSN: 2231-038X (Online), ISSN: 2348-7887 (Print)

Parallel Computing

The process of performing optimization can be improved using three ways [5]:

1. Newer theories which can effectively perform the process of optimization.
2. Better programs to efficiently use the hardware resources.
3. Multicore processor computers having higher hardware configuration.

"Optimization of large structures with thousands of members subjected to actual constraints of commonly used codes requires an inordinate amount of computer processing time and can be done only on multiprocessor super computers" [2]. This sentence clearly emphasizes the need for large computational power to perform the optimization of structures. High degree of parallel computing can be exploited using metaheuristic algorithms. The nature inspired swarm intelligence evolutionary algorithms such as Aqua Search, Firefly and so on have enormous potential to be used on multicore processor computers such as Dell T7910®, and Fujitsu Celsius R940®. Running a parallel computing program using evolutionary algorithms to optimize the need of computational power is a demanding task. The current analysis is done on Intel i7-6700, 3.5 GHz processor computer. The design variable is the relative density of each element. The present study uses two search engines in parallel to find the optimal distribution of material within the given design domain.

ANALYSIS

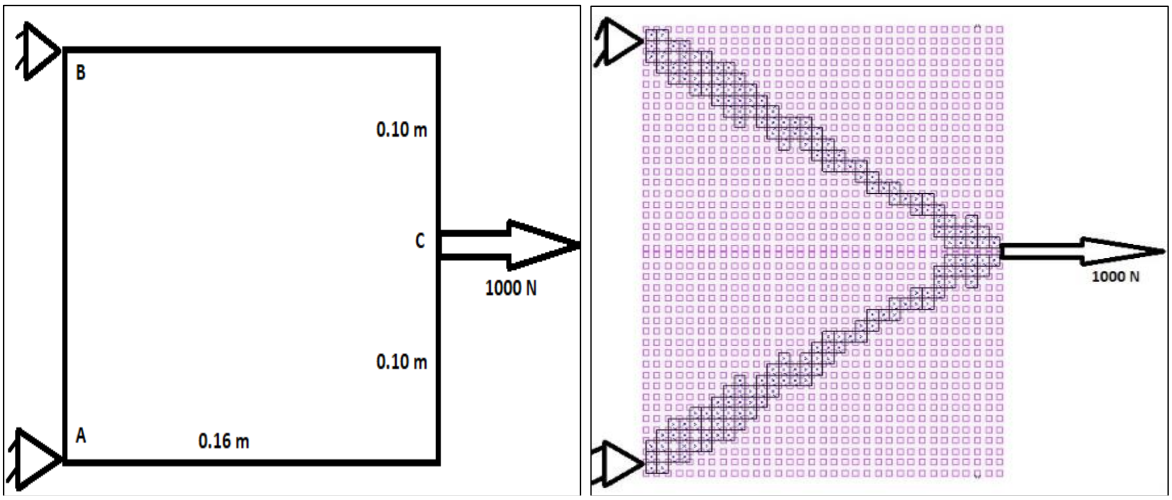
A cantilever plate 0.16 m×0.10 m as shown in the Figure 2 carries a point load of 1000 N acts at the centre of the right edge along X-direction. The entire design domain is discretized into 693 nodes and 640 elements first order four noded quadrilateral elements in plane stress condition. The Young's modulus of elasticity is taken as 200 GPa and the Poisson's ratio is taken as 0.33. The weight density of the material is 78700 N/m^3. The thickness of the plate is 0.010 m. The NURBS basis functions of first order are used here. The size of the Xi vector is equal to 37, and the size of the Eta vector is taken as 25. For the sake of symmetry, only half of the domain is analyzed.

