

MASTER OF SCIENCE THESIS

Overcoming the curse of dimensionality in composite laminate blending

A CA-based algorithm for blending laminated composite plates with a large amount of sections

E. van den Oord

Faculty of Aerospace Engineering · Delft University of Technology

Overcoming the curse of dimensionality in composite laminate blending

**A CA-based algorithm for blending laminated composite plates
with a large amount of sections**

MASTER OF SCIENCE THESIS

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

E. van den Oord

28-08-2018



Copyright © E. van den Oord
All rights reserved.

DELFT UNIVERSITY OF TECHNOLOGY
FACULTY OF AEROSPACE ENGINEERING
DEPARTMENT OF AEROSPACE STRUCTURES AND MATERIALS

GRADUATION COMMITTEE

Dated: 28-08-2018

Chair holder:

Dr.ir. R. de Breuker

Committee members:

Dr.ir. J.M.J.F. van Campen

Dr.ir. G.N. Saunders-Smits

Preface

This thesis is written to fulfil the graduation requirements for the master Structures & Computational Mechanics at the faculty of Aerospace engineering at the TU Delft.

I would like to thank my supervisor Dr.ir. J.M.J.F. van Campen for his guidance and support throughout the completion of this thesis. His positivity and enthusiasm gave me confidence in my own abilities.

Furthermore, I would like to thank the assessment committee members, Dr.ir. R. de Breuker and Dr.ir. G.N. Saunders-Smits for taking the time to review this thesis.

The process of writing my thesis included a challenging personal journey. I would like to thank the people that never stopped believing in me. Their encouragement motivated me to succeed. As my best friend Anne wrote:

*When your road is dark
eyes see naught but fear.
Bright sparkling people
are lanterns to hold dear.*

I hope you enjoy reading this thesis.

Delft, University of Technology
28-08-2018

E. van den Oord

“In the midst of chaos, there is also opportunity.”
— *Sun Tzu*

Abstract

The specific properties of fibre reinforced laminated composite plates can be optimized by tailoring the local stacking sequences to approach the locally required stiffness and strength. Stacking sequence incompatibilities may occur between neighbouring sections by individually optimizing each section. Stacking sequence incompatibilities may result in a lack of structural integrity. Blending of the stacking sequences enforces continuation of some or all plies from one stacking sequence to another to ensure structural integrity and manufacturability. The optimization of a blended laminated composite plate is a combinatorial optimizing problem which suffers from the curse of dimensionality. An increase in the number of sections causes an exponential increase in the amount of design solutions. Therefore, the effectivity of the blending strategy becomes important when the number of sections increases.

The objective of this thesis is to overcome the dimensionality in blending laminated composite plates by creating an optimization tool that can effectively blend composite laminated plates with a large amount of sections.

An innovative algorithm is proposed. The proposed optimization strategy combines a genetic algorithm (GA) and a Cellular Automaton (CA). A GA is used to provide the local optima. The global blending constraints are translated to local dependencies. A CA-based algorithm evolves the local optima into a blended configuration by a straightforward set of rules based on these local dependencies. The algorithm pursues blended transitions with a minimum amount of modifications. The effectivity of a blending procedure is defined as a function of the dimensionality and the computational time. The proposed blending strategy is based on local dependencies to increase the effectivity of blending plates with a large amount of sections.

A benchmark case in the shape of a horseshoe pattern is used to compare the results of the proposed method to the state-of-the-art methods. The optimal design provided by the proposed algorithm is comparable to the state-of-the-art blending methods in terms of weight. The horseshoe benchmark case consists of 18 sections. A new benchmark case is proposed to prove the effectivity for blending a larger amount of sections. The results provide a locally blended configuration for 3D problems. However, the final configuration does not guarantee a feasible patch interpretation on a global level. This can be solved by a more global approach

in the set of rules which eliminate butted edges. The computational time of the proposed algorithm increases linearly with the amount of sections, while the dimensionality increases exponentially. Therefore, the proposed algorithm overcomes the dimensionality in blending laminated composite plates with a large amount of sections.

Table of Contents

Preface	vii
Nomenclature	xvii
1 Introduction	1
2 Background	3
2.1 Principles of laminated composites	3
2.1.1 Fibre reinforced laminated composites	4
2.1.2 Classical lamination theory	4
2.2 Blending laminated composites	6
2.2.1 Guidelines for blending a laminated composite plate	7
2.2.2 Dimensionality of the optimization of blended laminated composite plates	8
2.3 State-of-the-art methods for blending laminated composites	9
2.3.1 Origin of blending laminated composites	10
2.3.2 Guide-based blending	11
2.3.3 Stacking sequence tables	12
2.3.4 Other blending methods	14
2.3.5 Limitations of current blending methods for plates with a large number of sections	15
2.4 Conclusions	16
3 Overview of the GA-CA-based optimization strategy	17
3.1 The proposed optimization method	17
3.2 Effectivity of a blending method	19
3.3 Conclusions	20

4	GA for local optimization	21
4.1	Chromosome representation of an individual	21
4.2	Fitness evaluation of an individual	22
4.2.1	Objective and constraints	23
4.2.2	Fitness function and fitness ranking	23
4.3	Evolution process of a genetic algorithm	24
4.3.1	Initialization of a population	24
4.3.2	Parent selection	24
4.3.3	Genetic operators	25
4.3.4	Termination criteria	27
4.4	Parameter settings and verification of the GA	27
4.4.1	Parameter settings of the fitness function	28
4.4.2	Parameter settings of the genetic operators	30
4.4.3	Settings that bound the evolution process	30
4.4.4	Verification of the GA	31
5	CA-based blending method	33
5.1	Representation of a multi section plate in a CA	33
5.1.1	Division of the cells according to a chessboard pattern	34
5.1.2	Range of layers included in the analysis of a ply	38
5.2	Evolution strategy	40
5.2.1	Foundation of Patches	40
5.2.2	Minimum patch size	45
5.2.3	Elimination of butted edges	50
5.3	Global patch interpretation	54
5.4	Verification of the CA-based algorithm	56
5.4.1	Verification of the evolution stages for 2D problems	56
5.4.2	Verification of the evolution stages for 3D problems	58
5.4.3	Verification of the GA-CA-based algorithm	60
6	Results and Discussion	63
6.1	Horseshoe pattern benchmark case	63
6.1.1	Generated results	64
6.1.2	Results and discussion	68
6.2	2D problems	69
6.2.1	Results and discussion	69
6.2.2	Effectivity of the 2D algorithm	70
6.2.3	Stability analysis	72
6.3	3D problems	73
6.3.1	Problem definition	74
6.3.2	Results and discussion	74
6.3.3	Effectivity of the 3D algorithm	77
6.3.4	Stability analysis	79
6.4	Conclusions	80

7	Conclusions and recommendations	81
7.1	Conclusions	81
7.2	Recommendations	82
7.2.1	Recommendations to improve the proposed algorithm	82
7.2.2	Recommendations for future research	82
A	Critical buckling load	89
B	Detailed scenarios for 2D problems	91
C	Results 2D plates	93
D	Results stability analysis	97
E	Benchmark case horseshoe pattern	99
F	All unique cross-sections of a 5 layer laminate	101

Nomenclature

List of abbreviations:

CA	Cellular Automaton
CLT	Classical Lamination Theory
DDOA	Double Distribution Optimization Algorithm
DVZ	Design Variable Zone
FE	Finite Element
GA	Genetic Algorithm
GSLB	Global Shared Layers Blending
IHR	Improved Hit and Run
LP	Lamination Parameter
NC	Neighbour Count
PDS	Ply Drop Sequence
SL	Sub-laminate
SLB	Shared Layers Blending
SST	Stacking Sequence Table

List of symbols

C	Size of the design problem
E_1	Longitudinal modulus
E_2	Transverse modulus
G	Number of genes
G_{12}	Shear modulus
g_{min}	Buckling constraint
L	Number of plies
M	Moment resultant
m_a	Slope of the algorithm
m_b	Slope of the baseline

N	Number of sections
N	Stress resultant
O	Fibre orientation options
P	Population size
S	Number of stacking sequences per section
T	Computational time
t	Thickness
U_i	Material invariant
W	Weight
z	Trough-the-thickness coordinate
$[A]$	Membrane stiffness matrix
$[B]$	Membrane-bending coupling stiffness matrix
$[D]$	Bending stiffness matrix
$[Q]$	Stiffness matrix
β	Penalty parameter
γ	In-plane shear strain
ε	In-plane tensile strain
ϵ	Bonus parameter
θ	Fibre orientation angle
θ	Fitness value
κ	In-plane curvature
ν_{12}	Poisson's ratio
σ	In-plane normal stress
τ	In-plane shear stress

Chapter 1

Introduction

The high stiffness-to-weight ratio and the high strength-to-weight ratio make the use of fibre reinforced composites popular in the aerospace and automotive industry. These specific properties can be optimized. A laminated composite plate subjected to a non-uniform load distribution is divided into sections. The specific properties are optimized by tailoring the stacking sequence of each section to approach the locally required stiffness and strength. Stacking sequence incompatibilities may occur between neighbouring sections by individually optimizing each section. Stacking sequence incompatibilities may result in a lack of structural integrity and manufacturability. Blending of the stacking sequences enforces continuation of some or all plies from one section to another to ensure structural integrity and manufacturability [15]. The optimization of a blended laminated composite plate is a combinatorial optimizing problem which suffers from the curse of dimensionality. This means that an increase in the number of sections causes an exponential increase in the amount of design solutions. Therefore, the effectivity of the blending strategy becomes important when the number of sections increases.

The objective of this thesis is to overcome the dimensionality in blending laminated composite plates by creating an optimization tool that can effectively blend composite laminated plates with a large amount of sections. A literature study is performed to define the dimensionality in blending laminated composite plates. The state-of-the-art blending methods are reviewed to determine the limitations of these methods for blending laminated composite plates with a large amount of sections. A blending strategy is proposed to overcome the dimensionality in blending laminate composite plates, including a method to measure the effectivity of the blending procedure. An algorithm is developed in MATLAB to implement the proposed blending strategy. A benchmark case in the shape of a horseshoe pattern is used to compare the results of the proposed method to the methods described in literature. A new benchmark case is proposed to test the effectivity of the blending algorithm for plates with a larger amount of sections than currently found in literature.

The structure of this report is as follows. The background of blending laminated composites and the limitations of the state-of-the-art blending methods are described in chapter 2. The proposed blending strategy to overcome the dimensionality of the blending problem is

described in chapter 3. This blending strategy combines a genetic algorithm (GA) and a cellular automaton (CA). The GA is used to determine the local optima, which is further described in chapter 4. A CA-based algorithm is applied to evolve these local optima to a blended configuration. An elaboration of the CA-based algorithm can be found in chapter 5. The results of the proposed algorithm are presented in chapter 6. The results include a comparison to the state-of-the-art and a new proposed benchmark case to test the effectivity of blending a larger amount of sections. The conclusions and recommendations can be found in chapter 7.

Chapter 2

Background

The specific properties of fibre reinforced laminated composite plates, the stiffness-to-weight ratio and the strength-to-weight ratio, can be optimized by tailoring the local stiffness and local strength to match the varying load distribution in a plate. In the optimization process of a composite structure, the structure is divided into several sections with varying loads. By individually optimizing each section, stacking sequence incompatibilities may occur between neighbouring sections. Stacking sequence incompatibilities may result in a lack of structural integrity and manufacturability. Blending enforces continuation of some or all plies from one section to another, to ensure structural integrity and manufacturability. Effectivity in blending becomes important when the number of variables increases, also referred to as the curse of dimensionality.

This chapter provides the background information about blending laminated composite plates. The basic principles of laminated composites are presented in section 2.1. A detailed description of blending laminated composites and the corresponding dimensionality is presented in section 2.2. The state-of-the-art methods for blending laminated composite plates are described in section 2.3, including the limitations of the current blending methods for plates with a large amount of sections. The concluding remarks on the optimization problem and the limitations of state-of-the-art methods are summarized in section 2.4.

2.1 Principles of laminated composites

Laminated composites consist of layers of composite material. The stiffness can be tailored to specific loading conditions by adapting the composition of the laminate. This results in a high stiffness-to-weight ratio and high strength-to-weight ratio. In section 2.1.1, a description is given of fibre reinforced laminated composites. The Classical Lamination Theory (CLT) is described in section 2.1.2. CLT is an analytical method that relates external loads to the deformation of a laminated composite plate.

2.1.1 Fibre reinforced laminated composites

Composite material contains at least two constituents, reinforcement material and matrix material. Layers of composite material stacked together form a composite laminate. The layers in laminated composites are also called laminae or plies [14]. In fibre reinforced laminated composites, fibres act as the reinforcement and are kept in place by matrix material; a light weight thermoset or thermoplastic resin [14]. Fibres can be continuous or discontinuous, diffused or aligned. Fibres can be made out of, amongst others glass, carbon, aramid or boron [5, 14]. The plies commonly used in fibre reinforced laminated composite are unidirectional plies and fabric plies [14]. All fibres in a unidirectional ply are continuous and have the same orientation. The fibres in a fabric ply are continuous and orientated in two perpendicular directions. A composite ply is orthotropic, which means it has different stiffness properties in three mutually orthogonal directions [14]. The stiffness in those directions depends on the fibre orientation. For pure membrane deformation, the main design variables of a laminate are the number of plies and the fibre orientation of the plies. However, the through-thickness location of a ply influences the bending stiffness of a laminate due to the orthotropic property of the plies [14]. Therefore, the stacking order of the plies becomes significant for bending deformation and coupling between membrane and bending behaviour. The laminate can be optimized to specific loading conditions by reordering the plies. This is of particular interest for plates, since plates are buckling critical design problems. The critical buckling load is dependent on the bending stiffness of the plate. Therefore, an optimized stacking sequence can improve the buckling load without an increase in weight. This gives fibre reinforced laminated composites its ability of a high stiffness-to-weight ratio and high strength-to-weight ratio [14].

2.1.2 Classical lamination theory

The Classical Lamination Theory (CLT) describes an analytical method that relates external loads to the deformation of a laminated composite plate. The theory assumes the plies are perfectly bonded and there is a perfect bond between the fibres and the matrix within all plies. The laminate is assumed to be a homogenous material with averaged material properties of those of all constituents, also called smeared properties [14]. A schematic representation of the ply and laminate axis system by Kassapoglou [14] is shown in figure 2.1. The elastic stiffness and strength of a laminated composite are defined in the laminate reference system 1,2. Each ply has a fibre orientation θ relative to laminate reference system. CLT assumes a composite laminate is orthotropic and in a state of plane stress. An orthotropic laminate has three planes of symmetry, therefore, an orthotropic laminate can have different material properties in three mutual orthogonal directions. A laminate consisting of plies with different fibre orientations is orthotropic in case the laminate is balanced and symmetric with respect to the laminate reference system. If the thickness of a laminate is thin relative to the other dimensions, the out-of-plane stresses σ_z , τ_{yz} and τ_{xz} are negligible compared to the in-plane stresses σ_x , σ_y and τ_{xz} and the laminate is assumed to be in a state of plane stress.

The in-plane stress along the laminate axis of the k-th layer of a laminate is a combination of membrane strain and curvature as shown by equation 2.1.

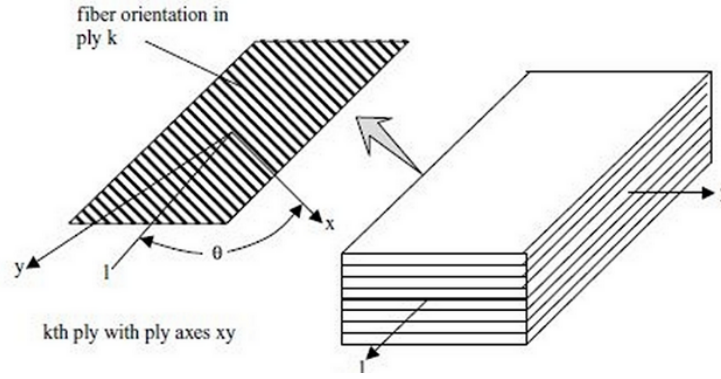


Figure 2.1: A schematic representation of the ply and laminate axis system [14]

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix}_k = \begin{bmatrix} \bar{Q}_{11} & \bar{Q}_{12} & \bar{Q}_{16} \\ \bar{Q}_{12} & \bar{Q}_{22} & \bar{Q}_{26} \\ \bar{Q}_{16} & \bar{Q}_{26} & \bar{Q}_{66} \end{bmatrix}_k \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{Bmatrix} + \begin{Bmatrix} z\kappa_x \\ z\kappa_y \\ z\kappa_{xy} \end{Bmatrix} \quad \text{where } z_{k-1} < z < z_k. \quad (2.1)$$

The stiffness matrix $\bar{Q}$ is defined by the material invariants U_i and the fibre orientation of the ply as shown in equation 2.2.

$$\begin{aligned} \bar{Q}_{11} &= U_1 + U_2 \cos(2\theta) + U_3 \cos(4\theta) & \bar{Q}_{22} &= U_1 - U_2 \cos(2\theta) + U_3 \cos(4\theta) \\ \bar{Q}_{12} &= U_4 - U_3 \cos(4\theta) & \bar{Q}_{66} &= U_5 - U_3 \cos(4\theta) \\ \bar{Q}_{16} &= \frac{1}{2} U_2 \sin(2\theta) + U_3 \sin(4\theta) & \bar{Q}_{26} &= \frac{1}{2} U_2 \sin(2\theta) - U_3 \sin(4\theta) \end{aligned} \quad (2.2)$$

According to equation 2.1, stacking plies with the same fibre orientation results in a constant or linear stress distribution. Stacking plies with different fibre orientations result in a non-linear stress distribution. The in-plane stresses are constant through the thickness of each ply. However, the stresses may differ per ply which results in a stepwise stress distribution through the laminate thickness. By integrating the stresses through the thickness of the laminate, the relation between the force resultant and the laminate strain can be determined as shown in equation 2.3.

$$\begin{aligned} N_x &= \int_{-\frac{h}{2}}^{\frac{h}{2}} \sigma_x dz & N_y &= \int_{-\frac{h}{2}}^{\frac{h}{2}} \sigma_y dz & N_{xy} &= \int_{-\frac{h}{2}}^{\frac{h}{2}} \tau_{xy} dz \\ M_x &= \int_{-\frac{h}{2}}^{\frac{h}{2}} \sigma_x z dz & M_y &= \int_{-\frac{h}{2}}^{\frac{h}{2}} \sigma_y z dz & M_{xy} &= \int_{-\frac{h}{2}}^{\frac{h}{2}} \tau_{xy} z dz \end{aligned} \quad (2.3)$$

Combining equations 2.1 and 2.3 results in the generalized constitutive relations of a laminate. In equation 2.4, the generalized constitutive relations are provided, which are also called the ABD-matrix [14]. The A-matrix relates the in-plane loads to membrane deformation. The D-matrix describes the relation between the bending moments and curvatures. The assumptions of the standard Kirchhoff plate theory are used for a laminate subjected to bending; a plane section remains plane and perpendicular to the neutral axis after deformation [14]. The

B-matrix describes the coupling between bending and membrane behaviour. There is no coupling between bending and membrane behaviour in case a laminate is symmetric thus the B-matrix is zero.

$$\begin{Bmatrix} N_x \\ N_y \\ N_{xy} \\ M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{16} & B_{11} & B_{12} & B_{16} \\ A_{12} & A_{22} & A_{26} & B_{12} & B_{22} & B_{26} \\ A_{16} & A_{26} & A_{66} & B_{16} & B_{26} & B_{66} \\ B_{11} & B_{12} & B_{16} & D_{11} & D_{12} & D_{16} \\ B_{12} & B_{22} & B_{26} & D_{12} & D_{22} & D_{26} \\ B_{16} & B_{26} & B_{66} & D_{16} & D_{26} & D_{66} \end{bmatrix}_k \begin{Bmatrix} \varepsilon_{xo} \\ \varepsilon_{yo} \\ \gamma_{xyo} \\ \kappa_x \\ \kappa_y \\ \kappa_{xy} \end{Bmatrix} \quad (2.4)$$

The stiffness is assumed to be constant through the thickness of each ply. Therefore, equation 2.4 can also be written as a summation of all plies in the laminate. The summation is provided in equation 2.5, where z_{k-1} and z_k are the distances between the midplane of the laminate and respectively the bottom and top of the k-th layer.

$$\begin{aligned} A_{ij} &= \sum_{k=1}^n \bar{Q}_{ij}(z_k - z_{k-1}) \\ B_{ij} &= \sum_{k=1}^n \frac{\bar{Q}_{ij}}{2}(z_k^2 - z_{k-1}^2) \\ D_{ij} &= \sum_{k=1}^n \frac{\bar{Q}_{ij}}{3}(z_k^3 - z_{k-1}^3) \end{aligned} \quad (2.5)$$

According to Equation 2.5, the A-matrix is not dependent on the through thickness location of the plies, while the B-matrix and D-matrix are affected by the stacking sequence of the plies. By altering the stacking sequence, the bending stiffness D_{ij} can be optimized without changing the composition of the laminate. This is especially important for plates, since plates are buckling critical design problems. The critical buckling load is dependent on the bending stiffness of the plate. Therefore, an optimized stacking sequence can improve the buckling load without an increase in weight. Using stacking sequence optimization, fibre reinforced laminated composites can obtain a higher stiffness-to-weight ratio and a higher strength-to-weight ratio.

2.2 Blending laminated composites

The stiffness and strength of laminated composites can be tailored to specific loading conditions by altering the stacking sequence as discussed in section 2.1. However, the loads in a plate might not be constant. The required stiffness and strength of the laminated composite changes throughout the plate in case a plate has a varying load distribution. Thereby, the optimal configuration of the laminate changes. The laminate configuration should be adapted to match local loading conditions in order to achieve the high stiffness-to-weight-ratio and high strength-to-weight-ratio. The challenge is to optimize the local laminate configurations while retaining structural integrity of the plate. This can be achieved by blending. Blending enforces continuation of some or all plies to ensure structural integrity and manufacturability [15]. The guidelines to obtain a blended plate are described in section 2.2.1. Finding

the optimal blended plate design is complex due to the large amount of variables. The dimensionality of the optimization of composite laminated blended plates is defined in section 2.2.2.

2.2.1 Guidelines for blending a laminated composite plate

A laminated composite plate with a non-uniform load distribution is divided into several sections with varying loads. The loads are assumed to be constant in each section. The objective is to determine the stacking sequences that results in the lowest weight of the overall plate, while complying with structural and manufacturing constraints. A schematic 2D representation of three scenarios regarding stacking sequence compatibility, is depicted in figure 2.2. The plate is divided into 4 sections. In case no stacking sequence incompatibilities are allowed, the stacking sequence of the section with the highest loads is used in all sections. This results in a high weight plate configuration as shown in figure 2.2a. The local optimum of all sections are depicted in figure 2.2b. The stacking sequence of a local optimum is tailored to comply with the local loading conditions and is independent of the configuration of neighbouring sections. A plate consisting of local optima may lack structural integrity due to the absence of ply continuity. The blended configuration is shown in figure 2.2c. The stacking sequences are locally optimized while blending ensures sufficient continuous plies to comply with the structural and manufacturing constraints. The stacking sequences of the blended configuration may not be identical to the stacking sequences of the local optima.

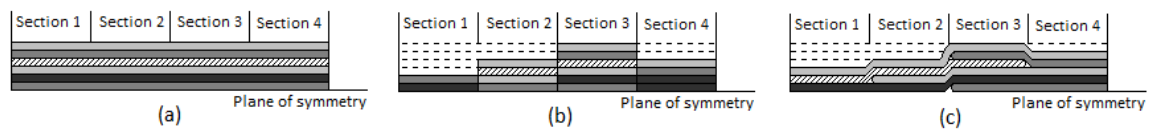


Figure 2.2: (a) Configuration with no stacking sequence incompatibilities, (b) Configuration consisting of the local optima, (c) Blended configuration

Blending ensures sufficient ply continuation to maintain the structural integrity of the plate. Ply drops are required to minimize the weight of a blended configuration. Ply drops create stresses in the laminate which may contribute to multiple failure mechanisms. Therefore, the amount of ply drops should be minimized and guidelines regarding ply drops are included in the blending guidelines. According to the ply drop guidelines, all plies should be dropped symmetrically with respect to the midplane of the laminate. A minimum distance between successive ply drops is required to avoid enhancement of stresses due to the interference of stresses from neighbouring ply drop sites. The minimum distance is 10 to 15 times the dropped height [14]. The ply drop layouts that may occur between neighbouring sections are presented in figure 2.3. The additional guidelines for the specific types of ply drop layouts are described below [14].

An external ply drop is depicted in figure 2.3a. External ply drops should be avoided if possible. Delamination may occur at the edge of the dropped ply. External ply drops can be used when the applied loads are sufficiently low. However, inner ply drops close to the midplane are preferred. An inner ply drop is depicted in figure 2.3b. Inner ply drops reduce the risk of delamination. Dropping multiple inner plies at the same location may result in failure due to interlaminar stresses. Interlaminar stresses are confined by limiting the

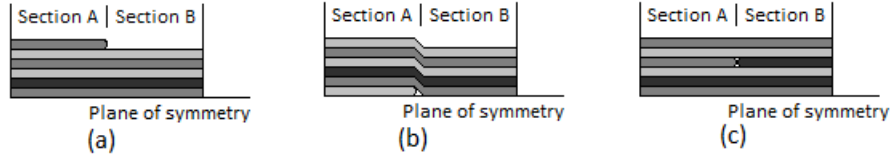


Figure 2.3: (a) External ply drop, (b) Inner ply drop, (c) Butted edges

maximum amount of plies dropped at the same location to 0.5mm worth of plies [14]. Butted edges are depicted in figure 2.3c. Butted edges are not preferred since butted edges cause stress concentrations. If multiple plies bridge the butted edges, butted edges may be deemed acceptable from a structural point of view. However, from a manufacturing point of view, butted edges should be avoided due to the required manufacturing accuracy [5].

2.2.2 Dimensionality of the optimization of blended laminated composite plates

The optimal stacking sequence of a section in a blended configuration is dependent on the stacking sequences of neighbouring sections. This global constraint makes the optimization of a laminated composite plate with a non-uniform load distribution a complex process. A simplified problem is presented in figure 2.4 to demonstrate the dimensionality of blending a laminated composite plate. A plate is divided in N number of sections. L_{min} is the minimum amount of layers that is analysed. L_{min} is derived from the amount of layers of the local optima. A maximum amount of layers L_{max} is given to all laminates to limit the design space in order to minimize the data to be analysed. The fibre orientations are limited to a discrete number of options O due to manufacturing constraints.

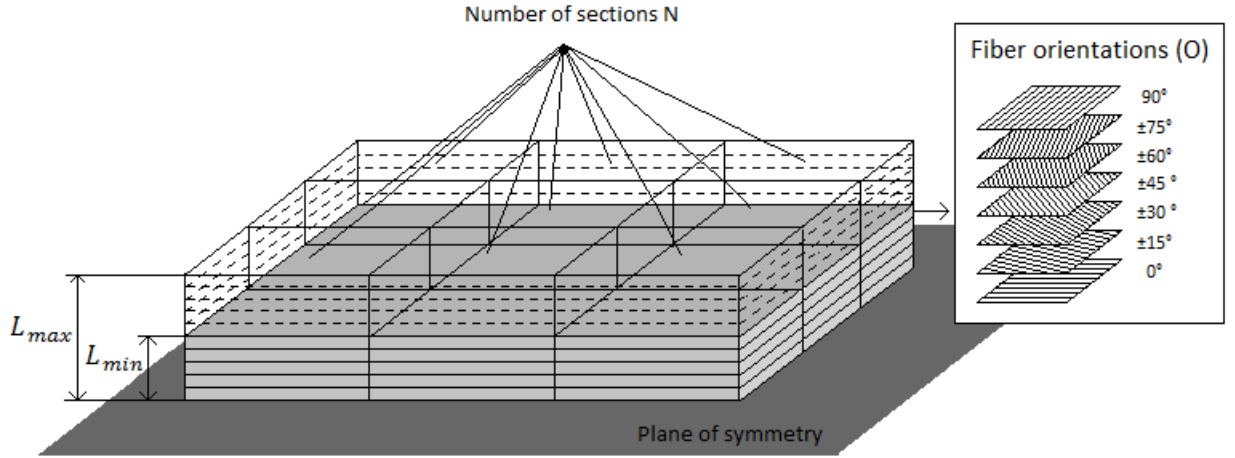


Figure 2.4: A schematic representation of a plate consisting of N sections

For a section with a minimum amount of layers L_{min} and a maximum amount of layers L_{max} , the total amount of stacking sequences per section to be considered is:

$$\text{Amount of stacking sequences per section (S)} = \sum_{i=L_{min}}^{L_{max}} O^i \quad (2.6)$$

A plate design consists of one stacking sequence per section. The combinations of all stacking sequences of all sections represent the total amount of plate designs to be analysed:

$$\text{Total amount of plate designs} = \prod_{i=1}^N S_{sectioni} \quad (2.7)$$

The plate in figure 2.4 is divided into 9 sections. Each layer can represent one of the 7 discrete fibre orientations; 0° , $\pm 15^\circ$, $\pm 30^\circ$, $\pm 45^\circ$, $\pm 60^\circ$, $\pm 75^\circ$ or 90° . For a symmetric laminate only half of the laminate is analysed. One half of the symmetric laminate in figure 2.4 has a minimum number of 5 layers per section and a maximum number of 10 layers, so $L_{min} = 5$ and $L_{max} = 10$. The amount of stacking sequences per section becomes:

$$\text{Amount of stacking sequences per section (S)} = \sum_{i=5}^{10} 7^i = 329551656$$

All 9 sections have 329551656 stacking sequence options in this example. The total amount of plate designs to be analysed becomes:

$$\text{Total amount of design solutions} = 329551656^9 = 4.5847059 \cdot 10^{76}$$

The dimensionality of the optimization of composite laminate blended plates is defined by the amount of design solutions. The set of design solutions, provided by equation 2.6, increases exponentially by increasing the number of sections. This makes blending a combinatorial optimization problem that suffers from the curse of dimensionality. For a multi-section plate with a large amount of sections, the finite set of design solutions exceeds the number of designs that can be analysed with the current computational power. Methods to obtain a (near) optimal plate design with lower computational time and effort are required.

2.3 State-of-the-art methods for blending laminated composites

Blending laminated composites is a combinatorial optimization problem which suffers from the curse of dimensionality. The amount of design solutions exceed the number of designs that can be analysed with the current computational power as described in section 2.2. Methods to obtain a (near) optimal plate design with lower computational time and effort are required. Widespread research has been performed on methods for blending laminated composite structures. The challenge is to minimize the number of design variables, while keeping the design space as larger as possible.

Multiple strategies are described in literature, where the combination of a customized blending method and an optimization algorithm is used to find (near) optimal plate designs. The origin of blending is described in section 2.3.1. The guide-based blending methods are described in section 2.3.2. The blending methods based on stacking sequence tables are described in section 2.3.3. In section 2.3.4 the blending methods based on other strategies are described. The limitations of the state-of-the-art blending methods for plates with a large amount of sections can be found in section 2.3.5.

2.3.1 Origin of blending laminated composites

Kristinsdottir et al. [15] introduced the term blending to ensure structural integrity and manufacturability of a laminated composite plate. The proposed blending method combines key regions with the "less-than-or-equal-to" rule as shown in figure 2.5. The "less-than-or-equal-to" rule described by Zabinsky [28], indicates the directions in which ply drops are permitted. Plies can be dropped, but when a ply is dropped, it cannot be readded in latter sections. The key region is defined as the thickest section and is denoted by a star in figure 2.5. All plies originate from the key region. Multiple key regions can be denoted for a large structure. Kristinsdottir et al. [15] used the optimization algorithm Improving Hit and Run (IHR) to find the optimum plate configuration based on the key region blending method.

	Axial					
	1	2	3	4	5	6
1	(1,1)	≤ (1,2)		≥	≥	≥ (1,6)
2	(2,1)	≤	≤	≥	≥	≥
3		≤	≤	★	≥	≥
4	(4,1)	≤	≤	≥	≥	≥ (4,6)

Figure 2.5: Representation of the blending rule described by Kristinsdottir et al. [15]

Optimization algorithms are used to find the (near) optimal plate design based on a customized blending method. The most commonly used optimization algorithm in blending laminated composites is the Genetic Algorithm (GA). A GA is probabilistic, nondeterministic optimization algorithm based on the Darwinian theory survival of the fittest. A detailed description of the GA can be found in chapter 4. McMahon et al. [19] describe how the standard GA can be used in the design process of laminated composite structures. McMahon et al. [20] improved the standard GA by introducing a parallel GA with migration. Each section is optimized by a different processor. However, the populations of all sections evolve simultaneously. Migrants are sent to adjacent sections during each generation. These migrants are adopted in the adjacent population to enhance ply continuity between adjacent sections. A decrease of computational effort is established compared to the standard GA. Adams et al. [1] developed a similar method. The main difference is the migrant implementation method. The migrants are not adopted in the adjacent population, but are stored and can be referred to in order to determine the degree of stacking sequence similarities. The stacking sequence similarities are defined based on the edit distance. This method produces a variety of designs which enables a trade-off between weight and continuity. Drawbacks of this method are the high computational cost necessary to obtain fully blended results and the tendency of the algorithm to converge to local optima.

