



TALLINN UNIVERSITY OF TECHNOLOGY
SCHOOL OF ENGINEERING
DEPARTMENT OF MECHANICAL AND INDUSTRIAL ENGINEERING

NUMERICAL ALGORITHMS FOR ORIENTATIONAL DESIGN OF 3D ORTHOTROPIC MATERIALS

NUMBRILISED ALGORITHMID 3D ORTOTROOPSE MATERJALI OPTIMAALSE ORIENTATSIOONI MÄÄRAMISEKS

MASTER THESIS

Student: Kallol Kumar Samaddar

Student code: 194288MARM

Supervisor: Jüri Majak, Professor

Tallinn 2021

AUTHOR'S DECLARATION

Hereby I declare, that I have written this thesis independently.

No academic degree has been applied for based on this material. All works, major viewpoints and data of the other authors used in this thesis have been referenced.

"24" May 2021

Author: Signed Digitally

/signature /

Thesis is in accordance with terms and requirements

"24" May 2021

Supervisor: Signed Digitally

/signature/

Accepted for defence

"....."20... .

Chairman of theses defence commission:

/name and signature/

Non-exclusive Licence for Publication and Reproduction of Graduation Thesis¹

I, Kallol Kumar Samaddar (date of birth: 14.10.1993) hereby

1. grant Tallinn University of Technology (TalTech) a non-exclusive license for my thesis

NUMERICAL ALGORITHMS FOR ORIENTATIONAL DESIGN OF 3D ORTHOTROPIC MATERIALS

NUMBRILISED ALGORITHMID 3D ORTOTROOPSE MATERJALI OPTIMAALSE ORIENTATSIOONI MÄÄRAMISEKS,

supervised by Prof. Jüri Majak,

1.1 reproduced for the purposes of preservation and electronic publication, incl. to be entered in the digital collection of TalTech library until expiry of the term of copyright;

1.2 published via the web of TalTech, incl. to be entered in the digital collection of TalTech library until expiry of the term of copyright.

1.3 I am aware that the author also retains the rights specified in clause 1 of this license.

2. I confirm that granting the non-exclusive license does not infringe third persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

¹ *Non-exclusive Licence for Publication and Reproduction of Graduation Thesis is not valid during the validity period of restriction on access, except the university's right to reproduce the thesis only for preservation purposes.*

Signed Digitally (*signature*)

"24" May 2021 (*date*)

Department of Mechanical and Industrial Engineering

THESIS TASK

Student: Kallol Kumar Samaddar (194288MARM)

Study programme, Industrial Engineering and Management (MARM)

main speciality:

Supervisor(s): Prof. Jüri Majak, +372 620 3265

Consultants:(name, position)

..... (company, phone, e-mail)

Thesis topic:

(in English) NUMERICAL ALGORITHMS FOR ORIENTATIONAL DESIGN OF 3D ORTHOTROPIC MATERIALS

(in Estonian) NUMBRILISED ALGORITMID 3D ORTOTROOPSE MATERJALI OPTIMAALSE ORIENTATSIOONI MÄÄRAMISEKS

Thesis main objectives:

- | |
|---|
| 1. Analyze and compare different optimization methods and algorithms for solving 3D optimal material orientation problems of linear elastic materials |
| 2. Determine the most computationally cost-effective algorithms for particular problems or classes of problems |

Thesis tasks and time schedule:

No	Task description	Deadline
1.	Topic proposal and structure of the thesis	25.01.2021
2.	Preliminary background studies and concept generation	23.02.2021
3.	Designing for the proposed solutions	18.03.2021
4.	Result analysis and conclusion	15.05.2021

Language: English **Deadline for submission of thesis:** "26" May 2020

Student: Kallol Kumar Samaddar Signed Digitally "24" May 2021
/signature/

Supervisor: Prof. Jüri Majak Signed Digitally "24" May 2021
/signature/

Consultant: ".....".....2021
/signature/

Head of study programme: ".....".....2021
/signature/

CONTENTS

LIST OF FIGURES	6
LIST OF TABLES	7
PREFACE	8
LIST OF ABBREVIATIONS AND SYMBOLS	9
1. INTRODUCTION.....	10
2. BACKGROUND STUDIES	13
2.1 Optimal design problem	13
2.2 Orientation problem in the composite structure	16
2.3 Optimization techniques in orientation problem	19
3. METHODS AND TECHNIQUES USED.....	22
3.1 Gradient method.....	22
3.2 Genetic algorithm	24
3.3 Particle swarm optimization	26
3.4 Lagrange multipliers method	29
4. OPTIMAL MATERIAL ORIENTATION OF 3D ORTHOTROPIC MATERIALS	30
4.1 Problem formulation.....	30
4.2 Selection of optimization methods and techniques	32
4.3 Necessary optimality conditions	33
4.4 Constrained optimization	33
4.4.1 Problem formulation	33
4.4.2 Problem solution using GA	34
4.4.3 Problem solution using Lagrange multipliers method.....	35
5. NUMERICAL RESULTS AND ANALYSIS.....	39
5.1 E-Glass/Vinylester.....	39
5.2 Graphite/Epoxy	46
SUMMARY.....	53
KOKKUVÕTE	54
LIST OF REFERENCES	55
APPENDICES	63

LIST OF FIGURES

Figure 2.1 Design of optimization process	14
Figure 2.2 Principal material coordinates of a typical lamina	17
Figure 2.3 State of stress at a point of a continuum	17
Figure 2.4 A scheme of a composite plate under in-plane stress	18
Figure 3.1 Steepest descent method, in the 1D case	23
Figure 3.2 Steepest descent method, in the 2D case	23
Figure 3.3 Genetic algorithm program flowchart	24
Figure 3.4 Roulette wheel selection	25
Figure 3.5 Double point crossover operator	26
Figure 3.6 The illustration velocity update of a particle in PSO	27
Figure 3.7 Flow diagram illustrating the particle swarm optimization algorithm.....	28
Figure 5.1 E-Glass/Vinilester, distribution of the strain energy as function of Euler angles θ_2 and θ_3 (θ_1 is constant)	42
Figure 5.2 E-Glass/Vinilester, distribution of the strain energy as function of Euler angles θ_3 and θ_1 (θ_2 is constant)	43
Figure 5.3 E-Glass/Vinilester, distribution of the strain energy as function of Euler angles θ_1 and θ_2 (θ_3 is constant)	44
Figure 5.4 Graphite/Epoxy, distribution of the strain energy as function of Euler angles θ_2 and θ_3 (θ_1 is constant)	49
Figure 5.5 Graphite/Epoxy, distribution of the strain energy as function of Euler angles θ_3 and θ_1 (θ_2 is constant)	50
Figure 5.6 Graphite/Epoxy, distribution of the strain energy as function of Euler angles θ_1 and θ_2 (θ_3 is constant)	51

LIST OF TABLES

Table 5.1 E-Glass/Vinilester, solutions using GA+Gradient, GA, PSO, and integer variables GA.....	40
Table 5.2 E-Glass/Vinilester, four equivalent optimal solutions.	41
Table 5.3 E-Glass/Vinylester, computing time	45
Table 5.4 Graphite/Epoxy, solutions using GA+Gradient, GA, PSO, and integer variables GA.....	47
Table 5.5 Graphite/Epoxy, four equivalent optimal solutions.	48
Table 5.6 Graphite/Epoxy, computing time	52

PREFACE

The topic of this thesis was offered by the Department of Mechanical and Industrial Engineering of Tallinn University of Technology.

I would like to express my profound gratitude to my supervisor, Professor Jüri Majak, Department of Mechanical and Industrial Engineering of Tallinn University of Technology, for his excellent guidance and support during this whole period. Under his supervision and collaboration, the research practice was coordinated and organized successfully.

At the same time, I am genuinely thankful to Professor Kristo Karjust, Head of the Department of Mechanical and Industrial Engineering of Tallinn University of Technology, for his valuable critique and direction to enhance the writing and presentation of the thesis.

In this research paper, the optimal material orientation problem is studied, considering the strain energy density as the objective function. The Euler angles corresponding to the global minimum of the strain energy density have been determined. Different evolutionary and gradient-based optimization methods are utilized, and the obtained results are compared. Finally, the best technique is determined based on accuracy and computational cost.

Keywords: Optimal material orientation, 3D orthotropic materials, Strain energy, Euler angles, and Global extremes

LIST OF ABBREVIATIONS AND SYMBOLS

GA: genetic algorithm.

GPa: gigapascal

HGA: hybrid genetic algorithm.

PSO: particle swarm optimization

1. INTRODUCTION

In recent decades, optimization of the material structure has been a significant research course for development in terms of time and expenses [10,65]. For the composite materials, structural applications have been expanding and generating the need to advance analysis and design techniques to keep up with the pace as the area evolving. Distinctive characteristics, such as high strength and stiffness, ultra-light in weight, and others, can be designed freely to suit the particular application in aircraft, automobiles, and infrastructure [34]. Still, considering composite material's design and mechanical properties seems complicated and arduous for structural design compared to the conventional material structure [27]. Moreover, attaining the maximum and minimum stiffness of materials while designing composite structures remains a practical challenge requiring continuous testing and experimentation to address various phases. On the other hand, the size, shape, topology, thermal effects, and structure of advanced composite materials for optimal stiffness/flexibility need to be determined [25, 26].

For academics worldwide, the optimal orientation problem of orthotropic materials has been a substantial area of research interest and potential to the future direction. Sahadevan et al. [63] explored the composite wing structure of laminate material design and offered the meta-heuristic weight reduction optimization solution. Banichuk proposed elastic energy density as a measurement criterion of the stress-strain state by an energy-based formulation [4] for resolving the optimal material orientation problem. Later many others also applied the approach in their research and analysis. In another study, Pedersen [57] suggested a strain-based method considering the strain field constant about the deviations of the fiber orientation variable. For the determination of oriental layout problems, divergent nonlinear elastic material models are introduced in [40,41,58], and effective strain (stress) has been considered as a scalar measure of the strain (stress) state. In [12,61,62,64], the authors denoted the stress, and strain tensors are coaxial at the optimum while analyzing orientational design problems for 3D orthotropic materials. Moreover, to illustrate periodic composites with the auxetic and isotropic response [39], topology optimization is used. However, designing an isotropic material system facilitating adjustable auxetic behavior for the non-complicated manufacturing process requires further analysis.

Previous optimization studies of composite structures pursued methods based on gradients of the objective and constraints functions concerning the design variables [2]. In [21], the optimization methods have been discussed to optimize composites laminates with constant stacking sequence through the intact structure. Various

optimization techniques involving gradient-based and direct search methods, specialized algorithms are developed, and the attributes for composite lay-up design are compared. These research results have a downside due to the constrained to limited values processed during the manufacturing and causes the design to fall over on a distinct optimization conundrum. Hence, as an alternative to the gradient-based methods, other optimization techniques have been tested. Nevertheless, the implementation of a genetic algorithm (GA) has exhibited superiority over others because of the adjustability nature of the composite optimization problems. To avoid early convergence and concentrate on the process's local optimal region, the GA applies numerous search points in the design space.

Ample experimental research has been conducted on resolving the orientation problem of the composite structure, and different approaches are introduced. However, a simplified algorithm that provides results faster and precisely deserves further assessment. In [50], Majak and Pohlak introduced the decomposition method to fill this gap for the strain energy density function. However, to better understand the optimization practices used in optimal material orientations, comparing the derived results and determining the necessary criteria has been considered necessary.

Therefore, the main goal of the current study is to analyze and compare different optimization methods/algorithms for solving 3D optimal material orientation problems of linear elastic materials to determine the most computationally cost-effective algorithms for particular problems or classes of problems.

In this paper, the optimal material orientation problem developed in [50] considering the strain energy density as a measure of the structure's stiffness has been revisited. Two numerical methodologies – constrained and unconstrained (real and integer variables)- are designed and evaluated in the study. Numerical algorithms based on local (gradient techniques) and global optimization techniques (GA, PSO, and hybrid GA) are applied as an alternative solution. Finally, the results are compared to discover the efficient algorithm in a similar design optimization problem. A MATLAB optimization toolbox is utilized to find the optimal numerical solution.

The paper is organized as follows. Chapter 2 presents the general background/literature overview of the composite structure, and later Chapter 3 illustrates different numerical optimization techniques applied in previous studies. Section 4 formulates the problem and necessary criteria for optimal material orientation of 3D orthotropic material. Problem formulation and analysis for constrained optimization have been overviewed.

Section 5 presents the test of the different optimization techniques for optimal material orientation of 3D orthotropic material, numerical analysis, and discusses the results. Finally, the summary summarises the results and findings of the study.

2. BACKGROUND STUDIES

The following chapter presents an overview of previous findings on the optimization problem of composite material and techniques to address. The primary purpose is to explore the topics that have been analyzed associated with the proposed study.

2.1 Optimal design problem

The problem formulation of composite material concerning design optimization generally focuses on attaining the optimal material design for the most superior performance in definite constraints environment. The minimization of the desired objective function has been the primary goal of the optimization process. The objective functions have been tailored, encompassing mass, volume, bending stiffness, buckling loads, natural frequency, and maximum deflection by choosing the individual integral materials, volume fragments, fiber orientation angles, and laminas measurement and quantity due to the manufacturing strategy [45, 47, 52]. In [27], for a particular orthotropic material maximum, principal material stresses before incurring failure have been studied, and the objective function is to determine the maximum stress failure criterion.

In engineering applications, fabrication cost, maximum product reliability, maximum stiffness/weight ratio, minimum aerodynamic drag, maximum natural frequencies, maximum critical shaft speeds, and so-forth. have been distinguished as noteworthy targets, whereas alignment, dimensions, and sizes of components, and material attributes as design variables [19,47,68]. For the design of structural components, key design variables are also stipulated in various studies. For instance, the covering of skin panels' thickness and space, size, and form of the transverse and longitudinal stiffeners are considered design variables in [47] for a similar function area. In the automobile design structure, the design variables are also defined independently. Apart from the size and thickness of covering skin panels, the size and shape of the transverse and longitudinal stiffeners are examined here [47]. The orientation of the fibers and the ratio can also be studied as additional variables for stiffeners composed of layered composites.

Figure 2.1 illustrates a general form of an approach to design and devise optimization problems with single and multicriteria objectives.

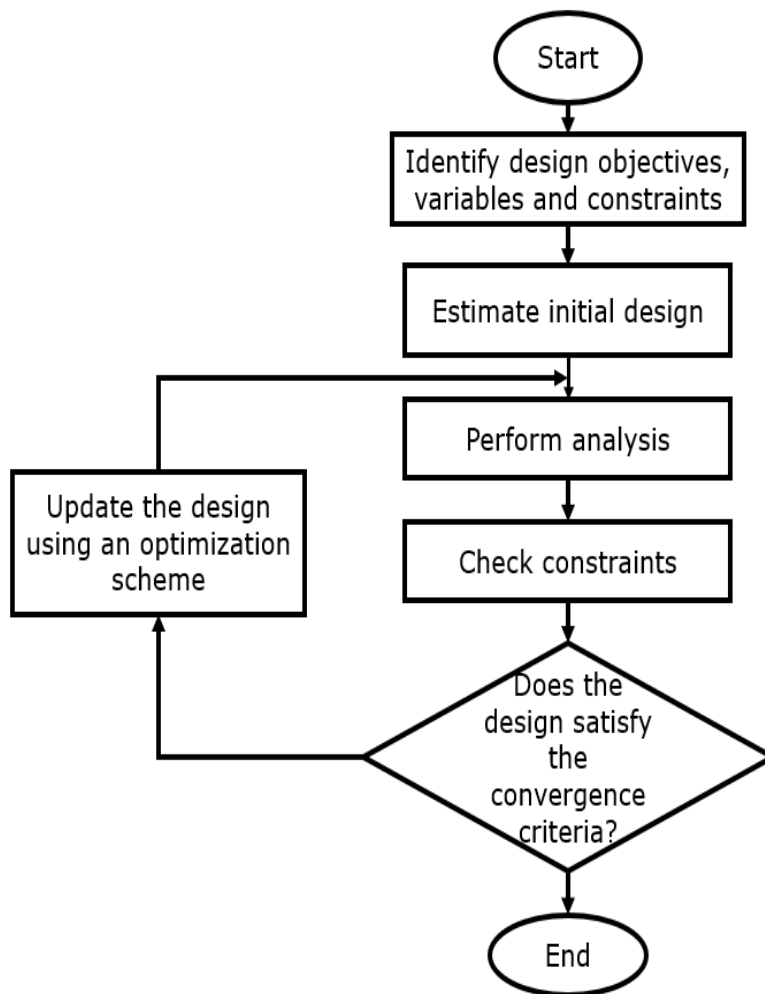


Figure 2.1 Design of optimization process [47]

The design optimization process primarily investigates the chosen design variables (X_n) to achieve the best results within the distinct boundary conditions ($g_j(X)$). Thus, considering specific environmental restrictions such as system performance, admissible stresses, geometrical properties, and other factors, the goal of optimality is determined as a vector of objective functions ($F_k(X)$). As mentioned above the Figure 2.1 represents a general understanding of the optimization procedures. However, there is additional consideration necessary to address while analyzing discrete problems or classes of problems. Therefore, several additional detailed suggestions can be outlined based on the literature analysis [X_1 , X_2].

1. Firstly, in most optimization problems, the ranges for design variables, i.e., design domain, are needed to specify. The limitations on the design domain can also be applied to determine local/global extremes in subdomains.
2. Secondly, in the case of evolutionary methods, the second block, "Estimate (in fact specify) initial design," is commonly "automated." Thus, the initial designs/solutions are generated randomly.
3. Thirdly, the update design block is mainly generic. It may include selecting optimization strategies like weighted summation, Pareto concept, application of various optimization techniques like gradient methods, evolutionary methods, hybrid methods, structural analysis for determining the strength and stiffness characteristics, response modeling of the objective and constraint functions, along with others.

The selection of suitable optimization techniques can be designed based on pair-wise analysis of the optimization criteria.

1. If the two optimality criteria appear not controversial, these criteria can be combined into one criterion by applying some weighted summation, etc., approaches.
2. If the two optimality criteria appear controversial, the Pareto approach should be applied.

The multicriteria optimization problem can be formulated as

Minimize functions,

$$F(x) = [F_1(x), F_2(x), \dots, F_r(x)] \quad (1)$$

Subjected to-

$$g_j(x) \leq 0, \quad j = 1, \dots, k, \quad (\text{inequality constrains}) \quad (2)$$

$$h_k(x) = 0 \quad k = 1, \dots, l, \quad (\text{equality constraints}) \quad (3)$$

$$x_{i*} \leq x_i \leq x_i^*, \quad i = 1, \dots, m, \quad (4)$$

where x_{i*} and x_i^* stand for the lower and upper limits of the design variables x_i .

If some of the original objective functions are subjected to maximization or the values of the objective functions are not in the same range, the following normalization can be performed.

$$f_i = \frac{F_i^* - F_i(x)}{F_i^* - F_{i*}} \quad , \quad (5)$$

$$f_i = \frac{F_i(x) - F_{i*}}{F_i^* - F_{i*}} \quad . \quad (6)$$

The formulas (5) can be applied when the original objective function $F_i(x)$ is subjected to maximization and formula (6) for depreciation. The F_i^* and F_{i*} in (5)-(6) denote the upper and lower estimates of the objective function.