Xi Vector

{0	0	0	0.031250.0625
	0.093750.125		0.156250.1875
	0.218750.25		0.281250.3125
	0.343750.375		0.406250.4375
	0.468750.5		0.531250.5625
	0.593750.625		0.656250.6875
	0.718750.75		0.781250.8125
	0.843750.875		0.906250.9375
	0.968751		1 1}

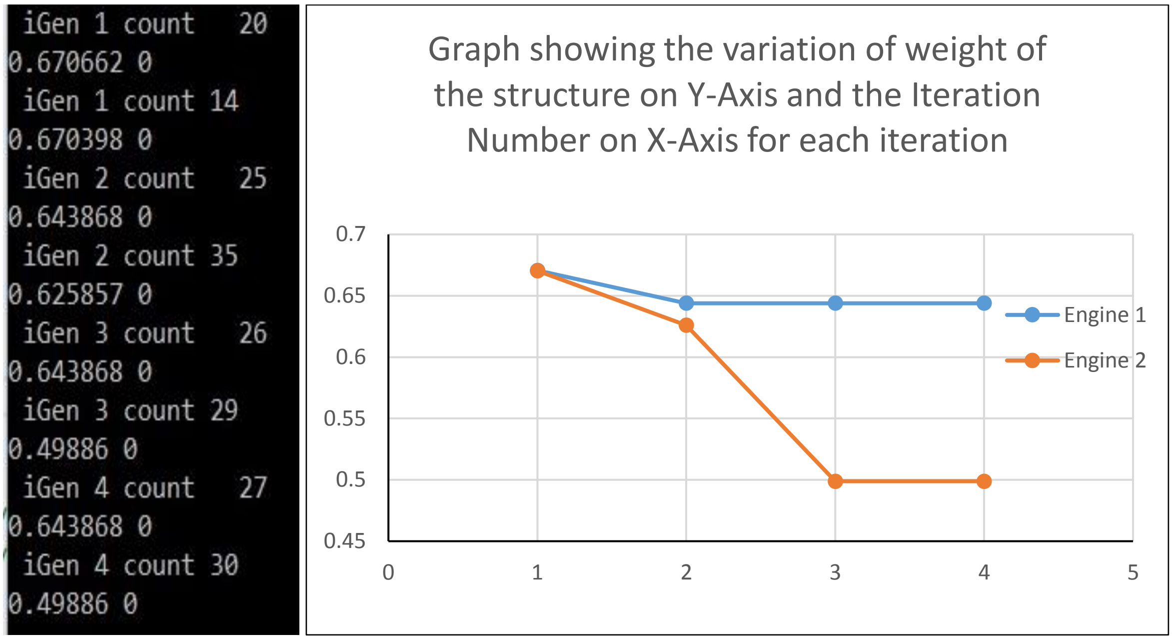
Eta Vector

{0	0	0	0.05	0.10	0.15
	0.20	0.25	0.30	0.35	0.40
	0.45	0.50	0.55	0.60	0.65
	0.70	0.75	0.80	0.85	0.90
	0.95	1	1	1}	



(a) Design Domain, *(b) Flip Vertical Image of Optimal Distribution*

Fig. 2: *Showing the Initial Design Domain and the Final Distribution of Material after the Optimization.*



(a) Parallel Engines Performing the Computation *(b) Iteration Curve*

Fig. 3: ***(a)*** *Showing the Weight for Two Search Engines,* ***(b)*** *Iteration Curve for Two Engines.*

The iteration-wise minimum weight is as shown in the Figure 3. The topology optimization process is performed using Aqua Search algorithm. The optimal distribution can be reached with fewer computations using two search engines in parallel.

Cantilever Carrying Point Load Y at the Centre

A cantilever plate 120 mm×120 mm which is fixed at left end and carries a load of 1000 N at the centre of the right edge. The design domain is meshed using 625 nodes and 576 elements first order four noded quadrilateral elements in plane stress condition. The Young's modulus of elasticity is taken as 200 kN/mm^2 and the Poisson's ratio is taken as 0.33. The density of the material is taken as 78700×10^{-9} N/mm^3. The NURBS basis functions of first order are used here. The size of the Xi Vector and Eta Vector each is equal to 29. The thickness of the plate is taken as 10 mm.