2.3.2 Guide-based blending

Adams et al. [2] proposed a guide based blending method using a GA to reduce the number of design variables and thereby the computational effort. A schematic overview of the guide based blending method as depicted by Adams et al. [2] is presented in figure 2.6. A guiding stack is defined by the stacking sequence of the thickest section. The stacking sequences of the other sections are represented by a segment of the guiding stack. Inner or outer plies can be dropped at each section, which results in respectively outwardly blended or inwardly blended designs. The guiding stack reduces the amount of design variables and guarantees fully blended designs throughout the design process. This method results in multiple near optimal designs that are substantially different, which can be beneficial for trade-off during the design process.

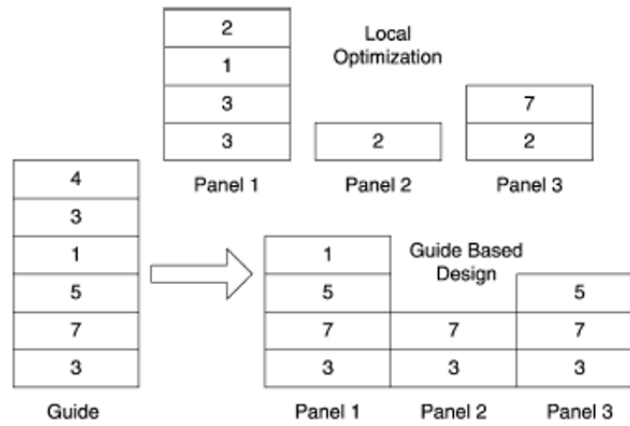


Figure 2.6: Guide based blending method depicted by Adams et al. [2]

IJsselmuiden et al. [9] improved the guide based blending method by eliminating the structural analysis from the GA to decrease the computational effort. The method described is a multistep approach. In the first step, optimization is performed with panel thickness and lamination parameters as continuous design variables to find the optimal stiffness and thickness distribution. The second step consists of the optimization of the discrete stacking sequence using the guide based blending method as described by Adams et al. [2]. Stacking sequence optimization is performed to approach the target stiffness and thickness distribution. The computational effort of the GA is reduced since the structural analysis is replaced by the less computationally expensive lamination parameters analysis. Jin et al. [11] extended the guide based blending method to a four step optimization method. Jin et al. [11] determined the lamination parameters of the guide laminate instead of defining the lamination parameters of all sections. The stacking sequence of the guide is determined using a GA with a form of least square fitting based on the guide lamination parameters. A parallel GA is used to determine the stacking sequences of all sections based on the inner/outer blending scheme, in which the stacking sequence of the guide and the thickness distribution are used as design variables. This method offers a wider application scope since the structural analysis is performed using real laminate configurations instead of lamination parameters. The number of design variables is less compared to the guide based blending method described by Adams et al. [2]. A

drawback of this four step optimization method is the high computational effort compared to the two step method proposed by IJsselmuiden et al. [9]. Seresta et al. [25] proposed a multi-chromosomal guide based blending framework. Instead of one chromosome that contains the thickness distribution of a panel, each section is described by a chromosome with equal length as the guiding stack. The chromosomes as depicted by Seresta et al. [25] is presented in figure 2.7. The first chromosome contains the stacking sequence of the guide laminate. The other chromosomes consists of binary numbers and describe the stacking sequences of a section. The number 1 indicates the existence of the corresponding guide laminate ply and 0 indicates the absence of the corresponding guide laminate ply. This method provides a global framework for inner and outer blending that eliminates the need for local improvements.

1 2 3 4 5 6 7 1 2 3 4 5	Guide stacking sequence
1 1 1 1 1 1 1 1 1 0 0 0	Panel 1
1 1 1 1 1 1 1 0 0 0 0 0	Panel 2

Figure 2.7: Individuals in a multi-chromosomal GA as depicted by Seresta et al. [25]

The previously described guide based blending methods are based on inner and outer blending, which only allows the inner or the outer plies to be dropped. This greatly reduces the design space since all ply drop options throughout the thickness of the laminate are eliminated. Van Campen et al. [4] introduced two new blending definitions:

"Generalized blending: *We consider two adjacent panels completely blended if all the layers from the thinner panel continue in the thicker one regardless of their position along the thickness of the laminate.*

Relaxed generalized blending: *We consider two adjacent panels completely blended if there are no dropped edges in physical contact."*

A schematic representation of these blending definitions as depicted by Van Campen [4] is presented in figure 2.8. These new blending definitions increase the design space without the need to adjust manufacturing methods. Van Campen [4] implemented these new blending definitions in a guide based blending method using a multi-chromosomal GA as described by Seresta [25]. The new blending definitions enable more design freedom. However, no conclusive results are obtained since the genetic search is thrown off course by the crossover operator of the GA. For the implementation of the new blending definitions, modifications are required.

2.3.3 Stacking sequence tables

Irisarri et al. [23] proposed a blending method based on a Stacking Sequence Table (SST). A SST describes an unique laminate between a minimum and a maximum number of layers as shown in table 2.1. A unique laminate is determined and the SST is built by adding plies one by one. When the thickness of a section is determined, the corresponding stacking sequence can be found in the SST. Irisarri et al. [23] used an evolutionary algorithm based on Double Distribution Optimization Algorithm (DDOA) to find the optimal design. The

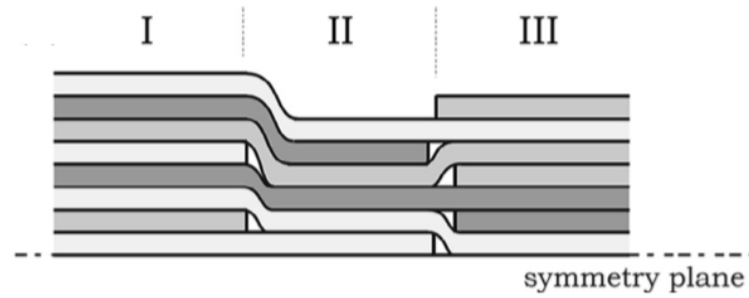


Figure 2.8: Examples of generalized blending (I-II) and relaxed generalized blending (II-III) as depicted by van Campen [5]

design variables of the SST-based method are the unique laminate, the ply insertion order and the number of layers in each section. The SST-based method gives a clear overview of the ply drop order and position, which may contribute to the design process. The designs are guaranteed to be fully blended. The design space is larger compared to inner and outer blending due to the variable ply insertion order.

Thickness [layers]	Nmin					Nmax
	3	4	5	6	7	8
			0	0	0	0
	+45	+45	+45	+45	+45	+45
					-75	-75
		90	90	90	90	90
				75	75	75
						90
	-45	-45	-45	-45	-45	-45
	0	0	0	0	0	0

Table 2.1: An example of a stacking sequence table

Yang et al. [27] proposed a similar method based on SST. The design variables in this method consist of the thickness distribution, the guiding stack and the Ply Drop Sequence (PDS). In this method the ply insertion variable is replaced by the PDS. The optimization of the laminate configuration is performed using a GA. Jing et al. [12] proposed a two level method called Global Shared Layer Blending (GSLB). The blending constraints are implemented at global level and at local level the stacking sequences are optimized. The method is based on the Shared Layer Blending concept as described by Liu et al. [17] which is further discussed in section 2.3.4. At global level, sets of shared layers are determined using an evolutionary algorithm that minimizes the ply drops and controls the transition of the sets to comply with blending constraints. At local level an SST is used to optimize the stacking sequence while complying with the design rules. Meddaikar et al. [21] proposed an optimization method using an adjusted GA based on the SST blending method as described by Irisarri et al. [23]. The GA is adjusted to take load redistribution into account using the optimization strategy "Improved Shepard's method" as described by Irisarri [10]. This strategy is used to decrease the required number of Finite Element (FE) analyses by substituting the structural responses by successive structural approximations. Fully blended designs result from this method using

a low number of FE analyses. Zein et al. [30] describe a two-step optimization process using a backtracking method. In step one, a gradient based optimization method determines the required thickness per orientation in each section. The thickness of each orientation is taken as a continuous variable. In step two, the ply orientation composition is transformed into a stacking sequence by means of permutation in a blending scheme similar to the SST scheme. A backtracking method is used to eliminate infeasible design branches from the enumeration tree to make the optimization based on enumeration efficient. The backtracking method guarantees fully blended designs throughout the design process.

2.3.4 Other blending methods

Liu et al. [16] developed a two-level method to enable a trade-off between plies discontinuity and weight. At global level the material compositions continuity is optimized, which is the amount of common layers divided by the total number of layers of one of the two sections. At local level a GA is used to transform material composition to the stacking sequence designs of all sections using stacking sequence continuity. Results show that by adding a little weight the continuity can be improved substantially, up to a limit. However, above the limit more continuity has a high weight penalty. This method does not guarantee blended results, since local optimization of the stacking sequences is highly constraint due to the applied material compositions constraint at global level. Soremekun et al. [26] created a two-step method that guarantees fully blended results. The method is based on Design Variable Zones (DVZ) and Sub Laminates (SL). During the first step all sections are individually optimized by a GA without taking into account the blending constraints. The lowest number of layers in each section is determined which is used as input for step two. In step two the DVZs and SLs are redefined based on the lowest number of layers, to implement the blending constraints as depicted by Soremekun [26] in figure 2.9. All sections can be simultaneously optimized using a GA. A fully blended design can be obtained with a small weight penalty compared to the individually optimized stacking sequences. However, the SL's restrict the design space compared to the relaxed generalized blending definition.

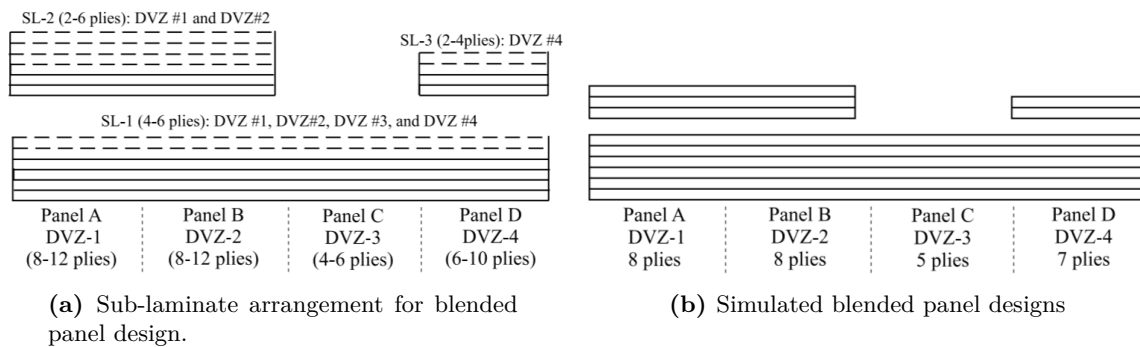


Figure 2.9: Illustration of the concept of SL's and DVZ's as depicted by Soremekun et al. [26]

Liu et al. [17] developed a bi-level optimization method based on lamination parameters and a shared layer blending scheme. The top level mathematical optimization is performed with panel thickness and lamination parameters as continuous design variables to find the optimal distribution of thickness and lamination parameters. At local level a permutation

GA in combination with a shared layer blending scheme is used to find the optimal stacking sequence. The shared layer blending scheme determines sets of shared layers between adjacent sections as depicted in figure 2.10. After a set of shared layers is determined, the GA is used to find the optimal stacking sequence of the set. The optimal stacking sequence is found by minimizing the difference between the stacking sequence lamination parameters and the optimal lamination parameters calculated in the top level. In this method the buckling analysis is replaced by the lamination parameters analysis which decreases the computation effort of the GA. The final designs are not guaranteed to be fully blended due to restricted design freedom resulting from the applied continuity constraints.

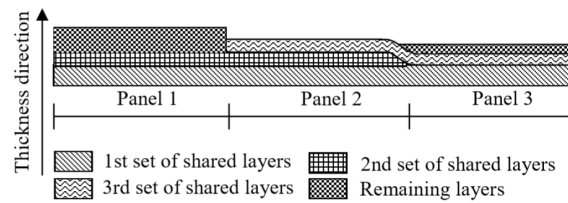


Figure 2.10: Illustration of the shared layers blending concept as depicted by Liu et al. [17]

Fan et al. [6] proposed an adjusted GA to implement continuity rules by using newly constructed chromosomes. The continuity rules are used to generate a less restricted design space, where two sections with the same thickness can have a different stacking sequence. The first rule ensures laminate composition continuity between adjacent sections, while the second rule ensures stacking sequence continuity between adjacent sections. To implement the continuity rules, new chromosomes are constructed which represent the ply composition and the ply ranking. Low weight designs are established. However, the continuity rules used in this method do not prevent butted edges. Therefore, this method does not produce blended designs according to the definition of relaxed generalized blending as described in section 2.3.2.

Macquart et al. [18] proposed a method based on continuous blending constraints to decrease computational effort. The use of continuous lamination parameters in a multi-step method [9, 17] resulted in a decrease of computational effort due to the elimination of the structural analysis from the GA. However, finding discrete stacking sequences that match the continuous lamination parameters remains computationally expensive. Expressing the blending constraints in the lamination parameter space reduces the mismatch between the discrete and continuous optimization steps. This method does not provide fully blended results.

2.3.5 Limitations of current blending methods for plates with a large number of sections

The high stiffness-to-weight ratio and high strength-to-weight ratio of laminated composite plates is accomplished by tailoring the local stiffness and local strength to varying loads in a panel. The size of the sections influences the extent to which the local stiffness can be approached. Increasing the number of sections can lead to a reduction in weight. Therefore, the optimization of a blended plate with a large amount of sections and a varying load distribution necessitates local optimization. Different regions of the plate may require different stiffness properties and thus different stacking sequences. A blending method is required that

allows stacking sequences to completely transform in different regions of a plate. A schematic representation of the required transformation of stacking sequences is presented in figure 2.11. The plate is fully blended while region 1 consist of different types of plies than region 2.

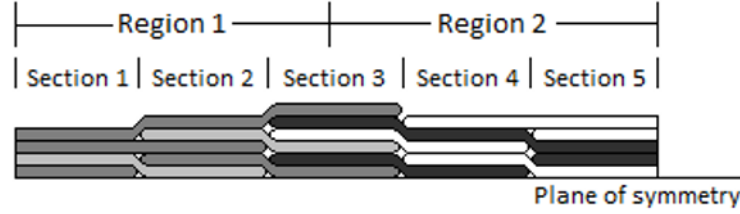


Figure 2.11: A schematic representation of the required transformation of stacking sequences

A blending strategy is required that can handle a large amount of sections and provides the design freedom required for local optimization. Many blending algorithms are based on the inner/outer blending definitions [2, 9, 11, 15, 25, 28]. Inner and outer blending only allows the inner or the outer plies to be dropped. This greatly reduces the design options since all ply drop options throughout the thickness of the laminate are eliminated. The decreased design space impedes local optimization. Generalized blending increase the design space since all plies can be dropped [4]. However, plies cannot be added. Therefore, generalized blending is not suitable for a varying load distribution. The design space of a SST based methods [21, 23, 27] is more suitable for a varying load distribution. However, the SST imposes the same guide laminates to all regions of the plate. It is not possible for two section with the same amount of layers to have a different stacking sequence. Therefore, the SST based methods do not provide the design freedom required for local optimization as depicted in figure 2.11. The relaxed generalized blending definition provides the design space required to approach local loading conditions in a plate [4]. However, implementation of relaxed generalized blending in a guide based method did not provide results [4].

2.4 Conclusions

Blending laminated composite plates is a combinatorial optimization problem that suffers from the curse of dimensionality. The dimensionality of the optimization of composite laminate blended plates is defined by the amount of design solutions. The amount of design solutions increase exponentially with an increase in the number of sections. The number of the sections influences the extent to which the local stiffness of a plate can be approached. Therefore, the optimization of a blended plate with a large amount of sections and a varying load distribution necessitates local optimization. The relaxed generalized blending definition describes the design freedom required to approach the local optima. The state-of-the-art does not provide a method to implement the relaxed generalized blending definition for a large amount of sections.

In conclusion, a blending strategy is required that effectively implements the relaxed generalized blending definition to optimize blended composite laminates with a large amount of sections. The proposed strategy is described in chapter 3.

Overview of the GA-CA-based optimization strategy

In this chapter, an optimization strategy is proposed to overcome the dimensionality in blending laminated composite plates. The optimization of a blended plate with a large amount of sections and a varying load distribution necessitates local optimization as discussed in chapter 2. The proposed blending strategy implements the relaxed generalized blending definition for local optimization. An overview of the proposed optimization strategy is described in section 3.1. The effectivity of the optimization algorithm becomes important when the amount of sections increases as discussed in chapter 2. A method to measure the effectivity of the blending procedure is described in section 3.2. Conclusive remarks are summarized in section 3.3

3.1 The proposed optimization method

In this research an innovative blending algorithm is proposed. The algorithm blends the stacking sequences of a multi-section laminated composite plate with a varying load distribution. The algorithm is designed for plates with a larger amount of sections than is currently found in literature. The global blending constraints are translated to local dependencies. This allows for more design freedom and thereby a smaller weight penalty from the global constraints. In this algorithm a Genetic algorithm (GA) and a Cellular Automaton (CA) are combined to evolve the local optima to a blended configuration. Figure 3.1 shows the flowchart of the GA-CA-based algorithm.

Local optima are the stacking sequences optimized for local loading conditions. These stacking sequences are independent of neighbouring sections. The local optima are produced by a GA. A GA is a widely used optimization tool for discrete design problems with a complex design space [3, 19]. A detailed description of the GA is found in chapter 4. A GA can obtain a (near) optimal solution with limited computational power and is therefore selected to construct the

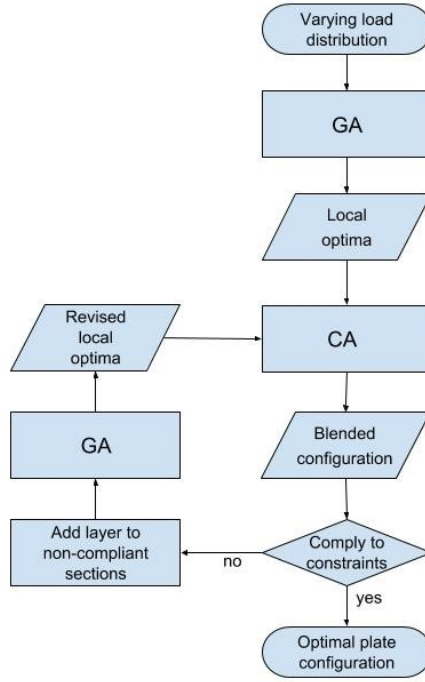


Figure 3.1: Flowchart of the GA-CA-based algorithm

local optima. Other methods for individually optimizing the stacking sequences can be used as well, as further discussed in chapter 7.

The local optima proceed as input for the CA-based blending algorithm. A CA is a discrete model, where cells evolve based on their state and the state of their neighbourhood. All sections of a composite laminated plate are represented by a cell in the proposed CA-based blending algorithm. The state of a cell is defined by its stacking sequence. Each cell evolves according to a straightforward set of rules based on the local dependencies. During each evolution cycle a single ply in each cell can be modified. The CA evolves the stacking sequences towards a blended transition between the cell and its neighbourhood. A plate is assumed locally blended in case all cell transitions comply to a blended transition. A detailed description of the CA-based algorithm is given in chapter 5.

The CA-based algorithm pursues blended transitions with a minimum amount of modifications. All stacking sequences of the blended plate configuration are examined for compliance to the mechanical constraints. In case one or multiple sections are non-compliant, their local optima are revised. A layer is then added to the local optima of the non-compliant sections. The GA provides the revised local optima for these sections. The revised local optima of non-compliant sections and the original local optima of compliant sections are the revised input for the CA. The CA evolves these stacking sequences towards a blended configuration. The extra layer provides a larger margin on the mechanical constraint at the critical sections. The process of adding layers is repeated until all sections in the blended configuration comply

to the mechanical constraints.

The output of the GA-CA-based algorithm is a locally blended laminate configuration. A global interpretation of the laminate is required to establish the global patch configuration. The global patch configuration defines the size of the patches and the layup required for manufacturing. The global patch configuration defines the degree of compliance to the blending guidelines and design constraints on a global level.

3.2 Effectivity of a blending method

Blending laminated composite plates is a combinatorial optimization problem that suffers from the curse of dimensionality. The effectivity in solving the dimensionality in the optimization of blending composite laminates is defined by the problem size (C) and the computational time (T). The amount of fibre orientation options (O) is set at 7, as discussed in chapter 2. Therefore, the size of a design problem is a function of the number sections (N) and the number of layers in the laminate (L) as indicated in equation 3.1.

$$C = 7^{N \cdot L} \quad (3.1)$$

A baseline is established to determine the degree of effectivity of the algorithm, for a varying problem size. The baseline is derived from the expected computational time of an analysis based on enumeration. A logarithmic scale is used, since the expected computational time increases exponentially with the problem size. The baseline is a extrapolation of the time required for the smallest design problem. The computational time of the algorithm and the baseline are normalised with the time required for the smallest design problem.

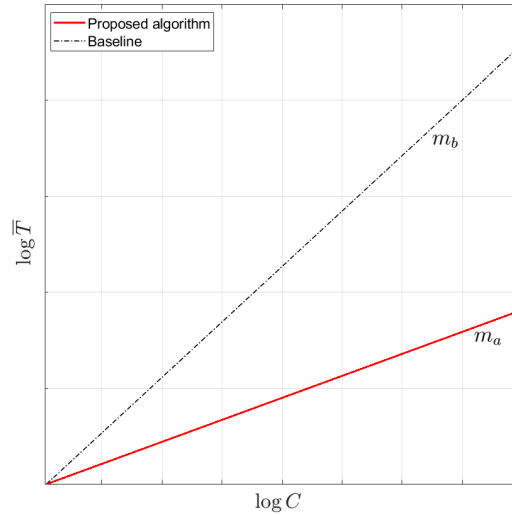


Figure 3.2: A logarithmic plot of the normalized time corresponding to the size of a design problem

The normalized time of the algorithm and the baseline are plotted against the size of the design problem on a logarithmic scale, as indicated in figure 3.2. The degree of effectivity

in solving the dimensionality of the optimization problem can be calculated according to equation 6.2. Where m_a is the slope of the algorithm and m_b is the slope of the baseline.

$$\text{Effectivity} = \frac{m_b}{m_a} \quad (3.2)$$

The dimensionality of a design problem increases exponentially with an increase in the problem size. The computational time of the proposed algorithm is proportional to the amount of sections rather than the amount of design solutions. Therefore, the effectivity of the proposed algorithm increases for design problems with a large amount of sections. The effectivity of the algorithm is further discussed in chapter 6.

3.3 Conclusions

An innovative algorithm is proposed to overcome the dimensionality in blending laminated composites. A GA is used to provide the local optima. The global blending constraints are translated to local dependencies. A CA-based algorithm evolves the local optima into a blended configuration by a straightforward set of rules, which are based on these local dependencies. The proposed algorithm is an iterative process in which layers can be added to ensure all stacking sequences in the blended configuration comply to the mechanical constraints.

The effectivity of a blending procedure is based on the dimensionality of a design problem and the corresponding computational time. The proposed blending strategy is based on local dependencies to increase the effectivity of blending plates with a large amount of sections.

GA for local optimization

The stacking sequence of each section is optimized independently from adjacent sections during local optimization. The locally optimized stacking sequences are input for the cellular automaton. Local optimization is performed by a Genetic Algorithm (GA). A GA is a probabilistic [24] and nondeterministic algorithm, which is based on the Darwinian theory of survival of the fittest. Generations evolve towards a fitter population, by giving fitter individuals a higher probability to procreate. [2, 7, 19]. The flowchart of a GA is depicted in figure 4.1. Just as in the biological evolution process a population consists of multiple individuals. In this case an individual is a design solution and a population is thus a set of design solutions. Initially, the population consists of a randomly created set of individuals. This initial population enters an iterative process, where each iteration represents a generation. All individuals in the population are given a fitness value based on the objective function and design constraints. A higher fitness value indicates a better design. The fittest individuals have the highest probability to be selected as parents during each generation [26]. This increases the likelihood of a fitter next generation. During the reproduction process, the selected parents are modified by genetic operators to create the next generation. This process is repeated until a termination criterion is met [7].

A detailed description of the GA is provided in this chapter. An individual symbolizes a design solution. The chromosomic representation of a design solution is described in section 4.1. The fitness of each individual is evaluated to create a fitness ranking. The fitness evaluation process is described in section 4.2. The population of individuals with corresponding fitness ranking is input for the reproduction process. The reproduction process is described in section 4.3. Tuning of the GA parameters and the verification of the algorithm are described in section 4.4.

4.1 Chromosome representation of an individual

An individual represents a design solution. The discrete variables of an individual, in a GA for local optimization, are the number of plies (L) and their corresponding ply orientations

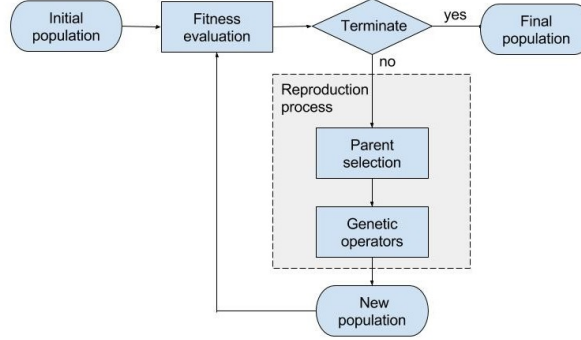


Figure 4.1: Flowchart of the GA

(O). These two variables combined describe the stacking sequence. The stacking sequence is represented by a chromosome based on the encoding scheme of McMahon et al. [20]. A chromosome consists of genes, where each gene describes the ply orientation of a layer. The discrete ply orientations 0° , $\pm 15^\circ$, $\pm 30^\circ$, $\pm 45^\circ$, $\pm 60^\circ$, $\pm 75^\circ$ and 90° are represented by the integer values 1-7. All ply orientations except for 0° and 90° can be positively oriented or negatively oriented. The integer value 0 is used to denote an empty ply. Empty plies are shifted to the end of the chromosome to eliminate invalid designs. An example of the encoding of the stacking sequence of a symmetric and semi-balanced laminate is given below.

$$\left[45/30/-30/0/-45/75/0/-75/90 \right]_s \rightarrow \left[4 \ 3 \ 3 \ 1 \ 4 \ 6 \ 1 \ 6 \ 7 \ 0 \ 0 \ 0 \right]$$

The laminate is assumed to be symmetric and semi-balanced. A chromosome requires only one half of the stacking sequence to describe a symmetric laminate. The sign changes for alternate occurrences of the same orientation in a semi-balanced laminate [20]. The first time an orientation occurs in a chromosome, it is assumed to be positively oriented. The next occurrence of the same orientation is assumed to be negatively oriented and so on. The laminate is completely balanced if the total amount of layers in each direction is even.

4.2 Fitness evaluation of an individual

A GA evolves generations towards a fitter population. To evaluate the fitness of an individual, the objective and the constraints are combined in a fitness function. The fitness function is designed to reward a better design with a higher fitness value. The individuals are ranked according to their fitness values. This fitness ranking is input for the reproduction process. The objective and constraints are described in section 4.2.1. The fitness function and the fitness ranking are described in section 4.2.2.

4.2.1 Objective and constraints

The objective of local optimization is to minimize weight W without violating imposed mechanical constraints. An important mechanical constraint for the optimization of a laminated composite plate is the buckling load, since plates are buckling critical design problems as described in chapter 2. The critical buckling load is dependent on the stacking sequence, therefore, optimization is essential. Other mechanical constraints can be implemented in the fitness function if required. The buckling constraint is an inequality constraint that prohibits the laminated composite plate to buckle under pre-set loading conditions. The method for calculating the critical buckling load is described in appendix A. The inequality constraint for buckling is formulated such that a feasible design has a positive margin as shown in equation 4.1. If $g_{min} \geq 0$ the plate will not buckle and the buckling constraint is satisfied.

$$g_{min} = \frac{N_0}{N_x} - 1 \quad (4.1)$$

In summary, the objective can be described by the objective function

$$\text{minimize } W$$

Subjected to the buckling constraint

$$g_{min} \geq 0.$$

4.2.2 Fitness function and fitness ranking

The objective and the constraints are combined into a fitness function to evaluate the fitness of an individual. The objective is to minimize the weight. The imposed constraint is the buckling constraint which is represented by $g_{min} \geq 0$ as described in section 4.2.1. The fitness function assigns a fitness value θ to each individual. The fitness function is given in equation 4.2.

$$\begin{aligned} \theta &= W + \beta g_{min} & \text{if } g_{min} < 0 \\ \theta &= W + \varepsilon g_{min} & \text{if } g_{min} \geq 0 \end{aligned} \quad (4.2)$$

Where W is the normalized weight, β is the penalty parameter and ε is the bonus parameter. A higher fitness value θ indicates a better design. The penalty parameter β is applied to decrease the fitness value of designs that violate the buckling constraint. The bonus parameter ε is applied on all feasible designs to increase the fitness value of the designs that have a larger margin to the buckling constraint. Tuning of the penalty and bonus parameter is required to reward the best design with the highest fitness value. Tuning of these parameters is further discussed in section 4.4. Each individual in a population is ranked from best to last, based on the assigned fitness value. This generates a fitness ranking from $i = 1 : P$, where P is the number of individuals in a population. The population and the corresponding fitness ranking are the input for the reproduction process.

4.3 Evolution process of a genetic algorithm

The evolution process is an iterative process. Each iteration reproduces a new population. The input of the reproduction process is a population with the corresponding fitness ranking. Parents are selected based on the fitness ranking. Genetic operators are applied to the selected parents to create a child population. Initially, the population consists of a randomly created set of individuals which is further discussed in section 4.3.1. The parent selection process is described in section 4.3.2. A description of the genetic operators can be found in section 4.3.3. The termination criterion of the evolution process is discussed in section 4.3.4.

4.3.1 Initialization of a population

The initial population is a randomly generated set of individuals. An individual is represented by a chromosome with a preset number of genes. Each gene is filled with a randomly chosen ply orientation which is described by an integer value between 1 and 7. Two input variables are required to obtain an initial population; the population size (P) and the number of genes in a chromosome (G). An example of an initial population with P=3 and G=8 is given in table 4.1.

Initial population	P=3, G=8
Individual 1	[5 6 2 4 5 5 3 2]
Individual 2	[1 5 2 6 4 2 6 2]
Individual 3	[3 6 2 6 4 1 3 3]

Table 4.1: An initial population of 3 individuals consisting of 8 genes

The length of a chromosome defines the maximum amount of plies. The amount of genes does not vary during the evolution process. The required amount of genes depends on the optimization problem which is further discussed in section 4.4. The population size remains constant during the evolution process. A large population size increases the amount of genetic information in a population and thereby increases the probability of finding the optimal result. However, an oversized population will slow down the algorithm. The tuning of the population size is described in section 4.4.

4.3.2 Parent selection

The individuals with the highest fitness ranking are given a higher probability to be selected as parents [26]. This increases the likelihood of a fitter next generation. Parent selection is accomplished by a roulette wheel selection process [1,2,24]. An individual with a higher fitness ranking is awarded a larger fraction of the roulette wheel. This increases the probability to be selected as a parent. The fraction of the roulette wheel is determined by the following equation:

$$\text{Fraction of roulette wheel} = \frac{2(P + 1 - i)}{P(P + 1)} \quad (4.3)$$

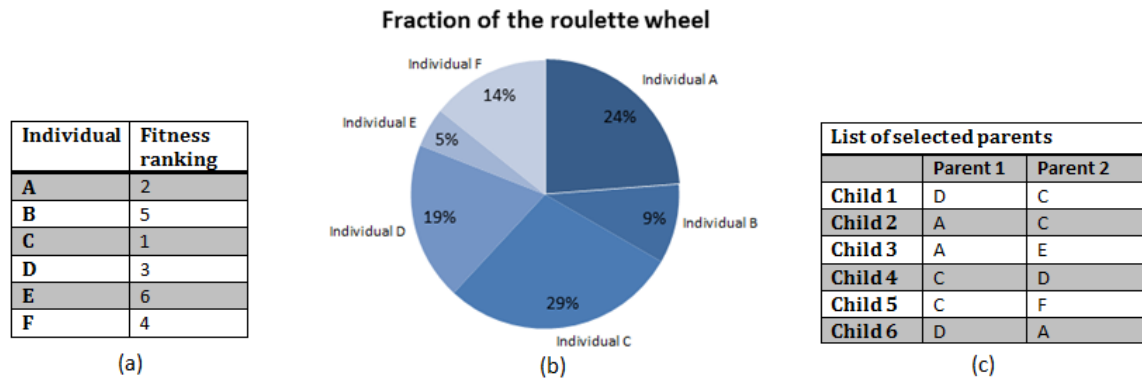


Figure 4.2: Roulette wheel selection process of a population consisting of 6 individuals

Where the individual occupies the i -th place on the fitness ranking in a population of size P . The roulette wheel selection process of a population consisting of $P=6$ individuals is depicted in figure 4.2. The fitness ranking as described in section 4.2.2 is given in figure 4.2a. The fraction of the roulette wheel for each individual according to equation 4.3 is shown in figure 4.2b. A list of parents is generated based on the probability distribution of the roulette wheel as depicted in figure 4.2c. The selection of a parent is independent on previously selected parents, thus an individual can be selected multiple times. The genetic operators require one or two parents to create a child. Therefore, the list of selected parents consists of two parents per child, where the amount of children equals the population size. This ensures sufficient parents to create a next generations of the same population size. A population and the list of selected parents are sent to the reproduction step. During reproduction the chromosomes of the selected parents are combined and/or modified to create a child chromosome.