2.2 Orientation problem in the composite structure

Material scientists usually customize laminated composite by experimenting with the laminae's thickness, number, and orientation to develop material with excellent mechanical properties. However, the manufacturing and experimental data limitations show that a small set of values is available for the ply thickness and orientation angles [38]. Hence, discovering optimum strength designs for fiber-reinforced composite laminates has been a challenge for academics. The relation between fiber orientation and unidirectional lamina strength has been studied in [5], and the result revealed that it could be maximized under in-plane stresses. The findings of another research [74] showed that optimal fiber orientation contingent on the relative value of transverse and in-plane shear strengths and the stress state of laminated material. The optimization method can be intricate and specific while analyzing for maximum strength of a multidirectional composite laminate.

In various studies, the stresses in each lamina in the principal material coordinates are primarily considered to analyze the failure of laminated composite materials [34]. Figure 2.2 below is a generalized schema that has been drawn to illustrate the principal material coordinates and fiber direction in composite laminates. The maximum stress theory explains that the failure can be projected after maximum stress in the principal material coordinates exceeds the respective strength.

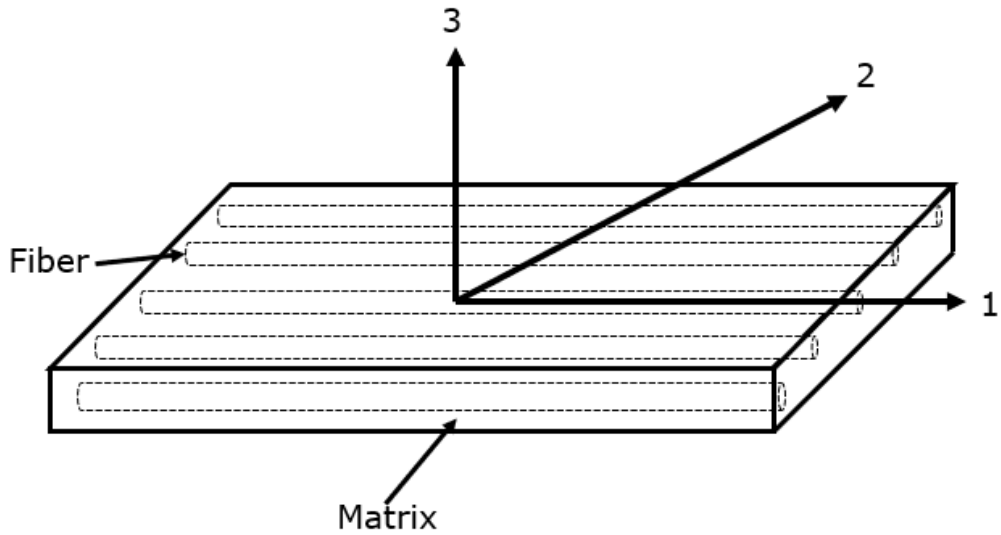


Figure 2.2 Principal material coordinates of a typical lamina [43]

For analyzing a laminate, composite plate nine stress components σ_{ij} ($i,j=1,2,3$) can be used to articulate the stress state at a point in a general continuum. As illustrated in Figure 2.3, the elements can operate parallelly on an elemental cube to the axes, referring to a coordinate system.

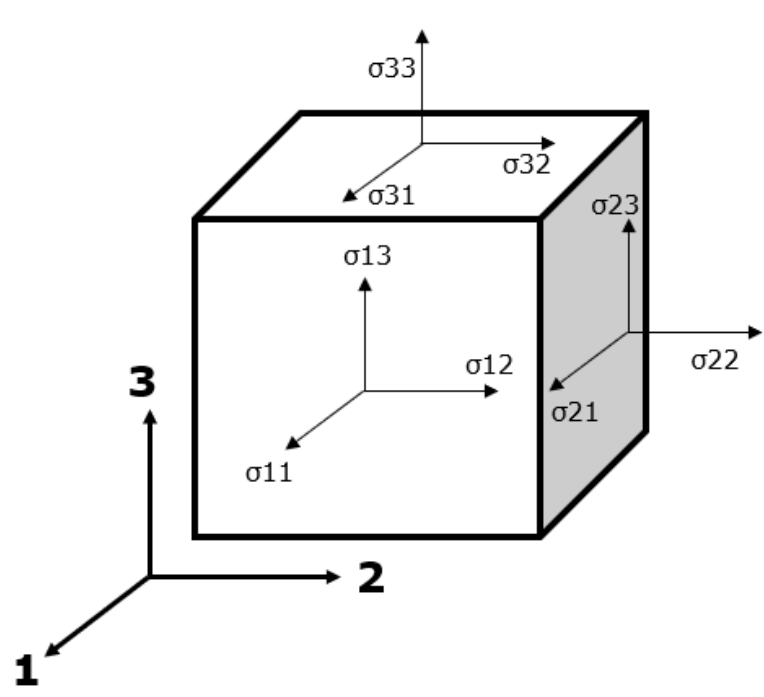


Figure 2.3 State of stress at a point of a continuum [13]

In general, the classical laminate theory is applied for resolving in-plane stress problems. In this case, plane stress components are assumed to be zero. As depicted

in Figure 2.4, in-plane stress components correlate to the strain components regarding the coordinate system [1].

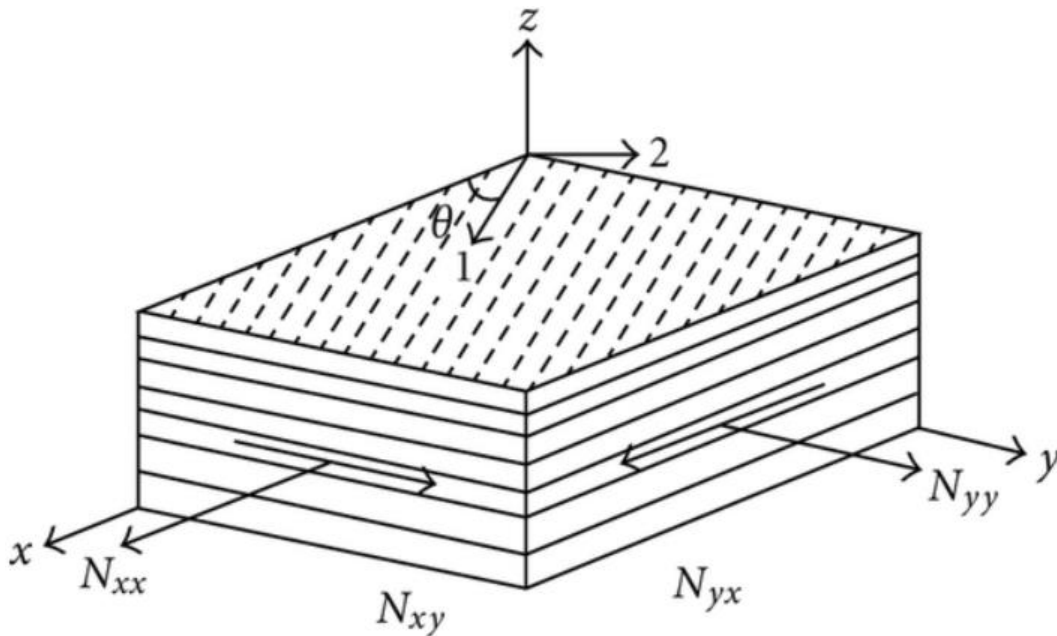


Figure 2.4 A scheme of a composite plate under in-plane stress [1]

The orientation of the fibers has a significant impact on the material properties of composite materials. Therefore, the fiber-reinforced composite structure has a continuous research goal to enhance the properties concentrating on multiple directions and woven fibers at different angles. Furthermore, achieving material behavior similar to isotropic materials is another desirable goal. Still, in some cases, anisotropic material properties are likely to be more compatible [60]. Thus, the direction of fibers is preferred to be composed in the same direction. The material's microstructure embodies the favorable and unfavorable factors of the fiber and the matrix material when required.

The fiber orientation is used as the design parameters in [30], and variables like a fixed set of values, the fiber matrix ratio, and the stacking configuration of the associated lamina for a microstructure are considered. The study offers a solution for the optimum fiber distribution of microstructure orientation. Suzuki and Kikuchi (1991) developed a direct approach to the design problem [69]. The finding reveals that when orientations of fibers are in the same direction, the principal stress results in the best stiffness regarding a microstructure. In another research, Gea and Luo (2004) discovered that the method provides acceptable results limited to the orthotropic materials with limited and a few intense shear stress [20].

In [8], multiple loaded cases have been considered for computing microstructure orientation angles, and in [14], eigenvector problems are studied. Bruyneel and Fleury (2002) and Lindgaard and Lund (2011) used the Continuous Fibre Angle Optimization (CFAO) method to express the optimal orientation of the fibers in the microstructure of the material, and the orientation angle was studied as the design variable [6,42]. Conversely, the optimization problem associated with numerical treatment confronts various challenges when using continuous design angle. Nomura et al. (2014) introduced an orientation optimization method, a three-dimensional formulation, for any desired material property.

For 3D orthotropic materials, orientational design problems have been studied in [12, 61, 62, 64] and revealed that the potential energy of deformation and the specific elastic energy density are subjected to depreciation. Seregin and Troitski (1982) derived the optimality conditions for general orthotropic material and discussed the solution modes [64]. Euler angles have been used to formulate the optimization problem in [61] and derive the symmetric properties of principal orientations of stress and strain at the optimum from the stationary condition of the strain energy density. In addition, the study offers an analytical solution for a body with cubic symmetry regarding strains.

In [23,28,44,70], researchers have described the mechanical response of transversely isotropic materials by proposing fundamental models. Specifically, a model was introduced based on hyperelastic or viscoelastic theories to incorporate transverse anisotropy under finite deformation [32]. For biological materials, an anisotropic hyper-elastic constitutive model has been formed in [59]. Moreover, describing compressible soft tissues, an isotropic visco-hyperelastic model is proposed in [75]. However, as the mentioned models are based on elasticity, they have limited application in large inelastic deformation. Thus, to address the shortcoming, a transversely isotropic model for porous materials based on a neo-Hookean strain energy function has been proposed [23, 24]. However, the parameter values that correspond with experiments differ with the direction of loading.

2.3 Optimization techniques in orientation problem

An optimization problem to explore in an n-dimensional space where the minimum value of the overall objective function operates within the boundary conditions representing the constraint functions and resolving iterative techniques is usually applied [47]. In the design space, a succession of guided design adjustments is rendered among the consecutive points. In [46], researchers classified optimization techniques based on the

approach of selecting the search direction. The study employed numerical optimization methods such as conjugate gradients, conjugate directions, and random search.

As the gradient-based methods provide better computational efficiency for general-purpose industrial applications, the algorithms are commonly used in optimization software. However, probabilistic and non-gradient have been studied to a broad extent. For solving global optimization problems, the heuristic methods are considered to find the global optimum with a high probability and accuracy at the same time. The genetic algorithms (GAs) and the simulated annealing technique are the heuristic methods that analogize to the physiological and biological phenomenon to approach the global optimum [47]. Based on natural genetics and natural selection principles, GAs do not employ any gradient information [54,73]. The metaheuristic optimization technique involves undertaking crossover, mutation, and selection operations to evolve candidate solutions via progressing a population inspired by Darwinian evolution [33].

When the objective achieves the desired value, or converges to extreme, or end of a fixed number of maximum generations, the algorithm terminates. GAs have demonstrated a high degree of robustness in achieving ideal solutions to complex optimization problems by selecting well-designed operators and optimal parameters [22,29]. Notably, researchers applied GA and various failure mechanisms in [36] to attain an optimal composite structure regarding different failure criteria. However, some limitations, like the tendency to reach the solution close to global optimum but not to exact optimally, require comparatively prolonged calculating time. However, it has been the most efficient stochastic technique for global optimization. Besides, the sampling capability of GAs is greatly affected by population size and local search algorithms such as a memetic algorithm.

Katoch et al. combined Baldwinian, Lamarckian, and local search with GAs to address the above issues. [35]. This hybridization optimization technique offers better solution quality, efficiency, guarantee of feasible solutions, and optimized control parameters [16]. However, the parameter setting is another challenge in GAs to find control parameters. Thus, a hybrid GA is developed in [48] at both local and global levels using a symbolic-numerical algorithm based on reducing nonlinear relations.

Moreover, for solving complex problems of optimal structural design, another probabilistic search algorithm called particle swarm optimization algorithm (PSOA) is also investigated in numerous research. Kennedy and Eberhart [15,36] first introduced the algorithm, which is inspired by the behavior of a group of species, such as a swarm of insects like wasps, bees, ants, and a school of fish or a flock of birds [47]. The method is particularly appropriate for solving problems containing optimal solutions in a

multidimensional space of the parameter. The particles, inspired from the natural analogy, act as an agent are illustrated considering both a position and a velocity allowing to move around in the search space [51]. As a metaheuristic nature, the method allows finding results for non-differentiable problems as well. The PSO algorithm can ameliorate predicaments with irregular, noisy, or dynamically changing with time in versatile computer science and applied mathematics domains.

Several scholars studied the PSO algorithm in [11,37,66,67] to solve various multimodal mathematical problems with little focus on practical challenges. However, Fourie and Groenwold applied PSO for structural and multidisciplinary optimization, considering shape and size optimization [18] and topology optimization [17]. Venter and Sobieszczanski-Sobieski (2003) studied both continuous and integer/discrete versions of the cantilevered beam problem, focusing on enhancing the basic PSO algorithm [72]. In [55], authors used the algorithm to reach a required strength of a composite structure under different failure criteria while minimizing weight and total cost.

In this chapter, optimal design problems of the composite structures are revisited, and the optimization techniques applied in the various studies are overviewed. From the background studies, it is evident that several types of research have taken place to achieve optimal criteria for composite materials and still require comparative experimentation with optimization methods in design research.

3. METHODS AND TECHNIQUES USED

In chapter 3, a short overview of methods and techniques used in this study has been presented. Based on the discussion in 2.3, the primary attention is paid to global optimization techniques for covering both real and integer design optimization problems. The gradient method is considered one of the fastest solutions, and the Lagrange multipliers method is introduced mainly to derive the necessary optimality conditions. Amongst many global optimization techniques available from literature are selected two widely used methods in engineering (GA, PSO).

3.1 Gradient method

Diverse gradient methods for optimization like the steepest descent method, Newton method, quasi-Newton method, and others. have been applied in many studies.

In the current thesis, the gradient method is not applied separately since

- Strongly nonlinear optimization problem has been considered, including a large number of local extremes.
- Gradient methods possess the limited capability to find global extreme; instead, they converge to the nearest extreme.

However, in this thesis, the gradient method is combined with a genetic algorithm to exploit the advantages of both. There is no significant preference for applying a specific gradient method for the problem designed in the study. In a hybrid algorithm, a substantial amount of computing time is consumed by GA. Additionally, considered gradient methods are principally faster than GA. When applying gradient methods, one solution is updated, but the population is updated in GA.

In the steepest descent method, the solution is modified incrementally by the following rules [53,71] (see Figure 3.1 for 1D and Figure 3.2 for 2D)

$$x_{n+1} = x_n - \gamma_n \nabla F(x_n), \quad n = 1, 2, \dots, \quad (7)$$

where γ_n is a stepsize and $\nabla F(x_n)$ is a gradient for the objective function. The starting point x_1 (assumed) is opted as an initial value. In (7) x is a vector of design variables (in the 1D problems case scalar).

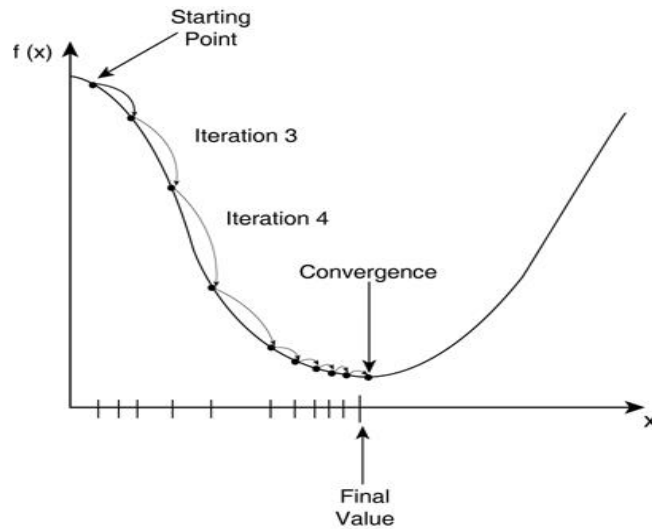


Figure 3.1 Steepest descent method, in the 1D case [71]

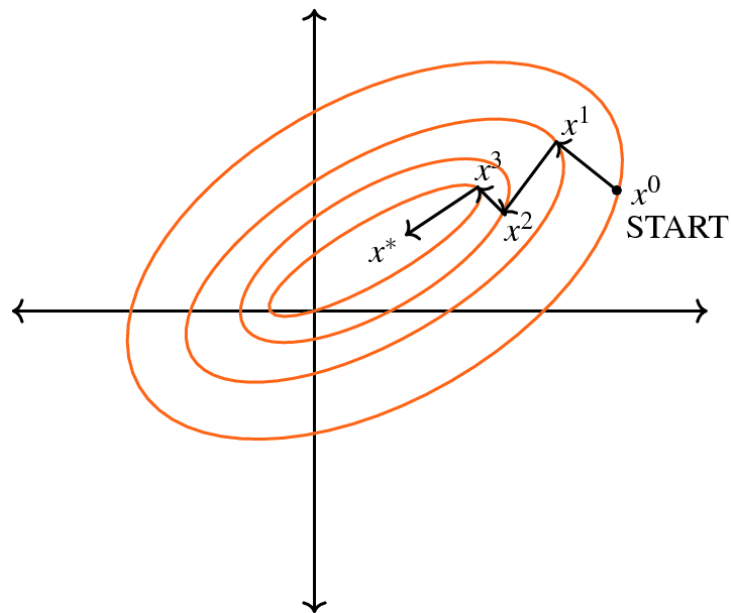


Figure 3.2 Steepest descent method, in the 2D case [53]

The value of the objective function in each line of Figure 3.2 is the same (level lines).

Similarly, in the Newton method, the rule for modification of the design variables is outlined as follows-

$$x_{n+1} = x_n - \gamma_n H^{-1}(x_n) \nabla F(x_n), \quad n = 1, 2, \dots, \quad (8)$$

where $H^{-1}(x_n)$ stand for Hess matrix.

The end part of the algorithm of the steepest descent and Newton methods can be determined by the value of the gradient or the difference between x_{n+1} and x_n (includes in formulas (7,8)). When the algorithm is not convergent, the maximum

number of iterations is defined and applied to control continuous run and computation time.

Both discussed methods have advantages since the steepest descent method has a broader convergence area, and the Newton method has a higher convergence speed, i.e., is faster. However, in the hybrid algorithm, the steepest descent method can be preferred since the computing time is determined mainly by GA.