Xi Vector ={ 0 0 0
0.041666667 0.083333333 0.125
0.166666667 0.208333333 0.25
0.291666667 0.333333333 0.375
0.416666667 0.458333333 0.5
0.541666667 0.583333333 0.625
0.666666667 0.708333333 0.75
0.791666667 0.833333333 0.875
0.916666667 0.958333333 1
1 1}

Eta Vector ={ 0 0 0
0.041666667 0.083333333 0.125
0.166666667 0.208333333 0.25
0.291666667 0.333333333 0.375
0.416666667 0.458333333 0.5
0.541666667 0.583333333 0.625
0.666666667 0.708333333 0.75
0.791666667 0.833333333 0.875
0.916666667 0.958333333 1
1 1}

Figure 4 shows the distribution of material using Firefly algorithm. The distribution of the material is analyzed using a commercial standalone package Marc ®. The analysis clearly shows that the structure is safe in stress and displacement. Figure 5 shows the iteration wise weight of the structure using both firefly algorithm and the Aqua search algorithm. The Figure 6 clearly shows that the Aqua Search algorithm can perform the optimization process effectively and efficiently to reduce the weight of the structure. The search process of intensification and diversification has been effectively performed to locate the optimal.

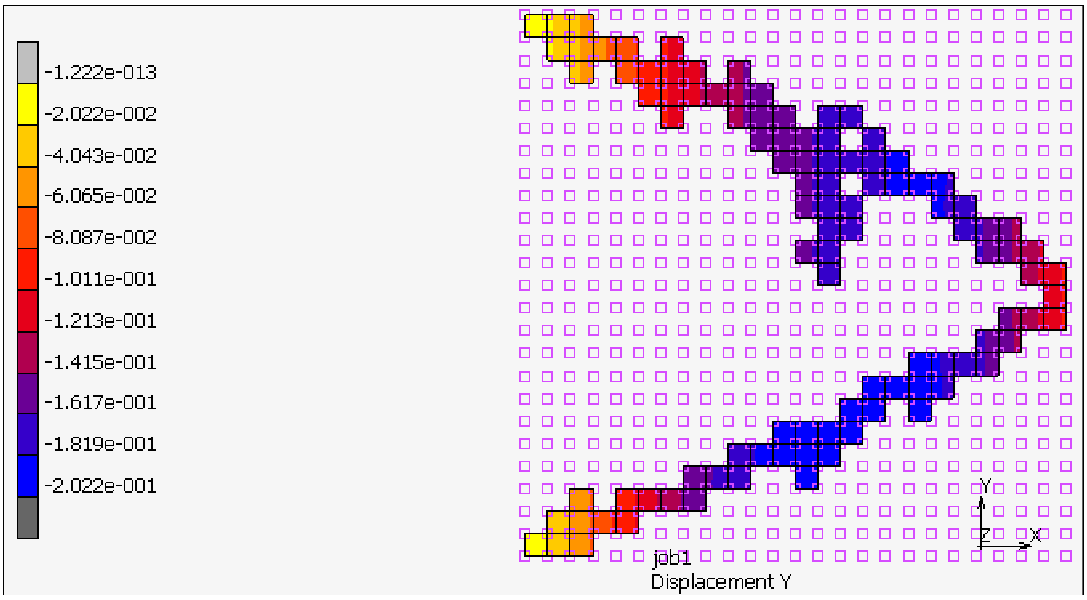
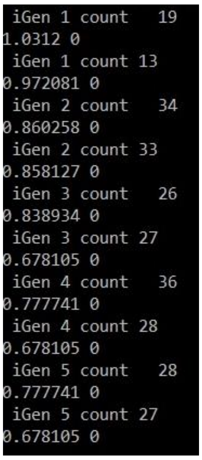
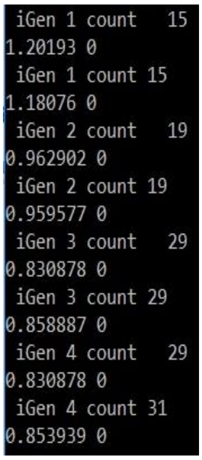


Fig. 4: *Optimal Distribution in Marc ® using FFA and Parallel Computing.*





(a) Minimum Weight Using AS (b) Minimum Weight Using the FFA
Fig. 5: *The Iteration-wise Minimum Weight Using AS and FFA.*