4.3.3 Genetic operators

A new generation is created in the reproduction process. Reproduction is performed by genetic operators that combine and/or modify parent chromosomes to create a child chromosome. Each operator is applied with a given probability to ensure the algorithm converges to a (near) optimum solution. The first operator is selected based on the application probabilities of the genetic operators. The selected operator is applied on the first set of parents which results in one or two child chromosomes. The child chromosomes are stored in a new population. A second operator is selected and applied to the second set of parents. This process is repeated until the population size is reached and thus a new population is created. Commonly used genetic operators are crossover, mutation, permutation, swap, ply addition, ply deletion, and elitist selection [2, 19, 20, 26]. The techniques of the operators are described below.

Crossover

Multiple crossover techniques exist. This algorithm uses the most common crossover technique: single point crossover [26]. This genetic operator is applied to two parents and generates two child chromosomes. A single crossover point is selected for both parents. The crossover point is a randomly selected location on the chromosome. The crossover point

splits the parents in two parts. The first child is created by combining the left part of parent 1 and the right part of parent 2. The combination of the remaining parts results in the second child. The single point crossover technique is depicted in the following example.

Parent		Child
[436 145127000]	→	[436 723412300]
[453 723412300]	→	[453 145127000]

Mutation

The mutation operator modifies a single parent to generate a child. A gene at a randomly selected locus is replaced by a new gene. The new gene is a randomly generated gene, thus it consists of a randomly chosen integer value between 1 and 7. This is illustrated in the following example, the gene at locus 3 is replaced by a randomly generated gene.

Parent		Child
[43 <u>6</u> 145127000]	→	[43 <u>2</u> 145127000]

Permutation

Ply permutation is a genetic operator that is applied to a single parent. Ply permutation reorders part of the gene sequence in the chromosome. Two random locations on the chromosome are selected. The sequence of the genes in between these locations is flipped. The chromosome with the reordered part of the gene sequence represents the child chromosome as depicted in the example below.

Parent		Child
[43 6145 127000]	→	[43 5416 127000]

Swap

Swap is a genetic operator that is applied to a single parent. The genes at two randomly chosen loci in the chromosome are swapped to create the child chromosome. The example below depicts this genetic operator.

Parent		Child
[<u>4</u> 3614 <u>5</u> 127000]	→	[<u>5</u> 3614 <u>4</u> 127000]

Ply addition

Ply addition is applied to a single parent. Ply addition adds a randomly generated gene at a random locus in the chromosome. Ply addition is depicted in the following example.

Parent		Child
[4361 45127000]	→	[4361 24512700]

Deletion

Ply deletion is applied to a single parent. A randomly chosen gene is deleted from the parent chromosome as shown in the example below.

Parent		Child
[43 <u>6</u> 145127000]	→	[431451270000]

Elitist selection

Elitist selection is implemented to ensure the fittest individuals are preserved. The lowest ranked individuals of the child population are replaced by the highest ranked individuals from the parent population.

Each genetic operator has an application probability. Crossover is usually applied with a high probability, since the primary function of crossover is to explore the design space. [26] Mutation prevents the loss of genetic information, thereby preventing a population to become uniform which may result in a local optimum. Ply addition, ply deletion and ply permutation can be applied with a small probability to increase the efficiency of the algorithm [26]. Elitist selection prevents the loss of the fittest individuals. However, if the number of preserved individuals is too high, a population may become uniform and converge to a local optimum. The tuning of the application probabilities of these parameters is further discussed in section 4.4.

4.3.4 Termination criteria

The evolution process is an iterative process. The process is terminated when a maximum number of generations is reached. The population should be converged to the global optimum within this number of generations. The rate of converging is dependent on the population size and application probabilities of the genetic operators. Determination of maximum number of generations is further discussed in section 4.4.3.

4.4 Parameter settings and verification of the GA

Tuning of the parameters is performed for each optimization problem. Tuning is required for the algorithm to converge to the global optimum and to increase the efficiency of the algorithm. Tuning is performed in three steps. First the parameters of the fitness function are tuned, which is described in section 4.4.1. The second step can be found in section 4.4.2, where the application probabilities of the genetic operators are tuned. The third step is described in section 4.4.3. The third step contains the parameter settings that bound the evolution process in order to limit the computational time and effort. The verification of the algorithm is described in 4.4.4.

4.4.1 Parameter settings of the fitness function

Tuning of the parameters in the fitness function is required to ensure the algorithm converges to the global optimum. A review of the fitness function is given in equation 4.4, where the critical buckling load is the imposed mechanical constraint. The inequality constraint for buckling is formulated such that a feasible design has a positive margin as indicated by $g_{min} \geq 0$.

$$\begin{aligned} \theta &= W + \beta g_{min} & \text{if } g_{min} < 0 \\ \theta &= W + \varepsilon g_{min} & \text{if } g_{min} \geq 0 \end{aligned} \quad (4.4)$$

All individuals in the initial population have the maximum amount of layers and thereby the maximum weight. During the evolution process, the weight of the individuals decreases while the algorithm converges to the optimal laminate. In the fitness function the weight is normalised with the maximum weight. The fitness value increases while the weight decreases. When the constraints are no longer satisfied, the penalty parameter decreases the fitness value. The penalty parameter should be high enough to counteract the increase in fitness value due to the weight reduction. The bonus parameter awards feasible individuals with the highest margin. An excessive bonus parameter impedes the convergence of the algorithm since gaining a larger margin would have more effect on the fitness value than minimizing the weight. The values of the normalized weight and the margins vary for different optimization problems. Therefore, the values of the bonus parameter and the penalty parameter are problem dependent. Tuning of the parameters is described for the example problem given in figure 4.3.

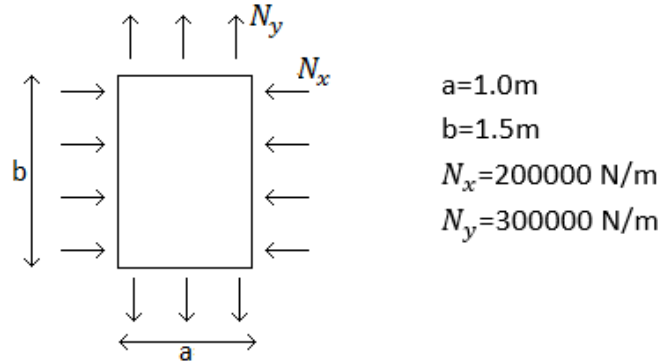


Figure 4.3: Panel used for tuning the parameters

The optimal laminate is determined by enumeration to demonstrate the behaviour of the GA for a range of values for the bonus and penalty parameters. The optimal laminate is:

$$\begin{aligned} \text{Optimal laminate} & \quad [30^\circ / -30^\circ / 30^\circ / 0^\circ / 45^\circ / 90^\circ]_s \\ \text{Chromosomic representation:} & \quad [333147] \end{aligned}$$

The application probabilities of the genetic operators influence the required population size and the number of generations. The application probabilities implemented for the tuning

of the fitness function parameters are based on literature [2, 26]. The tuning of the application probabilities is performed in section 4.4.2. To account for the disruption of the GA that originates from the implemented application probabilities, tolerances are applied on the population size and the number of generations. A GA is a probabilistic algorithm; therefore, multiple runs are required to evaluate the results. A variety of combinations between the penalty parameter and the bonus parameter are analysed. The results are generated for a population of 100 individuals after 50 generations. The algorithm is executed 100 times for each combination of variables. The number of runs that results in the optimal laminate is indicated in table 4.2.

	$\varepsilon = 0.00001$	$\varepsilon = 0.0001$	$\varepsilon = 0.001$	$\varepsilon = 0.01$	$\varepsilon = 0.1$
$\beta = 1$	0	0	0	0	0
$\beta = 2$	0	0	0	94	0
$\beta = 3$	88	93	96	88	0
$\beta = 4$	90	89	89	91	0
$\beta = 5$	92	97	93	95	0
$\beta = 6$	88	92	90	98	0
$\beta = 7$	93	93	96	92	0
$\beta = 8$	95	96	92	89	0
$\beta = 9$	93	95	93	91	0
$\beta = 10$	92	90	88	88	0

Table 4.2: Amount of runs that result in the optimal laminate from the total of 100 runs

Table 4.2 can be divided in three regions. All runs in region 1 are converged to designs that violate the buckling constraint. The majority of runs in region 2 resulted in the global optimum. The remaining runs in region 2 resulted in near optimal designs. In region 3, the algorithm did not converge.

Region 1 indicates that penalty parameter has a the lower limit of $\beta = 3$, except for the case where $\varepsilon=0.01$. In this case the penalty parameter is not high enough to counteract the effect of the weight reduction. However, the bonus parameter of $\varepsilon=0.01$ is high enough to increases the fitness value for the feasible designs, so the feasible design are ranked higher than the infeasible designs. Therefore, the algorithm does converge to a (near) optimal design with this combination of values.

Region 3 indicates the bonus parameter has an upper limit of $\varepsilon=0.01$. In case the bonus parameter is $\varepsilon=0.1$, the last generation of all runs contains individuals consisting of the maximum amount of layers. This indicates that the algorithm did not converge, since the effect of a higher margin is larger than the effect of minimizing the weight.

All combination of values in region 2 result in the correct fitness ranking. Note that region 2 may be limited by the amount of significant figures that are included during computation of the fitness value. In case the amount of significant figures used during computation is insufficient, multiple designs have the same fitness value. In order to avoid this, minimization of the amount of significant figures of the parameters is preferred. For this example a combination of values that satisfy these conditions is $\beta=5$ and $\varepsilon=0.01$.

4.4.2 Parameter settings of the genetic operators

The application probabilities of the genetic operators are tuned to ensure convergence to the global optimum and to increase the efficiency of the GA. The sum of all application probabilities is one. A variety of application probabilities is analysed. A GA is a probabilistic algorithm; therefore, 100 runs are performed for each set of application probabilities. The set of application probabilities that provided the most stable and efficient results is given in figure 4.4.

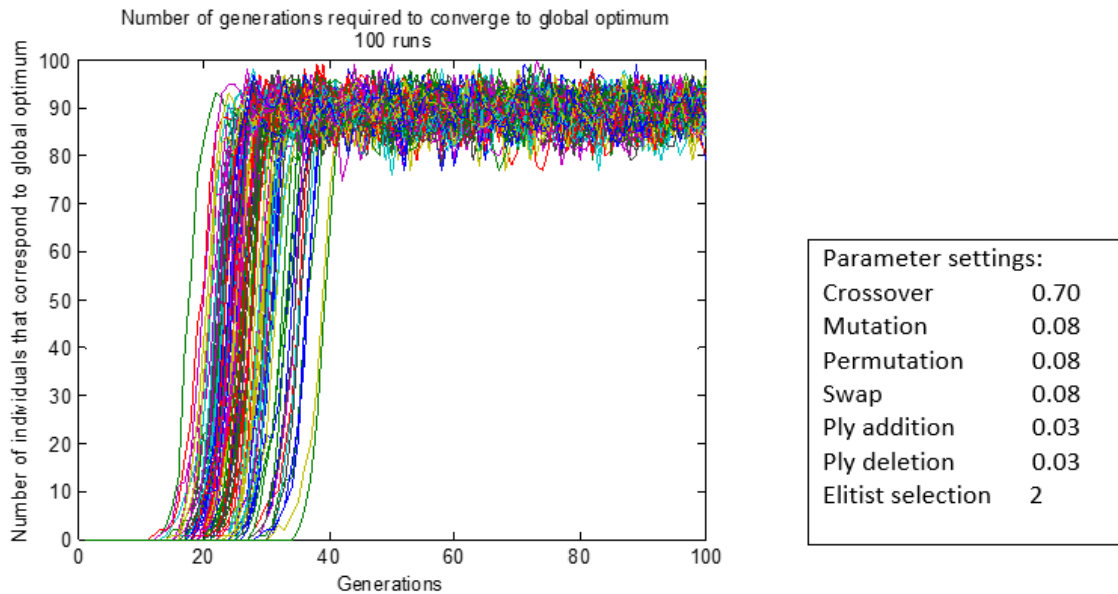


Figure 4.4: The set of application probabilities that provides the most stable and efficient results

All runs converge to the global optimum as illustrated in figure 4.4. The generation in which the first global optimum occurs ranges from 12-34, due to the probabilistic nature of the GA. The noise after convergence is due to the implementation of new genetic information. The majority of individuals remain the global optimum since no fitter individuals exist. The largest effect is observed when varying the crossover application probability. A higher crossover probability increases the range of first occurrences with a higher average number of required generations. A low crossover probability resulted in an increased the bandwidth of the noise which impedes differentiation between the converged optimum and the noise. Adjustment of the ply deletion probability alters the lowest number of required generations, while the highest number of required generations remains equal. Adjustment of the mutation, permutation or swap probability affects the noise bandwidth and the range of first occurrences. However, this effect is small and the GA converged to the global optimum with all analysed probabilities.

4.4.3 Settings that bound the evolution process

Bounds are set to the evolution process to limit the computational effort. These bounds are represented by the chromosome size, the population size and the number of generations. The size of the chromosome limits the amount of layers in a laminate. The required number

of layers is not known beforehand. The required number of layers is based on comparable laminates including a safety margin. The GA starts analysing laminates with the maximum amount of layers and converges to the required amount of layers. For the example problem depicted in figure 4.3, the chromosome consist of 10 genes and the optimal laminate can be described by 6 genes.

A larger population contains more genetic information. Therefore, the population size affects the amount of generations required to converge to the global optimum. The required amount of generations for a population consisting of 50 individuals and a population consisting of 100 individuals are depicted in figure 4.5. For a population of 50 individuals, 70 generations are required for all runs to converge to the global optimum. For a population of 100 individuals, the maximum required number of generations is reduced to 43. The computational effort required for both settings is similar. The stability of the algorithm with a population size of 50 is higher since the range in which the algorithm converges to the global optimum is smaller. This increases the reliability of the algorithm. Further increasing the population size up to 200, decreases the maximum amount of required generations to 38. The range of required generations does not further decrease, so the reliability remains similar. The reduction of 5 generations does not outweigh the increase of computational effort. For this optimization problem the populations size is set to 100 individuals and the evolution process is terminated after 50 generations.

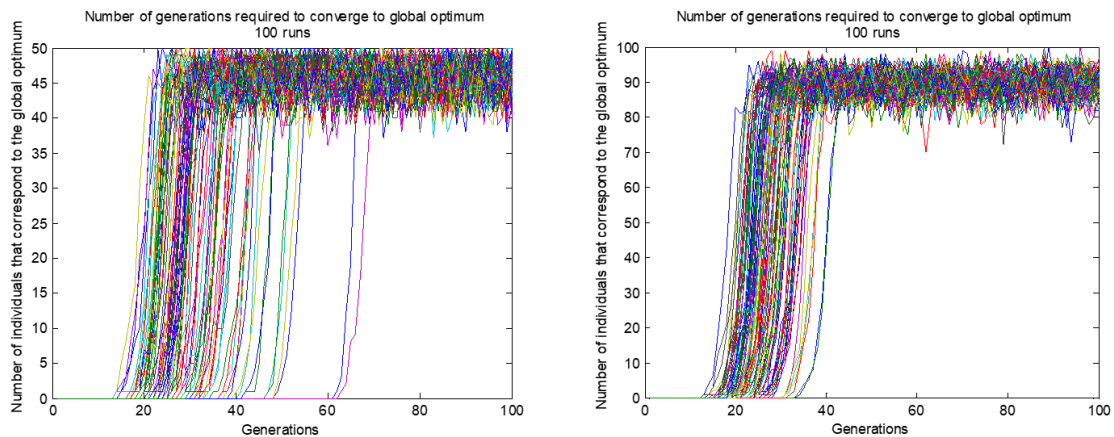


Figure 4.5: The amount of generations required to converge to the global optimum for different population sizes

4.4.4 Verification of the GA

The GA is verified by using a section from the horseshoe pattern benchmark problem. The horse shoe pattern benchmark problem is elaborated in chapter 6. The section that used for the verification is illustrated in figure 4.6. The buckling constraint g_{min} is the applied mechanical constraint as discussed in section 4.2. The GA parameter settings are provided in table 4.3.

The algorithm is probabilistic and may provide multiple design solutions during multiple runs. The algorithm is executed 100 times to evaluate the results. The design solution of

Application probabilities	
Crossover	0.75
Mutation	0.05
Permutation	0.05
Swap	0.05
Ply addition	0.05
Ply deletion	0.05
Elitist selection	1

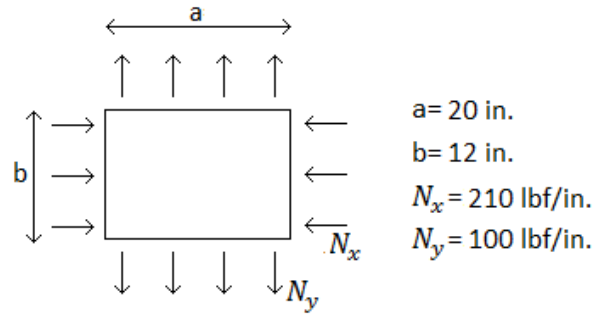


Table 4.3: GA parameter settings

Figure 4.6: Panel used for verification

a run is defined as the stacking sequence with the highest buckling margin g_{min} in the final population. The design solutions for a population of 150 individuals after 100 generations are presented in table 4.4.

Results	Amount of runs out of the 100 performed runs	Stacking sequences	g_{min}
Design solution 1	78	$[60^\circ / \pm 60^\circ / 75^\circ / -75^\circ / -60^\circ / 75^\circ / 60^\circ]_s$	0.2186
Design solution 2	22	$[\pm 60^\circ / 75^\circ / \pm 60^\circ / 60^\circ / \pm 75^\circ]_s$	0.2182

Table 4.4: Design solutions provided by the GA in 100 runs

The GA provided design solution 1 in 78 out of the 100 performed runs. Design solution 1 is equal to the optimum stacking sequence provided in literature [1,2]. The algorithm resulted 22 times in design solution 2. The stacking sequence of both design solutions consist of 16 layers. The critical margin of the design solution 2 is comparable to the critical margin of the optimum design solution. Therefore, design solution 2 is a near optimal design solution. Design solution 2 has a higher critical margin compared to near optimum solutions presented in literature [26].

The algorithm performed 100 runs for the example illustrated in figure 4.6. The algorithm provided either design solution 1 or design solution 2. Therefore, the algorithm resulted in a (near) optimal solution during each run, which concludes the verification of the algorithm.

CA-based blending method

A CA is a discrete model where cells evolve based on their state and the state of their neighbourhood. The proposed CA-based blending algorithm translates the global blending constraints to local dependencies. The sections of a multi-section laminated composite plate are represented by cells. The state of a cell is initially defined by the local optimum. The CA evolves these stacking sequences towards a blended transition between the cell and its neighbourhood. A flowchart of the CA-based blending algorithm is depicted in figure 5.1. The algorithm consist of three stages with different objectives. Each cell evolves towards the objective of the stage, based on a straightforward set of rules. During each evolution cycle a single ply in each cell can be modified. This provides a homogeneous distribution of modifications throughout the plate. The algorithm pursues blended transitions with a minimum amount of modifications. A plate is assumed locally blended in case all cell transitions comply to a blended transition. The interpretation of the laminates at a global level is required to define the layup for manufacturing. The representation of a multi-section plate in a CA is described in section 5.1. The evolution strategy of a cell is presented in section 5.2. The global patch interpretation is described in section 5.3. The verification of the CA-based algorithm can be found in section 5.4, including the verification of the the combined GA-CA-based algorithm.

5.1 Representation of a multi section plate in a CA

A 3D representation of a plate divided in 12 sections is shown in figure 5.2. The plate is divided in rectangular sections. A rectangular grid corresponds to blending problems currently found in literature and is therefore used to demonstrate the algorithm. Other patterns can be implemented in this algorithm if required. The stacking sequence of each section is initially described by the local optima. The stacking sequences of cells evolve towards a blended transition between the cell and its neighbourhood. Ideally, all cells evolve simultaneously to obtain a homogeneous evolution throughout the plate. A homogeneous evolution prevents prioritization of cells or regions. However, the dependency between neighbouring cells prohibits simultaneous evolution of all cells. For a regular grid, a chessboard pattern is applied

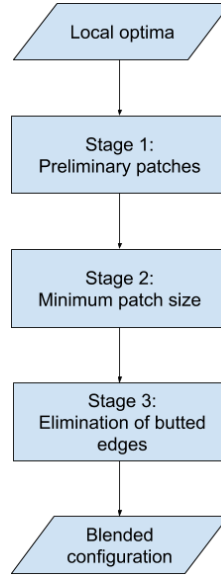


Figure 5.1: Flowchart of the CA-based algorithm

to divide the cells in groups of independent cells. These groups evolve successively to obtain a homogeneous evolution without simultaneous evolution of interconnected cells. The division of cells according to a chessboard pattern is further discussed in section 5.1.1.

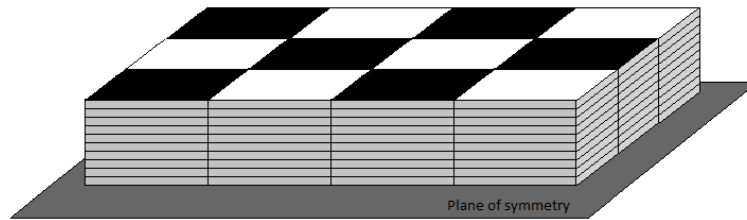


Figure 5.2: A 3D representation of a plate in which the cells are divided according to a chessboard pattern.

The cells are divided in groups of independent cells. As a consequence, the neighbourhood of a cell remains unaltered during the evolution of a cell. A schematic overview of a cell and its neighbourhood is presented in figure 5.3. Cell X represents the analysed cell. The neighbourhood of cell X consists of 4 cells. The evolution of a cell consists of the modification of a ply in the stacking sequence. All plies are analysed to determine which ply modification is preferred. The amount of layers included in the analysis of a ply limits the utilized data during the analysis. The utilized data restricts the freedom of a patch, which results in the applied definition of a patch. The amount of layers included in the ply analysis is further discussed in section 5.1.2.

5.1.1 Division of the cells according to a chessboard pattern

The dependency between cells and their neighbourhood prohibits simultaneous evolution of all cells. Cells are divided in groups to separate interconnected cells. The division of cells

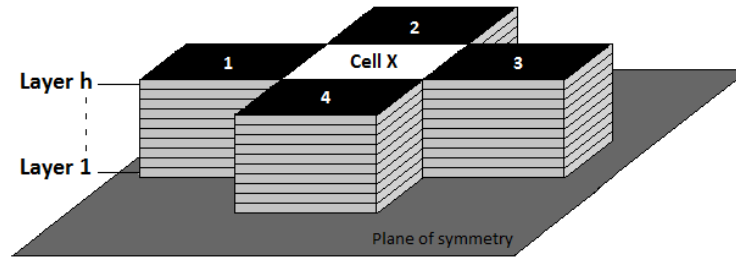


Figure 5.3: A 3D representation of a cell and its neighbourhood

in a rectangular grid is based on a chessboard pattern. The neighbourhood in a layout with varying cell dimensions is defined in section 5.1.1.1. The classification of cells according to a chessboard pattern is described in section 5.1.1.2. The evolution cycle based on the classification of the cells is described in section 5.1.1.3.

5.1.1.1 Neighbourhood

Each cell is dependent on a set of adjacent cells, which is called the neighbourhood. Commonly used neighbourhood definitions in a CA are the von Neumann neighbourhood and the Moore neighbourhood [13], depicted in figure 5.4a and figure 5.4b respectively. These neighbourhood definitions are based on a regular grid. A regular grid restricts the cell layout of a plate. Therefore, these neighbourhood definitions do not represent a viable definition for the blending algorithm. A plate is assumed blended if all sections are blended with all adjacent sections. Therefore, the neighbourhood of the proposed blending method is defined as; all neighbours that share a boundary with the cell. The proposed neighbourhood definition is depicted in figure 5.4c.

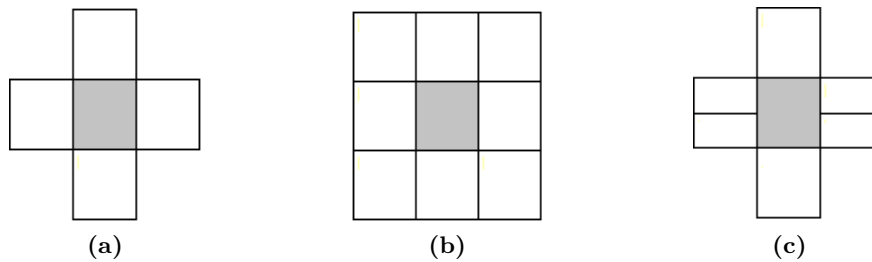


Figure 5.4: (a) von Neumann neighbourhood (b) Moore neighbourhood (c) Applied neighbourhood

A cell layout with varying cell dimensions is depicted in figure 5.5. The neighbourhood of these cells is given in table 5.1. The amount of neighbours varies per cell, since the layout is not restricted by a regular grid.

5.1.1.2 Chessboard patterns

The dependency between adjacent cells impedes simultaneous evolution of all cells. The division of cells in a rectangular grid is based on a chessboard pattern. Cells with similar

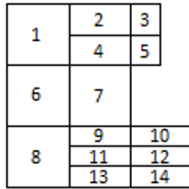


Figure 5.5: An irregular cell layout

Cell	Neighbourhood
1	2,4,6
2	1,3,4
3	2,5
4	1,2,5,7
5	3,4
6	1,7,8
7	4,6,9

Cell	Neighbourhood
8	6,9,11,13
9	7,8,10,11
10	9,12
11	8,9,12,13
12	10,11,14
13	8,11,14
14	12,13

Table 5.1: The neighbourhood corresponding to the cell layout in figure 5.5

colours are independent of each other and can evolve simultaneously. A rectangular grid of 16 cells is illustrated in figure 5.6. The cells are classified according to a chessboard pattern. The cells are either black or white. At time 0, all black cells evolve while all white cells remain in their initial condition. At time interval 1, all white cells evolve, while the black cells remain in their current state. This evolution cycle continues over time. A homogeneous evolution of the plate is established without simultaneous evolution of interconnected cells.

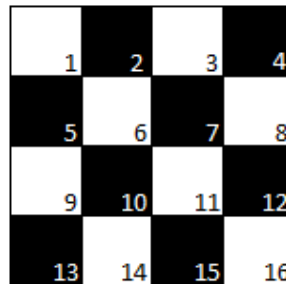


Figure 5.6: A 2D representation of a plate in which the cells are divided according to a chessboard pattern

The example depicted in figure 5.6 is based on a regular grid. The layout of a laminated composite plate may not be restricted to a regular grid. A single chessboard pattern is insufficient, to divide the cells in groups of independent cells, for an irregular grid. Therefore, a multilevel grid is implemented, where each grid contains a chessboard pattern. The first level grid is defined as the smallest grid with straight continuous lines that start and end at the periphery of the plate. Higher level grids are implemented if the cell dimensions vary throughout the plate. Each cell of the first level grid can contain a second level grid. A second level grid is defined in the same way as the first level grid, where the periphery is bound by the borders of the cell it is implemented in. Third or higher level grids can be implemented if required. An example of a plate layout that requires a two level grid is illustrated in figure 5.7.

A schematic representation of a plate divided into 14 sections is presented in figure 5.7a. The first level grid, according to the above mentioned definition, is shown in figure 5.7b. Since not all cells are defined by this grid, two types of second level grids are implemented. A two

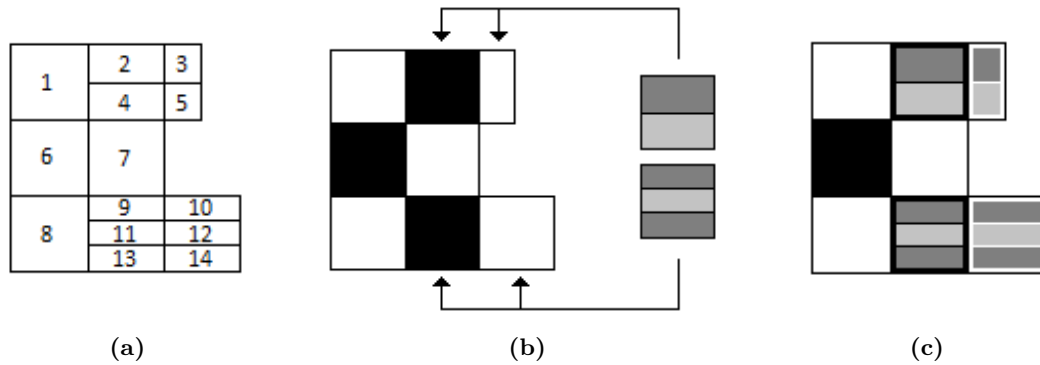


Figure 5.7: (a) Cell layout with varying cell dimensions, (b) 1st and 2nd level grids, (c) The two level grid in which all cells are defined

level grid is sufficient to define all cells according to figure 5.7c. The division of cells is based on the colours of the chessboard patterns. The division of cells is further described in section 5.1.1.3.

5.1.1.3 Division of cells

The division of cells prevents simultaneous evolution of interconnected cells. The cells are divided in groups of independent cells. These groups evolve successively during each evolution cycle. The group allocation is based on the assigned chessboard colours. The evolution cycle of a single chessboard pattern consists of two groups, first all white cells evolve after which all black cells evolve. The evolution cycle of a two-level grid is presented by the scheme in figure 5.8. The evolution cycle consists of 4 groups. The cells are sorted according to the combination of chessboard colours at each level.

One evolution cycle:	Group 1	Group 2	Group 3	Group 4
Grid level 1				
Grid level 2				
Evolve cells:	1,5,7,8,12	3,10,14	4,6,11	2,9,13

Figure 5.8: The evolution cycle of the plate layout presented in figure 5.7

The groups do not contain interconnected cells. The cells in each group are scattered over plate. Therefore, no cells or regions of the plate are prioritized. All cells evolve a single time during one evolution cycle. In conclusion, the division of cells based on a chessboard pattern results in a homogeneous evolution of the plate without simultaneous evolution of interconnected cells.

5.1.2 Range of layers included in the analysis of a ply

A single ply in the stacking sequence of a cell can be modified during an iteration of the evolution process. All plies are analysed to determine which ply is modified. The amount of surrounding layers included in the analysis of a ply, limits the amount of utilized data. The design space to blend a configuration follows from the utilized data. More freedom to form a blended transition may be beneficial to decrease the weight of the plate. However, the complexity to obtain a blended configuration increases. As a result, larger modifications or backtracking may be required to obtain a blended configuration. This may be counter-productive in the minimization of the weight and increases the computational effort of the algorithm. The optimum amount of included layers depends on the scenario that is pursued. The pursued scenario varies in each stage of the algorithm. The identifiable scenarios corresponding to the range of included layers are discussed in section 5.1.2.1. The optimum range is described in section 5.1.2.2 for all stages of the algorithm.

5.1.2.1 Scenarios corresponding to the number of included layers

The identifiable scenarios depend on the number of layers included in the analysis of a ply. A schematic representation of the transition between a cell and one of its neighbours is presented in figure 5.9. The included amount of layers is restricted to a single layer. The analysed ply is outlined. Three types of scenarios can be distinguished with the provided data. In the first scenario, the neighbouring ply has the same ply orientation as the analysed ply. This indicates the analysed ply is part of a continuous patch. The remaining scenarios indicate the discontinuation of a patch. With the limited amount of data, the edge is interpreted either as a butted edge or as a free edge. The edge is interpreted as a butted edge in case the analysed ply and the neighbouring ply have a different fibre orientation. The neighbouring ply of a free edge is an empty ply.

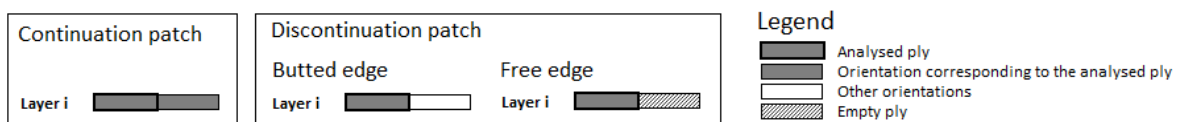


Figure 5.9: A schematic representation of the transition between two cells in case the ply analysis is restricted to a single layer

A patch may occupy multiple layers. The discontinued patch in figure 5.9 may be part of a patch that is present in an adjacent layer. This scenario can only be identified in case the included amount of layers is increased. The scenarios which can be distinguished with three included layers are shown in figure 5.10. Figure 5.10 consist of a larger amount of scenarios compared to figure 5.9. According to these scenarios, there are more options to continue the patch. Three layers provide sufficient data to differentiate between butted edges and covered edges according to figure 5.10. The amount of layers should be further increased to identify multiple ply drops at the same location.

The identifiable scenarios depend on the number of layers included in the analysis of a ply. Different stages in the evolution process pursue different scenarios and thereby require a different amount of layers. The applied range of included layers, for all stages of the evolution process, is discussed in section 5.1.2.2.