3.2 Genetic algorithm

There are a considerable number of different real and binary-coded GA algorithms available. However, for this research, a general scheme of the GA has been delineated as follows [7]:

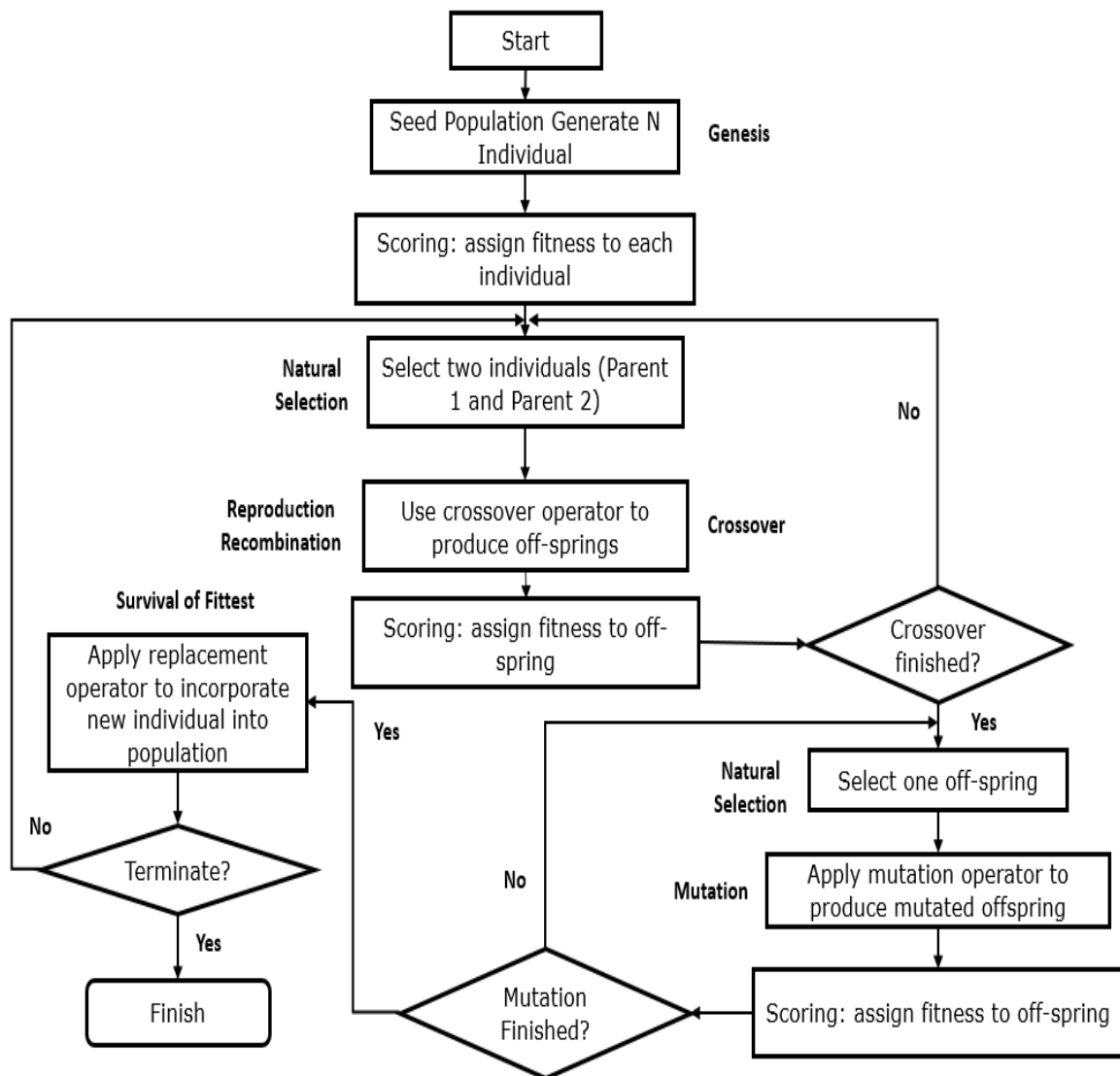


Figure 3.3 Genetic algorithm program flowchart [7]

At first, the initial population is generated using a random generator. In the case of classical binary coding, the bit value is generated by random. When population generated, the fitness function values are computed. The next step is employing the selection operator, and the principal task is to select parents for new children. Some widely used algorithms for selecting the parents can be listed as roulette wheel selection, group selection, tournament selection. In the roulette wheel selection method, the probability $p(k)$ of an individual, k is selected as a parent, proportional to the fitness function of the individual (Depicts in Figure 3.4).

$$p(k) = \frac{f(k)}{\sum_{i=1}^N f(i)}. \quad (9)$$

In (9) $f(k)$ and N stand for fitness function and size of the population, respectively.

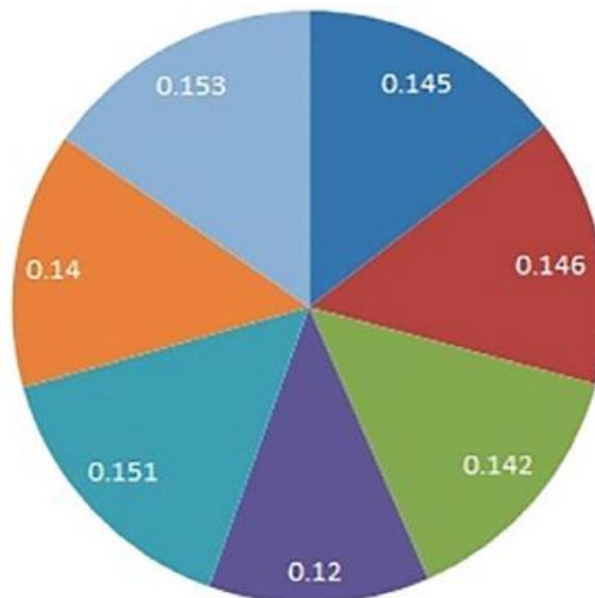


Figure 3.4 Roulette wheel selection [9]

The next operator is the crossover operator. Based on each two selected parents, two children (offspring) are composed. Each child receives chromosomes from two parents. However, the proportion of how many chromosomes come from each parent can be determined by applying randomly. Typical fragment on composing two children's (offspring) from the parents can be designed as follows-

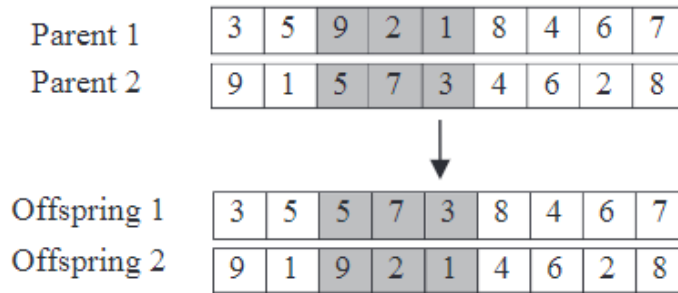


Figure 3.5 Double point crossover operator [3]

According to Figure 3.5, the first offspring gets the first two bits from parent 1, the subsequent three bits from parent 2, and the last four bits again from parent 1. The second offspring gets all data units not used for the first offspring. The next operator is the mutation operator. The mutation of a bit is meant to change its values from 0 to 1 and vice-versa. The mutation of chromosomes is performed with appointed mutation probability (mutation coefficient, small value 0.005). Mutations assist in finding new extremes during the convergence process.

After the mutation operator, the fitness values for each population member are computed, and the assigned number of survivors (population size) is determined based on fitness values. Finally, the termination condition can be defined by the number of generations and values of convergence parameters and others.

3.3 Particle swarm optimization

The population of outcomes is termed as a swarm and individuals as particles. Each particle has speed and coordination for moving in multidimensional space. The flying is adjusted according to the individual particle's flying experience and other particles present in the swarm. The particles accumulate memory, and each keeps track of its previous personal best (pbest) position. The current global best position of the population is denoted as gbest.

Eberhart and Shi (1998) introduced a standard PSO algorithm in [67] and the formula as follow:

$$v_{id}(t+1) = \omega v_{id}(t) + c_1 r_1 (\text{Localbest}(t) - x_{id}(t)) + c_2 r_2 (\text{Globalbest}(t) - x_{id}(t)), \quad (10)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (11)$$

Here,

- $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$; represents the position of i^{th} particle in D-dimension
- $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$; denotes the velocity of i^{th} particle in D-dimension direction of searching
- $i = 1, 2, \dots, N$; the population of the group particles
- $d = 1, 2, \dots, D$; the maximum number of iterations
- r_1, r_2 implies the random values between $[0,1]$ to maintain the group particles' diversity.
- c_1, c_2 entails the learning coefficient.
- ω represents inertia weight used to control the velocity, calculated by the equation (12).

$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{k_{max}} * k \quad (12)$$

Shi and Eberhart (1998) used inertia factor (ω), enhancing the performance of PSO convergence by regulating the influence on the present particle by the previous particle's velocity. In global searching proceeding, a more considerable value of ω has been shown comparatively better result whereas smaller ω during local search [67].

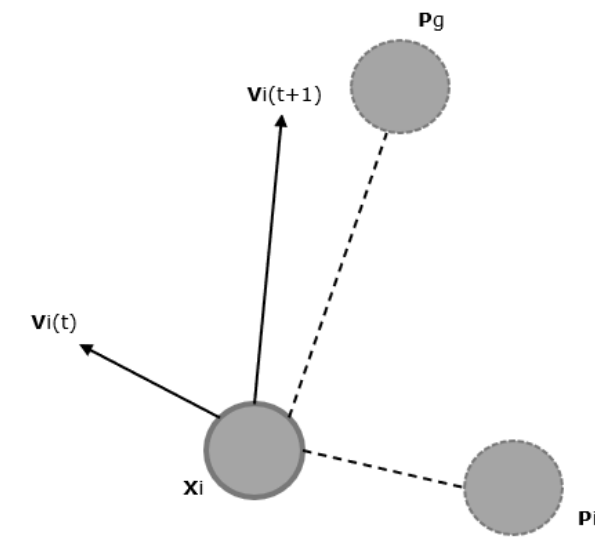


Figure 3.6 The illustration velocity update of a particle in PSO [56].

Figure 3.6 depicts a particle (x_i) that renews the velocity and position from the previous state in each step (t). With the summation of previous velocity ($v_i(t)$), the best position attained to this point by the particle (p_i) and the best position reached by the entire swarm (p_g), the new velocity ($v_i(t + 1)$) is derived. Finally, the new position can be calculated by adding the new velocity to the equation (11).

In [31], a general schema is introduced to illustrate the basic steps entailed in PSO and depicted as follows in Figure 3.7:

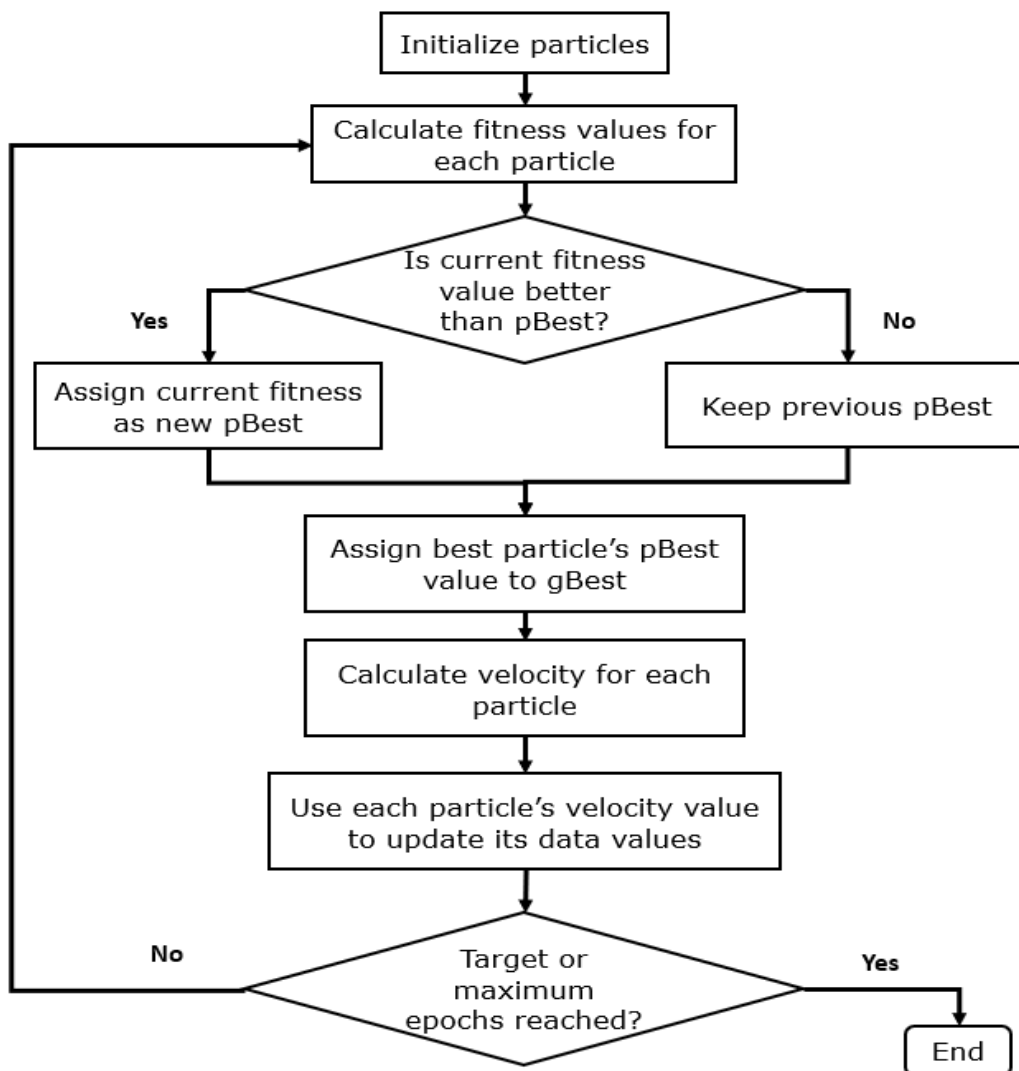


Figure 3.7 Flow diagram illustrating the particle swarm optimization algorithm. [31].

Usually, the PSO algorithm is more straightforward than GA due to the absence of crossover and mutation operators. However, the PSO has several variations, and some of them include mutation operators.

3.4 Lagrange multipliers method

As the Lagrange multipliers method requires the derivatives/gradients of the functions while analyzing numerical functions, it is applicable where objective and constraints functions are differentiable.

In the current research, the Lagrange multipliers method is developed for derivation of the necessary optimality conditions. The Lagrange function for optimization problem (13)-(15) with one objective function read.

$$L = F(x) + \sum_{k=1}^l \lambda_k h_k(x) + \sum_{j=1}^m \mu_j g_j(x). \quad (13)$$

The necessary optimality conditions can be obtained by:

- Equalizing the derivatives of the Lagrange function concerning design variables to zero i.e.

$$\frac{\partial L}{\partial x_i} = \frac{\partial F(x)}{\partial x_i} + \sum_{k=1}^l \lambda_k \frac{\partial h_k(x)}{\partial x_i} + \sum_{j=1}^m \mu_j \frac{\partial g_j(x)}{\partial x_i} = 0, i = 1, \dots, n, \quad (14)$$

- adding the equality constraints (3),
- adding the following equations

$$\mu_j g_j(x) = 0, \quad j=1, \dots, m. \quad (15)$$

The meaning of the equations (15) is that one of two possibilities take place:

- the inequality constraint will take the form of equality ($g_j(x) = 0$) or
- corresponding Lagrange multiplier is equal to zero ($\mu_j = 0$).

The necessary optimality conditions derived can be used to validate the local extreme of the objective function. The optimality conditions derived by the Lagrange multipliers method represent a set of algebraic equations (linear and nonlinear) that can be solved for determining the optimal solution of the initial optimization problem.

In this chapter, numerical methods have been briefly studied to approach both local and global optimum addressing the constrained and unconstrained optimization problem. The following two chapters will apply the referred techniques to resolve the designed problem.

4. OPTIMAL MATERIAL ORIENTATION OF 3D ORTHOTROPIC MATERIALS

Chapter 4 formulates the problem and necessary conditions for optimal material orientation for orthotropic material. The orthotropic materials, both 2D and 3D, are the most widely used reinforced composite due to axisymmetric properties. In 2D orthotropic materials, the closed-form analytical solution was derived by P.Pedersen [57]. Later constraint optimization has been analyzed considering GA and Lagrange multipliers techniques to draw a comparison.

4.1 Problem formulation

The strain energy density is subjected to minimization to obtain a material/structure with maximum stiffness properties [41].

$$J_{StrainEnergyDensity} = \frac{1}{2} \boldsymbol{\varepsilon}^T \mathbf{C} \boldsymbol{\varepsilon} \rightarrow \min, \quad (16)$$

where \mathbf{C} and $\boldsymbol{\varepsilon}$ stand for the orthotropic constitutive matrix and strain vector, respectively.

$$\mathbf{C} = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & 0 & 0 & 0 \\ C_{1122} & C_{2222} & C_{2233} & 0 & 0 & 0 \\ C_{1133} & C_{2233} & C_{3333} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{2323} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{3131} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{1212} \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{23} \\ 2\varepsilon_{31} \\ 2\varepsilon_{12} \end{Bmatrix}. \quad (17)$$

The elements of the constitutive matrix describe materials properties and can be determined from material tests.

In the case of 3D linear elasticity, the transformation formulas for strains are formulated as follows.

$$\begin{aligned} \varepsilon_{11} = & \left((e_1 - e_2) * \cos^2(\theta_3) - e_1 + e_3 \right) * \cos^2(\theta_2) + (e_1 - e_2) * \cos^2(\theta_3) + e_2 \\ & - e_3 \Big) * \cos^2(\theta_1) - 2 * \sin(\theta_1) * \sin(\theta_3) * \cos(\theta_2) * \cos(\theta_3) * (e_1 - e_2) \\ & * \cos(\theta_1) + \left((-e_1 + e_2) * \cos^2(\theta_3) + e_1 - e_3 \right) * \cos^2(\theta_2) + e_3, \end{aligned}$$

$$\begin{aligned} \varepsilon_{22} = & \left(-(\cos^2(\theta_2) + 1) * (e_1 - e_2) * \cos^2(\theta_3) + (e_1 - e_3) * \cos^2(\theta_2) - e_2 + e_3 \right) \\ & * \cos^2(\theta_1) + 2 * \sin(\theta_1) * \sin(\theta_3) * \cos(\theta_2) * \cos(\theta_3) * (e_1 - e_2) \\ & * \cos(\theta_1) + (e_1 - e_2) * \cos^2(\theta_3) + e_2, \end{aligned}$$

$$\begin{aligned} \varepsilon_{33} = & \cos^2(\theta_3) * \cos^2(\theta_2) * e_1 - \cos^2(\theta_3) * \cos^2(\theta_2) * e_2 - \cos^2(\theta_3) * e_1 + \\ & \cos^2(\theta_3) * e_2 - \cos^2(\theta_2) * e_1 + \cos^2(\theta_2) * e_3 + e_1, \end{aligned}$$

$$\begin{aligned} \varepsilon_{12} = & -2 * \sin(\theta_3) * \cos(\theta_2) * \cos(\theta_3) * (e_1 - e_2) * \cos^2(\theta_1) \\ & - \left((e_1 - e_2) * \cos^2(\theta_3) - e_1 + e_3 \right) * \cos^2(\theta_2) + (e_1 - e_2) * \cos^2(\theta_3) \\ & + e_2 - e_3 \Big) * \sin(\theta_1) * \cos(\theta_1) + \sin(\theta_3) * \cos(\theta_2) * \cos(\theta_3) \\ & * (e_1 - e_2), \end{aligned}$$

$$\begin{aligned} \varepsilon_{23} = & - \left(-\cos(\theta_2) * \left((e_1 - e_2) * \cos^2(\theta_3) - e_1 + e_3 \right) * \cos(\theta_1) + \sin(\theta_1) * \sin(\theta_3) \right. \\ & \left. * \cos(\theta_3) * (e_1 - e_2) \right) * \sin(\theta_2), \end{aligned}$$

$$\begin{aligned} \varepsilon_{31} = & \left(\sin(\theta_1) * \left((e_1 - e_2) * \cos^2(\theta_3) - e_1 + e_3 \right) * \cos(\theta_2) + \sin(\theta_3) * \cos(\theta_1) * \cos(\theta_3) * (e_1 - \right. \\ & \left. e_2) \right) * \sin(\theta_2). \end{aligned} \tag{18}$$

The Euler angles θ_1 , θ_2 and θ_3 mentioned above are design variables describing the orientation of the material.