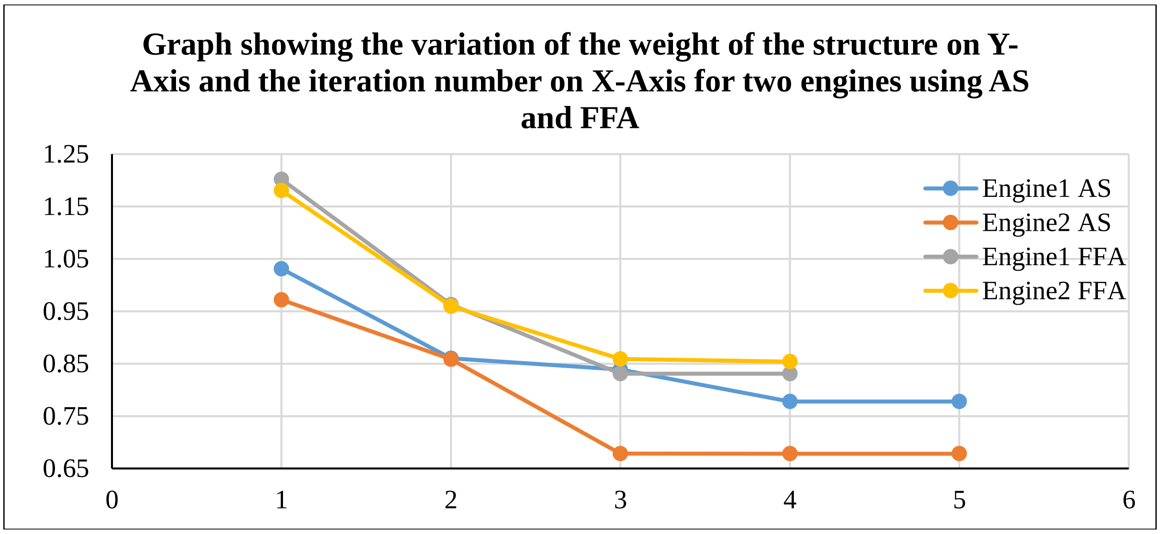


Fig. 6: *Showing the Iteration Curve Showing the Variation of Weight of the Structure vs. Iteration Number for Two Engines Using AS and FFA.*

CONCLUSIONS

The aim of this study is to perform and compare performance of two swarm intelligence algorithms namely Firefly algorithm and Aqua search algorithm using the concept of parallel computing by employing two engines in parallel. The analysis has been performed using isogeometric NURBS basis functions. The process of optimization of continuum structures has been performed using the flowchart as shown in Figure 1. The standard problems from the literature have been solved using IGA. The results have been compared and presented in the graphs as shown in the analysis section. The results show that the Aqua search algorithm performs better and is able to identify the optimal distribution of material in the design domain of the structure using two engines in parallel.

Further Study

The future study can be extended to optimize the bracing systems of the buildings, and optimal reinforcement distribution in the reinforced concrete structures. The study can be applied to fracture mechanics as well.

REFERENCES

1. Sahithi, Chandrasekhar KNV. Parallel Search Engines for Topology Optimization of an Arch Truss Using Evolutionary Algorithms. *RTCET*. Aug 2017; 7(2).
2. Adeli H, Park HS. Fully Automated Design of Super-High Rise Building Structures by a Hybrid AI Model on a Massively Parallel Machine. *AI Magazine*. 1996; 17(3): 87–93p.
3. Adeli H. Neural Networks in Civil Engineering: 1989-2000. *Comput-Aided Civ Infr Eng*. 2001; 16(2): 126–142p.
4. Chandrasekhar KNV. Multi-Population Search Engines for Topology Optimisation of Continuum Structures using Aqua Search Algorithm. *RTCET*. May 2017.
5. Chandrasekhar KNV, *et al*. Aqua Search: A New Metaheuristic Modified Firefly Algorithm for Topology Optimisation of Continuum Structures. *Journal of Offshore Structure and Technology (JoOST).* 2016; 3(3): 32–37p.
6. Hughes TJR, Cottrell JA, Bazilevs Y. Isogeometric Analysis: CAD, Finite Elements, NURBS, Exact Geometry and Mesh Refinement. *Comput Methods Appl Mech Eng*. 2005; 194: 4135–4195p.

Cite this Article

Sahithi NSS, Chandrasekhar KNV, Muralidhara Rao T. A Comparative Study on Evolutionary Algorithms to Perform Isogeometric Topology Optimization of Continuum Structures using Parallel Computing. *Journal of Aerospace Engineering & Technology*. 2018; 8(1): 51–58p.