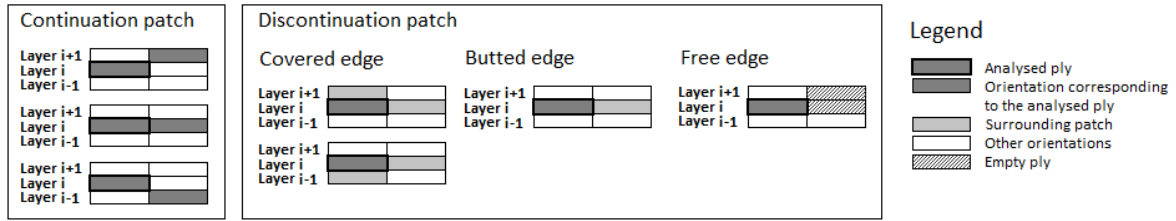


Figure 5.10: A schematic representation of the transition between cells in case three layers are included in the analysis of a ply

5.1.2.2 Optimum number of included layers for each evolution stage

The range of included layers determines the type of scenarios that can be distinguished during the analysis of a ply. The pursued scenarios differ for each stage of the evolution process in the proposed method. Therefore, the amount of included layers is adapted for each stage of the algorithm. The stages in the proposed algorithm consist of the assembly of preliminary patches, the minimum patch size and the elimination of butted edges. An elaborate description of the stages in the evolution process can be found in section 5.2. The applied range of included layers for each evolution stage is described below.

Foundation of patches: The assembly of preliminary patches is performed in a single layer. A straightforward set of rules is established which ensures a homogeneous distribution of modifications as described in section 5.2.1. The applied set of rules ensures all plies to become part of a patch. A single layer is sufficient to assemble preliminary patches. A higher number of included layers increases the amount of options for patch continuation as discussed in section 5.1.2.1. Consequently, it provides more design freedom to form a blended transition. However, with the applied method, the accompanying complexity is counterproductive in obtaining a blended plate configuration. The early created patches restrict the growth area of prospective patches, which may result in a scattered distribution of patches and isolated plies. Large modifications may be required to match the isolated plies to surrounding patches. This is counterproductive in the minimization of the alterations. Another drawback of isolated plies is the modification distribution throughout the plate. The distribution of modifications is not homogeneous since majority of the modification steps result from the isolated plies.

Minimum patch size: Manufacturing constraints include a minimum patch size. For 2D problems the minimum patch size can be seen as the minimum length of a patch. A single layer is sufficient in the applied method as described in section 5.2.2. However, including more layers may lower the amount of modification required. In a 2D problem a range of three layers provides blended results for 2D plates with realistic stacking sequence transitions. However, the probability of fractionation increases which may be counterproductive in the implementation of a minimum patch size. For 3D problems the patch size is defined by multiple parameters; the amount of cells and the distribution of the cells. The minimum patch size can be seen as the minimum cross-sectional length. Compact patches without extremities are preferred. The 2D strategy is extended to obtain method suitable for 3D problems. The amount of included layers is restricted to a single layer for the 3D method. A larger range may fractionate small patches, thereby increasing the amount of extremities.

Elimination of butted edges: A range of three layers is included in the analysis of a ply to eliminate butted edges. Three layers is the minimum amount of layers required to differentiate between butted and covered edges as discussed in section 5.1.2.1. Three layers restrict the amount of ply drops at the same location to one. This results in the definition of a patch during evolution for the proposed algorithm; a patch can continue in the same layer or an adjacent layer during each transition. Increasing the range of layers increases the maximum amount of ply drops at the same location. This can be beneficial for the weight minimization, however, it also increases the complexity to evolve to a blended configuration. A range of three layers is implemented in this algorithm to prove the concept of this method. Further research is required to investigate methods that allow for more flexibility concerning the amount of ply drops at the same location during evolution.

5.2 Evolution strategy

The evolution of a cell is divided in three stages. These stages pursue different objectives; foundation of patches, minimal patch size and the elimination of butted edges. Each stage provides a straightforward set of rules. These rules modify a ply in the stacking sequence of the analysed cell. A ply is only modified in case it increases the amount of blended transitions between a cell and its neighbours. Cells that are in a higher stage of blending are less likely to be modified. All cells proceed simultaneously to the succeeding stage to obtain homogeneous distribution of modification over the cells. Further research is required to combine all three stages in a single set of rules while maintaining a homogeneous distribution of modifications, which is further discussed in chapter 7. The first stage provides the foundation of the patches. The first stage is described in section 5.2.1. The algorithm proceeds to the second stage when all plies are part of a patch. During the second stage a minimum patch size is established. The second stage is discussed in section 5.2.2. The third stage identifies and eliminates butted edges, which is described in section 5.2.3.

5.2.1 Foundation of Patches

The first stage of the algorithm lays the foundation of patches. All plies become part of a preliminary patch during this stage. A preliminary patch is defined by all interconnected plies with the same fibre orientation within a single layer of the laminate. One ply in the stacking sequence of a cell can be modified during each evolution cycle. A modification consists of one step in orientation. Consequently, the modification options of each ply consist of one orientation step in the positive direction or one orientation step in the negative direction. A genotype is created, which is a representation of the stacking sequences of the analysed cell and the neighbourhood. The genotype is used to prioritize the modification options in a cell. The selected modification is subsequently applied to the stacking sequence of the analysed cell. The construction of the genotype is described in section 5.2.1.1. The motivation for the proposed genotype and the application are further elaborated in section 5.2.1.2.

5.2.1.1 Construction of the genotype

A genotype is constructed to prioritize the modification options in a cell. The genotype is a representation of the stacking sequences of the analysed cell and the neighbourhood. In this section, the transformation of the stacking sequences required to obtain the genotype is described. The description of the transformation is divided in two steps to clarify the underlying principle. In the first step the stacking sequences are transformed into a deviation scheme. The deviation scheme provides an overview of the amount of neighbouring plies per deviation distance. This deviation scheme is subsequently transformed to construct the genotype. The stacking sequences presented in table 5.2 are used to demonstrate construction of the genotype. Cell X represents the analysed cell. The stacking sequences consist of fibre orientations represented by integers 1-7 and empty layers represented by integer 0 as discussed in chapter 2.

Layer	Cell X	Neighbour			
		1	2	3	4
7	3	6	7	0	7
6	6	6	6	3	6
5	7	3	3	5	1
4	2	2	1	3	4
3	5	5	5	6	6
2	4	3	1	4	1
1	7	2	7	2	2

Table 5.2: Stacking sequences of the analysed cell and its neighbourhood

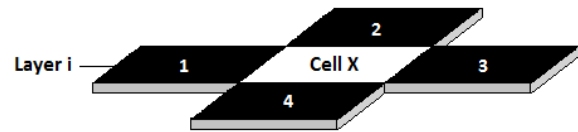


Figure 5.11: A representation of all plies included in the analysis of a ply

The deviation scheme is constructed from the stacking sequence of the analysed cell and the stacking sequences of the neighbourhood. The amount of layers included in the analyses of a ply is restricted to a single layer as described in section 5.1.2.2. Figure 5.11 provides a schematic representation of a single layer in cell X and the neighbourhood. A ply is represented by one of the 7 discrete fibre orientations. Therefore, one step in fibre orientation corresponds to a change of 15 degrees. The amount of steps required for the analysed ply to obtain a similar fibre orientation as a neighbouring ply is defined as the deviation. The deviation varies between 0 - 6 steps. The deviation scheme indicates the amount of neighbouring plies per deviation distance as shown in table 5.3. The direction of the deviation can be either positive or negative. The fibre orientations are equal in case the deviation is 0. In all other cases the deviation is indicated by the amount of steps, represented by I-VI.

The deviation scheme presented in table 5.3 is constructed from the stacking sequences in table 5.2. The transformation of layer 1 is elaborated to demonstrate the construction of the deviation scheme. A schematic overview of layer 1 is given in figure 5.12. Layer 1 of the analysed cell consists of a ply with fibre orientation 7. The neighbouring plies have fibre orientations 2, 7, 2 and 2. Accordingly, one neighbouring ply has an identical fibre orientation. This ply is represented by a 1 in the column that indicates the amount of neighbours with equal fibre orientations. The remaining three neighbouring plies consist of fibre orientation

Cell X	Amount of neighbouring plies per deviation distance													
	Equal	I step		II steps		III steps		IV steps		V steps		VI steps		
		+	-	+	-	+	-	+	-	+	-	+	-	
Layer 7						1		2						
Layer 6	3						1							
Layer 5					1				2				1	
Layer 4	1	1	1	1										
Layer 3	2	2												
Layer 2	1		1				2							
Layer 1	1										3			

Table 5.3: The deviation scheme corresponding to the stacking sequences in table 5.2

2. The analysed ply has a fibre orientation 7 and therefore requires a deviation of 5 steps in the negative direction to obtain fibre orientation 2. These three neighbouring plies are indicated by a 3 in the column that specifies the amount of neighbouring plies that require a deviation of V steps in the negative direction. This concludes the transformation of layer 1. The empty cells in table 5.3 indicate the absence of neighbouring plies for the corresponding deviation distance. Empty layers in the stacking sequences are not represented in the scheme which is apparent in layer 7. According to table 5.2, layer 7 of neighbour 3 is an empty layer. Therefore, layer 7 contains three neighbouring plies in table 5.3, while the total amount of neighbouring cells is four.

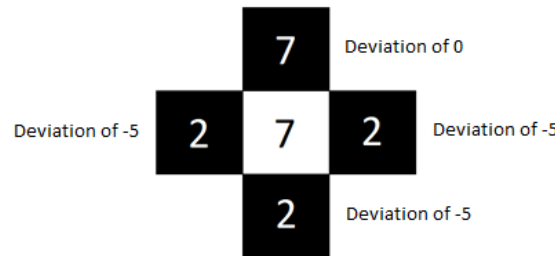


Figure 5.12: A schematic representation of the fibre orientations in layer 1 and the corresponding deviation distances

The deviation scheme provides an overview of the amount of neighbouring plies per deviation distance. The amount of neighbouring plies per deviation distance is divided in two groups; the neighbours that require a positive deviation and the neighbours that require a negative deviation. The genotype combines these groups and provides the Neighbour Count (NC) per deviation distance for each layer. The NC is defined as the amount of neighbouring plies that require a positive deviation minus the amount of neighbouring plies that require a negative deviation. The NC provides the dominant sign and an absolute value. The dominant sign indicates the direction of the modification and the absolute value is used to prioritize the modification options. A higher value indicates a better option which is further discussed in section 5.2.1.2.

The genotype, which is constructed from the deviation scheme in table 5.3, is presented in table 5.4. The transformation for a deviation distance of I step is discussed to demonstrate the

Cell X	Neighbour Count per deviation distance						
	Equal	I step	II step	III step	IV step	V step	VI step
Layer 7				+1	+2		
Layer 6	3			-1			
Layer 5			-1		-2		-1
Layer 4	1	0	+1				
Layer 3	2	+2					
Layer 2	1	-1		-2			
Layer 1	1					-3	

Table 5.4: The genotype for the assembly of preliminary patches

construction of the genotype. Layers 2, 3 and 4 contain neighbouring plies according to table 5.3. Layer 2 contains one neighbouring ply. The ply requires a deviation in negative direction. The NC becomes $0-1=-1$ as indicated in table 5.4. Layer 3 contains two neighbouring plies. Both neighbouring plies require a deviation in positive direction. The NC becomes $2-0=2$ as indicated in table 5.4. Layer 4 contains two neighbouring plies that require a deviation in opposite direction. The NC becomes $1-1=0$. The zero indicates that no modification direction is suggested since no distinction can be made between both directions. This concludes the transformation from the deviation scheme to the genotype, for a deviation distance of I step. The application of the genotype is further discussed in section 5.2.1.2.

5.2.1.2 Application of the genotype

The genotype is used to prioritize all modification options in the analysed cell. A feasible modification option with the highest priority is selected and implemented in the stacking sequence of the analysed cell. Each NC in the genotype indicates a modification option. The prioritization of the modification options and the feasibility of those options are described in this section. The order of priority of the modification options is defined by the following list.

1. **Options with the smallest required deviation.** Modifications that create or extend a preliminary patch with the smallest required deviation distance are given priority over modifications that provide a high absolute NC value. All options initiate a formation or extension of a preliminary patch. However, a large deviation distance results in an increased probability of interference from the evolution of the neighbourhood, which is further discussed in this paragraph. The genotype provides an intermediate representation of the stacking sequences during evolution. The modification of a cell during one evolution cycle consists of the alteration of one ply. The ply can change one step in orientation. A modification that requires multiple steps is therefore spread out over multiple evolution cycles. During each evolution cycle, the neighbourhood changes which may interfere with the intended modification. This may result in excessive modifications. Therefore, the options with the smallest required deviation are given priority. This means all modification options that require a deviation of I step have priority over the options that require II steps and so on.
2. **Options with the highest NC.** Prioritization of the options that require a similar

deviation distance is based on the NC. The sign of the NC indicates the direction of the modification and the absolute value is used to prioritize the modification options. The NC is defined as the amount of neighbouring plies in the positive direction minus the amount of neighbouring plies in the negative direction. Therefore, the option with the highest absolute NC value is given the highest priority to be modified. An option is considered infeasible in case the current amount of neighbouring plies with the same ply orientation is equal or higher compared to the NC. These modifications would not progress patch formation.

3. **The inner most layer.** The inner most layer is selected in case multiple options have a similar NC. Modifications in the inner most layers have the smallest effect with respect to the bending stiffness as discussed in chapter 2. The modifications in the outer layers are delayed and evolution may eliminate the required modifications in the outer layers.

The example described in section 5.2.1.1 is used to demonstrate the application of the genotype. In table 5.5, an overview is given of the stacking sequences provided in table 5.2 and the corresponding genotype. The modification options are analysed according to their priority. Not all options are feasible, since the current amount of neighbouring plies with a similar fibre orientation can eliminate an option. A feasible option with the highest priority is selected to be modified. This modification is subsequently applied to the stacking sequence of the analysed cell.

Layer	Cell X	Neighbour			
		1	2	3	4
7	3	6	7	0	7
6	6	6	6	3	6
5	7	3	3	5	1
4	2	2	1	3	4
3	5	5	5	6	6
2	4	3	1	4	1
1	7	2	7	2	2

Cell X	Neighbour Count per deviation distance						
	Equal	I step	II step	III step	IV step	V step	VI step
Layer 7				+1	+2		
Layer 6	3			-1			
Layer 5			-1		-2		-1
Layer 4	1	0	+1				
Layer 3	2	+2					
Layer 2	1	-1		-2			
Layer 1	1					-3	

Table 5.5: An overview of the stacking sequences presented in table 5.2 and the corresponding genotype.

First, the options with the smallest required deviation distance are analysed. The smallest required deviation distance is I step. The options for a deviation distance of I step consist of layers 2, 3 and 4 according to table 5.5. Layer 3 has the highest absolute NC value. However, layer 3 is not a feasible option since the NC equals the current amount of neighbouring layers with equal ply orientation. Subsequently layer 2 is analysed. Layer 2 is also an infeasible option since the absolute NC value equals the current amount of neighbouring layers with the same ply orientation. Layer 4 is not a feasible option since the NC equals zero. A deviation of I step does not provide a feasible option. Therefore, the options corresponding to a deviation of II steps are analysed. The options for a deviation of II steps consist of layer 4 and layer 5. Both layers have an equal absolute NC value. Therefore, the innermost ply has priority. Layer 4 is the innermost ply, however, layer 4 is not a feasible option since its value equals the current amount of neighbouring layers with the same ply orientation. Subsequently, layer 5 is analysed. Layer 5 is a feasible option. The dominant sign is negative, therefore, this option specifies a ply modification in the negative direction. This modification is implemented in

the stacking sequence of the cell X. The fibre orientation of the ply in layer 5 corresponds to the discrete value 7 according to the stacking sequence of cell X presented in table 5.5. A modification in the negative direction results in a fibre orientation corresponding to discrete value 6. In conclusion, the modification of cell X in the current evolution cycle changes the fibre orientation of layer 5 into fibre orientation 6.

All cells are analysed during each evolution cycle in order to find feasible modifications. No modification is performed in case there is no feasible option in a cell. The evolution is completed in case no new modifications are performed during a complete evolution cycle. An evolution cycle may still contain recurrent modifications arising from an infinite loop. An infinite loop may start in case a layer contains a neighbouring ply with an equal fibre orientation and multiple similar neighbouring plies with a deviation distance that is equal or larger than III steps. In case the neighbouring plies remain similar during the evolution process, the analysed ply will alternate between its current orientation and one orientation step towards the other neighbouring plies. An infinite loop is not a common scenario, since an infinite loop can only start if the deviation distance is equal or larger than III steps and all plies have evolved into their final orientation. This scenario may occur at the end of the evolution process, therefore, the effect of the infinite loop on the evolution is negligible.

The evolution of the cells is completed in case no new modifications are performed during a complete evolution cycle. However, the stage of constructing preliminary patches is not completed since cells may still contain isolated plies. Cells may contain isolated plies as a result of an infinity loop, or the NC of a ply remains zero after the evolution of all other cells is completed. The isolated plies are merged with the patch that requires the smallest modification. In case of equivalent option in both directions, the ply is merged with the patch that requires a modification in positive direction. Thereby the algorithm remains deterministic. The assembly of patches is completed after all isolated plies are eliminated. The updated stacking sequences continue to the next stage of the algorithm; minimum patch size.

5.2.2 Minimum patch size

All plies become part of a preliminary patch during the first stage of the evolution strategy. During the second stage, a minimum size is applied to the preliminary patches. The amount and the shape of preliminary patches after the first stage depend on the diversity of fibre orientations in the local optima. A lot of diversity results in small preliminary patches and extremities. The minimum patch size is a manufacturing constraint, however, it may also be required to obtain a blended configuration. Small preliminary patches result in a high number of edges. The amount of edges may exceed the number of edges that can be covered by other plies. The remaining butted edges impede a blended configuration. The method to implement the minimum patch size in 2D problems is based on the amount of cells in a patch. Non-compliant preliminary patches are merged with surrounding patches, which is described in section 5.2.2.1. The patch size in 3D problems is dependent on the amount of cells and the distribution of the cells. The minimum patch size can be seen as the minimum cross-sectional length of a patch. Compact patches without extremities are preferred. The 2D strategy is extended to provide a method for 3D problems, which is further described in section 5.2.2.2.

5.2.2.1 Method for 2D problems

The minimum patch size in 2D problems is obtained by merging non-compliant patches to surrounding patches. Increasing the amount of layers included in the analysis of a ply increases the probability of fractionating surrounding layers. This may be counterproductive in the elimination of non-compliant patches. The preferred amount of included layers depends on the diversity of the patches in a laminate. Laminates with a high diversity are more susceptible to patch fractionation. This is further explained in section 5.3, where the interpretation of the final patch design is discussed. The order in which cells evolve during the evolution stages is based on a chessboard, as described in section 5.1.1.2. However, patches are not restricted to the neighbourhood of the analysed cell. Therefore, all cells need to evolve successively to avoid interference. The evolution order according to the chessboard pattern still provides a homogeneous evolution. In this section, the determination of merge options is described in case a single layer is included in the analysis of a ply. Thereafter, the determination of merge options is described in case multiple layers are included in the analysis of a ply. Determination of the patch that is merged is based on the merge option of all non-compliant plies, which is further described at the end of this section.

Merge options in case the amount of included layers in the analysis of a ply is restricted to 1

The number of included layers is restricted to a single layer to eliminate the fractionation of surrounding patches. A 2D plate which consist of 8 cells is illustrated in figure 5.13. The analysed ply is outlined, which is located in layer 4 of cell 5. The layers which are not included in the analysis of this ply are crossed out as indicated figure 5.13. The patch size is defined as the amount of interconnected cells with similar fibre orientation that contain the analysed ply. The patch size of the outlined ply is 3, in case a single layer is included in the analysis of the ply. The patch corresponding to the analysed ply has two merge options, the adjacent patch in cells 1-3 and the adjacent patch in cells 7-8. The maximum amount of merge options, for an analysis within a single layer, is two options. Edges of a patch that coincides with the boundary of the plate do not provide merge options.

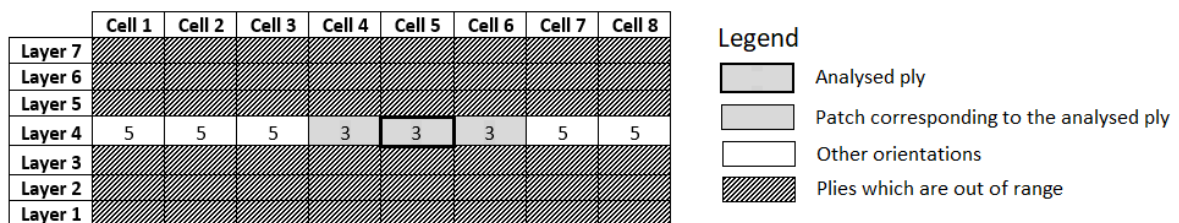


Figure 5.13: Patch size and merge options in case the number of included layers in the analysis is restricted to a single layer

Merge options in case the amount of layers included in the analysis of a ply is restricted to 3.

Multiple layer can be included in the analysis of a ply to implement a minimum patch size. The amount of merge options increases, therefore, the required amount of modifications may decrease. However, the probability of fractionating surrounding patches increases. Fractionation may result in small non-compliant patches, which is counterproductive in the implementation of a minimum patch size. Laminates with a high diversity are more susceptible to patch fractionation, which is further discussed in section 5.3. Plies are not dedicated to specific patches during the evolution stages. Therefore, a ply may be part of multiple patch options. The patch size is defined as the maximum feasible patch size that contains the analysed ply. A patch can only deviate one layer during each cell transition, according to the definition of a patch described in section 5.1.2.2. In figure 5.14, the same 2D plate is presented as in figure 5.13. However, the number of layers included in the analysis of the outlined ply is increased to 3 layers. The plies that are out of range to form a patch with the analysed ply are eliminated from the analysis as indicated figure 5.14. The plies that form a patch with the analysed ply are indicated in figure 5.14. The patch size corresponding to the analysed ply is 4 cells. The patch consists of plies from cell 3-6. Note that in case multiple plies in the same cell can be part of the patch, only one ply is defined as part of the patch. The patch starts at the analysed ply and continues in the same layer as long as possible. A transition to an adjacent layer only occurs if the transition is required to obtain a patch with the maximum feasible patch size. This is apparent in cell 4, layer 4 and 5 both provide plies that can be part of the patch corresponding to analysed ply. Layer 4 is selected since the patch can continue in layer 4, until the transition to layer 3 is required in cell 3 to obtain maximum feasible patch size.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8
Layer 7	6	6						5
Layer 6	1	1	1				7	7
Layer 5	2	2	3	3		4	4	4
Layer 4	5	5	5	3	3	3	5	5
Layer 3	4	4	4	6		6	1	1
Layer 2	7	7	7				2	2
Layer 1	3	3						7

Legend

- Analysed ply
- Patch corresponding to the analysed ply
- Other orientations
- Plies which are out of range

Figure 5.14: Patch size in case 3 layers are included in the analysis

A patch can deviate one layer during each cell transition. Therefore, a patch has a maximum of 3 merge options for each edge. The merge options for the patch corresponding to the analysed ply are indicated in figure 5.15. An edge that coincides with an outer cell of the plate does not provide merge options. An edge provides 2 merge options in case the edge is located in the most inner or most outer layer of the laminate.

Strategy to implement a minimum patch size based on the merge options

All cells evolve according to the chessboard pattern as discussed in section 5.1.1.2. In figure 5.16, a complete overview is given of the plate configuration presented in figure 5.14 and 5.15. The analysis of cell 5 is discussed to present the minimum patch size strategy. The patch size is determined for all plies in the analysed cell to establish the non-compliant patches, as indicated in figure 5.16. Note that the patches in layers 4 and 5 are constructed with the

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8
Layer 7	6	6						5
Layer 6	1	1	1				7	7
Layer 5	2	2	3	3		4	4	4
Layer 4	5	5	5	3	3	3	5	5
Layer 3	4	4	4	6		6	1	1
Layer 2	7	7	7				2	2
Layer 1	3	3						7

Legend

- Analysed ply
- Patch corresponding to the analysed ply
- Merge options corresponding to the analysed patch
- Other orientations
- Plies which are out of range

Figure 5.15: Merge options in case 3 layers are included in the analysis

same plies in cell 3 and 6. This may occur since the interpretation of patches is based on local dependencies. An elaboration of the patch interpretation on a global level can be found in section 5.3.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8
Layer 7	6	6	6	6	6	6	5	5
Layer 6	1	1	1	5	5	7	7	7
Layer 5	2	2	3	3	3	4	4	4
Layer 4	5	5	5	3	3	3	5	5
Layer 3	4	4	4	6	6	6	1	1
Layer 2	7	7	7	7	7	2	2	2
Layer 1	3	3	3	3	1	1	7	7

	Patch size
Layer 7	6
Layer 6	2
Layer 5	4
Layer 4	4
Layer 3	3
Layer 2	5
Layer 1	2

Figure 5.16: Patch sizes corresponding to all plies in the analysed cell

The merge options are defined for the smallest non-compliant patches to minimize the amount of modifications. The modification of larger non-compliant patches is delayed since evolution of other cells may eliminate the need for these modifications. The minimum patch size in this example is defined as 3 plies. Consequently, layer 1 and layer 6 contain the smallest non-compliant patches. The edges of the patch in layer 1 are located in the inner most layer and therefore provide 2 merge options per edge. The patch in layer 6 provides 3 merge options per edge. The merge options for these non-compliant patches are indicated in table 5.6. The merge options are described by the amount orientation steps required for the non-compliant patch to obtain the fibre orientation of the merge option.

	Patch size	Merge option 1	Merge option 2	Merge option 3	Merge option 4	Merge option 5	Merge option 6
Layer 7	6	-	-	-	-	-	-
Layer 6	2	1 step	4 steps	2 steps	1 step	2 steps	1 step
Layer 5	4	-	-	-	-	-	-
Layer 4	4	-	-	-	-	-	-
Layer 3	3	-	-	-	-	-	-
Layer 2	5	-	-	-	-	-	-
Layer 1	2	6 steps	2 steps	No option	1 step	6 steps	No option

Table 5.6: The merge options for the non-compliant patches

The non-compliant patch corresponding to the merge option with the smallest deviation is selected to be merged during the current evolution cycle. In case multiple patches provide

equivalent merge options, the inner most non-compliant patch is selected. Modifications of outer patches have more impact with respect to the bending stiffness as described in chapter 2. Delay may eliminate the need for these modifications due to evolution of surrounding cells. In case multiple merge option of the inner most non-compliant patch provide the smallest deviation distance, the first options that is analysed is selected. All plies of the non-compliant patch are modified and thereby merged with the selected patch. According to table 5.6, both non-compliant patches have merge option that require 1 step in orientation. Therefore, the most inner non-compliant patch is selected, which is the patch in layer 1. Merge option 4 is the option that requires 1 step in orientation. The patch of merge option 4 has a fibre orientation 2. Both plies of non-compliant patch in layer 1 are modified to fibre orientation 2 and thereby merged with the selected patch as indicated in figure 5.17. Figure 5.17 also indicates this the merge resulted in the fractionation of a patch. This is further discussed in section 5.3 which elaborates on the interpretation of the final patches design.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8
Layer 7	6	6	6	6	6	6	5	5
Layer 6	1	1	1	5	5	7	7	7
Layer 5	2	2	3	3	3	4	4	4
Layer 4	5	5	5	3	3	3	5	5
Layer 3	4	4	4	6	6	6	1	1
Layer 2	7	7	7	7	7	2	2	2
Layer 1	3	3	3	3	1	1	7	7

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8
Layer 7	6	6	6	6	6	6	5	5
Layer 6	1	1	1	5	5	7	7	7
Layer 5	2	2	3	3	3	4	4	4
Layer 4	5	5	5	3	3	3	5	5
Layer 3	4	4	4	6	6	6	1	1
Layer 2	7	7	7	7	7	2	2	2
Layer 1	3	3	3	3	2	2	7	7

Figure 5.17: Implementation of the selected merge

5.2.2.2 Method for 3D problems

The patch size in 3D problems is defined by multiple parameters; the amount of cells and the distribution of the cells. Compact patches without extremities are preferred due to the minimization of manufacturing effort and cost. The minimum patch size can be seen as the minimum cross-sectional length of a patch. The amount of included layers is restricted to a single layer. A larger range of included layers may fractionate the patches in 2 dimensions, thereby, increasing the amount of extremities. This is counterproductive in implementation of a minimum cross-sectional length of the patches. The 2D strategy is extended to provide a method suitable for 3D problems. The 2D strategy where the analysis of a ply is restricted to a single layer can be found in section 5.2.2.1. The 2D strategy is applied to all unique cross sections of a 3D plate. A representation of all unique 2D cross sections in a 3D plate consisting of 6 cells is provided in figure 5.18.

The evolution order according to the chessboard pattern does not provide a feasible evolution strategy. The evolution of a cell merges part of a patch in one of the two dimensions. This results in fractionated patches in the other dimensions. The evolution order is adjusted during this stage to minimize the fractionation of patches. Probability of fractionation decreases if the evolution of neighbouring cells is successive, since neighbouring cells are similar and have a higher probability to make similar modifications. The minimum patch size is first implemented in all unique cross section in the X-direction after it is implemented in all unique cross section in the Y-direction. The cross-sections in the X-direction are analysed in an ascending order. This means a minimum patch size is first implemented in cross section X1, where all non-compliant patches are merged according to the 2D strategy discussed in section 5.2.2.1. After the minimum patch size is implemented in cross section X1, cross section X2 is analysed and

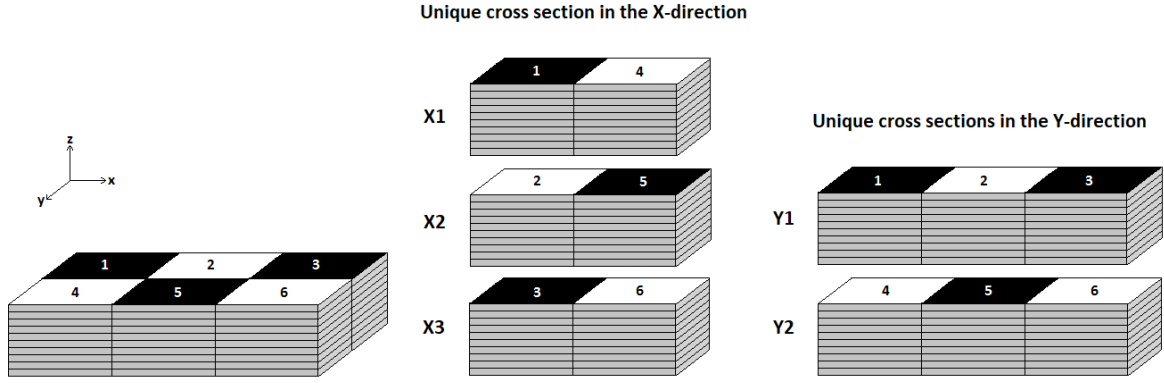


Figure 5.18: A representation of all unique cross sections in a 3D plate consisting of 6 cells

so on. When all non-compliant patches in the X-direction are eliminated, the minimum patch size is applied to all cross-section in Y-direction. All cross sections in the Y-direction are analysed in ascending order. The minimum patch size implementation in the Y-direction can influence the patch size in the X-direction. Therefore, this evolution process is repeated until all non-compliant patches are eliminated or a maximum amount of repetitions is reached.

5.2.3 Elimination of butted edges

The plate configuration consists solely of preliminary patches in this stage of the algorithm. A cell is assumed locally blended if the transition to each neighbour consists of blended transition scenarios. Therefore, butted edges are identified and eliminated. The transition scenarios that can be distinguished are illustrated in figure 5.19. The analysed ply is outlined. Three layers are required in the analysis to differentiate between the type of edges as discussed in section 5.1.2.2. A ply is assumed to be an edge if the ply is not part of a continuous patch. An edge complies to a blended transition scenario either if the edge is covered by another patch or the edge is free. All remaining scenarios are assumed to be butted edges. Elimination of butted edges for 2D problems is divided in three categories based on the layout of surrounding patches. A different modification strategy is applied for each category as described in section 5.2.3.1. The amount of surrounding patch scenarios for 3D problems is neither restricted by the amount of neighbours nor the layout of the neighbourhood. As a result, the 2D method does not provide a viable strategy for 3D problems. The method applied in 3D problems is described in section 5.2.3.2. This method is also applicable for 2D problems. However the required computational effort is increased compared to the 2D method.

5.2.3.1 2D problem

The arrangement of neighbours is similar for each cell in a 2D problem. This limits the amount of scenarios regarding patches which surround a butted edge. The layout of surrounding patches are classified in three categories as depicted in figure 5.20. These categories describe the location of the butted edge with respect to one of the surrounding patches. Category 1 contains scenarios in which the edges of the two patches overlap. Category 2 contains scenarios in which the butted edge is surrounded by a continuous patch. Category 3 consists

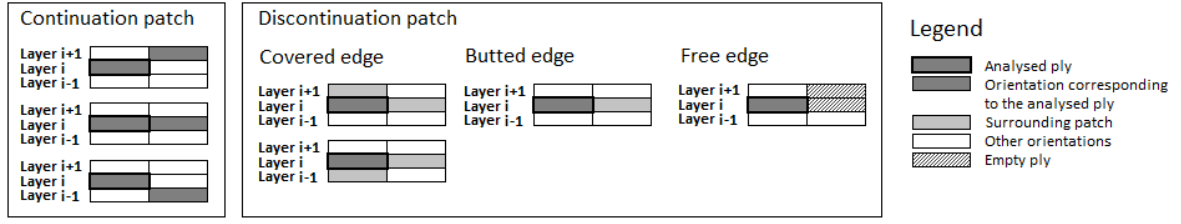


Figure 5.19: Transition scenarios that can be distinguished in case 3 layers are included in the analysis of a ply

of scenarios where two butted edges are situated at the same location in adjacent layers. Detailed scenarios of all categories that differentiate between the neighbours as well as the surrounding patches can be found in Appendix B. Each category requires a different strategy to eliminate the butted edge. The strategy of eliminating butted edges for each category is described below.