4.2 Selection of optimization methods and techniques

As for complex theoretical and practical optimization problems, selecting suitable and effective optimization methods and techniques is crucial.

Let proceed from suggestions outlined in section 2 to select optimization methods and techniques for considered optimal orientation.

1. Selection of optimization criteria, variables.

The optimization criterion, the design variables, the design domain, and constraints are determined by problem formulation stated in section 4.1. The constraints on the design domain will be applied to determine extremes in subdomains since a large number of solutions exist. In addition, some of them correspond to the same value of the objective function. The latter fact is caused due to symmetric properties of the orthotropic material.

2. Selection of optimization methods.

Since the considered problem is strongly nonlinear, the gradient methods are not suitable (fast but as a rule converges to the nearest extreme). In general, the evolutionary algorithms (EA) provide convergence to the global extreme (not guaranteed). Herein two widely used EA-s: genetic algorithms and particle swarm optimization algorithms, are selected. Another reason for the selection of EA algorithms because there are two options of the solution:

- Solution in terms of real variables
- Solution in terms of integer variables.

The solution procedure can speed up using a hybrid algorithm (GA+gradient or PSO+gradient) in a real variable problem. In both cases, the first stage of the algorithm (GA or PSO) provides convergence to the global extreme. The second stage of the algorithm (Gradient method) provides higher convergence speed and avoids perturbations near extreme, habitual for GA. Concerning an integer variables-based solution, the GA/PSO can be combined with integer programming methods (Hill climbing, cutting plane methods, and others). Herein, the hybrid approach is not applied regarding the integer problem since the solution set is substantially restricted, which provides faster convergence.

4.3 Necessary optimality conditions

Before the optimality analysis, it is essential to define the optimality conditions and operational arrangements. The necessary optimality conditions are needed to check and ensure that the solution determined corresponds to the extreme value of the objective function. Even so, the proposed condition may not ensure that the extreme will be global. Moreover, the necessary optimality conditions are applicable when the problems have differentiable objective and constraint functions and real continuous variables. Thus, it is not applicable in mixed-integer variables or even for the real variables with discrete values.

With regards to the unconstrained optimization problem formulated in section 4.1, the necessary optimality conditions can be derived by equalizing derivatives of the objective function concerning design variables (Euler angles) to zero i.e.

$$\frac{\partial J_{Energy}}{\partial \theta_i} = \{\varepsilon\}^T [C] \left\{ \frac{\partial \varepsilon}{\partial \theta_i} \right\} = 0, \quad i = 1, 2, 3. \quad (19)$$

The conditions (19) can be used to check the correctness of the solution and solve an initial optimization problem. However, despite to formal simplicity of the conditions (19), these conditions are strongly nonlinear equations with a large number of solutions (the strain components equations (18) will assist in understanding the complexity of the conditions (19). Herein, the optimality conditions (19) are applied for validation of the solution. For analytical calculation derivatives (19), the MAPLE function diff has been used, and the obtained final formulas are copied to MATLAB.

4.4 Constrained optimization

In general, the constrained optimization is more complex in comparison with unconstrained ones. However, the approach largely depends on a specific problem and the corresponding method used.

4.4.1 Problem formulation

In the current formulated problem in section (4.1), an alternate approach has been designed, which is likely to be effective in some instances. Namely, the posed problem can also be solved in two stages:

- First, determine an optimal solution in terms of strains,
- Second, compute Euler angles based on known strain components values.

Thus, the strain components are considered design variables, and formulas (18) will not be substituted (16). The energy expression (16) can be expressed in detail as

$$J_{Energy} = (C^{1111}\varepsilon_{11} + C^{1122}\varepsilon_{22} + C^{1133}\varepsilon_{33})\varepsilon_{11} + (C^{1122}\varepsilon_{11} + C^{2222}\varepsilon_{22} + C^{2233}\varepsilon_{33})\varepsilon_{22} + \quad (20)$$

$$+ (C^{1133}\varepsilon_{11} + C^{2233}\varepsilon_{22} + C^{3333}\varepsilon_{33})\varepsilon_{33} + 4C^{2323}\varepsilon_{23}^2 + 4C^{3131}\varepsilon_{31}^2 + 4C^{1212}\varepsilon_{12}^2$$

However, the constrained optimization problem is formulated in terms of strains for strain components to hold good,

$$\begin{cases} \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33} - \varepsilon_I - \varepsilon_{II} - \varepsilon_{III} = 0 \\ \varepsilon_{11}\varepsilon_{22} + \varepsilon_{11}\varepsilon_{33} + \varepsilon_{22}\varepsilon_{33} - \varepsilon_{12}^2 - \varepsilon_{23}^2 - \varepsilon_{31}^2 - \varepsilon_I\varepsilon_{II} - \varepsilon_I\varepsilon_{III} - \varepsilon_{II}\varepsilon_{III} = 0, \\ \varepsilon_{11}\varepsilon_{22}\varepsilon_{33} + 2\varepsilon_{12}\varepsilon_{23}\varepsilon_{31} - \varepsilon_{11}\varepsilon_{23}^2 - \varepsilon_{22}\varepsilon_{31}^2 - \varepsilon_{33}\varepsilon_{12}^2 - \varepsilon_I\varepsilon_{II}\varepsilon_{III} = 0 \end{cases} \quad (21)$$

where ε_I , ε_{II} and ε_{III} stand for principal strains.

Finally, in terms of strains, the optimization can be formulated to minimize the objective function (20) subjected to constraints (21).

4.4.2 Problem solution using GA

The strain components ε_{11} , ε_{22} , ε_{33} , ε_{12} , ε_{23} and ε_{31} are considered as design variables, and the objective function (20) is minimized considering constraints (21) by applying some optimization techniques. Herein the genetic algorithm is applied (MATLAB software functions are used. and details in Appendix 3). In GA, the population size 350 and number of generations 350 have been employed. As results of the solution, the optimal values of six strain components are obtained, also the value of the objective function:

$$\varepsilon_{11} = -0.7598 , \varepsilon_{22} = -0.9819 , \varepsilon_{33} = -3.2592 , \varepsilon_{12} = 5.1268 , \varepsilon_{23} = 4.5126 , \varepsilon_{31} = 4.6664 ,$$

$$J_{Energy} = 795.4273 \text{ GPa.}$$

Here E-Glass/Vinilester is used, detailed material data are presented in the next chapter.

As a second step, the optimal values of the Euler angles can be computed utilizing the following formulas [67]

$$\begin{aligned}\sin^2(\theta_2) &= \frac{(\varepsilon_{33} - \varepsilon_{III})(\varepsilon_{33} + \varepsilon_{III} - \varepsilon_I - \varepsilon_{II}) + \varepsilon_{31}^2 + \varepsilon_{23}^2}{(\varepsilon_I - \varepsilon_{III})(\varepsilon_{III} - \varepsilon_{II})}, \\ \sin^2(\theta_1) &= \frac{\varepsilon_{23}^2 - \varepsilon_{22}\varepsilon_{33} - \varepsilon_{11}\varepsilon_{III} + \varepsilon_{III}(\varepsilon_I + \varepsilon_{II})}{\sin^2(\theta_2)(\varepsilon_I - \varepsilon_{III})(\varepsilon_{III} - \varepsilon_{II})}, \\ \cos(2\theta_3) &= 2 \frac{\varepsilon_{III} - \frac{1}{2}(\varepsilon_I + \varepsilon_{II}) - (\varepsilon_{III} - \varepsilon_{33}) / \sin^2(\theta_2)}{(\varepsilon_{II} - \varepsilon_I)}.\end{aligned}\tag{22}$$

The formulas (22) are derived by combining the equations (18).

An approach where the above posed constrained optimization problem is solved directly using GA is the most time-consuming as the average computing time is ~100 sec. In addition, it is 30-300 times slower than other solutions considered. Therefore, in most cases, unconstrained optimization is preferable if it can solve the same problem. In practice, usually, apart from few problems, both approaches can be applied. Therefore, the considered optimal material orientation problem is instead an exception where different formulations are available.

4.4.3 Problem solution using Lagrange multipliers method

There are several approaches to resolving constrained optimization problems like constraints in the form of penalties, the Lagrange multipliers method, etc. The GA algorithm is used in this study, which satisfies the linear and bound constraints of applying mutation and crossover functions that exclusively generate feasible points even if the nonlinear constraints fail to satisfy during the convergence process. However, ultimately the algorithm can reach expected results in the final solution.

In this study, the Lagrange multipliers method is applied to derive the Karush-Kuhn-Tucker necessary optimality conditions. First, the Lagrange function summarizes the objective function and the constraints multiplied by corresponding Lagrange multipliers.

$$\begin{aligned}
L = & (C^{1111}\varepsilon_{11} + C^{1122}\varepsilon_{22} + C^{1133}\varepsilon_{33})\varepsilon_{11} + (C^{1122}\varepsilon_{11} + C^{2222}\varepsilon_{22} + C^{2233}\varepsilon_{33})\varepsilon_{22} + \\
& + (C^{1133}\varepsilon_{11} + C^{2233}\varepsilon_{22} + C^{3333}\varepsilon_{33})\varepsilon_{33} + 4C^{2323}\varepsilon_{23}^2 + 4C^{3131}\varepsilon_{31}^2 + 4C^{1212}\varepsilon_{12}^2 + \\
& \lambda_1(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33} - \varepsilon_I - \varepsilon_{II} - \varepsilon_{III}) + \\
& \lambda_2(\varepsilon_{11}\varepsilon_{22} + \varepsilon_{22}\varepsilon_{33} + \varepsilon_{33}\varepsilon_{11} - \varepsilon_{12}^2 - \varepsilon_{23}^2 - \varepsilon_{31}^2 - \varepsilon_I\varepsilon_{II} - \varepsilon_{II}\varepsilon_{III} - \varepsilon_{III}\varepsilon_I) + \\
& \lambda_3(\varepsilon_{11}\varepsilon_{22}\varepsilon_{33} + 2\varepsilon_{12}\varepsilon_{23}\varepsilon_{31} - \varepsilon_{11}\varepsilon_{23}^2 - \varepsilon_{22}\varepsilon_{31}^2 - \varepsilon_{33}\varepsilon_{12}^2 - \varepsilon_I\varepsilon_{II}\varepsilon_{III}).
\end{aligned} \tag{23}$$

In (23) λ_1 , λ_2 and λ_3 stand for Lagrange multipliers. The necessary optimality conditions can be obtained by equalizing the derivatives of the Lagrange function L concerning strain components with zero i.e.

$$\frac{\partial L}{\partial \varepsilon_{11}} = 0, \quad \frac{\partial L}{\partial \varepsilon_{22}} = 0, \quad \frac{\partial L}{\partial \varepsilon_{33}} = 0, \quad \frac{\partial L}{\partial \varepsilon_{12}} = 0, \quad \frac{\partial L}{\partial \varepsilon_{23}} = 0, \quad \frac{\partial L}{\partial \varepsilon_{31}} = 0 \tag{24}$$

The problem includes nine variables (6 strain components+3 Lagrange multipliers) and nine equations (6 equation in (24) +3 constraints (21)). In general, applying the Lagrange multipliers method will convert the initial optimization problem into a system of algebraic equations. The obtained algebraic system can be solved analytically or numerically depending on the complexity of the formulated specific problem. In the case of considered problems and orthotropic material, the algebraic system is firmly nonlinear and complicated. The analytical solutions can be determined for particular cases where at least one shear strain equals zero [41]. In the most complicated case where all shear strains are nonzero, the numerical solution can be utilized. However, based on the literature, the global minimum of the strain energy density can be achieved in this most complicated case, where the numerical solution is required [49].

Instead of a direct solution of the optimality conditions, it is reasonable to perform the first detailed analysis of the optimality conditions, and if possible, to simplify these conditions by eliminating a set of variables. By analyzing the problem, the three Lagrange multipliers and the shear strains can be eliminated. The expressions for the shear strain can be presented as follows [67]-

$$\varepsilon_{12}^2 = \frac{F_3}{D_3} \frac{F_2}{D_2}, \quad \varepsilon_{23}^2 = \frac{F_1}{D_1} \frac{F_3}{D_3}, \quad \varepsilon_{31}^2 = \frac{F_1}{D_1} \frac{F_2}{D_2}. \tag{25}$$

In (25) D_1 , D_2 and D_3 are given in terms of material parameters as

$$D_1 = 2(E^{2323} - E^{3131}), \quad D_2 = 2(E^{3131} - E^{1212}), \quad D_3 = 2(E^{1212} - E^{2323}), \tag{26}$$

but F_1 , F_2 and F_3 include the strain components ε_{11} , ε_{22} , ε_{33} and materials parameters

$$\begin{aligned} F_1 &= (E^{1111} - E^{1122} - 2E^{1212})\varepsilon_{11} - (E^{2222} - E^{1122} - 2E^{1212})\varepsilon_{22} + (E^{1133} - E^{2233})\varepsilon_{33}, \\ F_2 &= (E^{1122} - E^{1133})\varepsilon_{11} + (E^{2222} - E^{2233} - 2E^{2323})\varepsilon_{22} - (E^{3333} - E^{2233} - 2E^{2323})\varepsilon_{33}, \\ F_3 &= (E^{1133} - E^{1111} + 2E^{3131})\varepsilon_{11} + (E^{2233} - E^{1122})\varepsilon_{22} + (E^{3333} - E^{1133} - 2E^{3131})\varepsilon_{33}. \end{aligned} \quad (27)$$

The equations (27) are linear concerning strain components. By combining initial optimality conditions (24) and the strain invariants (21), the simplified optimality conditions can be derived as [49]

$$\begin{aligned} \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33} - \varepsilon_I - \varepsilon_{II} - \varepsilon_{III} &= 0, \\ \varepsilon_{11}\varepsilon_{22} + \varepsilon_{22}\varepsilon_{33} + \varepsilon_{33}\varepsilon_{11} - \frac{F_2 F_3}{D_2 D_3} - \frac{F_1 F_2}{D_1 D_2} - \frac{F_1 F_3}{D_1 D_3} - \varepsilon_I \varepsilon_{II} - \varepsilon_{II} \varepsilon_{III} - \varepsilon_{III} \varepsilon_I &= 0, \end{aligned} \quad (28)$$

$$\varepsilon_{11}\varepsilon_{22}\varepsilon_{33} + 2\frac{F_1 F_2 F_3}{D_1 D_2 D_3} - \varepsilon_{11}\frac{F_1 F_3}{D_1 D_3} - \varepsilon_{22}\frac{F_1 F_2}{D_1 D_2} - \varepsilon_{33}\frac{F_2 F_3}{D_2 D_3} - \varepsilon_I \varepsilon_{II} \varepsilon_{III} = 0.$$

The simplified optimality conditions (28) can be solved concerning three strain components ε_{11} , ε_{22} and ε_{33} . However, further simplification of the equations (28) is available, leading to lengthy derivations and extremely huge expressions. Herein the system (28) is solved numerically using a MATLAB solver. (Appendix 3).

If strain components ε_{11} are determined, the shear strains can be computed using formulas (25). After deriving all strain components, the strain energy density can be evaluated using (20). Finally, the Euler angles can be computed only for an optimal solution using formulas (22). Compared with the solution obtained in section (4.5), the average computing time has been reduced several magnitudes from 100 sec to 3 sec.

In the case of E-Glass/Vinilester (detailed material data are presented in the next chapter), the Euler angles corresponding to optimal material orientation are given as

$$\theta_1 = 0.7456 (42.72^\circ) \quad \theta_2 = 1.5261 (87.44^\circ), \quad \theta_3 = 0.5326 (30.52^\circ).$$

Here the Euler angles are presented in radians and the optimal value of the obtained strain energy density.

$$J_{Energy} = 795.4174 \text{ GPa}$$

The obtained value of the strain energy density is in good agreement with the corresponding value found in section 4.6 using a different approach (performing optimization with constraints).

In this chapter, for the 3D orthotropic material objective function, constraints and design variables have been formulated to address the orientational design problem. Later necessary optimality condition is also set for the optimality analysis. In addition, constrained optimization analysis was performed to draw a comparison applying the Lagrange multiplier and GA for a composite E-Glass/Vinilester. From the result, it is evident that constrained optimization is more complex and requires higher computation time. Compared with the GA and Lagrange multipliers method, the latter method is more effective as it operates within less computation time.

5. NUMERICAL RESULTS AND ANALYSIS

In Chapter 5, the numerical analysis has been conducted as a problem formulated in chapter 4 by applying the optimization techniques outlined in chapter 3. For this purpose, two different orthotropic materials are considered E-Glass/Vinylester and Graphite/Epoxy, and computation time has been compared concerning the applied methods.

5.1 E-Glass/Vinylester

The properties of the material are described with the following values of engineering parameters $E_1 = 25$ GPa, $E_2 = 24.8$ GPa, $E_3 = 8.5$ GPa, $G_{12} = 6.5$ GPa, $G_{13} = 4.2$ GPa, $G_{23} = 4.5$ GPa, $\nu_{12} = 0.1$, $\nu_{13} = 0.28$, $\nu_{23} = 0.3$ (stiffness characteristics). The compliance matrix can be computed based on engineering parameters as

$$S = \begin{bmatrix} 1/E_1 & -\nu_{12}/E_1 & -\nu_{13}/E_1 & 0 & 0 & 0 \\ -\nu_{12}/E_1 & 1/E_2 & -\nu_{23}/E_2 & 0 & 0 & 0 \\ -\nu_{13}/E_1 & -\nu_{23}/E_2 & 1/E_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/G_{23} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/G_{31} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/G_{12} \end{bmatrix} = \begin{bmatrix} 0.040 & -0.004 & -0.011 & 0 & 0 & 0 \\ -0.004 & 0.040 & -0.012 & 0 & 0 & 0 \\ -0.011 & -0.012 & 0.118 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.222 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.238 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.154 \end{bmatrix}$$

The constitutive matrix can be computed as the inverse of the compliance matrix.

$$C = S^{-1} = \begin{bmatrix} 2.614 & 0.345 & 0.284 & 0 & 0 & 0 \\ 0.345 & 2.604 & -0.301 & 0 & 0 & 0 \\ 0.284 & 0.301 & 0.908 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.450 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.420 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.650 \end{bmatrix}$$

And used in the objective function (16).

The principal strain values for this composite are presented as $\varepsilon_I = 8$, $\varepsilon_{II} = -7$, $\varepsilon_{III} = -6$.

Table 5.1 illustrates the optimal values of the Euler angles, minimum strain energy density in GPa-s according to the different methods used in the analysis. The detailed MATLAB code is available in Appendix 1 and 2.