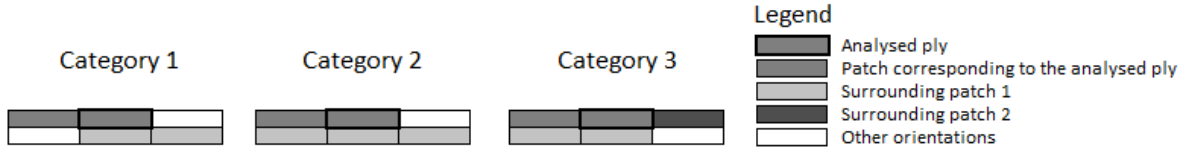


Figure 5.20: Three categories defining the layout of a patch that surrounds a butted edge

Category 1 contains scenarios in which the edges of the two patches overlap. The butted edge can be eliminated by swapping the outer plies as shown in figure 5.21. The butted edge of the analysed cell is eliminated. The status of the other edge is not taken into account in this analysis. However, this modification strategy impedes butted edges for both patches after modification.

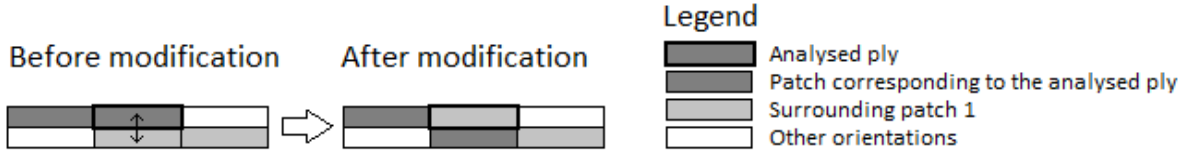


Figure 5.21: Modification of category 1: scenarios in which the edges of the two patches overlap.

Category 2 contains scenarios in which the butted edge is surrounded by a continuous patch. There is no feasible modification that eliminates the butted edge within a single iteration of the analysed cell. All modifications fractionate the patches or relocate the butted edge. Therefore, the proposed modification strategy swaps the patches during multiple evolution cycles. The modification strategy for this category consists of two stages; the initiation and the continuation of a patch swap. The initiation of the patch swap is indicated in figure 5.22. The butted edge is swapped with a ply from the continuous patch. This creates the scenario of transversely linked patches.

The transversely linked patches form a scenario that can be identified during the analyses of other cells. The modification strategy for the continuation of a patch swap is indicated in figure 5.23. This scenario is given priority over other modification scenarios to ensure

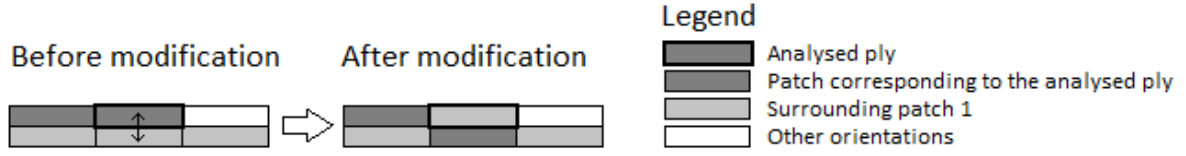


Figure 5.22: Modification of category 2: scenarios in which the butted edge is surrounded by a continuous patch

the patch swap process continues after initiation. The chessboard pattern evolution cycle ensures the succeeding modification occurs in the neighbouring cell. Therefore, the patch swap continues until the patches are no longer transversely linked.

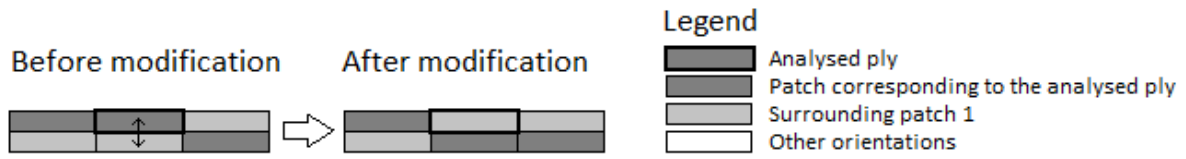


Figure 5.23: Modification of a transversely linked patch scenario.

A representation of the scenarios after the completion of the patch swap is given in figure 5.24. In case both patches end at the same cell, the complete patches are swapped. No further modifications are required after one of the patches is discontinued as illustrated in figure 5.24. The edge of the discontinued patch is covered by the continuous patch.



Figure 5.24: Representation of the possible scenarios of two patches after the completion of a patch swap.

The transversely linked patch scenario is assumed to be the result of a patch swap initiation. However, this scenario may occur without being part of a patch swap initiation. In that case, the unrequired patch swap is deemed acceptable, since the effect of the patch swap is negligible. In most common scenarios the patch swap strategy neither fractionates the patches nor relocates butted edges. The scenarios in which this does occur are not accounted for since the probability and the effect on the complete configuration are deemed negligible.

Category 3 consists of scenarios where two butted edges are situated at the same location in adjacent layers. The fibre orientation of the analysed ply is modified to match the patch that continues in the same layer as shown in figure 5.25. This creates a scenario that corresponds to category 1. The overlapping edges are swapped to eliminate the butted edges.

The sequence in which the categories are modified is based on the required modifications. First all category 1 scenarios are modified, after which scenarios from category 2 are modified. A category 2 modification may create a transversely linked patch scenario which has priority over scenarios from category 1. The remaining scenarios to be modified are the scenarios from category 3. The scenarios from category 3 may require a smaller modification than the scenarios in category 2. However, modifying multiple butted edges in close proximity of each

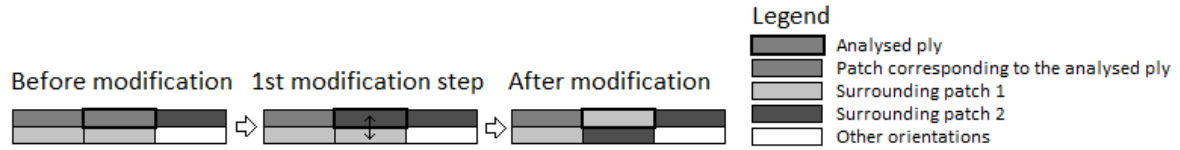


Figure 5.25: Modification of category 3: scenarios where two butted edges are situated at the same location in adjacent layers

other may fractionate and scatter these patches during multiple evolution cycles. This may throw the algorithm off course. The category 3 scenarios are therefore modified if all other modifications are completed. A cell may contain multiple scenarios during an evolution cycle. In case a cell contains multiple scenarios from the same category, the largest patches are given priority. The outermost plies have the largest effect on the buckling constraints as discussed in chapter 2. Therefore, the outermost patch is selected in case of multiple equivalent options.

5.2.3.2 Method for 3D problems

The amount of neighbours and the layout of the neighbourhood may differ for each cell in a 3D problem with an irregular grid. Consequently, the amount of surrounding patch scenarios that may occur in 3D problems are neither restricted by the amount of neighbours nor the layout of the neighbourhood. This increases the amount of scenarios significantly compared to 2D problems. The modifications required for the extended amount of scenarios cannot be provided by a straightforward set of rules. Backtracking would be required for this method to obtain a blended result. Backtracking may results in a limitation of the maximum amount of cells and is therefore undesirable in a method for blending a large amount of cells. An alternative method is created which is suitable for 3D problems with a large amount of cells. This method is also applicable for 2D problems. However, the required computational effort is increased compared to the method described in section 5.2.3.1.

Stacking sequence of Cell X		Amount of butted edges when modifying the orientation of a ply						
		1	2	3	4	5	6	7
Layer 7	6	7	7	6	7	7	5*	7
Layer 6	3	9	9	5*	9	9	8	9
Layer 5	6	5	5	5	5	5	5*	5
Layer 4	6	6	6	5	6	6	5*	6
Layer 3	6	5	5	5	5	5	5*	5
Layer 2	6	7	7	4	7	5	5*	7
Layer 1	6	5	5	2	5	5	5*	5

*Amount of butted edges for the current stacking sequence.

Table 5.7: Total amount of butted edges present in the analysed cell in case of modifying the orientation of a ply

The method for 3D problems is based on the minimization of the total amount of butted edges in a cell during each iteration. The maximum amount of butted edges per ply equals the amount of neighbours. A ply modification can create or dissolve butted edges in three

layers for each neighbour. The amount of butted edges per ply are added to form the total amount of butted edges in the analysed cell. The total effect of a modification can thereby be analysed. In case a cell contains butted edges, all modification options are identified. An overview of the modification options in a cell is provided in table 5.7. All plies in cell X are analysed. A ply can consist of one of the 7 fibre orientations as discussed in chapter 2. The fibre orientation of a ply can be transformed to the remaining 6 orientations. For each option the total amount of butted edges in the analysed cell is indicated. Five butted edges are present in the current stacking sequence of cell X as indicated in table 5.7. The modification that minimizes the total amount of butted edges is applied. For the example described in table 5.7, the minimum amount of butted edges that can be obtained by modifying a ply in this stacking sequence is two. The modification of cell X consist of changing layer 1 into a fibre orientation 3, since this modification decreases the amount of butted edges to two. Note that the amount of butted edges of the original stacking sequence is also considered. If the original stacking sequence results in the minimum amount of butted edges, no modification is applied. In case multiple modifications options result in a minimum amount of butted edges, the option with the smallest deviation in fibre orientation is selected. Modifications of the inner most plies minimize the effect on the bending stiffness as discussed in chapter 2. Thus, the inner most ply is selected in case multiple plies provide equivalent options. All modification options are analysed to maximize the decrease of butted edges. As a result, the amount of alterations is minimized and thereby the probability of interference between the alterations is minimized. Interference between alterations can throw the algorithm off course and impede a blended result. The required computational effort to analyse a cell is independent on the total amount of cells. This method is therefore suitable for blending a large number of sections as opposed to methods that require backtracking. However, the required computational effort in this stage is high compared to the previous stages and a fully blended results can not be guaranteed. Future research should investigate the ability of other methods to decrease the amount of required modifications and the accompanying computational effort. This is further discussed in chapter 7.

5.3 Global patch interpretation

The evolution process evolves the stacking sequences based on local dependencies. The locally evolved interconnections provide options to obtain a globally blended laminate. However, the local interconnections do not define the patches in the global laminate design. The final patch design is created on a global level. The interpretation of the patch designs at a global level and the selection criteria of the final patch design are described in this section. Multiple patch designs can be extracted from the stacking sequences provided by the evolution process. In figure 5.26a an example is given of the final stacking sequences of a 2D plate after the evolution is completed, where each colour indicates a different fibre orientation. The minimum patch size was set at 2 plies. The stacking sequences are fully blended at local level. This means that at each transition either a patch can continue in one of the three surrounding layers or the ply is an edge which can be covered by a surrounding patch. There are multiple options to interpret the patch design at a global level. Three interpretations of a global patch design are provided in figure 5.26. These global patch interpretations are manually chosen by the author. An interpretation which contains a butted edge is illustrated in figure 5.26b. The butted edge is located in layer 4, in the transition between section 5 and 6. This global

patch interpretation does not provide a fully blended result at global level and is therefore not selected as the final patch design. A global patch interpretation which is fully blended at global level is illustrated in figure 5.26c. However, in this interpretation the minimum patch size of 2 plies is not maintained. An isolated ply is present in section 5 of layer 2. Therefore, this patch interpretation does not comply to the manufacturing constraints. Figure 5.26d provides a fully blended global patch interpretation, in which all plies comply to the minimum patch size.

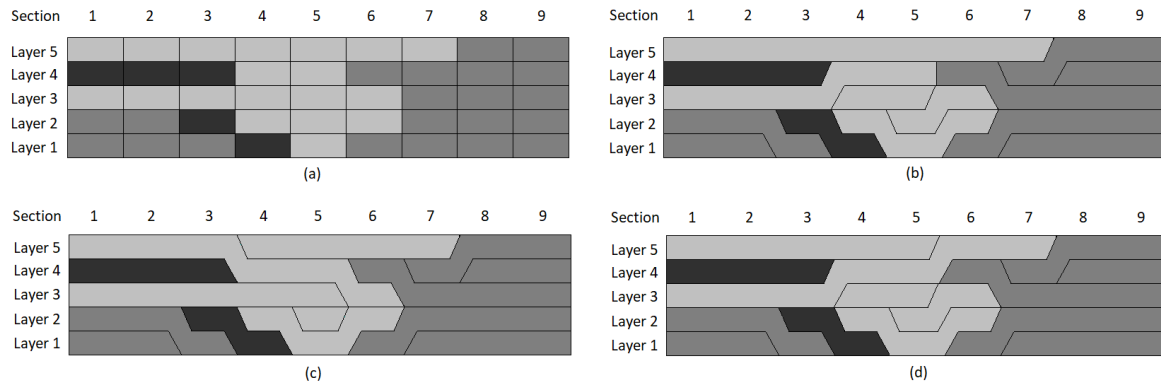


Figure 5.26: Multiple interpretations of the global patch definition for a set of locally blended stacking sequences

The selection of the global patch design is dependent on the compliance to the blending guidelines, ply drop guidelines, the structural constraints and the manufacturing constraints. Note that for 3D problems, these selection criteria apply to all unique cross-sections. Future research is required to obtain a standardized method to interpret the final patch design at global level as further discussed in chapter 7. The selection criteria of the global patch design are briefly discussed below.

Minimum patch size: A minimum patch size, imposed by the manufacturing constraints, is to be maintained in the global patch interpretation. Figure 5.26c provides a global patch interpretation in which the minimum patch size is not maintained while figure 5.26d provides a global patch interpretation in which the minimum patch size is maintained. The minimum patch size in this example can be maintained by a different interpretation of the global patches. However, in case of a high diversity of patches, fractionation of patches can not be resolved by a different interpretation of the patches. Therefore, fractionation is minimized as discussed in section 5.2.2.

Butted edges: The edges of a patch are assumed to be covered if a surrounding patch can cover the edge during evolution. However, this does not guarantee an edge is covered at a global level. Two patches can continue into one ply during local analysis due to the restricted amount of data used in the analyses of a ply. A ply can only be part of one patch in the global patch interpretation. A patch interpretation is required which eliminates or minimizes the amount of butted edges at global level.

Amount of plies dropped at the same location: The evolution algorithm is restricted by the amount of layer included in the analysis. The algorithm makes modifications to obtain a single ply drop, since multiple dropped plies can not be identified in the included amount of layers. However, multiple dropped plies at the same location are allowed according to the

blending guidelines. This provides more freedom to design the patches at a global level. Note that the maximum amount of plies dropped at the same location is restricted by the ply drop guidelines as discussed in chapter 2. Therefore, the maximum amount of plies dropped at the same location should be included in the selection criteria of the global patch design.

Decoding of the chromosomes: The final stacking sequences of each section can be determined when the global patch design is defined. The stacking of the local optima were assumed symmetric and semi balanced as discussed in chapter 2. The decoding of the final stacking sequences can not be performed separately, since the sign of the fibre orientation is specified by the patch. The stacking sequence in a section is not guaranteed to remain semi blended. This should be taken into account during the definition of the patches at a global level. Plies can be added to ensure the stacking sequences are blended in the global patch design.

5.4 Verification of the CA-based algorithm

The verification of the GA-CA-based blending algorithm is described in this section. The GA and the CA are verified separately before the combination of the algorithms is verified. The verification of the GA can be found in chapter 4. The input for the CA in the GA-CA-based algorithm is a set of stacking sequences which is generated by the GA. However, the input for the evolution process of the CA is adjusted for purpose of verification. The verification of the CA evolution process for 2D problems is described in section 5.4.1. The verification of the evolution process of the CA for 3D problems can be found in section 5.4.2. In section 5.4.3, the verification of the GA-CA-based algorithm is described.

5.4.1 Verification of the evolution stages for 2D problems

The evolution process of the CA consists of three stages; foundation of patches, minimum patch size and the elimination of butted edges. The verification of the three stages is performed separately. The verification of the foundation of patches is described in section 5.4.1.1. The verification of the minimum patch size can be found in section 5.4.1.2. The verification of the elimination of butted edges is described in section 5.4.1.3.

5.4.1.1 Foundation of patches

The first stage of the algorithm conducts the assembly of preliminary patches. The local optima, intermediate stacking sequences and the final stacking sequences of a configuration consisting of 2 cells is illustrated in figure 5.27. Both plies in each layer deviate a similar but opposite amount of steps from fibre orientation 4. The assembly of patches is performed within a single layer. As a result, all layers evolve to the fibre orientation 4. The plies in layer 5 remain unaltered since the plies already form a patch. Layer 3 and 4 provide equivalent options regarding the step size. The intermediate stacking sequences after the first evolution cycle are depicted in figure 5.27b. Layer 3 has evolved into a patch. Layer 3 is the most inner layer of both options and is therefore given priority. The intermediate stacking sequences after the third evolution cycle are depicted in figure 5.27c. Layer 1 evolves towards a patch. The

modification of the plies consist of a single orientation step during each evolution cycle. The performed modifications and the sequence of the modifications comply with the description given in section 5.2.1.

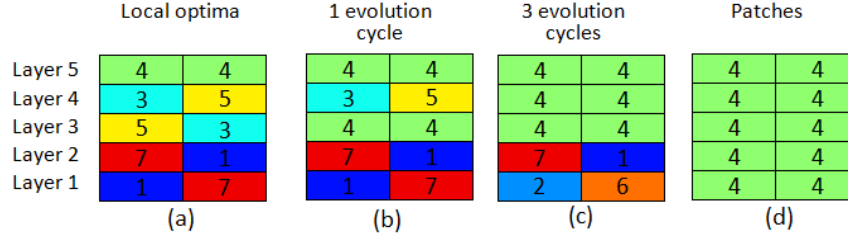


Figure 5.27: (a) local optima, (b) stacking sequences after 1 evolution cycle, (c) stacking sequences after 3 evolution cycles (d) preliminary patches

The patches which follow from the evolution of 5 randomly generated stacking sequences are presented in figure 5.28a. The modification options and the order of the modifications are identical to the example depicted in figure 5.27. However, the amount of modification options per cell is increased since a cell can have 2 neighbours. The constructed patches depicted in figure 5.28b show the modifications are performed in the correct order. The plate configuration contains patches consisting of 2, 3 and 5 plies. All isolated plies are eliminated. Therefore, patches consisting of 4 plies are not feasible in a configuration of 5 cells.

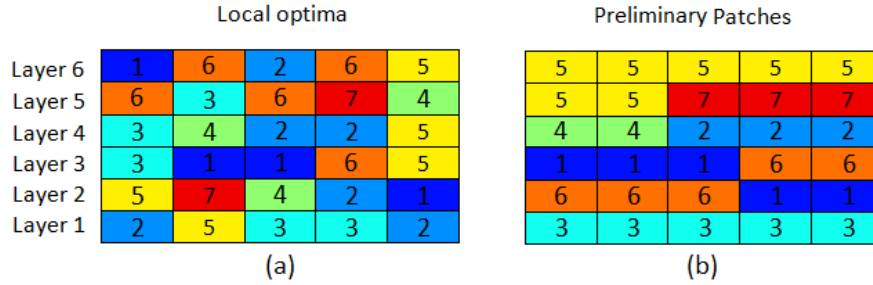


Figure 5.28: (a) local optima, (b) preliminary patches

5.4.1.2 Minimum patch size

A configuration of 8 cells, which is generated in the first stage of the algorithm, is presented in figure 5.29a. All plies are part of a patch. The minimum patch size in this example is defined as 3 plies. All patches consisting of a maximum feasible patch size of 2 plies do not comply with the minimum patch size. These patches are merged with surrounding patches in this stage of the algorithm. Non-compliant patches are found in layers 1,3 and 6. The algorithm correctly identifies the patch in layer 4 as compliant. This patch can continue in the adjacent layer. Therefore, the maximum feasible patch size of this patch is 4 plies. The final configuration of this stage in case the merge option are restricted to a single layer is given in figure 5.29b. The final configuration of this stage in case the merge options are provided by three surrounding layers is depicted in figure 5.29c. Both configurations show the non-compliant patches have merged with a surrounding patch that presents the smallest deviation in fibre orientation.

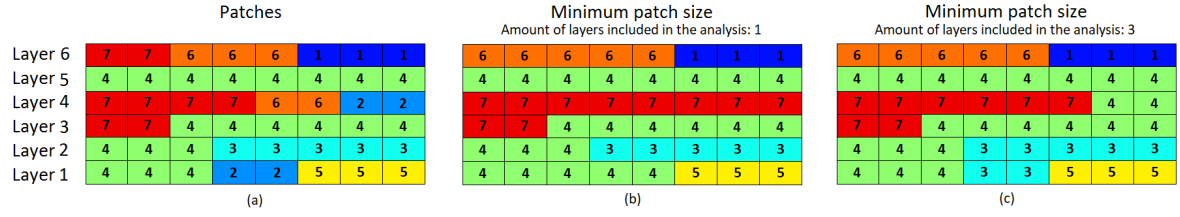


Figure 5.29: Implementation of the minimum patch size for a varying amount of layers included in the analysis of a ply

5.4.1.3 Elimination of butted edges

The final stage of the algorithm eliminates the butted edges. A configuration containing a butted edge scenario from each category is presented in figure 5.30. In addition to the butted edge scenarios, a transversely linked patch scenario is included. This configuration verifies the applied modification for each category and the order of the modifications.

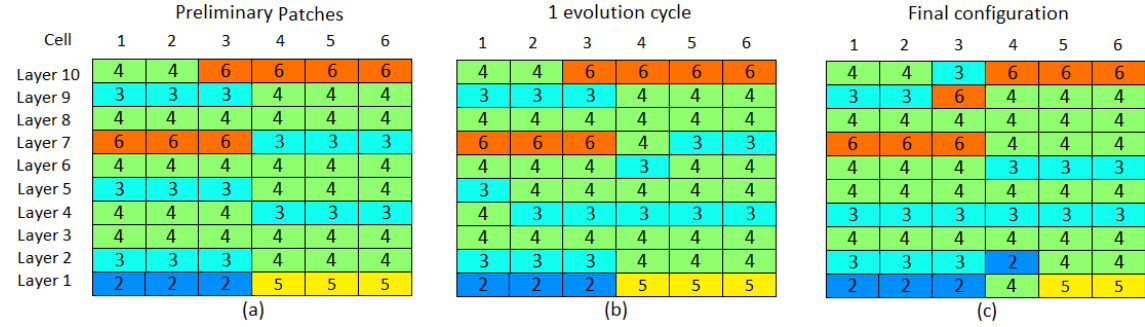


Figure 5.30: Modifications during the evolution in stage 3: the elimination of butted edges

The first modification is performed in cell 3, since cell 1 does not contain butted edges. Cell 3 contains multiple scenarios. The swap between layer 4 and 5 in cell 3 confirms the transversely linked patch scenario is given priority over all other scenarios. The patch swap process continues in this evolution cycle during the evolution of cell 2, as depicted in figure 5.30b. The evolution of cell 4 initiates a patch swap between layer 6 and 7. In the second evolution cycle the patch swap between layer 4 and 5 is concluded in cell 1. Cell 3 contains scenarios from 2 categories. According to figure 5.30c, category 1 is given priority over category 3, since the layer 9 and layer 10 of cell 3 are swapped. The category 3 scenario is present in the transition between cell 3 and 4. The category 3 scenario is eliminated in cell 4, since no other scenarios are present in cell 4. The patch swap between layers 6 and 7 continues during the evolution of cell 5 and concludes after the evolution of cell 6. All modifications are completed within 2 evolution cycles. All butted edges are eliminated in the final configuration. The applied modifications and the order of the modifications is consistent with the description presented in section 5.2.3.

5.4.2 Verification of the evolution stages for 3D problems

The evolution process of the CA consists of three stages; foundation of patches, minimum patch size and the elimination of butted edges. The verification of the three stages is performed

separately. The verification of the foundation of patches is described in section 5.4.2.1. The verification of the minimum patch size can be found in section 5.4.2.2. The verification of the elimination of butted edges is described in section 5.4.2.3.

5.4.2.1 Foundation of patches

The assembly of patches is the similar for 2D and 3D problems. The verification of this stage for 2D problems is described in section 5.4.1.1. In a 3D problem the order of preferred modifications is similar. The amount of modification options per cell is increased due to a higher number of neighbours. An exploded view of a 3D problem consisting of 9 sections and 5 layers is given in figure 5.31. Multiple evolution cycles of the patch assembly process are depicted. The stacking sequences in figure 5.31a represents the local optima. These local optima are randomly generated for the purpose of verification. The plate configurations after the first and second evolution cycle confirm the order of preferred modifications is not influenced by the increased amount of neighbours.

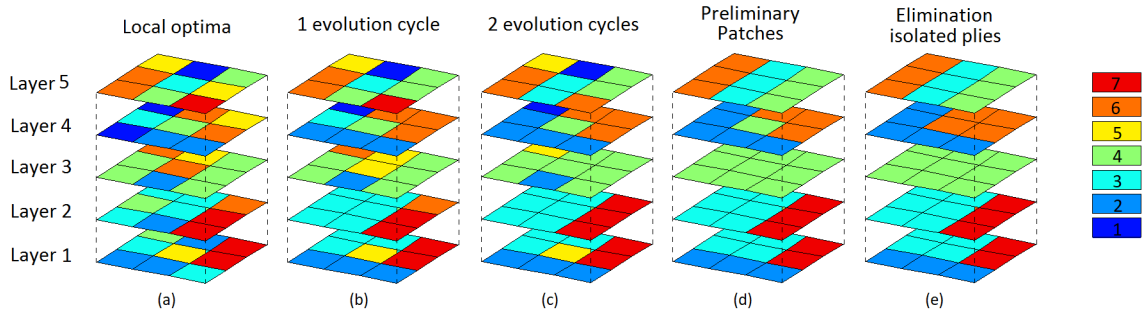


Figure 5.31: Exploded view of the layers in a 3D problem during the assembly of preliminary patches

The configuration after the 5th evolution cycle is depicted in figure 5.31d. The evolution of plies is concluded and all patches are formed. An isolated ply remains in layer 4. The ply has 2 equivalent options and remains isolated after the evolution of all plies is concluded. The final step in this stage is the elimination of remaining isolated plies as described. The isolated ply has merged with the option that requires a modification in the positive direction as indicated in figure 5.31e.

5.4.2.2 Minimum patch size

The 2D strategy to implement a minimum patch size is extended to provide a 3D strategy. Figure 5.32 shows a configuration before and after the implementation of the minimum patch size. Each layer consist of patches with different shapes and sizes. The amount of included layers is restricted to one. Therefore, modifications in different layers do not influence each other. The patches are first merged in X-direction, after which the patches are merged in Y-direction. The minimum patch size is set at 3. The cells are analysed in ascending order. Consequently, for a plate consisting of 3 by 3 cells, all plies are merged with the ply which is located at $x=3$ and $y=3$ in the same layer. According to the configuration depicted in figure 5.32b, the algorithm performs as described in section 5.2.2.2.

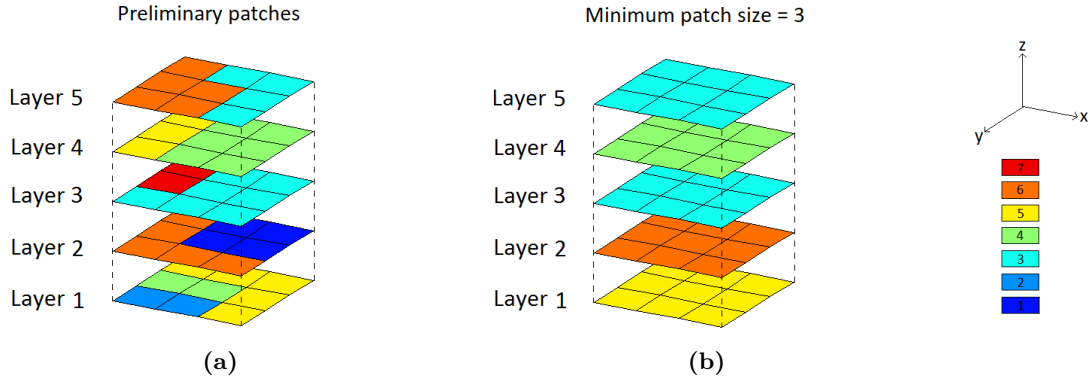


Figure 5.32: A configuration before (a) and after (b) the implementation of the minimum patch size

5.4.2.3 Elimination of butted edges

The algorithm for the elimination of butted edges in 3D problems is different than 2D problems. The strategy is based on minimization of the butted edges in each cell. An example of 3D plate divided in 9 cells is provided in figure 5.33. The amount of layers is 2 for the purpose of verification. The preliminary patches provided by the previous stages are depicted in figure 5.33a. The configuration after the elimination of the butted edges is depicted in figure 5.33b. All butted edges are eliminated at local level since all edges can be covered by a surrounding patch. The plies in section 1,5 and 7 of layer 1 are modified in evolution cycle 1. Note that these sections are modified in accordance with the evolution order based on the chessboard pattern. No butted edges remain after the first evolution cycle. The algorithm performs as described in section 5.2.3.

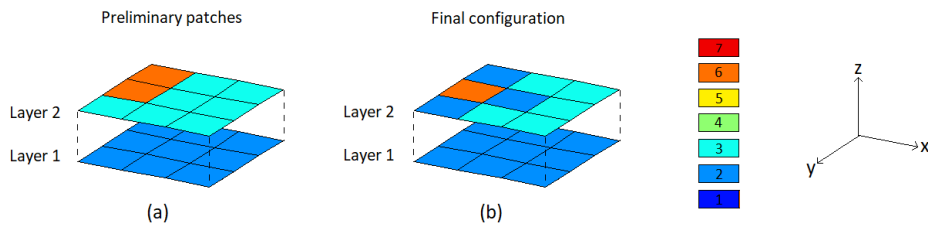


Figure 5.33: A configuration before and after the elimination of the butted edges

5.4.3 Verification of the GA-CA-based algorithm

The proposed GA-CA-based algorithm combines a GA and a CA. A review of the flowchart of the GA-CA-based algorithm is provided in figure 5.34. The verification of the GA can be found in chapter 4. The verification of the CA is described in section 5.4.1 and 5.4.2. This section provides the verification of the combined GA-CA-based algorithm.

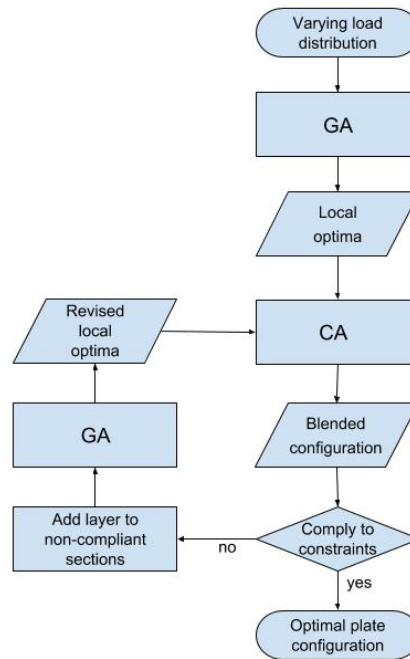


Figure 5.34: Flowchart of the GA-CA-based algorithm

The example presented in figure 5.35 is used to verify the CA-GA-based algorithm. These section originate from a benchmark problem in a horseshoe pattern [26] which is further described in chapter 6. The plate presented in figure 5.35 consists of two sections. The boundaries of each section are assumed simply supported. The applied mechanical constraint is the buckling load as described chapter 2.

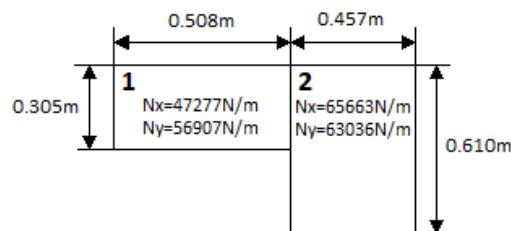


Figure 5.35: A representation of a plate consisting of 2 sections

An overview of the results is presented in figure 5.36. The local optima consist of respectively 10 and 14 layers. The CA evolves these local optima to a blended configuration. The plies in each layer will evolve to a patch since no isolated plies are allowed. The blended configuration is presented in figure 5.36. The stacking sequence of section 1 in the blended configuration does not comply to the buckling constraint. A layer is added to section 1 and the revised local optima is determined by the GA as depicted in figure 5.36. The CA evolves the revised

local optima to a blended configuration. The stacking sequence of both sections comply to the buckling constraints which concludes the iteration process. The GA-CA-based algorithm performs according to the description given in chapter 3.

		1 st iteration				2 nd iteration			
		Local optima		Blended configuration		Revised local optima		Blended configuration	
Layer 14			3		3		3		3
Layer 13			3		3		3		3
Layer 12			3		3		3		3
Layer 11			3		3		3		3
Layer 10		6	3	4	4	7	3	5	5
Layer 9		6	3	5	5	6	3	4	4
Layer 8		6	3	4	4	6	3	5	5
Layer 7		6	3	5	5	6	3	4	4
Layer 6		6	3	4	4	6	3	5	5
Layer 5		6	3	4	4	6	3	4	4
Layer 4		6	3	5	5	6	3	5	5
Layer 3		6	3	4	4	6	3	4	4
Layer 2		6	3	5	5	6	3	5	5
Layer 1		6	3	4	4	6	3	4	4
Nr of layers		10	14	10	14	11	14	11	14
Buckling load ratio		1.0129	1.1000	0.8990	1.0530	1.2824	1.1000	1.2246	1.0203

Figure 5.36: An overview of the results of the GA-CA-based algorithm for the plate depicted in figure 5.35

Results and Discussion

The results of the proposed blending method are discussed in this chapter. The results of proposed blending method are compared to the results of state-of-the-art methods using a comparative example introduced by Soremekun et al. [26]. The comparative example is a plate in the shape of a horseshoe. The results of this benchmark problem are further discussed in section 6.1. The results of the algorithm for 2D problems are discussed in section 6.2, including the effectivity of the algorithm for 2D problems. The horseshoe pattern benchmark case does not provide a feasible example to demonstrate the effectivity for 3D problems, due to the limited amount of sections. A new benchmark case is proposed to test the effectivity of the blending algorithms for plates with a larger amount of sections than currently found in literature. The results and the effectivity of the proposed blending method for 3D problems are discussed in section 6.3. The conclusions can be found in section 6.4.

6.1 Horseshoe pattern benchmark case

The results of the proposed blending method are compared to the results described in literature using a comparative example introduced by Soremekun et al. [26]. This comparative example is a horseshoe pattern divided in 18 sections as depicted in figure 6.1. The boundaries of each section are assumed to be simply supported. The applied loads vary per section. The applied loads and the material properties can be found in appendix E. The applied mechanical constraint is the buckling constraint. The determination of the critical buckling load for simply supported boundary conditions is described in chapter 2. The objective of the benchmark case is to find a blended laminate configuration with the lowest weight that complies to the buckling constraint. The results generated by the proposed blending method are described in section 6.1.1. The results of the proposed blending method are discussed and compared to the state-of-the-art methods in section 6.1.2.

1	2	3	4	5
		6	7	8
9	10			
11	12	13	14	15
		16	17	18

Figure 6.1: Section layout and section numbering of the horseshoe pattern benchmark case

6.1.1 Generated results

The generated results of the horseshoe pattern benchmark case are described in this section. The local optima are generated by the GA as described in Chapter 4. The parameter setting of the GA can be found in appendix E. The number of layers and the buckling load ratio of the local optima are provided in table 6.1. The local optima are depicted in figure 6.2a. Note that the local optima in figure 6.2a are presented by ascending section numbers. These section numbers indicate the location of each section in the horseshoe configuration presented in figure 6.1. The CA evolves these local optima to a blended configuration. The buckling load ratio of each section changes due to the evolution of the stacking sequences. Sections 3, 5, 7, 16 and 17 failed to comply with the buckling constraint, in the blended configuration after the first iteration, as indicated in table 6.1. The revised local optima for these sections are provided by the GA. These revised local optima and the local optima of the other sections are the new input for the CA. The CA evolves these stacking sequences to a blended configuration in the second iteration. Sections 4 and 8 failed to comply with the buckling constraint in the blended configuration of the second iteration. Therefore, a third iteration is performed. All stacking sequences in the blended configuration of the third iteration comply with the buckling constraint as indicated in table 6.1. Therefore, the iteration process is terminated after 3 iterations. The amount of added layers and the performed modification steps per section are included in table 6.1.

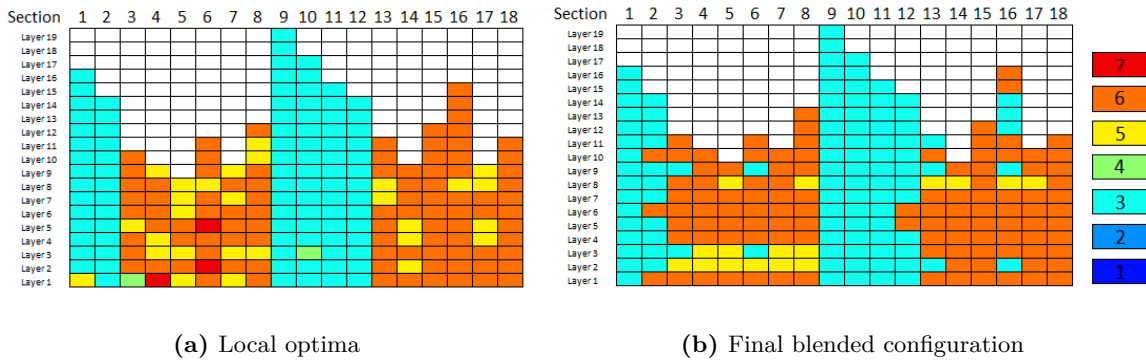


Figure 6.2: Stacking sequences in each section

Section	1 st Iteration			2 nd iteration		3 rd iteration			
	Local Optima		Blended configuration	Blended configuration		Blended configuration			
	Nr. of layers	Buckling load ratio		Nr. of layers	Buckling load ratio	Nr. of layers	Buckling load ratio	Modification steps	Amount of extra layers
1	16	1.0249	1.0249	16	1.0249	16	1.0249	2	0
2	14	1.1000	1.0740	14	1.0256	14	1.0255	9	0
3	10	1.0135	0.8869	11	1.2536	11	1.2534	8	1
4	9	1.0652	1.0507	9	0.9159	10	1.2706	3	1
5	8	1.2186	0.8837	9	1.2230	9	1.4293	2	1
6	11	1.2100	1.0133	11	1.1266	11	1.1264	10	1
7	9	1.0276	0.9605	10	1.1483	10	1.1614	2	1
8	12	1.0525	1.0483	12	0.9834	13	1.1961	4	1
9	19	1.1044	1.1044	19	1.1044	19	1.1044	0	0
10	17	1.0106	1.0107	17	1.0107	17	1.0107	1	0
11	15	1.1659	1.1659	15	1.1659	15	1.1659	0	0
12	14	1.0952	1.0641	14	1.0641	14	1.0641	6	0
13	11	1.2629	1.0489	11	1.0489	11	1.0489	10	0
14	9	1.1384	1.1352	9	1.1352	9	1.1352	4	0
15	12	1.0323	1.0323	12	1.0323	12	1.0323	0	0
16	15	1.1170	0.8943	16	1.1329	16	1.1334	15	1
17	9	1.0366	0.8846	10	1.1819	10	1.1819	5	1
18	11	1.1021	1.1021	11	1.1021	11	1.1021	0	0

Table 6.1: The buckling load ratio during each iteration of the GA-CA-based algorithm

The locally blended stacking sequences of all sections, provided by the CA, are depicted in figure 6.2b. The stacking sequences are presented by ascending section numbers. These stacking sequences provide locally blended transitions in the horseshoe configuration depicted in figure 6.1. A global patch interpretation is manually constructed by the author to evaluate the results. The global patch interpretation is required to determine the final stacking sequences as discussed in section 5.3. The global patch interpretation is illustrated in figures 6.3-6.4. The unique cross-sections in the y-direction are depicted in figure 6.3. The stacking sequences of sections 3, 4 and 5 are similar to the stacking sequences of sections 6, 7 and 8 respectively. The stacking sequences of sections 13, 14 and 15 are similar to the stacking sequences of sections 16, 17 and 18 respectively. Therefore, these cross-sections in the x-direction primarily consist of continuous plies. The cross-sections, which require a global patch interpretation in the x-direction, are depicted in figure 6.4. The stacking sequences corresponding to the global patch interpretation are provided in table 6.2. The global patch interpretation provides a fully blended laminate configuration, in which all stacking sequences are symmetric and semi-balanced.

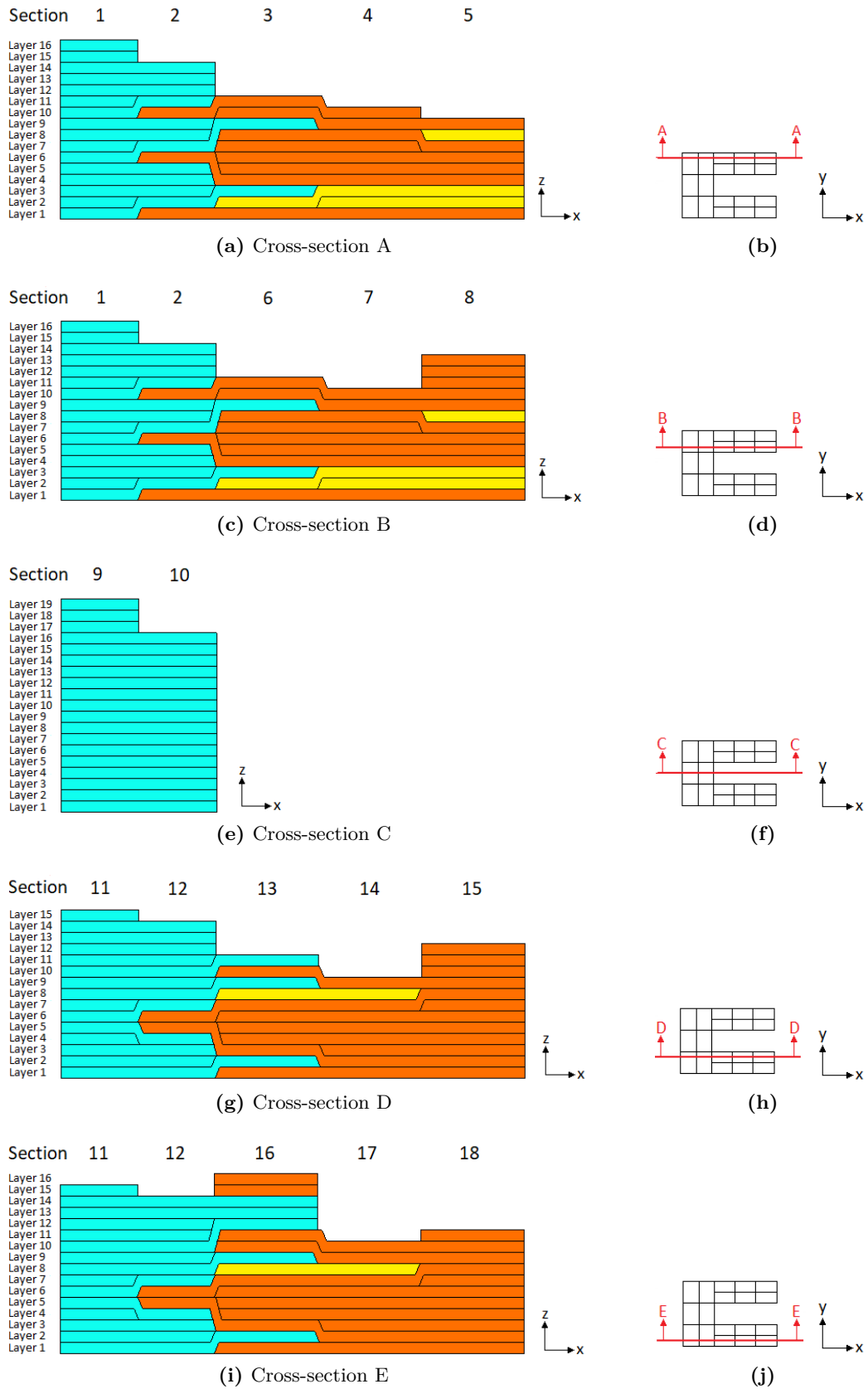


Figure 6.3: Cross-sections in the y-direction

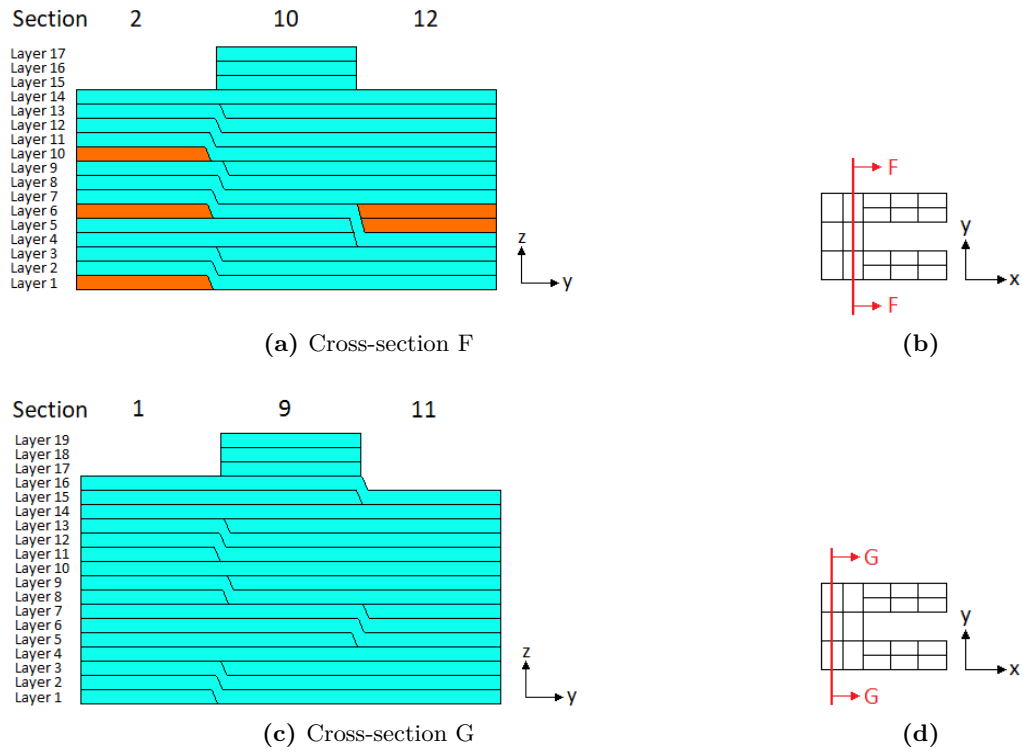


Figure 6.4: Cross-sections in the x-direction

Section	Stacking Sequence	Nr. of Layers	Buckling load ratio
1	$[-30/(\pm 30)_2/-30/(\pm 30)/30/(\pm 30)/-30/(\pm 30)/30/30]_s$	32	1.0249
2	$[-30/(\pm 30)/30/-75/-30/(\pm 30)/-75/-30/(\pm 30)/30/75]_s$	28	1.0255
3	$[-75/75/-30/-75/(\pm 75)_2/30/60/75]_s$	22	1.2534
4	$[-75/(\pm 75)_3/(\pm 60)/75]_s$	20	1.2706
5	$[75/60/-75/-75/(\pm 75)/(\pm 60)/75]_s$	18	1.4293
6	$[-75/75/-30/-75/(\pm 75)_2/30/60/75]_s$	22	1.1264
7	$[-75/(\pm 75)_3/(\pm 60)/75]_s$	20	1.1614
8	$[(\pm 75)_2/75/60/-75/-75/(\pm 75)/(\pm 60)/75]_s$	26	1.1961
9	$[(\pm 30)_3/-30/(\pm 30)/30/(\pm 30)_2/-30/30/(\pm 30)/30]_s$	38	1.1044
10	$[(\pm 30)_2/-30/(\pm 30)/30/(\pm 30)_2/-30/30/(\pm 30)/30]_s$	34	1.0107
11	$[-30/-30/-30/(\pm 30)/30/(\pm 30)/-30/(\pm 30)/30/(\pm 30)/30]_s$	30	1.1659
12	$[-30/-30/(\pm 30)/30/(\pm 30)/30/(\pm 75)/-30/(\pm 30)/30]_s$	28	1.0641
13	$[30/-75/-30/60/(\pm 75)_2/-75/30/75]_s$	22	1.0489
14	$[-75/60/(\pm 75)_3/75]_s$	18	1.1352
15	$[(\pm 75)_3/-75/(\pm 75)_2/75]_s$	24	1.0323
16	$[-75/75/-30/30/30/(\pm 75)/-30/60/(\pm 75)_2/-75/30/75]_s$	32	1.1334
17	$[(\pm 75)/60/(\pm 75)_3/75]_s$	20	1.1819
18	$[-75/(\pm 75)_2/-75/(\pm 75)_2/75]_s$	22	1.1021

Table 6.2: The stacking sequences corresponding to the global patch definition

6.1.2 Results and discussion

The results of the proposed method and the results of the state-of-the-art methods [2, 9, 23, 25, 26] are compared in table 6.3. All stacking sequences presented in table 6.3 are symmetric. The total amount of layers resulting from the proposed method is comparable to the results described in literature. Note that the manufacturing effort of the generated laminate is higher due to the large amount of patches. The large amount of patches result from the relatively small amount of sections and the layout of these sections in the horseshoe configuration. The proposed method is created for plates with a larger amount of section than is currently found in literature. In addition, the local optima in a plate with a varying load distribution change gradually from one section to another. The benchmark case is a set of combined panels with simply supported boundaries. The load distribution in the benchmark case causes three areas in which the local optima are similar. Area 1 consists of sections 1, 2, 9, 10, 11 and 12. Area 2 consists of sections 3-8 and area 3 consists of sections 13-18. Butted edges are concentrated in the transition between the 3 areas. As a consequence, there is no homogeneous distribution of modifications performed by the algorithm. Most modifications are performed in sections 3, 6, 13 and 16.

	Proposed method	Soremekun et al.	Seresta et al.	Isselmuiden et al.		Adams et al.	Irisarri et al.
	Semi balanced	Balanced	Balanced	Balanced	Unbalanced	Semi balanced	Balanced
Section 1	32	34	34	34	34	34	34
Section 2	28	30	28	30	30	28	30
Section 3	22	22	22	22	22	22	22
Section 4	20	20	20	18	18	18	18
Section 5	18	16	16	18	16	16	18
Section 6	22	22	22	22	22	22	22
Section 7	20	20	20	20	18	18	18
Section 8	26	24	26	26	26	26	26
Section 9	38	40	38	40	38	38	38
Section 10	34	36	36	36	36	36	38
Section 11	30	32	30	34	30	30	30
Section 12	28	30	28	30	30	28	30
Section 13	22	22	22	22	22	22	22
Section 14	18	20	20	18	18	18	18
Section 15	24	24	26	26	26	26	26
Section 16	32	32	30	34	32	38	30
Section 17	20	20	20	18	18	18	18
Section 18	22	22	26	22	22	22	22
Total amount of layers	456	466	464	470	458	460	460

Table 6.3: Results of the proposed method compared to the state-of-the-art [2, 9, 23, 25, 26]

In conclusion, the results of the proposed algorithm, for the horseshoe pattern benchmark case, are comparable to the methods described in literature in terms of weight. The amount of patches for this configuration is higher compared to the state-of-the-art, due to the relatively small amount of sections. Configurations consisting of a larger amount of sections are further discussed in section 6.3.

6.2 2D problems

The results of the 2D algorithm are demonstrated for a realistic design problem. This 2D design problem is derived from a square plate under uni-axial compression. The results for the realistic design problem are discussed in section 6.2.1. The maximum amount of sections in the square plate under uni-axial compression is limited. Therefore the effectivity of the algorithm is based on plates with randomly generated stacking sequences. The effectivity of the algorithm is described in section 6.2.2. The stability analysis of the algorithm can be found in section 6.2.3.

6.2.1 Results and discussion

The CA-based algorithm is demonstrated for a realistic 2D design problem of 21 sections. The 2D problem consists of one row of sections, which originates from a square plate under uni-axial compression. The square plate under uni-axial compression is elaborated in section 6.3. The local optima are the input for the CA-based algorithm. The local optima and the configurations after each stage of the algorithm are presented in figure 6.5.

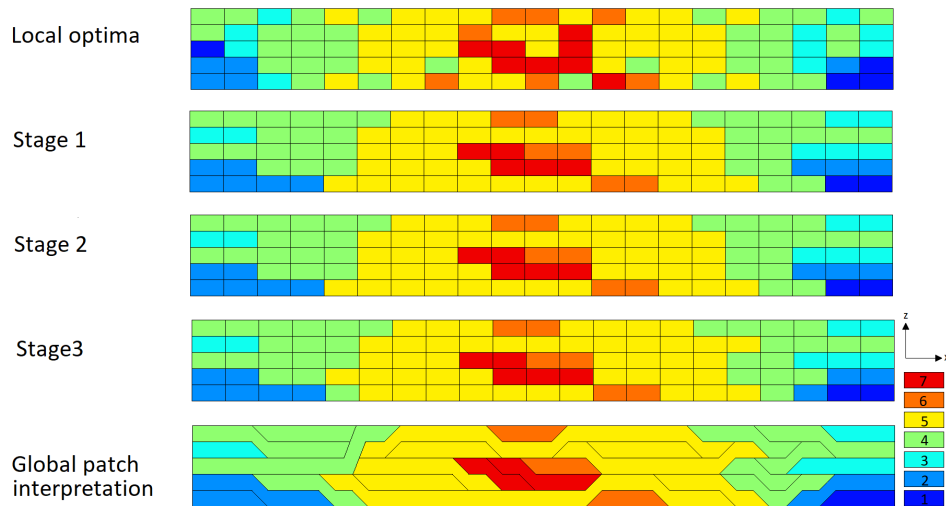


Figure 6.5: The configuration after each stage of the algorithm

The locally optimized stacking sequences change gradually from one section to another. Therefore, the amount of modifications steps required to obtain preliminary patches is relatively low, as shown in figure 6.6. Note that the implementation of a minimum patch size is not required to eliminate all butted edges. However, it may be imposed by a manufacturing constraint. Therefore, the amount of modification steps required to obtain a minimum patch size of 3 plies is also indicated in figure 6.6. The preliminary patches are further evolved to eliminate the butted edges. The configuration contains approximately 3 butted edges, depending on the global patch interpretation. All butted edges are eliminated in stage 3. The laminate configuration after stage 3 represents the final configuration of the locally blended laminate. A global interpretation of the patches, which is manually constructed by the author, is included in figure 6.5. The critical buckling load of the final configuration cannot be

determined, since the local optima are part of a 3D plate. The effect of the modification on the buckling load is further discussed for 3D plates in section 6.3.

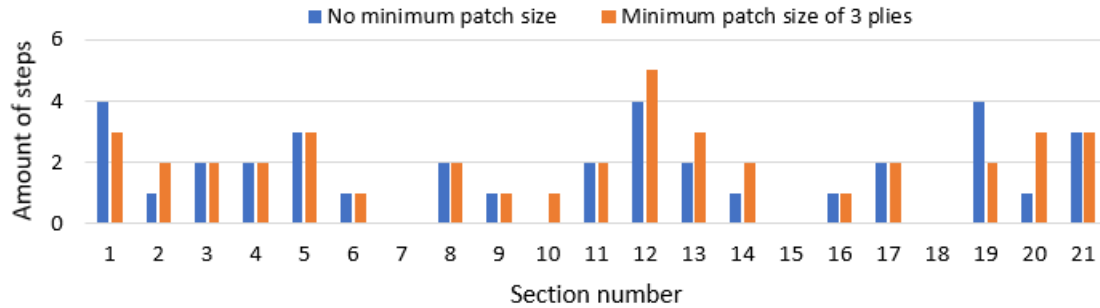


Figure 6.6: The amount of modification steps in each section of the plate presented in figure 6.5

The computational time required for each stage of the algorithm is presented in table 6.4. The first stage takes up the least amount of time. Although the amount of performed modifications is lower in stage 2 and 3, the analysis is more complex in these stages. The computational time required to complete all stages of the CA is 1.37 seconds. The computational time is used to determine the effectivity of the algorithm. The effectivity of the 2D algorithm is further discussed in section 6.2.2 for plates with a varying amount of sections.

	Computational time [s]
Stage 1	0.36
Stage 2	0.57
Stage 3	0.44

Table 6.4: Computational time for the plate presented in figure 6.5

6.2.2 Effectivity of the 2D algorithm

The effectivity of the algorithm for 2D problems is determined for plates with a varying amount of sections. The results of the varying problem sizes are shortly discussed, after which the effectivity of the algorithm is determined. The effectivity of the algorithm is based on plates with randomly generated stacking sequences, since the maximum amount of sections in the square plate under uni-axial compression is limited. Note that randomly generated stacking sequences do not gradually change from one section to another and are therefore not representative for a realistic design problem. Randomly generated stacking sequences generally require more modifications steps compared to gradually changing staking sequences. Therefore, randomly generated stacking sequences provide a conservative indication of the effectivity of the algorithm for realistic design problems.

The algorithm is applied to a set of design problems with a varying amount of sections and layers. The amount of section varies between 30, 40, 50 and 100 sections. The amount of layers varies between 5, 10 and 20 layers. Randomly generated stacking sequences are prone to small patches, due to the diversity and scatter of the fibre orientations. Small plies may impede a blended configuration. Therefore, the minimum patch size is set at 3 plies.

All design problems result in locally blended configurations. An illustration of these locally blended configurations can be found in appendix C. The blended configurations of three design problems with diverse problem sizes are presented in figure 6.7.

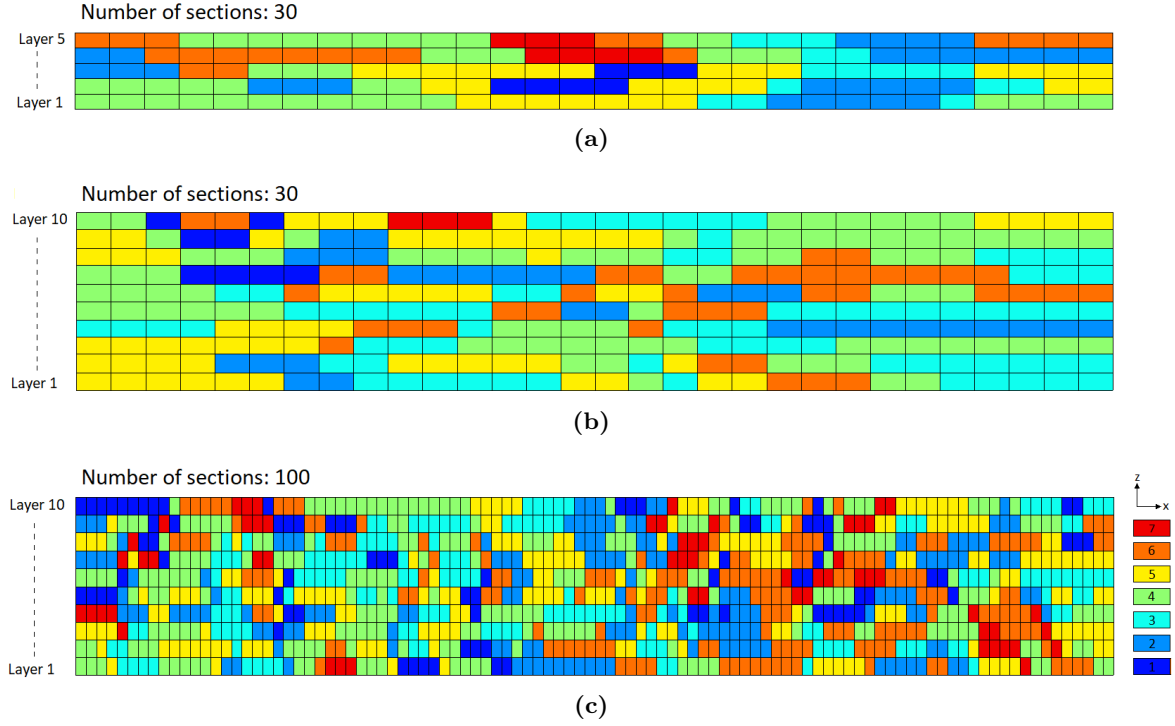


Figure 6.7: Locally blended configurations of three of the design problems

The total amount of modification steps, for each design problem, is indicated in table 6.8. The amount of modification steps is higher for the randomly generated stacking sequences compared to a realistic design problem. The realistic design problem described in section 6.2.1 is a laminated composite plate consisting of 21 sections and 5 layers. The total amount of modification steps for this design problem is 36, which is significantly lower compared to the design problems presented in table 6.8. This is expected since randomly generated stacking sequences do not gradually change from one section to another. As a result more modification steps are required compared to realistic design problems.

	30 sections	40 sections	50 sections	100 sections
5 layers	201	292	340	680
10 layers	468	616	720	1370
20 layers	813	1172	1460	2940

Figure 6.8: Total amount of modification steps for varying design problems

The effectivity in solving the dimensionality in the optimization of blending composite laminates is defined by the problem size (C) and the computational time (T) as described in section 3.2. The normalized time of the algorithm and the baseline are plotted against the size of the design problems on a logarithmic scale, as shown in figure 6.9. The effectivity can be defined by equation 6.1, as described in section 3.2.

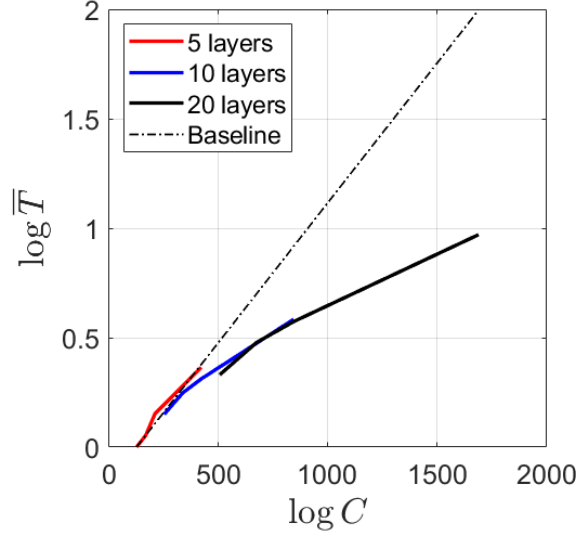


Figure 6.9: The effectivity of the proposed algorithm for randomly generated laminates with a varying amount of amount of layers

$$\text{Effectivity} = \frac{m_b}{m_a} \quad (6.1)$$

Where m_b is the slope of the baseline and m_a is the slope of the algorithm. The slope of the algorithm is not linear for small design problems. The computational time may differ for different configurations of the same problem size. The effect thereof is larger in small design problems compared to large design problems, which may cause the non-linearity in the domain of small design problems. In addition, small design problems encounter an increase of computational time due to the boundary conditions of the laminate. The outer most stacking sequences evolve towards one neighbour, while the inner stacking sequences evolve towards two neighbours. Note that the amount of layers imposes a similar effect of the boundaries. The boundary conditions have a larger impact on the computational time of small design problems compared to larger design problems. The slope of the algorithm becomes linear in the domain of larger design problems. The effectivity is based on the linear part of the slope, which is in the domain of $\log(C) > 1000$. The effectivity of the algorithm for randomly generated stacking sequences in this domain is 2.72.

6.2.3 Stability analysis

A stability analysis is performed to determine if the algorithm converges to a stable configuration. The analysis consist of an iterative process in which the results of each iteration are the input for the next iteration. Therefore, the input during each iteration is a locally blended configuration. The stability analysis is performed on a plate consisting of 15 sections and 5 layers. The local optima are randomly generated stacking sequences. The blended configuration, resulting from the evolution of the randomly generated stacking sequences, is depicted in figure 6.10a. This blended configuration is input for the first iteration of the stability analysis. The algorithm converges to a stable configuration after 11 iterations. No

modifications occur in iterations that follow after the 11th iteration. This final configuration is depicted in figure 6.10b. The intermediate configurations can be found in appendix D.

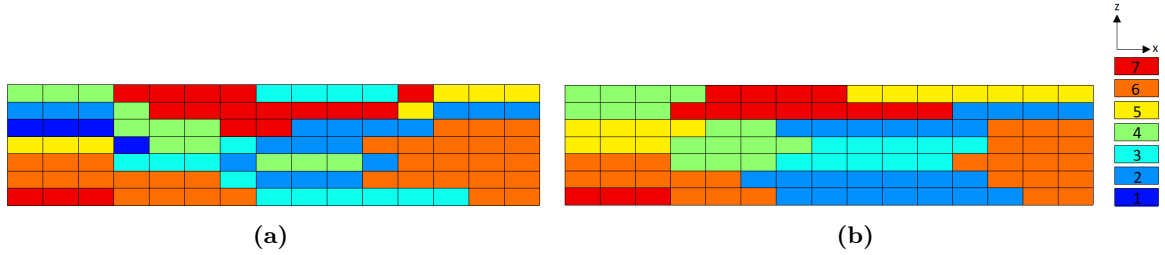


Figure 6.10: (a) Input of the stability analysis of a plate consisting of 7 layers (b) Final configuration of the stability analysis

The amount of required iterations depends on the distribution of patches and the length of the preliminary patches. An increase in the amount of layers increases the probability of a larger amount of required iterations. However, the effect of the amount of layers is negligible compared to the effect of the distribution and length of the patches. The first and final configuration of the stability analysis of a plate consisting of 15 layers is depicted in figure 6.11. The configuration converges to the stabilized configuration in 7 iterations.