Table 5.1 E-Glass/Vinilester, solutions using GA+Gradient, GA, PSO, and integer variables GA

Method	Min. strain energy density (GPa)	θ_1^*	θ_2^*	θ_3^*
GA+Gradient	795.4174	0.7456 (42.72°)	1.5261 (87.439°)	0.5326 (30.516°)
GA	795.4188	0.7490 (42.915°)	1.5305 (87.691°)	0.5324 (30.504°)
PSO	795.4173	0.7456 (42.72°)	1.5261 (87.439°)	0.5326 (30.516°)
GA, Integer, 1 degree	795.4699	43°	88°	31°
GA, Integer, 5 degrees	795.6845	45°	90°	30°
GA, Integer, 10 degrees	797.6317	40°	80°	30°
GA, Integer, 15 degrees	795.6845	45°	90°	30°

First of all, it can be observed from Table 5.1 that the results obtained by applying GA and PSO are in good agreement. The first to the third row of the results corresponding to real variable-based optimization. If the HGA (GA+Gradient) method was applied, the results obtained using different algorithms' runs would coincide. However, if GA is used (row 2), some perturbations appear in the optimal solution values because of the mutation operator used in GA. The results presented in the first two rows of Table 5.1 derived from applying HGA are in good agreement with the results exhibited in [50].

The last four rows of Table 5.1 represent integer solutions corresponding to a minimal change of Euler angles of 1, 5, 10, and 15 degrees. In practice, orientation angles are used with different step sizes depending on a specific problem. In general, the nearest integer values to real solutions (rounding) may not lead to the best integer solution. For the step size of Euler angles 1,5 and 15 degrees, the integer solutions are close to available to real value-based solutions. However, it is not the case in the case of step size of Euler angles 10 degrees.

Due to the symmetric nature of the objective function, four different equivalent solutions exist for the posed problem (for real variables). Since the objective function values coincide for these solutions, even EA algorithms are inadequate to discover all these solutions if the design domain is not restricted temporarily. The four equivalent solutions and their calculating formulas are exhibited in Table 5.2 for the HGA (GA+ Gradient) method. There is no justification for repeating all other solutions because their values can be calculated by similar formulas that remain effective for real variable solutions only.

Table 5.2 E-Glass/Vinilester, four equivalent optimal solutions.

Method	Min. strain energy density (GPa)	θ_1^*	θ_2^*	θ_3^*
GA+Gradient	795.4174	0.7456 (42.72°)	1.5261 (87.439°)	0.5326 (30.516°)
GA+Gradient	795.4174	0.7456 (42.72°)	π -1.5261 (180°-87.439°)	π -0.5326 (180°-30.516°)
GA+Gradient	795.4174	π -0.7456 (180°- 42.72°)	1.5261 (87.439°)	π -0.5326 (180°-30.516°)
GA+Gradient	795.4174	π -0.7456 (180°- 42.72°)	π -1.5261 (180°-87.439°)	0.5326 (30.516°)

Despite four equivalent global optimal solutions, the considered problem has a considerable number of local extremes.

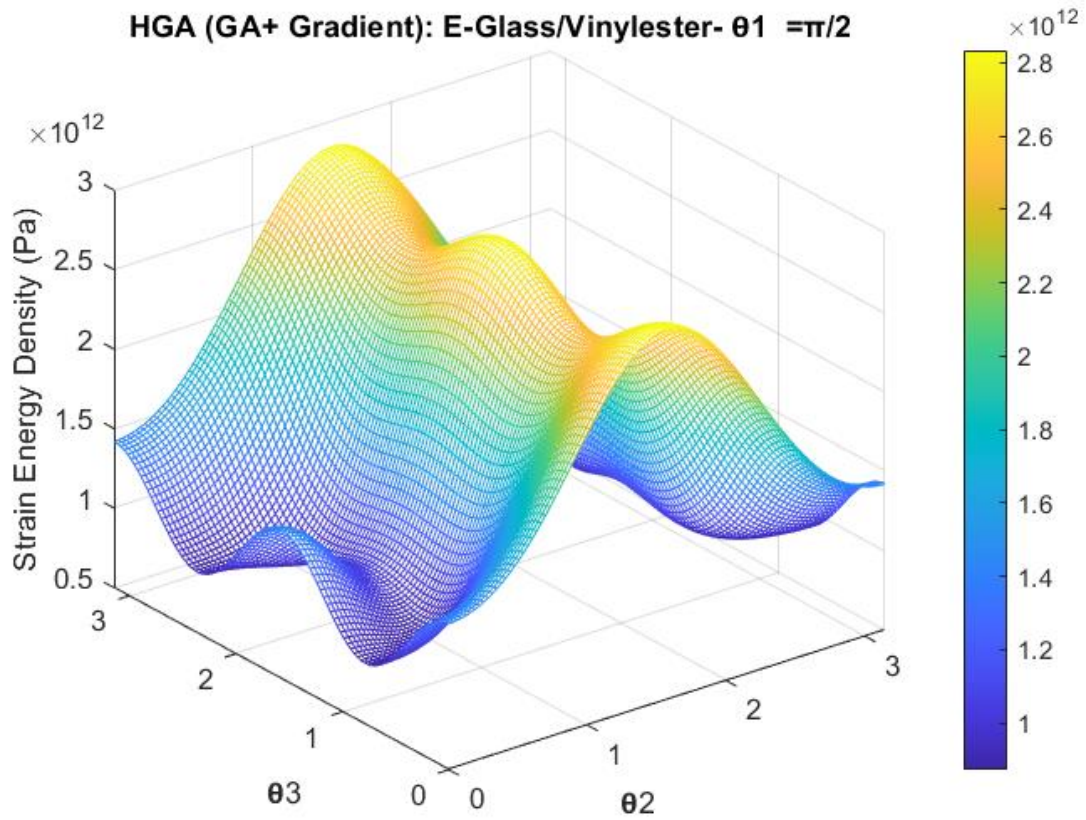


Figure 5.1 E-Glass/Vinylester, distribution of the strain energy as function of Euler angles θ_2 and θ_3 (θ_1 is constant)

Figure 5.1 depicts the strain energy density by fixing the Euler angle $\theta_1 = \frac{\pi}{2}$ to perceive the energy distribution behavior in 3D. However, the original objective function contains three design parameters, and the result is calculated in 4D. For this plot, the energy density function is a function of the Euler angle θ_2 and θ_3 . It is also apparent from the figure the presence of a number of local extremes.

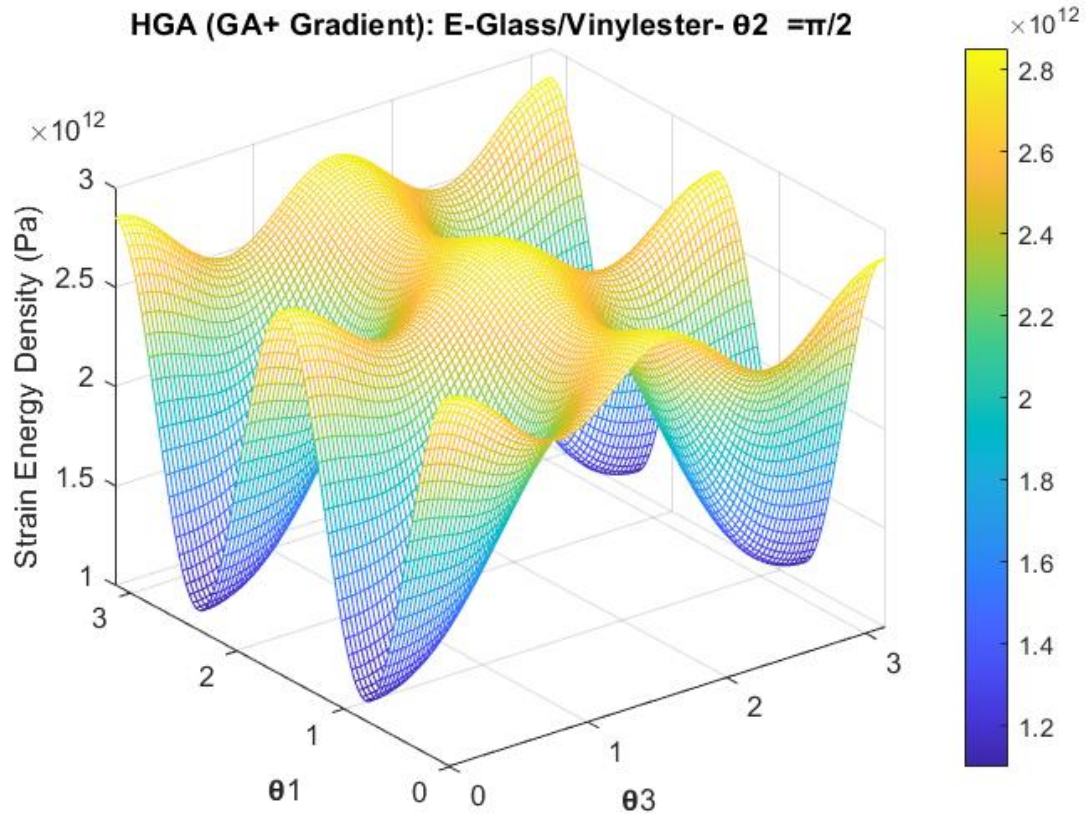


Figure 5.2 E-Glass/Vinylester, distribution of the strain energy as function of Euler angles θ_3 and θ_1 (θ_2 is constant)

Figure 5.2 represents the strain energy density by setting the Euler angle $\theta_2 = \frac{\pi}{2}$. For this plot, the energy density function is considered as a function of the Euler angle θ_3 and θ_1 to perceive the energy distribution behavior in 3D from a different perspective. The original objective function contains three design parameters, and the result is calculated in 4D. Similarly to Figure 5.1, it is also apparent from the figure that the presence of a number of local extremes and complexity of the designed problem.

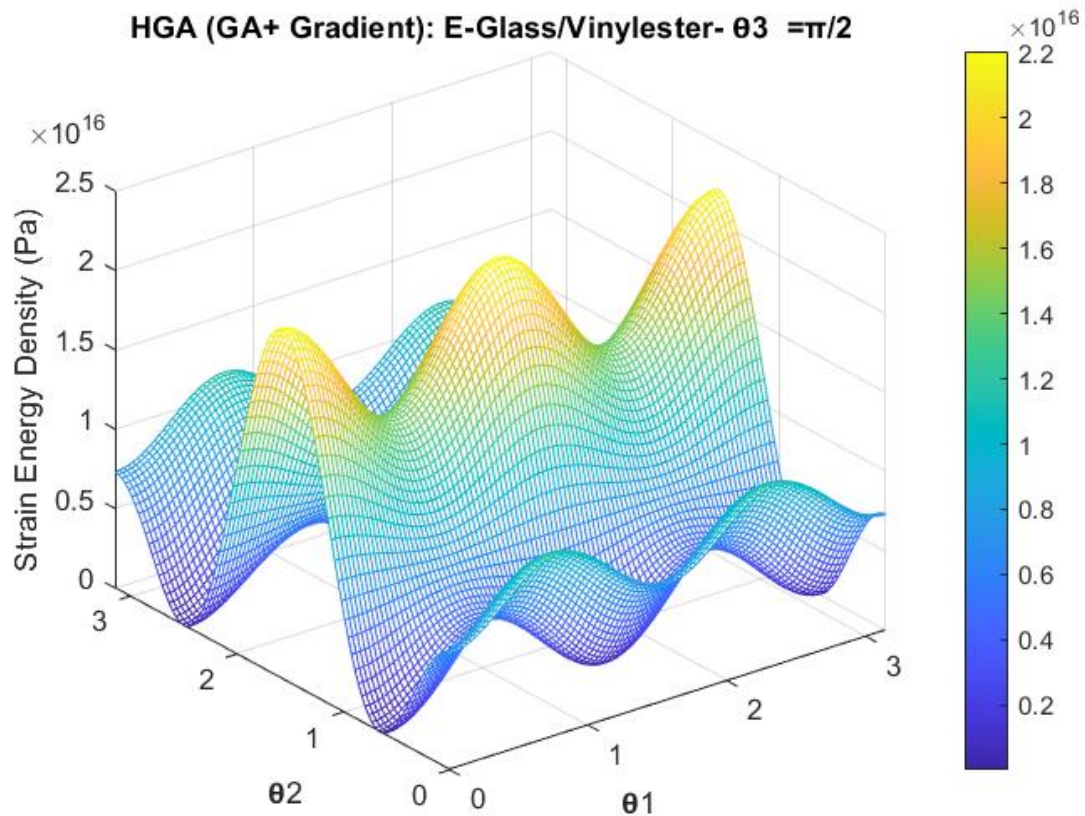


Figure 5.3 E-Glass/Vinylester, distribution of the strain energy as function of Euler angles θ_1 and θ_2 (θ_3 is constant)

Figure 5.3 depicts the strain energy density for the composite E-Glass/Vinylester by considering the Euler angle $\theta_3 = \frac{\pi}{2}$ to observe the energy distribution behavior in 3D. For this diagram, the energy density function is a function of the Euler angle θ_1 and θ_2 . The illustration would assist in comprehending the nature of the distribution because the original objective function contains three design parameters, and the result is calculated in 4D. The presence of numerous local extremes can also be visible, like in Figure 5.1 and Figure 5.2.

In table 5.3, average computation times from three consecutive runs are listed of the consecutive methods applied for the optimization analysis. The average value is necessary for the assessment since the studied algorithms operate as random, and a single run may not depict the real scenario.

Table 5.3 E-Glass/Vinylester, computing time

Method	Computing Time (Sec.)
GA+Gradient	1.7253
GA	2.5453
PSO	1.4032
GA, Integer, 1 degree	0.9834
GA, Integer, 5 degrees	0.8020
GA, Integer, 10 degrees	0.7345
GA, Integer, 15 degrees	0.7031

By considering the real value variables-based algorithms, it can be observed from Table 5.3 that the fastest method appears PSO algorithm and slowest GA algorithm. The HGA (GA+Gradient) is faster than GA since theoretically, the gradient method is faster than GA, and part of algorithm computing faster. In GA, HGA, and PSO, the equal population size 200 and number of generations 150 have been used. Also, the computing times for real value-coded algorithms are notably close, with minor differences.

In the case of integer coded algorithms, the maximum population size is not determined since it is determined indirectly by a combination of possible values of the design variables, i.e., 3^{Levels} Where levels are 90, 18, 9, and 6 for the step size 1, 5, 10, and 15, respectively, for this reason, it can be observed that reducing computing time when step size rises. Thus, in the context, a rise in the step size implies reducing the number of levels.

5.2 Graphite/Epoxy

In the following, another orthotropic reinforced composite Graphite/Epoxy is considered for this study. The properties of the Graphite/Epoxy can be given in terms of engineering parameters as $E_1 = 181$ GPa, $E_2 = 10.3$ GPa, $E_3 = 10.3$ GPa, $G_{12} = 7.17$ GPa, $G_{13} = 7$ GPa, $G_{23} = 3$ GPa, $\nu_{12} = 0.28$, $\nu_{13} = 0.27$, $\nu_{23} = 0.6$. The compliance matrix can be computed in terms of engineering.

$$S = \begin{bmatrix} 1/E_1 & -\nu_{12}/E_1 & -\nu_{13}/E_1 & 0 & 0 & 0 \\ -\nu_{12}/E_1 & 1/E_2 & -\nu_{23}/E_2 & 0 & 0 & 0 \\ -\nu_{13}/E_1 & -\nu_{23}/E_2 & 1/E_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/G_{23} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/G_{31} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/G_{12} \end{bmatrix} = \begin{bmatrix} 0.006 & -0.002 & -0.012 & 0 & 0 & 0 \\ -0.002 & 0.097 & -0.058 & 0 & 0 & 0 \\ -0.002 & -0.058 & 0.097 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.333 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.143 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.140 \end{bmatrix}$$

Next, the constitutive matrix is computed as the inverse of the compliance matrix as follows-

$$C = S^{-1} = \begin{bmatrix} 1.850 & 0.073 & 0.072 & 0 & 0 & 0 \\ 0.073 & 0.164 & 0.099 & 0 & 0 & 0 \\ 0.072 & 0.099 & 0.164 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.030 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.070 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.072 \end{bmatrix}$$

And substituted into the objective function (16).

The principal strain values introduced above are used. ($\varepsilon_I = 8$, $\varepsilon_{II} = -7$, $\varepsilon_{III} = -6$)

In addition, in Table 5.4, optimal Euler angles and minimum strain energy density in GPa are listed corresponding to the applied methods.

Table 5.4 Graphite/Epoxy, solutions using GA+Gradient, GA, PSO, and integer variables GA

Method	Min. strain energy density (GPa)	θ_1^*	θ_2^*	θ_3^*
GA+Gradient	964.3666	$\pi/2$ (90°)	0.8700 (49.847°)	$\pi/2$ (90°)
GA	964.3674	$\pi/2$ (90°)	0.8702 (49.859°)	$\pi/2$ (90°)
PSO	964.3666	$\pi/2$ (90°)	0.8700 (49.847°)	$\pi/2$ (90°)
GA, Integer, 1 degree	964.4766	90°	50°	90°
GA, Integer, 5 degrees	964.4766	90°	10°	90°
GA, Integer, 10 degrees	964.4766	90°	50°	90°
GA, Integer, 15 degrees	997.0620	45°	75°	15°

Regarding the real value-based optimization, the results obtained using HGA (GA+Gradient), GA, and PSO methods are presented in excellent agreement [50]. In HGA (GA+Gradient) and PSO methods, the results obtained in several runs coincide. However, for GA, the case is quite different, where each run may lead to different results. Usually, perturbation occurs near the global optimum, but GA converges to the local optimum in some cases.

Considering integer optimization, step size 1, 5, and 10 degrees are the nearest possible integer values to real value solution. In the case of step size of Euler angles 15 degrees, a different solution has been found. The values of design variables vary significantly, yet the change in the objective function is not substantial (3-4% deviation). This result confirms an acknowledged fact that, in general, the integer solution cannot be deduced

from the real value solution by rounding it to the nearest integer. It is essential to run several times for the integer GA to achieve global optimum.

Furthermore, four different equivalent solutions can be derived based on one global extreme (for real value solutions) to save computing time. These solutions demonstrate a preferable outcome due to the symmetry of the objective function.

Table 5.5 Graphite/Epoxy, four equivalent optimal solutions.

Method	Min. strain energy density (GPa)	θ_1^*	θ_2^*	θ_3^*
GA+Gradient	964.3666	$\pi/2$ (90°)	0.870 (49.847°)	$\pi/2$ (90°)
GA+Gradient	964.3666	$\pi/2$ (90°)	$\pi-0.870$ (180-49.847°)	$\pi/2$ (90°)
GA+Gradient	964.3666	$\pi/2$ (90°)	0.870 (49.847°)	$\pi/2$ (90°)
GA+Gradient	964.3666	$\pi/2$ (90°)	$\pi-0.870$ (180-49.847°)	$\pi/2$ (90°)

In table 5.5 are given results for HGA (GA+Gradient) method, but these rules are valid for real value variable-based solutions.