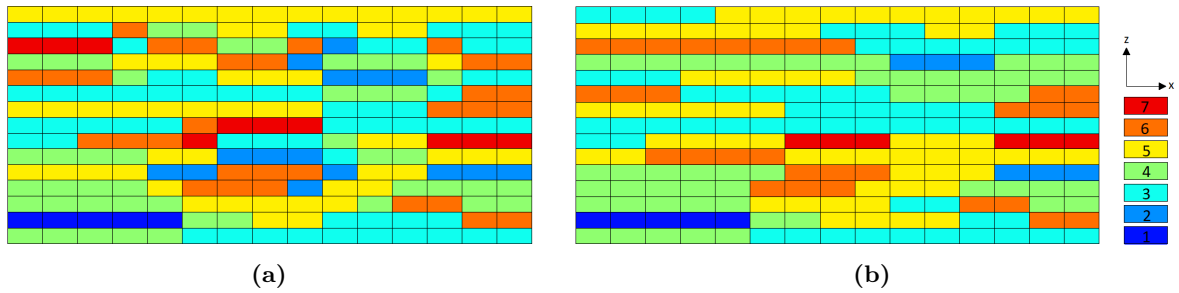


Figure 6.11: (a) Input of the stability analysis of a plate consisting of 15 layers (b) Final configuration of the stability analysis

6.3 3D problems

The proposed algorithm is demonstrated for a realistic design problem. The realistic design problem is a square plate under uni-axial compression with a varying amount of sections. The design problem is further elaborated in section 6.3.1. The generated results are presented in section 6.3.2. The square plate under uni-axial compression with a varying amount of sections is proposed a new benchmark problem to test the effectivity of a blending algorithm for plates with a large amount of sections. The effectivity of the algorithm for the proposed benchmark case is elaborated in section 6.3.3. The stability analysis of the 3D algorithm can be found in section 6.3.4.

6.3.1 Problem definition

The design problem to demonstrate the 3D algorithm is a square plate under uni-axial compression with a varying amount of sections. An illustration of the square plate with a cross-sectional dimension of 1m is provided in figure 6.12 [5]. The local optima, which are based on the Laminations Parameter (LP) distribution, are provided by van Campen [5] for a varying amount of sections and a varying amount of layers. These local optima are based on the following material properties: $E_1 = 181\text{GPa}$ (26.25Msi), $E_2 = 10.3\text{GPa}$ (1.49Msi), $G_{12} = 7.17\text{GPa}$ (1.04Msi) and $\nu_{12} = 0.28$.

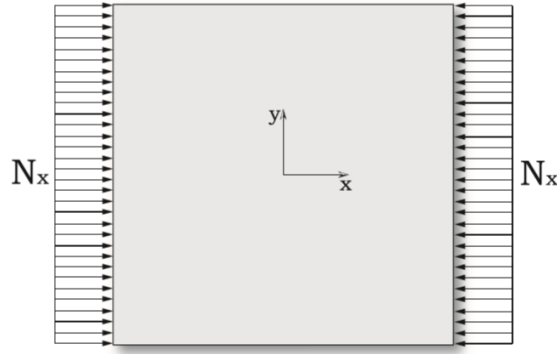


Figure 6.12: Square plate under uni-axial compression as depicted by van Campen [5]

The local optima approach the locally required stiffness and strength by increasing the amount of sections. An illustration of the effect of a varying amount of sections on the local optima is depicted in figure 6.13. The local optima in the same cross-section are provided for a varying amount of sections. The generated results, for a laminate consisting of 5 layers which is divided in 21x21 sections, are elaborated in section 6.3.2. The effectivity of the algorithm is determined based on a laminate consisting of 10 layers, which is divided in a varying amount of sections. The effectivity is further discussed in section 6.3.3.

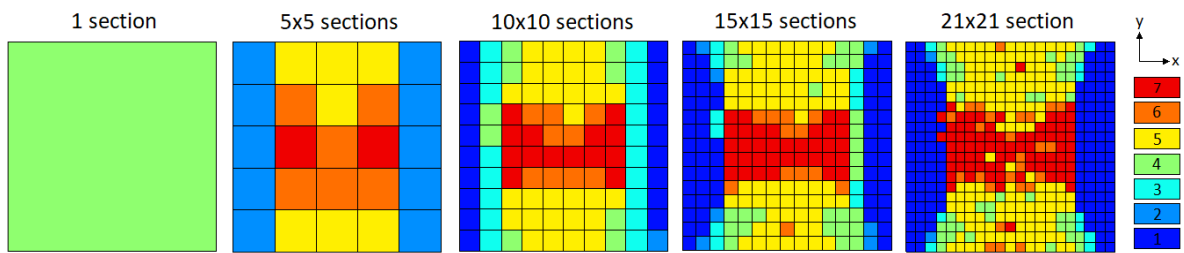


Figure 6.13: Illustrations of the local optima in the same cross-section for a varying amount of sections

6.3.2 Results and discussion

The square plate under uni-axial compression, consisting of 21x21 sections and 5 layers, is used to demonstrate the generated results. The local optima are the input for the CA-based algorithm. The local optima are depicted in figure 6.14. The laminate configurations and the

amount of modifications are presented after each stage of the algorithm. The configuration after the completion of stage 1 is depicted in 6.15. A minimum patch size of 3 plies is applied in stage 2. The configuration after the implementation of the minimum patch size is indicated in figure 6.16.

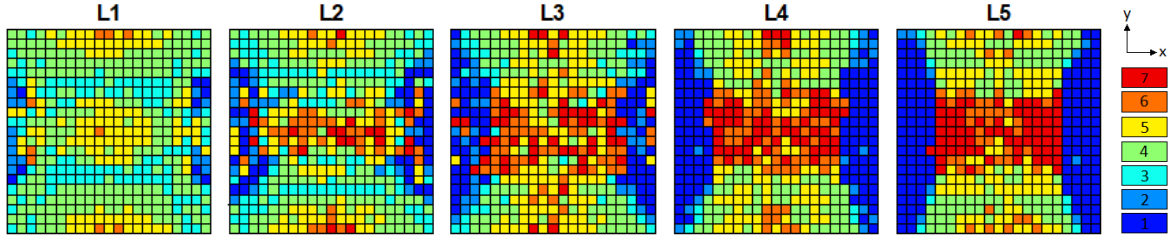


Figure 6.14: Local optima

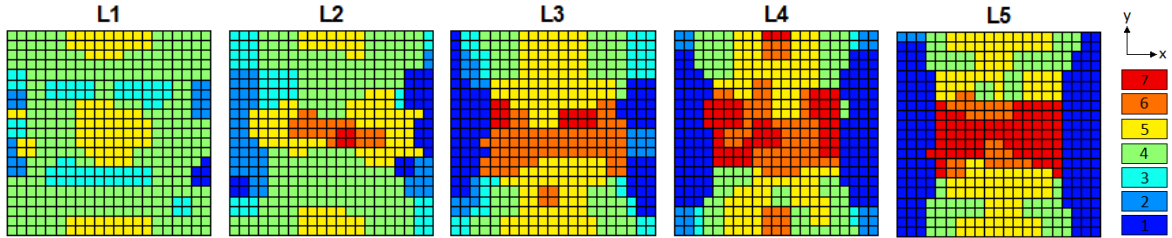


Figure 6.15: Results stage 1: preliminary patches

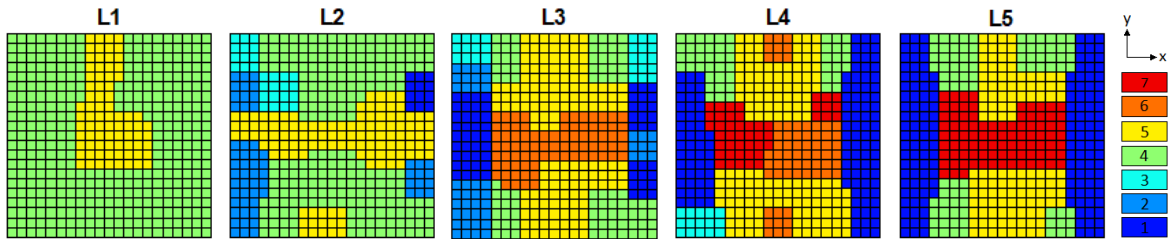


Figure 6.16: Results stage 2: Minimum patch size = 3 plies

Two cross-sections, in the x- and y-direction of the configuration after stage 2, are provided in figure 6.17. These cross-sections are representative of all unique cross-sections in this configuration. All unique cross-sections can be found in appendix F. Most transitions are locally blended due to the gradual change of the stacking sequences. However, some butted edges remain. These butted edges are eliminated in stage 3.

The laminate configuration after stage 3 is illustrated in figure 6.18. The configuration provides locally blended transitions in each unique x- and y-cross-section, as depicted in figure 6.19 for two of the unique cross-sections. All unique cross-sections can be found in appendix F. The configuration does not represent patches on a global level. This is a result of the local analysis. In the local analysis, a preliminary patch is not analysed as a whole. A single cross-section is analysed in each analysis. This strategy may result in the fractionation of the patches. The fractionation of patches impede a global interpretation of the patches for

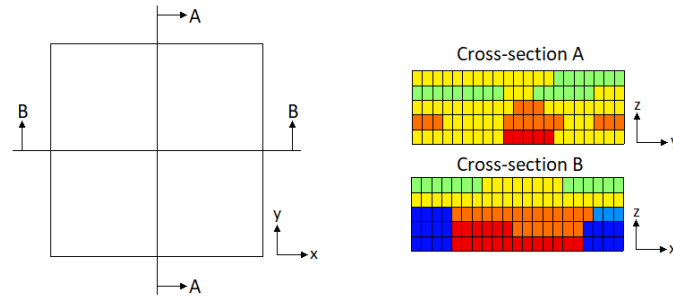


Figure 6.17: Cross-sections of the configuration after stage 2

this design problem. A more global approach in stage 3 could provide a solution to ensure a feasible global patch interpretation. This is further discussed in chapter 7.

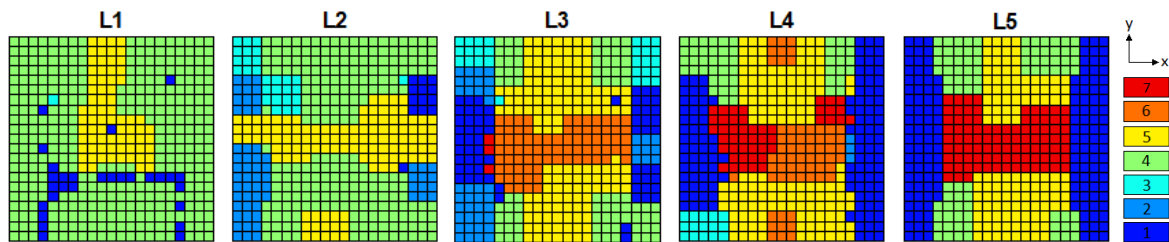


Figure 6.18: Results stage 3: Elimination of butted edges

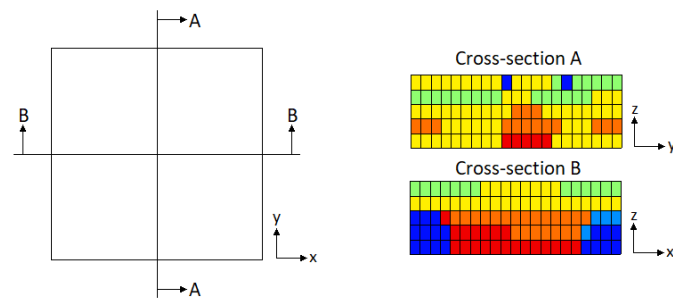


Figure 6.19: Cross-sections of the configuration after stage 3

The amount of modification steps in a section, after each stage of the algorithm, is illustrated in figure 6.20. More than half of the sections require 0, 1 or 2 modification steps in order to create preliminary patches in the first stage of the algorithm. The amount of modification steps is relatively low, since the stacking sequences gradually change from one section to another. Therefore, the influence of the modifications on the critical buckling load is assumed to be minor. The influence on the critical buckling load is further discussed in section 6.3.3 for a plate with varying amount of sections.

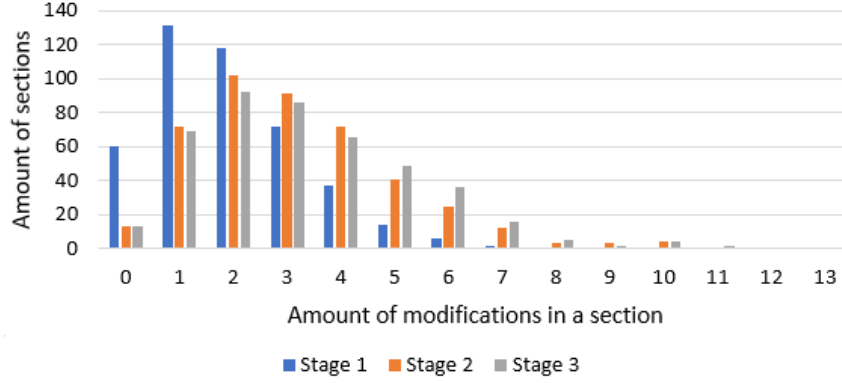


Figure 6.20: Amount of sections versus the amount of modification steps in a section

6.3.3 Effectivity of the 3D algorithm

The effectivity of the blending strategy is demonstrated by the square plate under uni-axial compression, which is introduced in section 6.3.1. The laminate consists of 10 layers. The laminate configuration is optimized for a varying amount of sections. The amount of sections in which the plate is divided, are identical in x- and y- direction. The amount of sections in each direction varies between 1 and 21 sections.

The dimensions of a single ply depend on the total amount of sections in a plate. The minimum patch size is defined by the amount of consecutive plies in each unique cross-section. No minimum patch size is implemented in the effectivity analysis, since this would distort the results due to a varying size of the sections. During stage 3, the algorithm is thrown off course due the fractionation of patches. Therefore, this stage is not included in the effectivity analysis. The final plate configuration in the effectivity analysis consists of the preliminary patches of stage 1.

The effectivity in solving the dimensionality in the optimization of blending composite laminates is defined by the problem size (C) and the computational time (T) as discussed in chapter 3. The normalized computational time of the algorithm and the baseline are plotted against the size of the design problem on a logarithmic scale, as indicated in figure 6.21. The effectivity in solving the dimensionality of the optimization problem is determined by equation 6.2. Where m_a is the slope of the algorithm and m_b is the slope of the baseline.

$$\text{Effectivity} = \frac{m_b}{m_a} \quad (6.2)$$

The slope of the algorithm is assumed to be linear in the domain of large design problems as discussed in section 6.2.2. The degree of effectivity is based on the slope in the domain: $3700 > \log(C) > 3000$. The effectivity of the algorithm for the proposed benchmark case in this domain is 27.37. The degree of effectivity is high since the algorithm is compared to a baseline which is derived from an analysis based on enumeration.

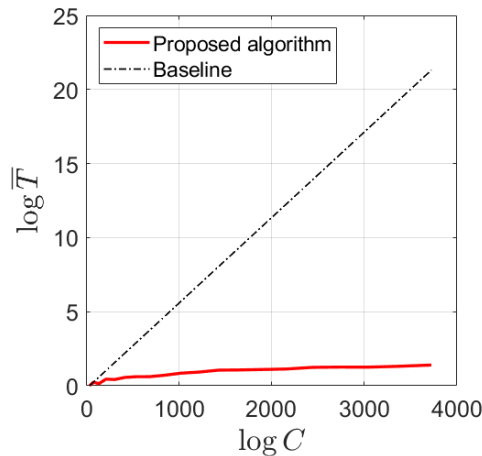


Figure 6.21: The effectivity for a square plate under uni-axial compression with an varying amount of sections

The described method to determine the effectivity of an algorithm can be used to compare the effectivity of blending strategies for a large amount of sections. When multiple algorithms can handle the same dimensionality, the quality of the results should be included in analysis. This recommendation is further discussed in chapter 7. The quality of the results can be described by the critical buckling load in this design problem. The critical buckling loads, corresponding to the number of sections in which the design problem is divided, are presented in figure 6.22. The critical buckling load of the local optima increases due to the increase of the amount of sections. The increase in the critical buckling load remains after the local optima are transformed to preliminary patches. This proves that the algorithm allows the design space required for local optimization, in plate configurations with a large amount of sections.

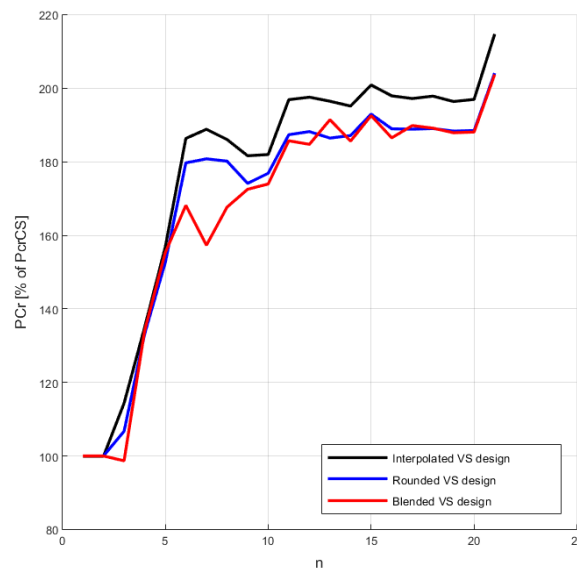


Figure 6.22: Critical buckling load for a varying amount of sections

6.3.4 Stability analysis

A stability analysis is performed to determine if the algorithm converges to a stable configuration. The analysis consists of an iterative process in which the results of each iteration are the input for the next iteration. Therefore, the input of each iteration is a locally blended configuration. The stability analysis is performed on the square plate under uni-axial compression, which is divided in 21x21 sections. The laminate consists of 10 layers. The locally blended configuration, resulting from the evolution of the local optima, is depicted in figure 6.23. This configuration is the input of the stability analysis. The algorithm converges to a stable configuration in 5 iterations. The stabilized configuration is depicted in figure 6.24.

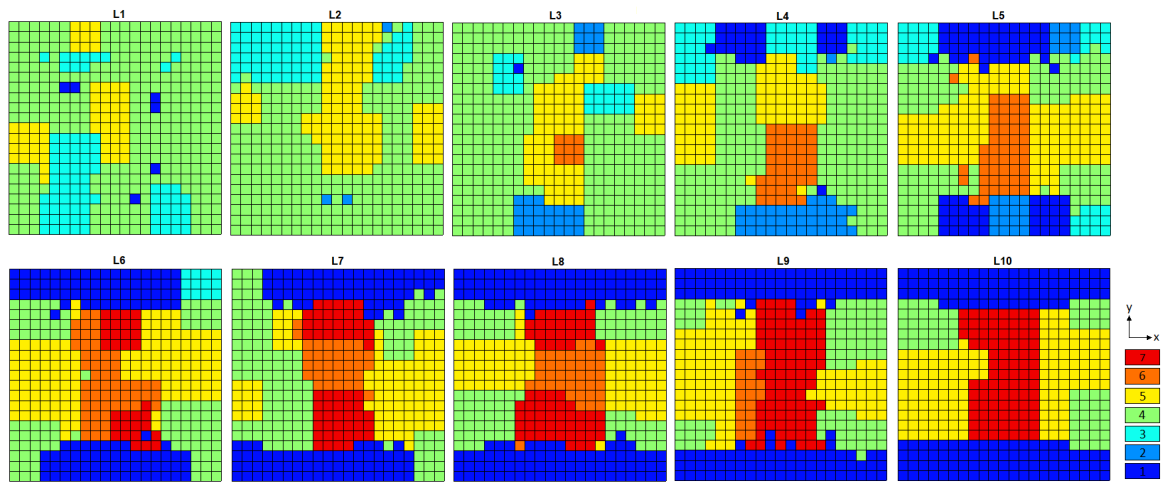


Figure 6.23: Input of the stability analysis

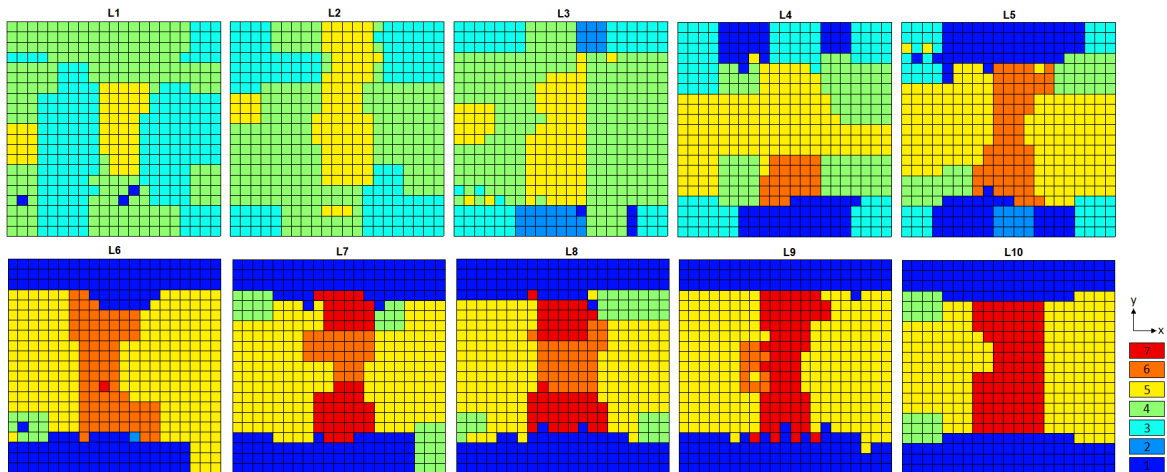


Figure 6.24: The stabilized configuration resulting from the stability analysis

The input of the stability analysis is a configuration which is locally blended. The configuration does not provide a global patch interpretation as discussed in 6.3.2. During the stability analysis, the amount of isolated plies and extremities decrease. However, the final configuration of the stability analysis still contains isolated plies and does not provide a global patch

interpretation. Although the results do not provide a global patch interpretation, the results converge to a stabilized configuration. Based on the results of the stability analysis, it can be concluded that the algorithm results in a stable solution.

6.4 Conclusions

The results of the proposed algorithm, for the horseshoe pattern benchmark case, are comparable to the results of the state-of-the-art methods in terms of weight. The CA-based algorithm is designed for plates with a large amount of sections. The horseshoe pattern benchmark case consists of a relatively small amount of sections. As a consequence, the results of the proposed algorithm consist of more patches, which increases the manufacturing effort compared to the state-of-the-art methods.

A new benchmark problem is proposed to demonstrate the effectivity of an algorithm for plates with a large amount of sections. The benchmark case is a square plate under uni-axial compression which is divided in a varying amount of sections. The computational time of the proposed algorithm increases linearly with an increase in the problem size. A method is established to determine the degree of effectivity of the algorithm for the proposed benchmark problem. The effectivity of blending methods, that can handle a similar dimensionality, can be compared by the degree of effectivity. The degree of effectivity of the CA-based algorithm is 27.37 in the domain $3700 > \log(C) > 3000$. The computational time of the proposed algorithm increases linear with an increase in the size of the design problem. Therefore, the degree of effectivity of the proposed algorithm is high compared to the baseline which is derived from an analysis based on enumeration. The quality of the results can be described by the critical buckling load. The critical buckling load increases for an increasing amount of sections. This proves the algorithm provides the design space required for local optimization. For 3D problems, a feasible global patch interpretation is not guaranteed. This can be solved by a more global strategy for the elimination of butted edges.

Conclusions and recommendations

The objective of this thesis is to overcome the dimensionality in blending laminated composite plates by creating an optimization tool that can effectively blend composite laminated plates with a large amount of sections. The conclusions are described in section 7.1. The recommendations are provided in section 7.2.

7.1 Conclusions

The dimensionality in the optimization of blended laminated composite plates is defined by the amount of design solutions. The amount of design solutions increases exponentially with an increase in the number of sections. The number of sections influences the extent to which the locally required stiffness in a plate can be approached. Therefore, the optimization of a blended plate with a large amount of sections and a varying load distribution necessitates local optimization. The state-of-the-art does not provide a method that allows local optimization for plates consisting of a large amount of sections, due to the dimensionality of the blending problem.

An innovative algorithm is developed in MATLAB to overcome the dimensionality in blending laminated composite plates. The proposed method is based on local dependencies between sections, which are derived from the global blending constraints. A CA-based algorithm evolves the local optima into a blended configuration by a straightforward set of rules based on these local dependencies.

The algorithm provides configurations with a globally blended interpretation for 2D problems. For 3D problems, the algorithm generates locally blended configurations. However, the final configuration does not guarantee a feasible patch interpretation on a global level.

The results of the proposed algorithm are compared to the results of the state-of-the-art methods for the horseshoe pattern benchmark case. The proposed algorithm provides a globally blended configuration, in which the weight is comparable to the configurations provided by the state-of-the-art methods.

A new benchmark case is proposed to demonstrate the effectivity of the blending algorithm for plates with a larger amount of sections than currently found in literature. The proposed benchmark case is a square plate under uni-axial compression with a varying amount of sections. A method is established to determine the degree of effectivity in order to compare the effectivity of blending algorithms that can handle a large amount of sections. The computational time of the proposed algorithm increases linear with an increase in the size of the design problem. Therefore, the degree of effectivity of the proposed algorithm is high compared to the baseline which is derived from an analysis based on enumeration.

7.2 Recommendations

The recommendations are described in this section. The recommendations are divided in two categories. The recommendations to improve the proposed algorithm are described in section 7.2.1. The recommendation regarding future research are described in section 7.2.2.

7.2.1 Recommendations to improve the proposed algorithm

1. Elimination of butted edges at a patch level

The 3D algorithm does not guarantee a globally blended result. A clustered distribution of butted edges can throw the algorithm off course. The algorithm provides locally blended transitions in each cross-section. However, the configurations do not represent patches at a global level. Implementing extension and/or swap of the patches is an effective way to solve the butted edges as demonstrated in the 2D algorithm. In the 2D algorithm, swap and extension are implemented by rules based on local dependencies. In the 3D algorithm, the patches may fractionate by rules based on the local dependencies. A more global approach to eliminate the butted edges is recommended, in which the patches are extended and/or swapped.

2. Determination of the local optima

The input of the proposed algorithm consist of the local optima. For the horseshoe pattern benchmark case, the assumption is made that the sections have simply supported boundaries. However, the proposed algorithm is designed to increase the amount of sections in a plate, in order to approach the locally required stiffness and strength. The assumption of simply supported boundaries conditions no longer remains viable when the amount of sections increases. It is therefore recommended that the configurations are analysed using a FE method.

7.2.2 Recommendations for future research

1. Specification of the structural requirements in complex structures

The proposed method implements the relaxed generalized blending definition. The design freedom for local optimization is therefore larger compared to the state-of-the-art methods. This may results in a larger amount of patches in a configuration. The distribution of the patches and the corresponding ply drops in a laminate configuration affect the integrity of a structure. The effect on the structural integrity becomes increasingly

important for structures which are more complex than plates. It is therefore recommended to test the effect of those variables for complex structures in order to further specify the structural requirements.

2. Include the manufacturing process in the optimization process

The manufacturing restrictions and accuracy influence the design freedom in the optimization process of blending laminated composites. The manufacturing restrictions and accuracy vary per manufacturing process. Combining the manufacturing process and the optimization process can therefore further optimize a blended configuration.

3. Include the quality of the results in the comparison of blending algorithms

The effectivity of an algorithm can be used to compare blending strategies for plates with a large amount of sections. When future research generates methods which have a similar effectivity, the quality of the results should be included in the comparison. Therefore, it is recommended to extend the method to compare algorithms by including the quality of the results.

References

- [1] Adams, D.B., Watson, L.T. and Gürdal, Z. (2003). "Optimization and Blending of Composite Laminates Using Genetic Algorithms with Migration." *Mechanics of Advanced Materials and Structures*, 10(3): 183-203.
- [2] Adams, D.B., Watson, L.T., Gürdal, Z. and Anderson-Cook, C.M. (2004). "Genetic algorithm optimization and blending of composite laminates by locally reducing laminate thickness." *Advances in Engineering Software* 35(1): 35-43.
- [3] An, W., Chen, D. and Jin, P., (2013). "A single-level composite structure optimization method based on a blending tapered model." *Chinese Journal of Aeronautics*, 26(4): 943-947.
- [4] van Campen, J.M.J.F., Seresta, O., Abdalla, M.M. and Gürdal, Z. (2008). "General Blending Definitions for Stacking Sequence Design of Composite Laminate Structures." 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 16th AIAA/ASME/AHS Adaptive Structures Conference, 10th AIAA Non-Deterministic Approaches Conference, 9th AIAA Gossamer Spacecraft Forum, 4th AIAA Multidisciplinary Design Optimization Specialists Conference, Schaumburg, IL
- [5] van Campen, J.M.J.F. (2011). "Optimum lay-up design of variable stiffness composite structures", Delft University of Technology, PhD thesis, **uuid:148c69e4-50a3-4990-bc2b-0e76a2e16bdf**
- [6] Fan, H.T., Wang, H. and Chen, X.H. (2016). "An optimization method for composite structures with ply-drops." *Composite Structures* 136: 650-661.
- [7] Giger, M., Keller, D. and Ermanni, P. (2008). "A graph-based parameterization concept for global laminate optimization." *Struct Multidisc Optim* 36: 289-305.
- [8] Ghiasi, H., Pasini, D. and Lessard, L. (2009). "Optimum stacking sequence design of composite materials Part I: Constant stiffness design." *Composite Structures*, 90(1): 1-11.

-
- [9] IJsselmuiden, S.T., Abdalla, M.M., Seresta, O. and Gürdal, Z.(2009). "Multi-step blended stacking sequence design of panel assemblies with buckling constraints." *Composites Part B: Engineering* 40(4): 329-336.
 - [10] Irisarri, F.X, Abdalla, M. and Gürdal, Z., (2011). "Improved Shepard's Method for the Optimization of Composite Structures." *AIAA Journal*, 49(12): 2726-2736.
 - [11] Jin, P., Zhong, X., Yang, J. and Sun, Z. (2016). "Blending design of composite panels with lamination parameters." *The Aeronautical Journal*, 120(1233): 1710-1725.
 - [12] Jing, Z., Fan, X. and Sun, Q. (2015). "Global shared-layer blending method for stacking sequence optimization design and blending of composite structures." *Composites Part B: Engineering*, 69: 181-190.
 - [13] Kari, J., (2005). "Theory of cellular automata: A survey." *Theoretical computer science*, 334(1-3): 3-33.
 - [14] Kassapoglou, C. (2010). "Design and Analysis of Composite structures with applications to Aerospace Structures, John Wiley & Sons Ltd
 - [15] Kristinsdottir, B.P., Zabinsky, Z.B., Tuttle, M.E. and Neogi, S. (2001). "Optimal design of large composite panels with varying loads." *Composite Structures*, 51(1): 93-102.
 - [16] Liu, B., and Haftka, R. T. (2001). "Composite wing structural design optimization with continuity constraints." *Proceedings of the 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Seattle, WA, USA.
 - [17] Liu, D., Toropov, V.V., Querin, O.M. and Barton, D.C. (2009). "Bi-level Optimization of Blended composite Panels." *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Palm Springs, CA, USA.
 - [18] Macquart, T., Bordogna, M.T., Lancelot, P. and De Breuker, R. (2016). "Derivation and application of blending constraints in lamination parameter space for composite optimisation." *Composite Structures* 135: 224-235.
 - [19] McMahon, M. T., Watson, L. T., Soremekun, G. A., Gürdal, Z., and Haftka, R. T. (1998). "A FORTRAN 90 genetic algorithm module for composite laminate structure design." *Engineering with Computers* 14(3): 260-273.
 - [20] McMahon, M.T. and Watson, L.T. (2000). "A distributed genetic algorithm with migration for the design of composite laminate structures." *Parallel Algorithms And Application*, 14(4): 329-362.
 - [21] Meddaikar, Y., Irisarri, F.X. and Abdalla, M. (2015). "Blended Composite Optimization combining Stacking Sequence Tables and a Modified Shepards Method." *11th World Congress on Structural and Multidisciplinary Optimization*, Sydney, Australia.
 - [22] Nagendra, S., Jestin, D., Gürdal, Z., Haftka, R. T., and Watson, L. T. (1996). "Improved genetic algorithm for the design of stiffened composite panels." *Computers and Structures* 58(3): 543-555.

-
- [23] Le Riche, R. and Irisarri, F.X. (2014). "Multiscale composite optimization with design guidelines." Palaiseau - IRT SystemX. In Journée d'étude: Optimisation des composites, Palaiseau, France.
 - [24] Seresta, O., Gürdal, Z., Adams, D. B., and Watson, L. T.,(2007). "Optimal design of composite wing structures with blended laminates." *Composites Part B: Engineering* 38(4): 469-480.
 - [25] Seresta, O., Abdalla, M. and Gürdal, Z. (2009). A genetic algorithm based blending scheme for design of multiple composite laminates. In 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 17th AIAA/ASME/AHS Adaptive Structures Conference, Palm springs, CA, USA.
 - [26] Soremekun, G., Gürdal, Z., Kassapoglou, C., and Toni, D. (2002). "Stacking sequence blending of multiple composite laminates using genetic algorithms." *Composite Structures*, 56(1): 53-62.
 - [27] Yang, J., Song, B., Zhong, X. and Jin, P. (2016). "Optimal design of blended composite laminate structures using ply drop sequence." *Composite Structures* 135: 30-37.
 - [28] Zabinsky, Z.B. (1994). "Global optimization for composite structural design." 35th Structures, Structural Dynamics, and Materials Conference, Hilton Head, SC, U.S.A.
 - [29] Zein, S., Basso, P. and Grihon, S. (2014). "A constraint satisfaction programming approach for computing manufacturable stacking sequences." *Computers and Structures*, 136: 56-63.
 - [30] Zein, S. and Bruyneel, M. (2015). "A bilevel integer programming method for blended composite structures." *Advances in Engineering Software* 79: 1-12.

Appendix A

Critical buckling load

In figure A.1 [14], a schematic representation is given of a composite laminated plate. The plate is assumed to be simply supported at all edges. The dimensions of the plate are defined by a and b . N_x and N_y represent the applied loads in x and y direction.

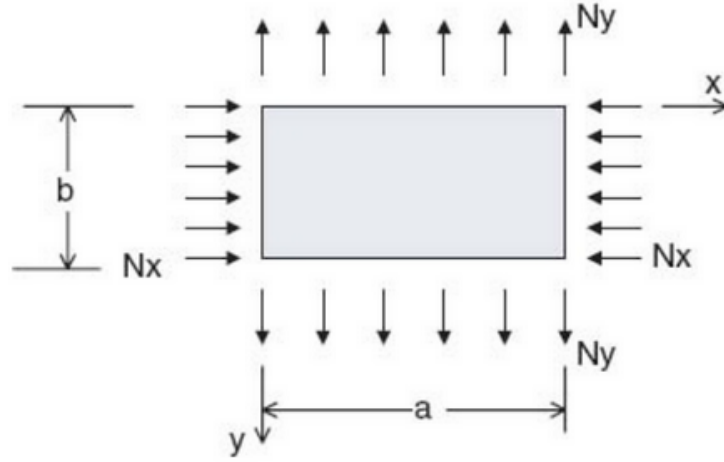


Figure A.1: A schematic representation of a laminated composite plate [14]

The method for calculating the critical buckling load N_0 in this research is dependent on the D-matrix. For a composite panel under biaxial loading, equation A.1 can be used to calculate N_0 .

$$N_0 = \frac{\pi^2 [D_{11}m^4 + (D_{12} + 2D_{66})m^2n^2(AR)^2 + D_{22}n^4(AR)^4]}{a^2(m^2 + kn^2(AR)^2)} \quad (\text{A.1})$$

Where m and n are the number of half-waves in x - and y -direction, $AR = \frac{a}{b}$ and $k = \frac{N_y}{N_x}$. If the loading ratio k is set, the applied load can be represented by N_x alone.

Appendix B

Detailed scenarios for 2D problems

A schematic overview of the search area during the analysis of a ply is given in figure B.1. Cell X indicates the analysed cell. The range of layers included in the analysis of a ply is 3 as described in section 5.1.2. The analysed ply is outlined. A cell in a 2D problem has a maximum of 2 neighbours. N1 and N2 indicate the neighbours of cell X. All plies which are given the same color are assumed to have similar fibre orientations. In case a ply does not contain a color, the ply can consist of all fibre orientations except the orientations that are already indicated by a color.

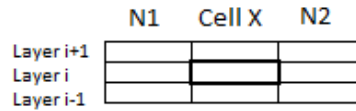


Figure B.1: schematic overview of the search area during the analysis of a ply

Category 1: Overlapping edges

Category 1 consists of scenarios with overlapping edges. All scenarios which are included in this category are depicted in figure B.2. In these scenarios the butted edge can be eliminated by swapping the edges.

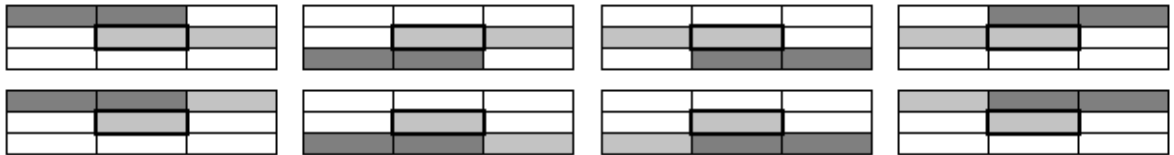


Figure B.2: Detailed scenarios of category 1

Note that not all overlapping scenarios are included in this category. The scenarios for which the swap results in fractionation of the patch are excluded. The analysed ply becomes an isolated ply after the swap according to the patch definition described in section 5.1.2. The excluded scenarios are depicted in figure B.3.



Figure B.3: Excluded scenarios of category 1

Category 2: Surrounded by a continuous patch

Category 2 contains the scenarios where the butted edge is surrounded by a continuous patch. The detailed scenarios of category 2 are depicted in figure B.4.

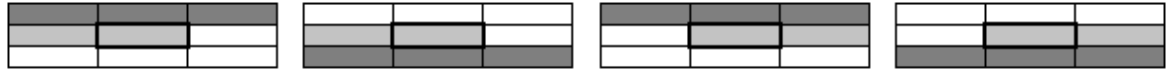


Figure B.4: Detailed scenarios of category 2

The modification in this category consists of swapping patches. Additional scenarios are required after the swapping is initiated; the transversely linked patches. The detailed scenarios of the transversely linked patches are given in figure B.5.



Figure B.5: Transversely linked patch scenarios

Category 3: Stacked butted edges

Category 3 consists of scenarios with two butted edges at the same location in adjacent layers. The detailed scenarios of the scenarios in this category are given in figure B.6.

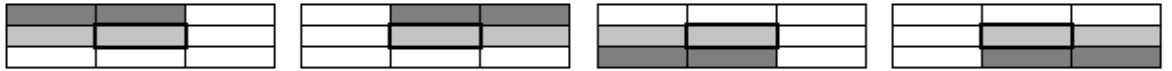


Figure B.6: Detailed scenarios of category 3

Appendix C

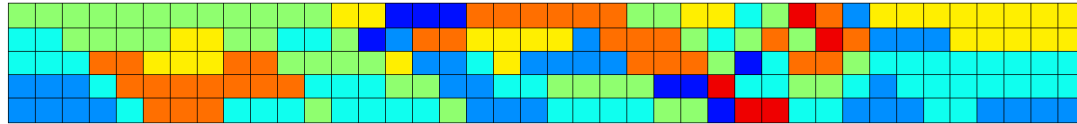
Results 2D plates

The results of the laminates consisting of 5 layers are presented in figure C.1. In figure C.2 the results for the laminates consisting of 10 layers are depicted. The results of the laminates consisting of 20 layers can be found in figure C.3.

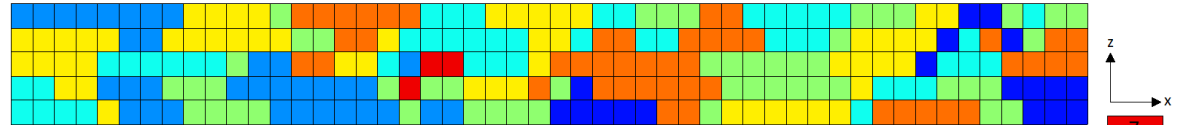
Number of sections: 30



Number of sections: 40



Number of sections: 50



Number of sections: 100

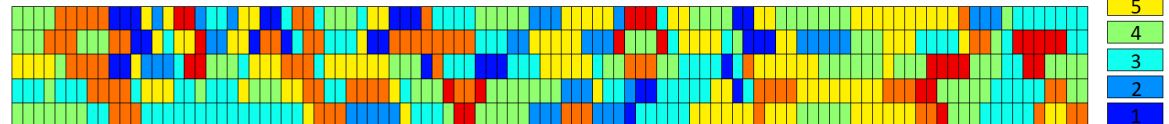
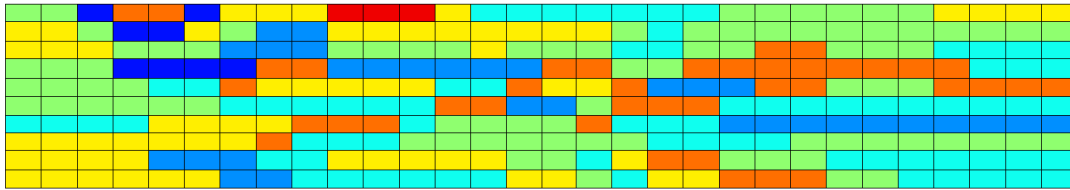
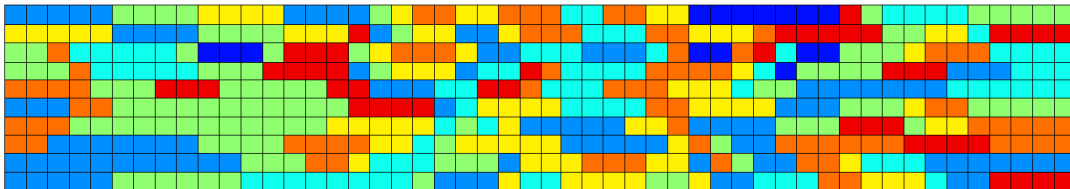


Figure C.1: Blended configuration of the laminates consisting of 5 layers

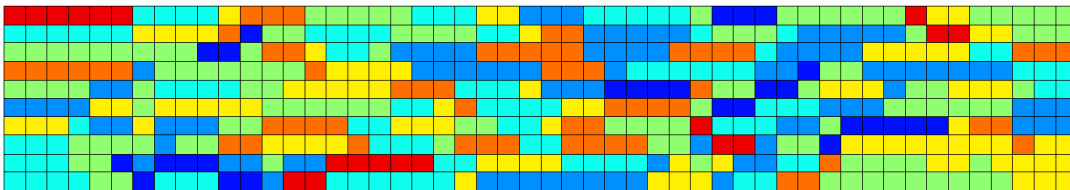
Number of sections: 30



Number of sections: 40



Number of sections: 50



Number of sections: 100

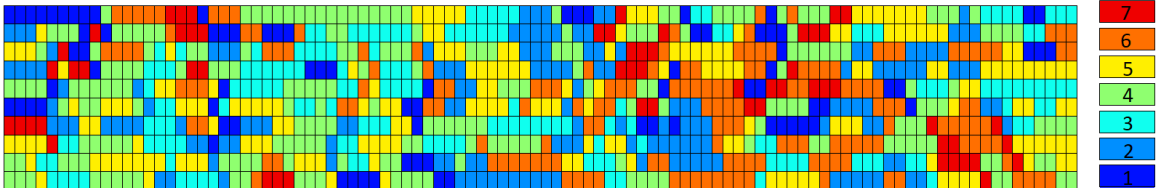
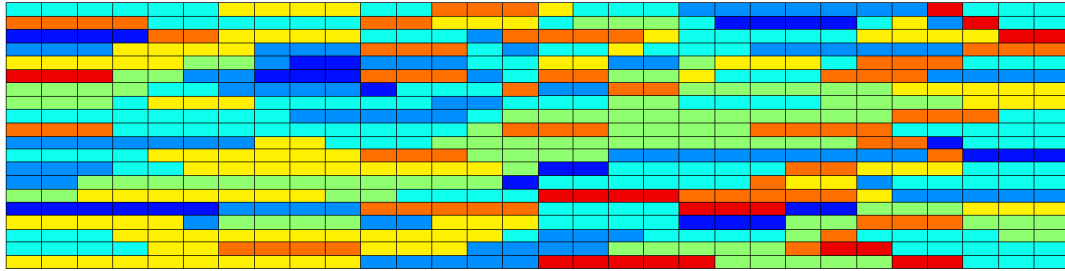
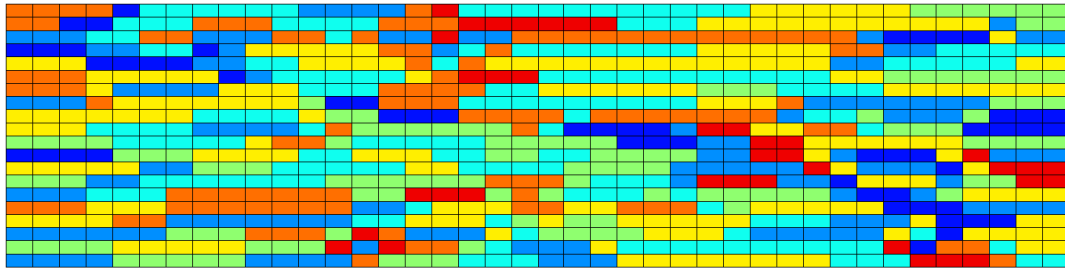


Figure C.2: Blended configuration of the laminates consisting of 10 layers

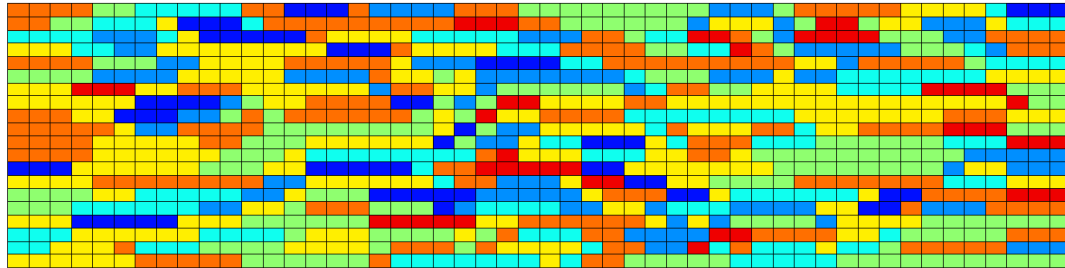
Number of sections: 30



Number of sections: 40



Number of sections: 50



Number of sections: 100

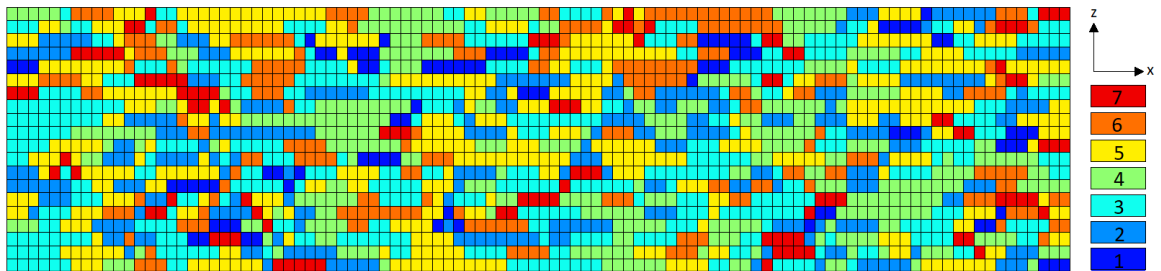


Figure C.3: Blended configuration of the laminates consisting of 20 layers

Appendix D

Results stability analysis

The intermediate results of the stability analysis are presented in figure [D.1](#). The results of each iteration are input for the next iteration. The results are converged after 11 iterations, no changes occur in subsequent iterations.

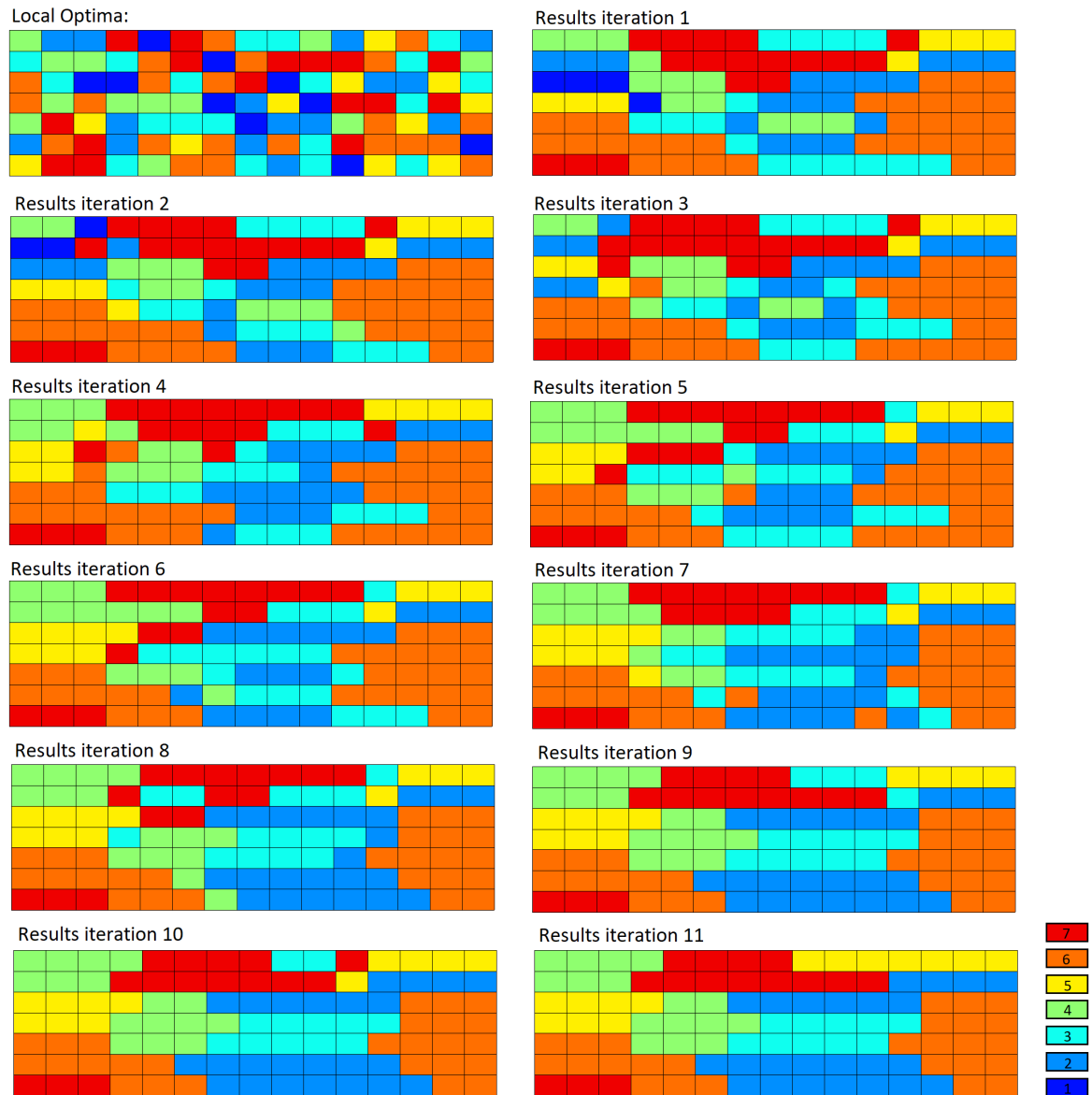


Figure D.1: The intermediate results of the stability analysis

Appendix E

Benchmark case horseshoe pattern

The geometry and the applied loads of the horseshoe pattern benchmark case are provided in figure E.1 [9, 26]. The material used in this design problem is a graphite-epoxy(IM7/8552). The properties of this material are: $E_1=141\text{GPa}$, $E_2= 9.03\text{GPa}$, $G_{12}=4.27\text{GPa}$, $v_{12}=0.32$ and $t_{ply}=0.191\text{mm}$.

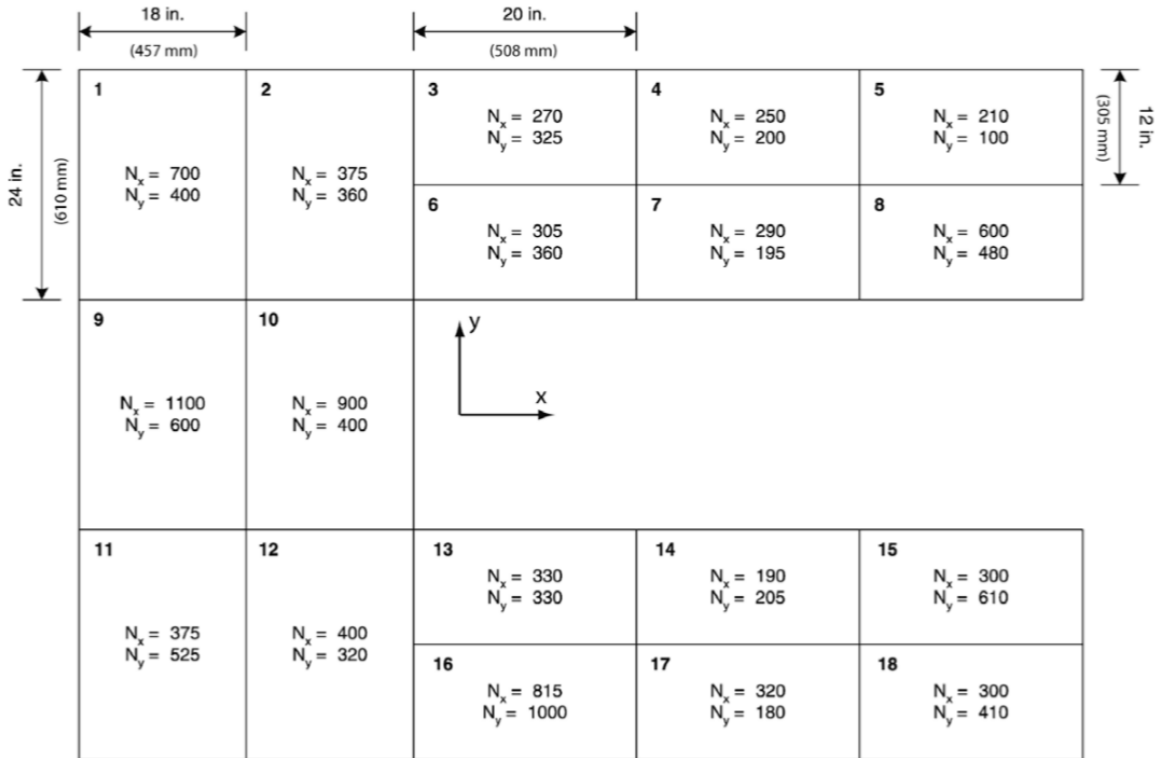


Figure E.1: Horseshoe pattern [9, 26], loads in lbf/in.

The parameter setting of the GA for the horseshoe pattern benchmark case are provided in table E.1.

Application probabilities		GA parameters	
Crossover	0.75	Number of generations	100
Mutation	0.05	Population size	75
Permutation	0.05	Maximum number of layers	25
Swap	0.05	β	5
Addition	0.05	ε	0.01
Deletion	0.05		
Elitist selection	1		

Table E.1: Parameter setting of the GA for the horseshoe pattern benchmark case

Appendix F

All unique cross-sections of a 5 layer laminate

The results of the design problem described in section 6.3.2 are presented in this appendix. The results of stage 2 are depicted in figure F.1. All unique cross-sections of the results from stage 2 are depicted in figure F.3 and F.4. The results of stage 3 are depicted in figure F.2. All unique cross-sections of the results from stage 3 are depicted in figure F.5 and F.6.

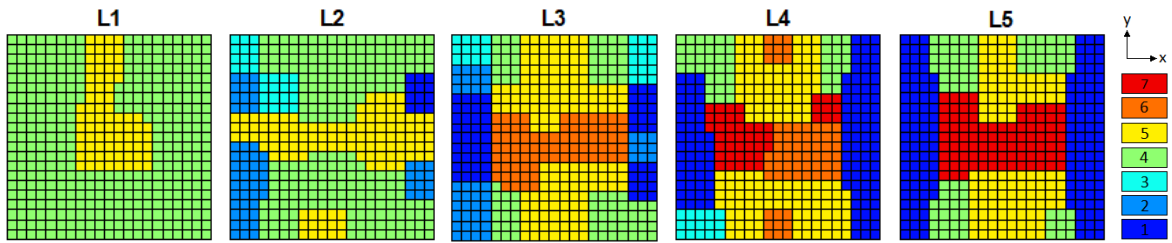


Figure F.1: Results stage 2: Minimum patch size = 3 plies

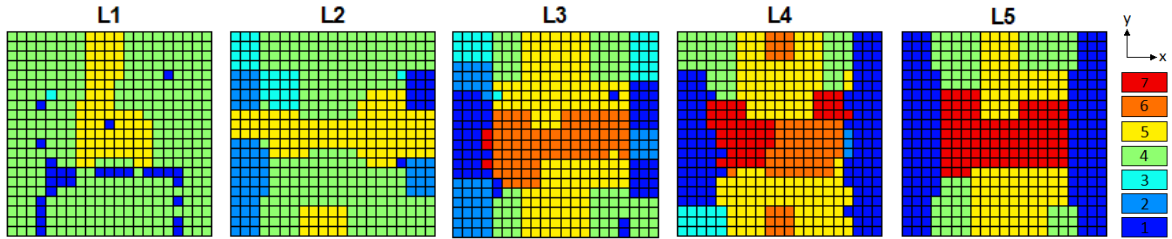


Figure F.2: Results stage 3: Elimination of butted edges

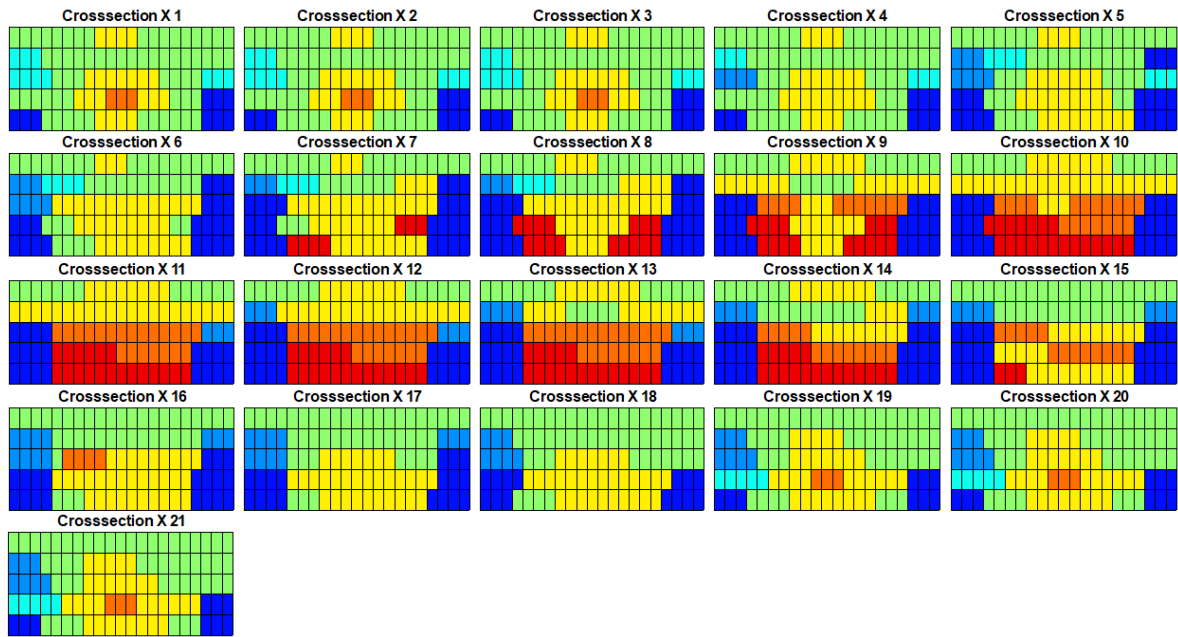


Figure F.3: Cross-sections in the x-direction after stage 2

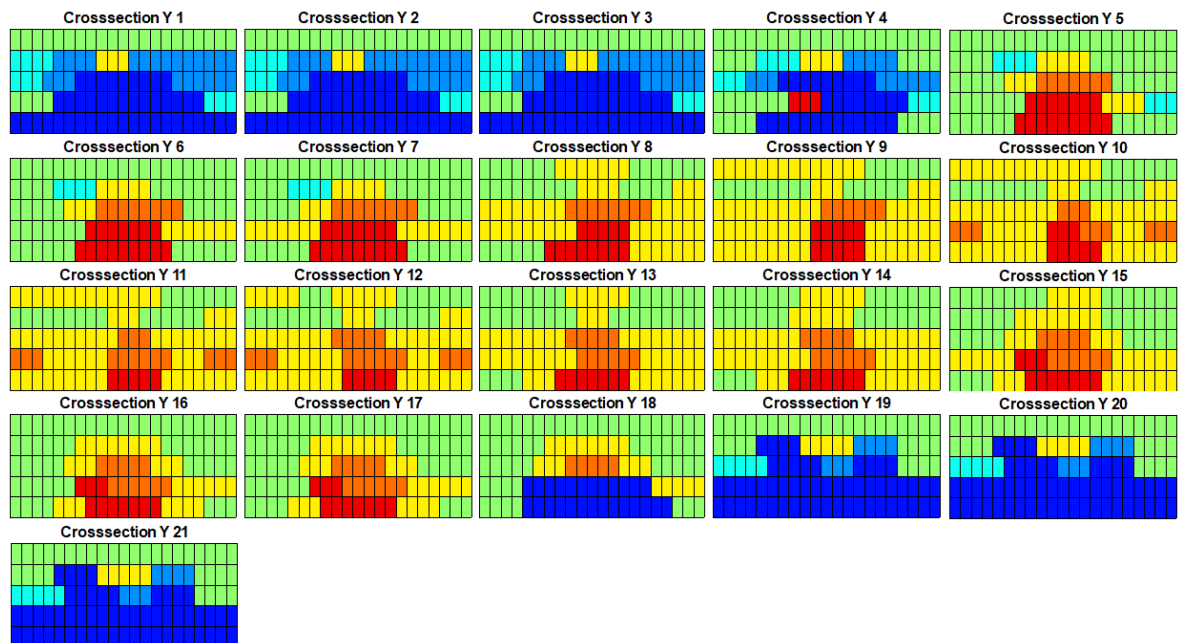


Figure F.4: Cross-sections in the y-direction after stage 2

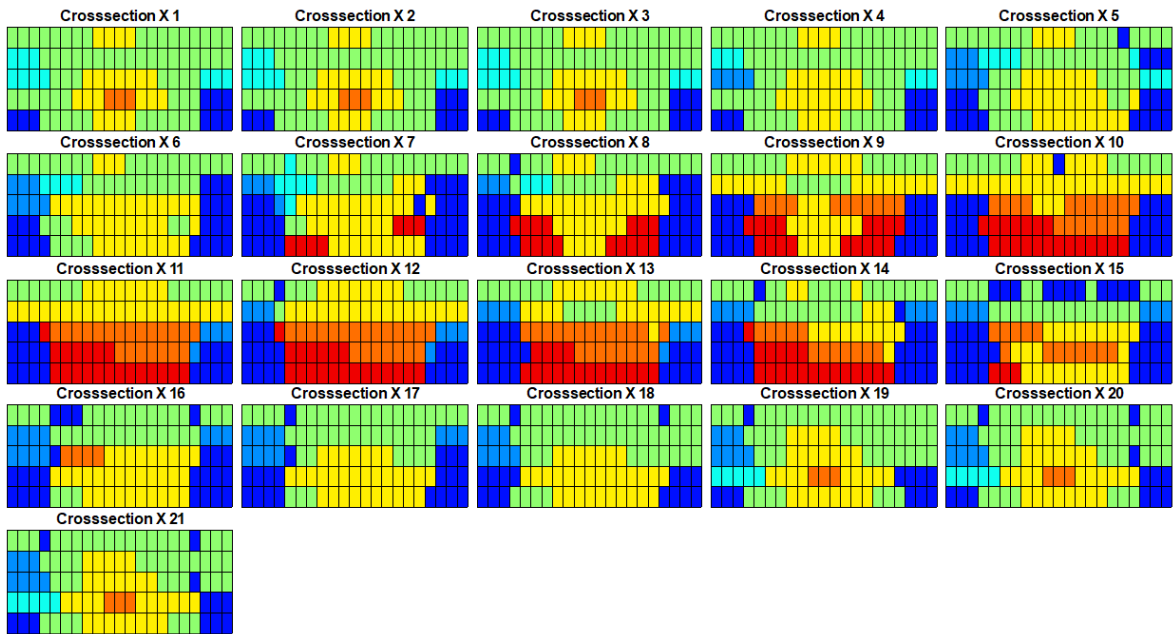


Figure F.5: Cross-sections in the x-direction after stage 3

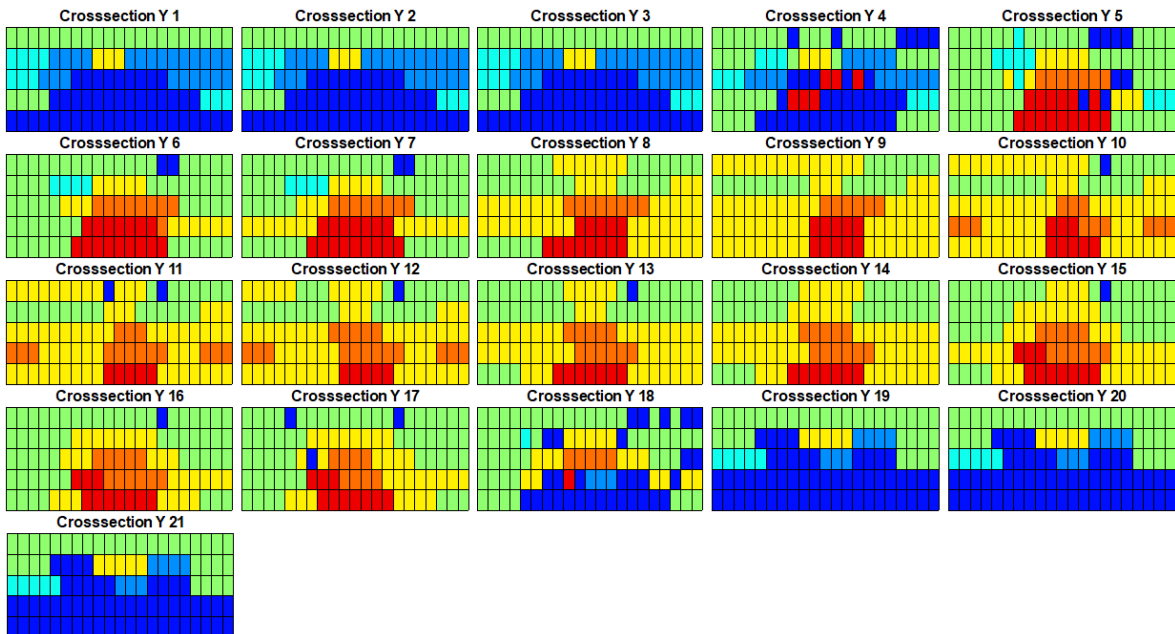


Figure F.6: Cross-sections in the y-direction after stage 3