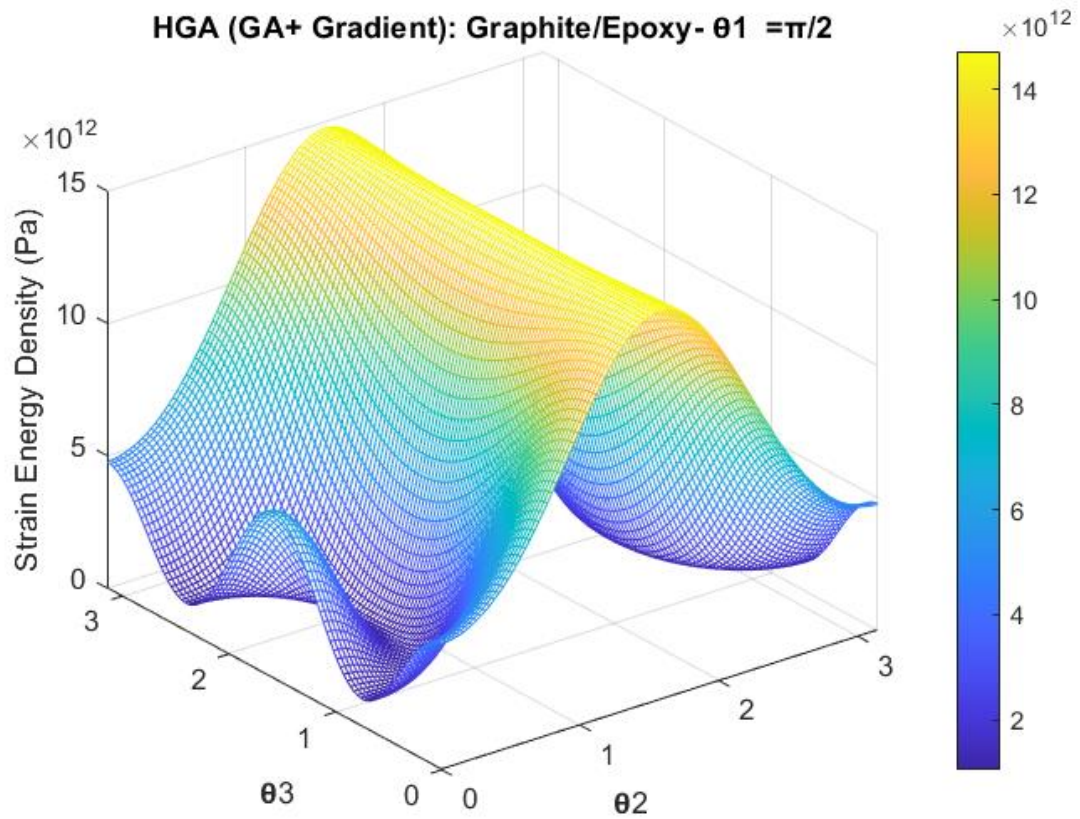


Figure 5.4 Graphite/Epoxy, distribution of the strain energy as function of Euler angles θ_2 and θ_3 (θ_1 is constant)

Figure 5.4 illustrates the strain energy density for the composite Graphite/Epoxy by fixing the Euler angle $\theta_1 = \frac{\pi}{2}$ to perceive the energy distribution behavior in 3D. However, the original objective function contains three design parameters, and the result is computed in 4D. For this plot, the energy density function is a function of the Euler angle θ_2 and θ_3 . It is also apparent from the figure the presence of a number of local extremes.

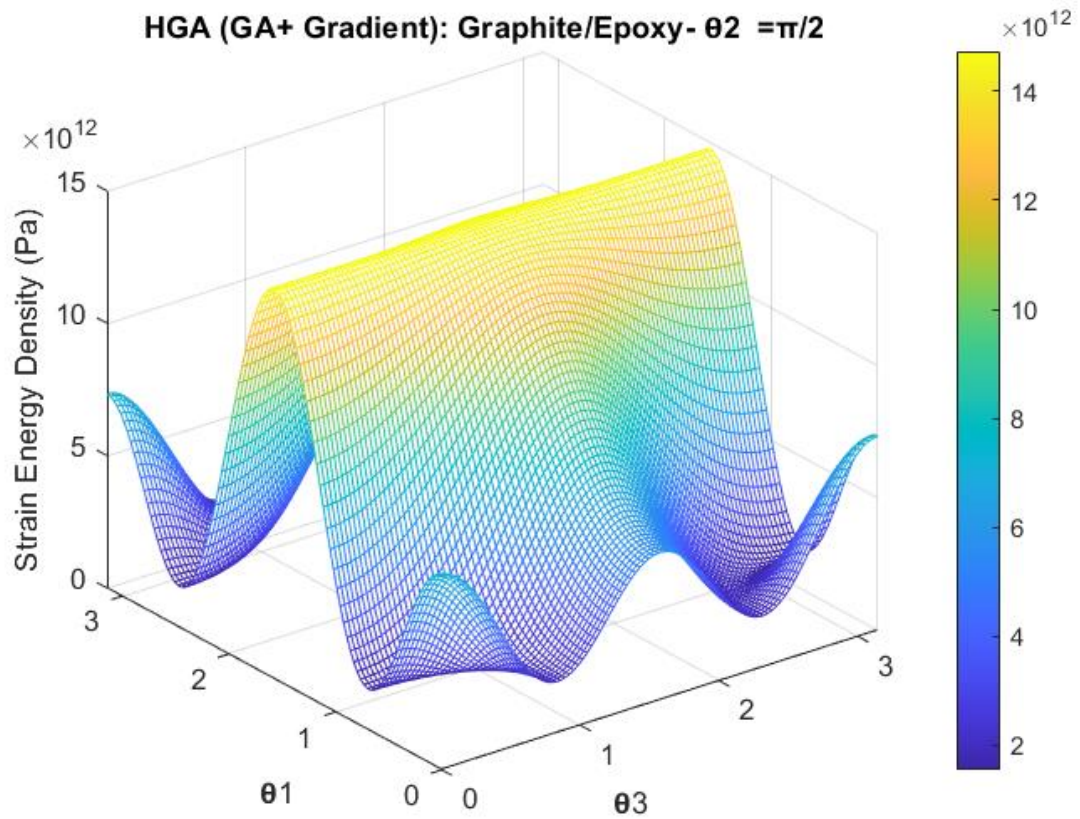


Figure 5.5 Graphite/Epoxy, distribution of the strain energy as function of Euler angles θ_3 and θ_1 (θ_2 is constant)

Figure 5.5 represents the strain energy density by fixing the Euler angle $\theta_2 = \frac{\pi}{2}$ to comprehend the energy distribution behavior in 3D. Whereas the initial objective function contains three design parameters, and the result is derived in 4D. For this plot, the energy density function is considered as a function of the Euler angle θ_3 and θ_1 . Similarly to Figure 5.4, it is also apparent from the figure that the presence of a number of local extremes and complexity of the designed problem.

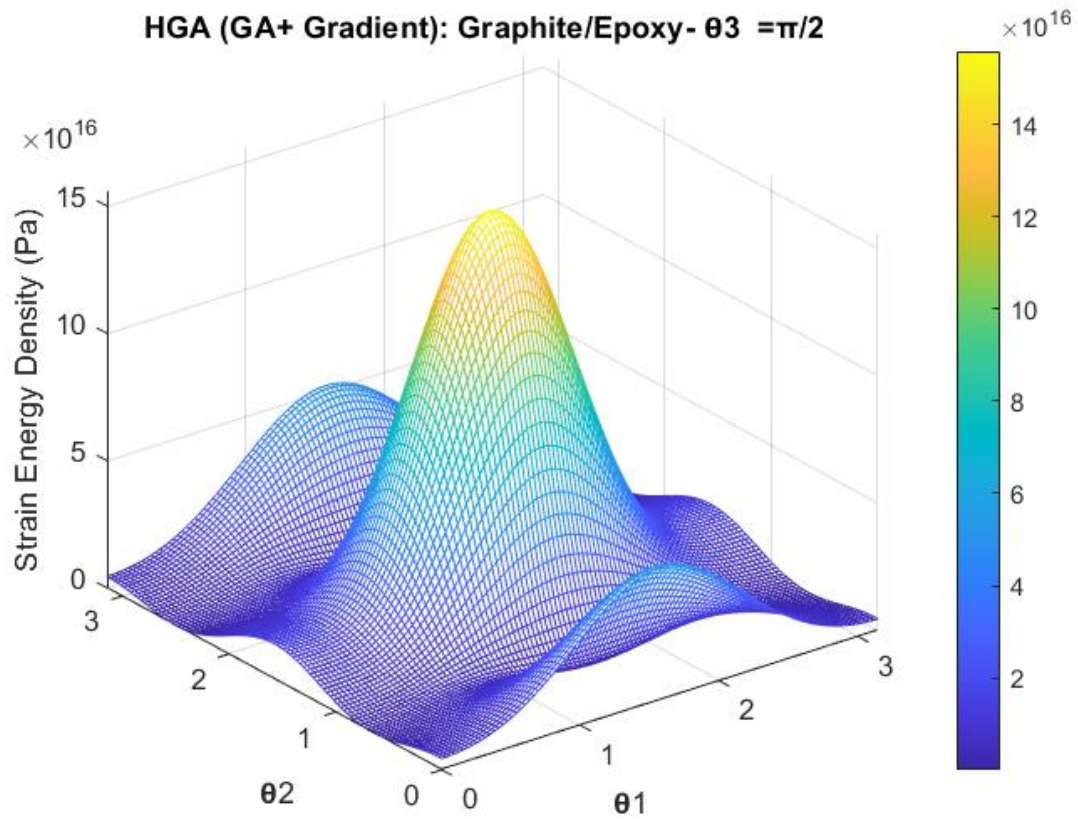


Figure 5.6 Graphite/Epoxy, distribution of the strain energy as function of Euler angles θ_1 and θ_2 (θ_3 is constant)

Figure 5.6 depicts the strain energy density considering the Euler angle $\theta_3 = \frac{\pi}{2}$ to observe the energy distribution behavior in 3D. For this plot, the energy density function is a function of the Euler angle θ_1 and θ_2 . Nevertheless, the original objective function contains three design parameters, and the result is calculated in 4D. The presence of several local extremes can also be perceptible, like in Figure 5.5 and Figure 5.6. for this material.

The computing times for each calculated methods are presented in Table 5.6

Table 5.6 Graphite/Epoxy, computing time

Method	Computing Time (Sec.)
GA+Gradient	0.6027
GA	2.0953
PSO	1.3021
GA, Integer, 1 degree	1.2284
GA, Integer, 5 degrees	0.7916
GA, Integer, 10 degrees	0.7389
GA, Integer, 15 degrees	0.7106

From Table 5.6, it can be observed that the HGA (GA+Gradient) algorithm appears to be the fastest, and the GA algorithm is the slowest among the real coded algorithms. In this case, PSO is slower than the HGA and faster than pure GA. Though PSO executes global and local searches concurrently, the gradient method, for this material, performed quite faster, combining with pure GA as HGA to reach global extremes. Moreover, the population size 200 and number of generations 150 have been considered regarding the real coded algorithms. Detailed MATLAB code is available in Appendix 1 and 2.

In integer coded algorithms, the maximum population size is given by 3^{Levels} , where levels values are 90, 18, 9, and 6 for the step size 1, 5, 10, and 15, respectively. Thus, reducing computing time can be followed from Table 5.6 when step size and max population size reduce.

In this chapter, unconstrained optimization has been examined considering both real and integer variables. For real coded variables, three different algorithms – GA, HGA, and PSO applied on the two different materials and regarding accuracy and computation time PSO algorithm performed better in a similar condition and framework. Additionally, for the designed integer variable, GA has been applied. The results revealed that the accuracy decreased as the step size increases, but the computation time reduced.

SUMMARY

The topic of the thesis is "Numerical algorithms for the orientational design of 3D orthotropic materials ". The optimal material orientation problem is one of the critical issues in the design of advanced composite materials. Although the problem is currently more studied in 2D anisotropic materials, 3D materials are yet less studied, especially orthotropic and general anisotropic materials.

The study's principal goal is to analyze and compare different optimization algorithms for solving 3D optimal material orientation problems to determine the most computationally cost-effective algorithms for particular problems or classes of problems.

The thesis is divided into five chapters. In chapter 1, an introduction to composite materials and optimization methods used is given. Chapter 2 presents the general background/literature overview covering the general formulation of the constrained multicriteria optimization problem, optimal material orientation problem in the composite structure, and techniques applied in previous studies for orientational design. In chapter 3 is given a detailed description of the four optimization methods used in the current study (gradient/steepest descent & Newton methods, genetic algorithm, particle swarm optimization algorithm, and Lagrange multipliers method). In chapter 4, the algorithms mentioned above are utilized for optimal material orientation of 3D linear elastic orthotropic materials. Finally, the obtained numerical results are discussed and analyzed in chapter 5, considering two different E-Glass/Vinilester and Graphite/Epoxy materials.

The main goal of the study is achieved. The results obtained using the above-considered algorithms are found to be in good agreement with each other in the case of both materials considered (E-Glass/Vinilester and Graphite/Epoxy). The results are in agreement also those available in the literature for the GA+Gradient method.

To sum, for the real variable design and considered problem or class of problems (orientational design of 3D materials), the most effective appears the PSO algorithm.

In the case of integer variable design, the GA algorithm was employed. The results obtained were close to that of real variable design when the step size of the Euler angle was 1 degree. However, the approach introduced in this study is novel as the results covering integer variable optimal design of 3D linear elastic orthotropic materials are not addressed in the previous literature.

KOKKUVÕTE

Antud magistritöö teemaks on „Numbrilised algoritmid 3D ortotroopse materjali optimaalse orientatsiooni määramiseks. Kaasaegsete komposiitmaterjalide projekteerimisel on optimaalne materjali orientatsioon üks võtmeküsimusi. Optimaalse orientatsiooni probleemi on uuritud peamiselt 2D anisotroopsete materjalide korral, 3D materjale on vähem uuritud, eriti ortotroopseid ja üldisi anisotroopseid.

Töö peamiseks eesmärgiks on analüüsida ja võrrelda erinevaid optimeerimise algoritme 3D ortotroopse materjali optimaalse orientatsiooni määramiseks, leida väikseima arvutusmahuga algoritm antud probleemi või probleemide klassi jaoks.

Töö on jaotatud viide peatükki. Esimene peatükk tutvustab komposiitmaterjale ja neile rakendatud optimeerimise algoritme. Teine peatükk sisaldab kitsendustega multikriteeriaalse optimeerimisülesande üldist formulatsiooni, põhjalikumalt kirjanduse ülevaadet materjali optimaalse orientatsiooni määramisest ning kasutatud optimeerimise meetoditest. Kolmanda peatükis on toodud detailsem kirjeldus antud töös kasutatud optimeerimismeetoditele (gradiendi meetod/kiireima languse & Newtoni meetod, geneetiline algoritm, osakeste parve algoritm ja Lagrange kordajate meetod). Neljandas peatükis on rakendatud eespooltoodud meetodeid 3D ortotroopse materjali optimaalse orientatsiooni määramiseks. Saadud numbrilised tulemused on kirjeldatud viiendas peatükis, kus on vaadeldud kahte materjali (E-Glass/Vinilester ja Graphite/Epoxy)

Töö eesmärk on täidetud. Erinevate algoritmide abil saadud tulemused on heas kooskõlas omavahel mõlema vaadeldud materjali korral ja on kooskõlas ka kirjanduses olemasolevate GA+gradiendi meetodi abil saadud tulemustega.

Peamine järeldus on et reaalarvulise lahendi korral osutus antud probleemi või probleemide klassi jaoks (3D ortotroopsed materjalid) parimaks osakeste parve algoritm.

Täisarvuliste muutujate korral on rakendatud GA algoritmi. Väikese sammu suuruse korral (1 kraad) saadud tulemused osutusid heas kooskõlas olevateks reaalarvulise lahendusega. Töö autorile teadaolevalt pole kirjanduses antud probleemi täisarvuliste lahendite korral uuritud.

LIST OF REFERENCES

1. Akbulut, M., & Sonmez, F. O. (2008). Optimum design of composite laminates for minimum thickness. *Computers & Structures*, 86(21–22), 1974–1982. <https://doi.org/10.1016/j.compstruc.2008.05.003>
2. Almeida, F., & Awruch, A. (2009). Design optimization of composite laminated structures using genetic algorithms and finite element analysis. *Composite Structures*, 88(3), 443–454. <https://doi.org/10.1016/j.compstruct.2008.05.004>
3. Arram, A., & Ayob, M. (2019). A novel multi-parent order crossover in genetic algorithm for combinatorial optimization problems. *Computers & Industrial Engineering*, 133, 267–274. <https://doi.org/10.1016/j.cie.2019.05.012>
4. Banichuk N.V. (1983) Optimization of Anisotropic Properties of Elastic Bodies. In: Haug E.J. (eds) Problems and Methods of Optimal Structural Design. *Mathematical Concepts and Methods in Science and Engineering*, vol 26. Springer, Boston, MA. https://doi.org/10.1007/978-1-4613-3676-1_5
5. Brandmaier, H. E. (1970). Optimum Filament Orientation Criteria. *Journal of Composite Materials*, 4(3), 422–425. <https://doi.org/10.1177/002199837000400314>
6. Bruyneel M, Fleury C (2002) Composite structures optimization using sequential convex programming. *Adv Eng Softw* 33(7-10):697711. [https://doi.org/10.1016/S09659978\(02\)00053-4](https://doi.org/10.1016/S09659978(02)00053-4)
7. Chakraborty, R. C. (2010, June 1). Fundamentals of Genetic Algorithms [Slides]. Myreaders. http://www.myreaders.info/09_Genetic_Algorithms.pdf (Accessed 12.03.2021)
8. Cheng H, Kikuchi N, Ma Z (1994) An improved approach for determining the optimal orientation of orthotropic material. *Struct Optimization* 8(2-3):101–112. <https://doi.org/10.1007/BF01743305>
9. Chunka, C., Goswami, R. S., & Banerjee, S. (2018). An efficient mechanism to generate dynamic keys based on genetic algorithm. *Security and Privacy*, e37. <https://doi.org/10.1002/spy2.37>
10. Christensen, P. W., & Klarbring, A. (2008). *An Introduction to Structural Optimization (Solid Mechanics and Its Applications, 153)* (2009th ed.). Springer.
11. Clerc, M. (1999). The swarm and the queen: towards a deterministic and adaptive particle swarm optimization. *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*. Published. <https://doi.org/10.1109/cec.1999.785513>

12. Cowin, S. C. (1994). Optimization of the strain energy density in linear anisotropic elasticity. *Journal of Elasticity*, 34(1), 45–68.
<https://doi.org/10.1007/bf00042425>
13. Daniel, I. O. M. I. (2021). *Engineering Mechanics Of Composite Material* (2nd ed.). Oxford University Press.
14. Díaz, A. R., & Bendsøe, M. P. (1992). Shape optimization of structures for multiple loading conditions using a homogenization method. *Structural Optimization*, 4(1), 17–22. <https://doi.org/10.1007/bf01894077>
15. Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 39–43.
<https://doi.org/10.1109/mhs.1995.494215>
16. El-mihoub, T. A., Alan Hopgood, A., Nolle, L., & Alan, B. (2006). Hybrid genetic algorithms: A review. *Engineering Letters*, 13(2).
http://www.engineeringletters.com/issues_v13/issue_2/EL_13_2_11.pdf
17. Fourie, P. C., & Groenwold, A. A. (2001). Particle swarms in topology optimization. *Proceedings of the Fourth World Congress of Structural and Multidisciplinary Optimization*, Liaoning Electronic Press, 1771–1776. (17)
18. Fourie, P., & Groenwold, A. (2002). The particle swarm optimization algorithm in size and shape optimization. *Structural and Multidisciplinary Optimization*, 23(4), 259–267. <https://doi.org/10.1007/s00158-002-0188-0>
19. Ganguli, R. (2013). Optimal design of composite structures : a historical review. *Journal of the Indian institute of science*, 93(4), 557-570.
<https://hdl.handle.net/10356/103207>
20. Gea, H., & Luo, J. (2004). On the stress-based and strain-based methods for predicting optimal orientation of orthotropic materials. *Structural and Multidisciplinary Optimization*, 26(3–4), 229–234.
<https://doi.org/10.1007/s00158-003-0348-x>
21. Ghiasi, H., Pasini, D., & Lessard, L. (2009). Optimum stacking sequence design of composite materials Part I: Constant stiffness design. *Composite Structures*, 90(1), 1–11. <https://doi.org/10.1016/j.compstruct.2009.01.006>
22. Goldberg, D. E. (1988). *Genetic Algorithms in Search, Optimization and Machine Learning* (13th ed.). Addison-Wesley Professional.
23. Guo, Z., Caner, F., Peng, X., & Moran, B. (2008). On constitutive modelling of porous neo-Hookean composites. *Journal of the Mechanics and Physics of Solids*, 56(6), 2338–2357. <https://doi.org/10.1016/j.jmps.2007.12.007>

24. GUO, Z., & CANER, F. C. (2010). MECHANICAL BEHAVIOUR OF TRANSVERSELY ISOTROPIC POROUS NEO-HOOKEAN SOLIDS. *International Journal of Applied Mechanics*, 02(01), 11–39. <https://doi.org/10.1142/s1758825110000494>
25. Gupta, A. K., & Kumar, L. (2007). Thermal effect on vibration of non-homogenous visco-elastic rectangular plate of linearly varying thickness. *Meccanica*, 43(1), 47–54. <https://doi.org/10.1007/s11012-007-9093-3>
26. Gupta, A. K., & Kaur, H. (2008). Study of the effect of thermal gradient on free vibration of clamped visco-elastic rectangular plates with linearly thickness variation in both directions. *Meccanica*, 43(4), 449–458. <https://doi.org/10.1007/s11012-008-9110-1>
27. Gürdal, Z., Haftka, R. T., & Hajela, P. (1999). *Design and Optimization of Laminated Composite Materials* (1st ed.). New York: John Wiley & Sons.
28. Hill, R. (1998). *The Mathematical Theory of Plasticity* (Oxford Classic Texts in the Physical Sciences) (2nd ed.). Oxford University Press.
29. Holland, J.H. (1975) *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor. (2nd Edition, MIT Press, 1992.)
30. Hull, D., Clyne, T. W., & Clarke, D. R. (Editor). (1996). *An Introduction to Composite Materials: 2nd (Second) edition*. Cambridge University Press.
31. *Introduction to Particle Swarm Optimization*. (2010). Mnemstudio.Org. <http://www.mnemstudio.org/particle-swarm-introduction.htm> (Accessed 10.04.2021)
32. Jemioło, S., & Telega, J. (2001). Transversely isotropic materials undergoing large deformations and application to modelling of soft tissues. *Mechanics Research Communications*, 28(4), 397–404. [https://doi.org/10.1016/s0093-6413\(01\)00189-6](https://doi.org/10.1016/s0093-6413(01)00189-6)
33. Jennings, P. C., Lysgaard, S., Hummelshøj, J. S., Vegge, T., & Bligaard, T. (2019). Genetic algorithms for computational materials discovery accelerated by machine learning. *Npj Computational Materials*, 5(1). <https://doi.org/10.1038/s41524-019-0181-4>
34. Jones, R. M. (1998). *Mechanics Of Composite Materials* (Materials Science & Engineering Series) (2nd ed.). CRC Press
35. Katoch, S., Chauhan, S. S., & Kumar, V. (2020). A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80(5), 8091–8126. <https://doi.org/10.1007/s11042-020-10139-6>
36. Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*. Published. <https://doi.org/10.1109/icnn.1995.488968>

37. Kennedy, J., & Spears, W. (1998). Matching algorithms to problems: an experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator. 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360). Published.
<https://doi.org/10.1109/icec.1998.699326>
38. Khandan, R., Noroozi, S., Sewell, P., Vinney, J., & Koohgilani, M. (2012). Optimum Design of Fibre Orientation in Composite Laminate Plates for Out-Plane Stresses. *Advances in Materials Science and Engineering*, 2012, 1–11.
<https://doi.org/10.1155/2012/232847>
39. Larsen, U., Signund, O., & Bouwsta, S. (1997). Design and fabrication of compliant micromechanisms and structures with negative Poisson's ratio. *Journal of Microelectromechanical Systems*, 6(2), 99–106.
<https://doi.org/10.1109/84.585787>
40. Lellep, J., & Majak, J. (1997). On optimal orientation of nonlinear elastic orthotropic materials. *Structural Optimization*, 14(2–3), 116–120.
<https://doi.org/10.1007/bf01812513>
41. Lellep, J., & Majak, J. (2000). Nonlinear constitutive behavior of orthotropic materials. *Mechanics of Composite Materials*, 36(4), 261–266.
<https://doi.org/10.1007/bf02262803>
42. Lindgaard, E., & Lund, E. (2010). Optimization formulations for the maximum nonlinear buckling load of composite structures. *Structural and Multidisciplinary Optimization*, 43(5), 631–646. <https://doi.org/10.1007/s00158-010-0593-8>
43. Lopez, R., Luersen, M., & Cursi, E. (2009). Optimization of laminated composites considering different failure criteria. *Composites Part B: Engineering*, 40(8), 731–740.
<https://doi.org/10.1016/j.compositesb.2009.05.007>
44. Lubliner, J. (2008). *Plasticity Theory (Dover Books on Engineering)* (Illustrated ed.). Dover Publications.
45. Lund, E. (2009). Buckling topology optimization of laminated multi-material composite shell structures. *Composite Structures*, 91(2), 158–167.
<https://doi.org/10.1016/j.compstruct.2009.04.046>
46. Maalawi, K. Y., & Badr, M. A. (2009). Design Optimization of Mechanical Elements and Structures: a Review with Application. *INSInet Publication*, 5(2), 221–231. https://www.researchgate.net/profile/Karam-Maalawi/publication/228348554_Design_Optimization_of_Mechanical_Elements_and_Structures_a_Review_with_Application/links/0fcfd5075b272ca1f8000000

- [/Design-Optimization-of-Mechanical-Elements-and-Structures-a-Review-with-Application.pdf](#)
47. Maalawi, K. (2019). Introductory Chapter: An Introduction to the Optimization of Composite Structures. *Optimum Composite Structures*. Published. <https://doi.org/10.5772/intechopen.81165>
 48. Majak, J., Toompalu, S., & Pohlak, M. (2008). Material parameters identification by use of hybrid GA. *Journal of Achievements of Materials and Manufacturing Engineering*, 27(1), 63–66. <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.533.6993&rep=rep1&type=pdf>
 49. Majak, J., & Pohlak, M. (2009). Optimal material orientation of linear and nonlinear elastic 3D anisotropic materials. *Meccanica*, 45(5), 671–680. <https://doi.org/10.1007/s11012-009-9262-7>
 50. Majak, J., & Pohlak, M. (2010). Decomposition method for solving optimal material orientation problems. *Composite Structures*, 92(8), 1839–1845. <https://doi.org/10.1016/j.compstruct.2010.01.015>
 51. Marini, F., & Walczak, B. (2015). Particle swarm optimization (PSO). A tutorial. *Chemometrics and Intelligent Laboratory Systems*, 149, 153–165. <https://doi.org/10.1016/j.chemolab.2015.08.020>
 52. Muc, A. (2007). Optimal design of composite multilayered plated and shell structures. *Thin-Walled Structures*, 45(10–11), 816–820. <https://doi.org/10.1016/j.tws.2007.08.042>
 53. Mishra, S. K., & Ram, B. (2019). *Introduction to Unconstrained Optimization with R* (1st ed. 2019 ed.). Springer. <https://doi.org/10.1007/978-981-15-0894-3>
 54. Narayana Naik, G., Gopalakrishnan, S., & Ganguli, R. (2008). Design optimization of composites using genetic algorithms and failure mechanism based failure criterion. *Composite Structures*, 83(4), 354–367. <https://doi.org/10.1016/j.compstruct.2007.05.005>
 55. Omkar, S., Senthilnath, J., Khandelwal, R., Narayana Naik, G., & Gopalakrishnan, S. (2011). Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures. *Applied Soft Computing*, 11(1), 489–499. <https://doi.org/10.1016/j.asoc.2009.12.008>
 56. Passaro, A., & Starita, A. (2008). Particle Swarm Optimization for Multimodal Functions: A Clustering Approach. *Journal of Artificial Evolution and Applications*, 2008, 1–15. <https://doi.org/10.1155/2008/482032>
 57. Pedersen, P. (1989). On optimal orientation of orthotropic materials. *Structural Optimization*, 1(2), 101–106. <https://doi.org/10.1007/bf01637666>

58. Pedersen, P. (1993). *Optimal Design with Advanced Materials*. Elsevier Science. 51–66.
59. Peng, X. Q., Guo, Z. Y., & Moran, B. (2005). An Anisotropic Hyperelastic Constitutive Model With Fiber-Matrix Shear Interaction for the Human Annulus Fibrosus. *Journal of Applied Mechanics*, 73(5), 815–824.
<https://doi.org/10.1115/1.2069987>
60. Petrovic, M., Nomura, T., Yamada, T., Izui, K., & Nishiwaki, S. (2017). Orthotropic material orientation optimization method in composite laminates. *Structural and Multidisciplinary Optimization*, 57(2), 815–828.
<https://doi.org/10.1007/s00158-017-1777-2>
61. Rovati, M., & Taliercio, A. (1991). Optimal Orientation of the Symmetry Axes of Orthotropic 3-D Materials. *Lecture Notes in Engineering*, 127–134.
https://doi.org/10.1007/978-3-642-84397-6_12
62. Rovati, M., & Taliercio, A. (2003). Stationarity of the strain energy density for some classes of anisotropic solids. *International Journal of Solids and Structures*, 40(22), 6043–6075. [https://doi.org/10.1016/s0020-7683\(03\)00371-8](https://doi.org/10.1016/s0020-7683(03)00371-8)
63. Sahadevan, V., Bonnefon, Y., & Edwards, T. (2006). A Meta-Heuristic Based Weight Optimisation for Composite Wing Structural Analysis. *Applied Mechanics and Materials*, 5–6, 305–314.
<https://doi.org/10.4028/www.scientific.net/amm.5-6.305>
64. Seregin, G., & Troitskii, V. (1981). On the best position of elastic symmetry planes in an orthotropic body. *Journal of Applied Mathematics and Mechanics*, 45(1), 139–142. [https://doi.org/10.1016/0021-8928\(81\)90022-8](https://doi.org/10.1016/0021-8928(81)90022-8)
65. Seyranian, A. P., Lund, E., & Olhoff, N. (1994). Multiple eigenvalues in structural optimization problems. *Structural Optimization*, 8(4), 207–227.
<https://doi.org/10.1007/bf01742705>
66. Shi, Y., & Eberhart, R. (1998a). A modified particle swarm optimizer. *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, 69–73.
<https://doi.org/10.1109/icec.1998.699146>
67. Shi, Y., & Eberhart, R. C. (1998b). Parameter selection in particle swarm optimization. *Lecture Notes in Computer Science*, 1447, 591–600.
<https://doi.org/10.1007/bfb0040810>
68. Sonmez, F. O. (2016). Optimum Design of Composite Structures: A Literature Survey (1969–2009). *Journal of Reinforced Plastics and Composites*, 36(1), 3–39. <https://doi.org/10.1177/0731684416668262>

69. Suzuki, K., & Kikuchi, N. (1991). A homogenization method for shape and topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 93(3), 291–318. [https://doi.org/10.1016/0045-7825\(91\)90245-2](https://doi.org/10.1016/0045-7825(91)90245-2)
70. Tagarielli, V., Deshpande, V., Fleck, N., & Chen, C. (2005). A constitutive model for transversely isotropic foams, and its application to the indentation of balsa wood. *International Journal of Mechanical Sciences*, 47(4–5), 666–686. <https://doi.org/10.1016/j.ijmecsci.2004.11.010>
71. The University of Texas at Austin. (2020, February 27). Methods for Local Optimization. Freshman Research Initiative: Computational Materials. http://fri.oden.utexas.edu/fri/Labs_2020/lab2/lab2B.php
72. Venter, G., & Sobieszczanski-Sobieski, J. (2003). Particle Swarm Optimization. *AIAA Journal*, 41(8), 1583–1589. <https://doi.org/10.2514/2.2111>
73. Walker, M., & Smith, R. (2003). A technique for the multiobjective optimisation of laminated composite structures using genetic algorithms and finite element analysis. *Composite Structures*, 62(1), 123–128. [https://doi.org/10.1016/s0263-8223\(03\)00098-9](https://doi.org/10.1016/s0263-8223(03)00098-9)
74. Wang, J., & Karihaloo, B. L. (1996). Optimum In Situ Strength Design of Composite Laminates. Part I: In Situ Strength Parameters. *Journal of Composite Materials*, 30(12), 1314–1337. <https://doi.org/10.1177/002199839603001202>
75. Zhurov, A. I., Limbert, G., Aeschlimann, D. P., & Middleton, J. (2007). A constitutive model for the periodontal ligament as a compressible transversely isotropic visco-hyperelastic tissue. *Computer Methods in Biomechanics and Biomedical Engineering*, 10(3), 223–235. <https://doi.org/10.1080/13639080701314894>

APPENDICES

The below MATLAB codes explicitly reveal the details of the proposed objective function and design variable to address the optimal orientational design problem for unconstrained (Real and Integer) and constrained optimization.

Appendix 1

Unconstrained Optimization (Real variable solution)

Genetic Algorithm (GA)

```
clear all
tic

%mat1 Graphite/Epoxy

E1=181e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3e9; G31=7e9; ny12=0.28;
ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester

% E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

A=[]; b=[];
Aeq=[]; beq=[];
lb=[0,0,0];
%ub=[pi,pi,pi];
ub=[pi/2,pi/2,pi/2];
%ub=[90,90,90];
numOfVar=3;
ea(1)=8; ea(2)=-7; ea(3)=-6; % principal strains

intcon=[]
options= optimoptions('ga','Generations',150,'PopulationSize', 200);

[x,Uval,exitf] =
ga(@(x)pr_ob1(x,C,ea),numOfVar,A,b,Aeq,beq,lb,ub,[],intcon, options);
format long
x=x
U=Uval
Computing_Time=toc
%d1=pi/2;
d2=pi/2;
%d3=pi/2;
D1=linspace(0,pi,100);
%D2=linspace(0,pi,100);
D3=linspace(0,pi,100);
%[d2,d3] = meshgrid(D2,D3);
%[d1,d2] = meshgrid(D1,D2);
```

```

d3,d1] = meshgrid(D3,D1);
Fu=plot1 (d1,d2,d3,C,ea);
%mesh (d2,d3,Fu)
mesh (d3,d1,Fu)
%mesh (d1,d2,Fu)

```

HGA (GA+Gradient)

```

clear all
tic

%mat1 Graphite/Epoxy

%E1=181.0e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3.0e9; G31=7.0e9;
ny12=0.28; ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester

E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

% For Local optima analyais(fmincon)

A=[]; b=[];
Aeq=[]; beq=[];
lb=[0.001,0.001,0.01]; % trivial solution x=0 is omitted
%ub=[pi,pi,pi];
ub=[pi/2,pi/2,pi/2];
%ub=[pi/4,pi/4,pi/4];
numOfVar=3;
ea(1)=8; ea(2)=-7; ea(3)=-6; % principal strains

%x0=[0.9, 0.5, 0.8];
%x0=[0.12, 0.09, 0.098];
%x0=[0.02, 0.07, 0.055];
x0=[0.6, 1, 1.5];
options= optimoptions('ga','Generations',150,'PopulationSize', 200);
[x,Uval,exitf] =
ga(@ (x)pr_ob1 (x,C,ea) ,numOfVar,A,b,Aeq,beq,lb,ub, [],options)

x0=x

[x,Uval,exitf] = fmincon (@ (x)pr_ob1 (x,C,ea) ,x0,A,b,Aeq,beq,lb,ub, []);
format long;
x=x
U=Uval
Computing_Time=toc
d1=pi/2;
%d2=pi/2;
%d3=pi/2;
%D1=linspace (0,pi,100);
D2=linspace (0,pi,100);
D3=linspace (0,pi,100);
[d2,d3] = meshgrid(D2,D3);
%[d1,d2] = meshgrid(D1,D2);

```



```

%[d3,d1] = meshgrid(D3,D1);
Fu=plot1(d1,d2,d3,C,ea);
mesh(d2,d3,Fu)
%mesh(d3,d1,Fu)
%mesh(d1,d2,Fu)

```

Particle Swarm Optimization (PSO)

```

clear all
tic

%mat1 Graphite/Epoxy

%E1=181e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3e9; G31=7e9;
ny12=0.28; ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester

E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

A=[]; b=[];
Aeq=[]; beq=[];
lb=[0,0,0];
%ub=[pi,pi,pi];
ub=[pi/2,pi/2,pi/2];
%ub=[pi/4,pi/4,pi/4];
numOfVar=3;
ea(1)=8; ea(2)=-7; ea(3)=-6; % principal strains

options =
optimoptions('particleswarm','SwarmSize',200,'MaxIterations',150);
% options =
optimoptions('particleswarm','SwarmSize',100,'HybridFcn',@fmincon);
[x,Uval,exitf] =
particleswarm(@ (x)pr_ob1(x,C,ea),numOfVar,lb,ub,options);
x=x
U=Uval
Computing_Time=toc

```

Objective functions

```
function [U] = pr_ob1(x,C,ea)
d1=x(1);
d2=x(2);
d3=x(3);
e1=ea(1);e2=ea(2); e3=ea(3);
aep12=-2*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)^2-((e1-e2)*cos(d3)^2
e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*sin(d1)*cos(d1)+sin(d3)*cos(d2)*cos(d3)*(e1-e2);
aep31=(sin(d1))*((e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)+sin(d3)*cos(d1)*cos(d3)*(e1-e2))*sin(d2);
aep23=-(-cos(d2))*((e1-e2)*cos(d3)^2-
e1+e3)*cos(d1)+sin(d1)*sin(d3)*cos(d3)*(e1-e2))*sin(d2);
aep11=((e1-e2)*cos(d3)^2-e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*cos(d1)^2-2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+((-
e1+e2)*cos(d3)^2+e1-e3)*cos(d2)^2+e3;
aep22=(-(cos(d2)^2+1)*(e1-e2)*cos(d3)^2+(e1-e3)*cos(d2)^2-
e2+e3)*cos(d1)^2+2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+(e1-
e2)*cos(d3)^2+e2;
aep33=cos(d3)^2*cos(d2)^2*e1-cos(d3)^2*cos(d2)^2*e2-
cos(d3)^2*e1+cos(d3)^2*e2-cos(d2)^2*e1+cos(d2)^2*e3+e1;
U=0.5*C(1,1)*aep11^2+ C(1,2)*aep11*aep22+
C(1,3)*aep11*aep33+0.5*C(2,2)*aep22^2+C(2,3)*aep22*aep33+0.5*C(3,3)*aep33
^2+2*C(6,6)*aep12^2+2*C(4,4)*aep23^2+2*C(5,5)*aep31^2;
End
```

Plot functions

```
function [U] = plot1(d1,d2,d3,C,e)
ea(1)=(e(1)+e(2))/2;
ea(2)=(e(2)-e(1))/2;
ea(3)=e(3)-ea(1);
aep12=(sin(d2).^2).*sin(2.*d1)*ea(3)/2+(cos(d2).*cos(2.*d1).*sin(2.*d3)+(
1+cos(d2).^2)*sin(2.*d1).*cos(2.*d3)/2)*ea(2);
aep31=sin(d2).*(sin(d1).*cos(d2)*ea(3)-
(cos(d2).*sin(d1).*cos(2.*d3)+sin(2.*d3).*cos(d1))*ea(2));
aep23=sin(d2).*(cos(d1).*cos(d2)*ea(3)-(cos(d2).*cos(d1).*cos(2.*d3)-
sin(2.*d3).*sin(d1))*ea(2));
aep11=sin(d2).^2*(1-
cos(2.*d1))*ea(3)+(cos(d2).*sin(2.*d1).*sin(2.*d3)+(cos(d2).^2*(1-
cos(2.*d1))-(1+cos(2.*d1))).*cos(2.*d3)/2)*ea(2)+ea(1);
aep22=sin(d2).^2*(1+cos(2.*d1))*ea(3)+(-
cos(d2).*sin(2.*d1).*sin(2.*d3)+(cos(d2).^2*(1+cos(2.*d1))-(1-
cos(2.*d1))).*cos(2.*d3)/2)*ea(2)+ea(1);

aep33=-sin(d2).^2.*(ea(3)-cos(2.*d3).*ea(2))+e(3);
U=0.5.*C(1,1).*aep11.^2+ C(1,2).*aep11.*aep22+
C(1,3).*aep11.*aep33+0.5.*C(2,2).*aep22.^2+C(2,3).*aep22.*aep33+0.5.*C(3,
3).*aep33.^2+2.*C(4,4).*aep23.^2+2.*C(5,5).*aep31.^2+2.*C(6,6).*aep12.^2;
end
```

Appendix 2

Unconstrained Optimization (Integer variable solution)

Step Size 1⁰

```
clear all
tic

%mat1 Graphite/Epoxy

%E1=181e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3e9; G31=7e9;
ny12=0.28; ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester

E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

A=[]; b=[];
Aeq=[]; beq=[];
lb=[0,0,0];
ub=[90,90,90];
numOfVar=3;
ea(1)=8; ea(2)=-7; ea(3)=-6; % principal strains

x0=[0.6, 0.15, 0.3];
intcon=[1,2,3]
[x,Uval,exitf] =
ga(@(x)pr_ob2Degrees(x,C,ea),numOfVar,A,b,Aeq,beq,lb,ub,[],intcon);
format long
x=x
U=Uval
Computing_Time=toc
```

Objective functions

```
function [U] = pr_ob2Degrees(x,C,ea)
d1=deg2rad(x(1));
d2=deg2rad(x(2));
d3=deg2rad(x(3));
e1=ea(1);e2=ea(2); e3=ea(3);
aep12=-2*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)^2-((e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*sin(d1)*cos(d1)+sin(d3)*cos(d2)*cos(d3)*(e1-e2);
aep31=(sin(d1))*((e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)+sin(d3)*cos(d1)*cos(d3)*(e1-e2))*sin(d2);
aep23=-(-cos(d2))*((e1-e2)*cos(d3)^2-
e1+e3)*cos(d1)+sin(d1)*sin(d3)*cos(d3)*(e1-e2))*sin(d2);
```

```

aep11=(( (e1-e2)*cos(d3)^2-e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*cos(d1)^2-2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+((-
e1+e2)*cos(d3)^2+e1-e3)*cos(d2)^2+e3;
aep22=(-(cos(d2)^2+1)*(e1-e2)*cos(d3)^2+(e1-e3)*cos(d2)^2-
e2+e3)*cos(d1)^2+2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+(e1-
e2)*cos(d3)^2+e2;
aep33=cos(d3)^2*cos(d2)^2*e1-cos(d3)^2*cos(d2)^2*e2-
cos(d3)^2*e1+cos(d3)^2*e2-cos(d2)^2*e1+cos(d2)^2*e3+e1;
U=0.5*C(1,1)*aep11^2+ C(1,2)*aep11*aep22+
C(1,3)*aep11*aep33+0.5*C(2,2)*aep22^2+C(2,3)*aep22*aep33+0.5*C(3,3)*aep33
^2+2*C(6,6)*aep12^2+2*C(4,4)*aep23^2+2*C(5,5)*aep31^2;
end

```

Step Size 5⁰

```

clear all
tic

%mat1 Graphite/Epoxy

%E1=181e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3e9; G31=7e9;
ny12=0.28; ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester

E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

A=[]; b=[];
Aeq=[]; beq=[];
lb=[0,0,0];
ub=[18,18,18];
numOfVar=3;
ea(1)=8; ea(2)=-7; ea(3)=-6; % principal strains

x0=[0.6, 0.15, 0.3];
intcon=[1,2,3]
[x,Uval,exitf] =
ga(@(x)pr_ob2Degrees5(x,C,ea),numOfVar,A,b,Aeq,beq,lb,ub,[],intcon);
format long
x=x
U=Uval
Computing_Time=toc

```

Objective functions

```
function [U] = pr_ob2Degrees5(x,C,ea)
d1=deg2rad(5*x(1));
d2=deg2rad(5*x(2));
d3=deg2rad(5*x(3));
e1=ea(1);e2=ea(2); e3=ea(3);
aep12=-2*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)^2-((e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*sin(d1)*cos(d1)+sin(d3)*cos(d2)*cos(d3)*(e1-e2);
aep31=(sin(d1))*((e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)+sin(d3)*cos(d1)*cos(d3)*(e1-e2))*sin(d2);
aep23=-(-cos(d2))*((e1-e2)*cos(d3)^2-
e1+e3)*cos(d1)+sin(d1)*sin(d3)*cos(d3)*(e1-e2))*sin(d2);
aep11=((e1-e2)*cos(d3)^2-e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*cos(d1)^2-2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+((-
e1+e2)*cos(d3)^2+e1-e3)*cos(d2)^2+e3;
aep22=(-(cos(d2)^2+1)*(e1-e2)*cos(d3)^2+(e1-e3)*cos(d2)^2-
e2+e3)*cos(d1)^2+2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+(e1-
e2)*cos(d3)^2+e2;
aep33=cos(d3)^2*cos(d2)^2*e1-cos(d3)^2*cos(d2)^2*e2-
cos(d3)^2*e1+cos(d3)^2*e2-cos(d2)^2*e1+cos(d2)^2*e3+e1;
U=0.5*C(1,1)*aep11^2+ C(1,2)*aep11*aep22+
C(1,3)*aep11*aep33+0.5*C(2,2)*aep22^2+C(2,3)*aep22*aep33+0.5*C(3,3)*aep33
^2+2*C(6,6)*aep12^2+2*C(4,4)*aep23^2+2*C(5,5)*aep31^2;
end
```

Step Size 10⁰

```
clear all
tic

%mat1 Graphite/Epoxy

%E1=181e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3e9; G31=7e9;
ny12=0.28; ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester

E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

A=[]; b=[];
Aeq=[]; beq=[];
lb=[0,0,0];
ub=[9,9,9];
numOfVar=3;
ea(1)=8; ea(2)=-7; ea(3)=-6; % principal strains

x0=[0.6, 0.15, 0.3];
intcon=[1,2,3]
[x,Uval,exitf] =
ga(@(x)pr_ob2Degrees10(x,C,ea),numOfVar,A,b,Aeq,beq,lb,ub,[],intcon);
format long
```

```
x=x
U=Uval
10*x
Computing_Time=toc
```

Objective functions

```
function [U] = pr_ob2Degrees10(x,C,ea)
d1=deg2rad(10*x(1));
d2=deg2rad(10*x(2));
d3=deg2rad(10*x(3));
e1=ea(1);e2=ea(2); e3=ea(3);
aep12=-2*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)^2-((e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*sin(d1)*cos(d1)+sin(d3)*cos(d2)*cos(d3)*(e1-e2);
aep31=(sin(d1)*(e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)+sin(d3)*cos(d1)*cos(d3)*(e1-e2))*sin(d2);
aep23=(-cos(d2)*(e1-e2)*cos(d3)^2-
e1+e3)*cos(d1)+sin(d1)*sin(d3)*cos(d3)*(e1-e2))*sin(d2);
aep11=((e1-e2)*cos(d3)^2-e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*cos(d1)^2-2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+((-
e1+e2)*cos(d3)^2+e1-e3)*cos(d2)^2+e3;
aep22=(-cos(d2)^2+1)*(e1-e2)*cos(d3)^2+(e1-e3)*cos(d2)^2-
e2+e3)*cos(d1)^2+2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+(e1-
e2)*cos(d3)^2+e2;
aep33=cos(d3)^2*cos(d2)^2*e1-cos(d3)^2*cos(d2)^2*e2-
cos(d3)^2*e1+cos(d3)^2*e2-cos(d2)^2*e1+cos(d2)^2*e3+e1;
U=0.5*C(1,1)*aep11^2+ C(1,2)*aep11*aep22+
C(1,3)*aep11*aep33+0.5*C(2,2)*aep22^2+C(2,3)*aep22*aep33+0.5*C(3,3)*aep33
^2+2*C(6,6)*aep12^2+2*C(4,4)*aep23^2+2*C(5,5)*aep31^2;
end
```

Step Size 15°

```
clear all
tic

%mat1 Graphite/Epoxy

E1=181e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3e9; G31=7e9;
ny12=0.28; ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester

E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

A=[]; b=[];
Aeq=[]; beq=[];
lb=[0,0,0];
ub=[6,6,6];
numOfVar=3;
ea(1)=8; ea(2)=-7; ea(3)=-6; % principal strains
```

```

x0=[0.6, 0.15, 0.3];
intcon=[1,2,3]
[x,Uval,exitf] =
ga(@(x)pr_ob2Degrees15(x,C,ea),numOfVar,A,b,Aeq,beq,lb,ub,[],intcon);
format long
x=x
U=Uval
15*x
Computing_Time=toc

```

Objective functions

```

function [U] = pr_ob2Degrees15(x,C,ea)
d1=deg2rad(15*x(1));
d2=deg2rad(15*x(2));
d3=deg2rad(15*x(3));
e1=ea(1);e2=ea(2); e3=ea(3);
aep12=-2*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)^2-((e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*sin(d1)*cos(d1)+sin(d3)*cos(d2)*cos(d3)*(e1-e2);
aep31=(sin(d1))*((e1-e2)*cos(d3)^2-
e1+e3)*cos(d2)+sin(d3)*cos(d1)*cos(d3)*(e1-e2))*sin(d2);
aep23=(-cos(d2))*((e1-e2)*cos(d3)^2-
e1+e3)*cos(d1)+sin(d1)*sin(d3)*cos(d3)*(e1-e2))*sin(d2);
aep11=((e1-e2)*cos(d3)^2-e1+e3)*cos(d2)^2+(e1-e2)*cos(d3)^2+e2-
e3)*cos(d1)^2-2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+((-
e1+e2)*cos(d3)^2+e1-e3)*cos(d2)^2+e3;
aep22=(-cos(d2)^2+1)*(e1-e2)*cos(d3)^2+(e1-e3)*cos(d2)^2-
e2+e3)*cos(d1)^2+2*sin(d1)*sin(d3)*cos(d2)*cos(d3)*(e1-e2)*cos(d1)+(e1-
e2)*cos(d3)^2+e2;
aep33=cos(d3)^2*cos(d2)^2*e1-cos(d3)^2*cos(d2)^2*e2-
cos(d3)^2*e1+cos(d3)^2*e2-cos(d2)^2*e1+cos(d2)^2*e3+e1;
U=0.5*C(1,1)*aep11^2+ C(1,2)*aep11*aep22+
C(1,3)*aep11*aep33+0.5*C(2,2)*aep22^2+C(2,3)*aep22*aep33+0.5*C(3,3)*aep33
^2+2*C(6,6)*aep12^2+2*C(4,4)*aep23^2+2*C(5,5)*aep31^2;
end

```

Appendix 3

Constrained Optimization

Genetic Algorithm (GA)

```
clear all
tic

%mat1 Graphite/Epoxy

%E1=181e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3e9; G31=7e9;
ny12=0.28; ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester

E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

A=[]; b=[];
Aeq=[]; beq=[];
lb=[-8,-7,-6, -8,-8,-8];
ub=[8,7,6,6,6,6];
numOfVar=6;
ea(1)=8; ea(2)=-7; ea(3)=-6; % principal strains

intcon=[]
options= optimoptions('ga','Generations',350,'PopulationSize', 350);
Aeq=[];
beq=[];
[x,Uval,exitf] =
ga(@(x)pr_ob1_constrained_opt(x,C,ea),numOfVar,A,b,Aeq,beq,lb,ub,@(x)nonl
in(x,ea),intcon, options);
format long
x=x
U=Uval

temp=((x(3)-ea(3))*(x(3)+ea(3)-ea(1)-ea(2))+x(4)^2+x(5)^2)/((ea(1)-
ea(3))*(ea(3)-ea(2)));
Theta2=asin(sqrt(temp));
temp2=(x(4)^2-x(2)*x(3)-x(1)*ea(3)+ea(3)*(ea(1)+ea(2)))/(temp*(ea(1)-
ea(3))*(ea(3)-ea(2)));
Theta1=asin(sqrt(temp2));
temp3=2*(ea(3)-0.5*(ea(1)+ea(2))-(ea(3)-x(3))/temp)/(ea(2)-ea(1));
Theta3=(acos(temp3))/2;
[Theta1,Theta2,Theta3]
Computing_Time=toc
e1=ea(1);
e2=ea(2);
e3=ea(3);
kontr=(x(4)*x(4)+x(5)*x(5)+(x(3)-e3)*(x(3)+e3-e1-e2))/((e1-e3)*(e3-e2))
dd2=asin(sqrt(kontr))
```


Objective functions

```
function [U] = pr_ob1_constrained_opt(x,C,ea)
aep11=x(1); aep22=x(2); aep33=x(3); aep12=x(6); aep23=x(4);
aep31=x(5);
U=0.5*C(1,1)*aep11^2+ C(1,2)*aep11*aep22+
C(1,3)*aep11*aep33+0.5*C(2,2)*aep22^2+C(2,3)*aep22*aep33+0.5*C(3,3)*aep33
^2+2*C(6,6)*aep12^2+2*C(4,4)*aep23^2+2*C(5,5)*aep31^2;
end
```

```
function [c,ceq] = nonlin(x,e)
c=[];
ceq(2)=x(1)*x(2)+x(2)*x(3)+x(3)*x(1)-x(4)^2-x(5)^2-x(6)^2-e(1)*e(2)-
e(2)*e(3)-e(3)*e(1); % second invariant
ceq(3)=x(1)*x(2)*x(3)+2*x(4)*x(5)*x(6)-x(1)*x(4)^2-x(2)*x(5)^2-
x(3)*x(6)^2-e(1)*e(2)*e(3);
ceq(1)=x(1)+x(2)+x(3)-e(1)-e(2)-e(3);
end
```

Lagrange multipliers method

```
clear all
tic
initime = cputime;

%mat1 Graphite/Epoxy
E1=181.0e9; E2= 10.3e9; E3=10.3e9; G12=7.17e9; G23=3.0e9; G31=7.0e9;
ny12=0.28; ny13=0.27; ny23=0.6;

%mat2 E-Glass/Vinylester
E1=25e9; E2=24.8e9; E3=8.5e9; G12=6.5e9; G23=4.5e9;
G31=4.2e9;ny12=0.1;ny13=0.28;ny23=0.3;

S=[1/E1, -ny12/E1, -ny13/E1,0,0,0;-ny12/E1, 1/E2, -ny23/E2, 0,0,0; -
ny13/E1,-ny23/E2, 1/E3,0, 0,0;...
0,0,0, 1/G23, 0,0;0,0,0,0,1/G31,0;0,0,0,0,0,1/G12];
C=S^-1;

e(1)=8; e(2)=-7; e(3)=-6; % principal strains
ea=e;
x0=[-0.8,-1,-3];
[x,fval,exitf] = fsolve(@ (x)OptConditions(x,C,e),x0)

f1=(C(1,1)-C(1,2)-2*C(6,6))*x(1)-(C(2,2)-C(1,2)-2*C(6,6))*x(2)+(C(1,3)-
C(2,3))*x(3);
f2=(C(1,2)-C(1,3))*x(1)+(C(2,2)-C(2,3)-2*C(4,4))*x(2)-(C(3,3)-C(2,3)-
2*C(4,4))*x(3);
f3=(C(1,3)-C(1,1)+2*C(5,5))*x(1)+(C(2,3)-C(1,2))*x(2)+(C(3,3)-C(1,3)-
2*C(5,5))*x(3);
D1=2*(C(4,4)-C(5,5));
D2=2*(C(5,5)-C(6,6));
D3=2*(C(6,6)-C(4,4));
x(4)=sqrt(f1*f3/(D1*D3));
x(5)=sqrt(f1*f2/(D1*D2));
```

```

x(6)=sqrt(f2*f3/(D2*D3));
temp=((x(3)-ea(3))*(x(3)+ea(3)-ea(1)-ea(2))+x(4)^2+x(5)^2)/((ea(1)-
ea(3))*(ea(3)-ea(2)));
Theta2=asin(sqrt(temp));
temp2=(x(4)^2-x(2)*x(3)-x(1)*ea(3)+ea(3)*(ea(1)+ea(2)))/(temp*(ea(1)-
ea(3))*(ea(3)-ea(2)));
Theta1=asin(sqrt(temp2));
temp3=2*(ea(3)-0.5*(ea(1)+ea(2))-(ea(3)-x(3))/temp)/(ea(2)-ea(1));
Theta3=(acos(temp3))/2;
[Theta1,Theta2,Theta3]

fintime = cputime;
fprintf('CPU TIME: %g\n', fintime - initime);
Computing_Time=toc

```

Objective functions

```

function [F] = OptConditions(x,C,e)
f1=(C(1,1)-C(1,2)-2*C(6,6))*x(1)-(C(2,2)-C(1,2)-2*C(6,6))*x(2)+(C(1,3)-
C(2,3))*x(3);
f2=(C(1,2)-C(1,3))*x(1)+(C(2,2)-C(2,3)-2*C(4,4))*x(2)-(C(3,3)-C(2,3)-
2*C(4,4))*x(3);
f3=(C(1,3)-C(1,1)+2*C(5,5))*x(1)+(C(2,3)-C(1,2))*x(2)+(C(3,3)-C(1,3)-
2*C(5,5))*x(3);
D1=2*(C(4,4)-C(5,5));
D2=2*(C(5,5)-C(6,6));
D3=2*(C(6,6)-C(4,4));
F(1) = x(1)+x(2)+x(3)-e(1)-e(2)-e(3);
F(2) = x(1)*x(2)+x(2)*x(3)+x(3)*x(1)-f2*f3/(D2*D3)-f2*f1/(D2*D1)-
f1*f3/(D1*D3)-e(1)*e(2)-e(3)*e(2)-e(3)*e(1);
F(3) = x(1)*x(2)*x(3)+2*f1*f2*f3/(D1*D2*D3)-x(1)*f1*f3/(D1*D3)-
x(2)*f1*f2/(D1*D2)-x(3)*f2*f3/(D2*D3)-e(1)*e(2)*e(3);
end

```