User Guide for the Stacking Sequence Retriever (SSR)

# Author’s Note

This document serves as an introduction to the Stacking Sequence Retriever (SSR) MATLAB tool which I developed for retrieving feasible stacking sequence (i.e. thicknesses and fibre angles) from lamination parameters. The SSR is based on a lamination parameter (LP) matching strategy employing a genetic algorithm which uses thicknesses (can be fixed as well) and fibre angles as design variables. This guide focuses on the use of this code rather than lengthy explanations of the code itself. The rest of this guide is divided into 2 sections. The introduction serves as a brief overview of the literature on stacking sequence retrieval strategies. A simple example is used to introduce the code in Section 3.

Requirement

- MATLAB 2014

- MATLAB Optimisation Toolbox

Current version limitations

- Only deals with symmetric laminates

- No option to enforced balanced laminates

- Ply thickness remains constant

Please try the SSR code out and feedback any issues to me.

# Introduction (can be skipped)

The significant weight saving potential achievable with tailored composite structures is now well-recognised amidst the scientific community. The incentive to manufacture strong yet lightweight structures is also resulting in the increasing use of composite materials in many engineering applications. Moving from metals to composite structures has, however, brought forward a considerable new set of challenges including new failure mechanisms, added complexity and increased number of design variables. These have led to the development of a broad range of composite design guidelines and optimisation methods [1, 2].

Over the last decade, it has become evident that optimising composite structures raises several difficulties. Amongst these, the non-convex fibre angle design space, mixed-integer design variables, and manufacturability constraints have been recognised as major obstacles [3, 4]. Single and multi-step optimisations have been proposed to solve the complex problems of composite structure optimisation. Single optimisation methods such as guide-based designs [5] and stacking sequence tables [4] are strictly limited to the generation of designs satisfying manufacturability constraints. Although successful on small scale problems, these approaches result in highly constrained optimisation with prohibitively high computational cost. On the other hand, multi-step algorithms divide the optimisation of composite structures into faster and simpler-to solve sub-optimisation problems [6, 7]. Commonly used are bi-step algorithms which separate the continuous and discrete optimisations [8, 9]. Employing intermediate design variables (e.g. lamination parameters), the initial problem is reformulated into a continuous convex design space [10]. Following the continuous optimisation, a highly constrained discrete optimisation is usually employed to retrieve ready-to-manufacture stacking sequences closely matching the continuous optimisation output results [8]. The SSR tool has been developed to retrieve feasible and manufacture-ready stacking sequences closely matching given lamination parameters.

# SSR Coding

The SSR code searches the laminate design space in order to match lamination parameters (LPs) provided by the user. These LPs are stored into the [LP2Match] matrix being composed of **LP2Match***p* vectors where *p* denote the laminate number. Two different stacking sequence retrieval strategies are available in SSR:

1. The first approach “LPMatch” matches the [LP2Match] by searching the fibre angle design space while keeping the number of plies constant (fixed by user).
2. The second strategy is a stacking sequence table “**SST**”approach. Within this framework, both the fibre angles and the number of plies are used as design variables.

**Notation Used**

The lamination parameters of a N-ply laminate built out of discrete plies of constant thicknesses and ply angles *θ* are defined as follows:

|  |  |  |
| --- | --- | --- |
|  |  | (.a)    (1.b)  (1.c) |

where, the upper scripts A,B and D refer to the LPs corresponding [A],[B],[D] stiffness matrices and:

|  |  |  |
| --- | --- | --- |
|  |  | () |

## First Approach: LPMatch

A simple 3-laminate fixed thickness example, shown in Figure 2, is now introduced for sake of simplicity. This example deals with 3 symmetric laminates with respective number of plies of 20, 18 and 12. In order to setup the SSR code the user must provide the following information:

* For each laminate,
  + Alaminate indexfor identification: *LamIndex*
  + Thelaminatenumber of plies:*Nplies*
  + The laminates lamination parameters: ***LP2Matchp***
* A lamination parameters index: *IndexLP* denoting which LPs are to be matched.

In our example, *IndexLP* =[1 3] meaning that only the LPs 1 and 3 shown in **bold purple** in Figure 1 will be accounted for during stacking sequence retrieval. Other LPs will be ignored.

*LamIndex* = 1

*Nplies* = 20

***LP2Match1***=

|  |  |
| --- | --- |
|  | **0.1821** |
|  | -0.3643 |
|  | **0.0667** |
|  | -0.1155 |
|  | 0 |
|  | 0 |
|  | 0 |
|  | 0 |
|  | 0.1699 |
|  | -0.2584 |
|  | 0.2261 |
|  | -0.3177 |

*LamIndex* = 2

*Nplies* = 18

***LP2Match2***=

|  |
| --- |
| **0.2102** |
| -0.2871 |
| **0.1539** |
| -0.1332 |
| 0 |
| 0 |
| 0 |
| 0 |
| 0.1227 |
| -0.1579 |
| 0.3518 |
| -0.2444 |

*LamIndex* = 3

*Nplies* = 12

***LP2Match3*** =

|  |
| --- |
| **0.3000** |
| -0.1732 |
| **0.1000** |
| -0.1732 |
| 0 |
| 0 |
| 0 |
| 0 |
| -0.0741 |
| 0.1131 |
| 0.4120 |
| -0.3811 |

Figure - 3-Laminate example as defined by user

The SSR code interprets the user input in the following fashion. According to the guide based strategy adopted herein, the thickest laminate is identified as the **guide** laminate. Other laminates will be obtained by dropping plies from the guide laminate. Since we are using symmetric laminates, 10 fibre angles are sufficient to fully define the guide laminate stacking sequence *Nplies* = 20. Furthermore, only 4 additional ply drop locations are necessary to describe the 12-ply laminates once the guide laminate is known. For instance, Figure 2 illustrates one possible solution of the 3-laminate example. The 14 design variables shown below the stacking sequence represent the coding of this solution. This coding of information into a vector form is also referred to as a genotype.

Plane of Symmetry

20-Ply Symmetric Laminate (Guide)

***θ6***

***θ7***

***θ8***

***θ9***

***θ10* = *θN/2***

***θ5***

***θ4***

***θ3***

***θ2***

***θ1***

*θ6*

*θ7*

*θ8*

*θ9*

*θ10* = *θN/2*

*θ5*

*θ4*

*θ3*

*θ1*

*θ3*

*θ7*

*θ8*

*θ9*

*θ10* = *θN/2*

*θ1*

18-Ply Symmetric Laminate (1st Drop)

12-Ply Symmetric Laminate (2nd Drop)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *θ1* | *θ2* | *…* | *θ10* | *2* | *4* | *5* | *6* |

Genotype:

Figure - 3-Laminate Example

The **generic genotype** representation of solutions **for the 3-laminate problem** is:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *θ1* | *θ2* | *…* | *θ10* |  |  |  |  |

where, *θ1* to *θ10*  describe the 10 angles of the guide laminate while to denote the ply-drop locations. This generic genotype is used to code individuals (i.e. potential solutions) which will be decoded and evaluated by the genetic algorithm. Once individuals are fed to the fitness evaluation function of the GA, they are decoded and their corresponding LPs are calculated and used to evaluate their fitness’s. The GA will find a solution minimising:

where, is a vector containing the lamination parameters provided by the user and is a vector of similar size containing the lamination parameters of the *p* laminates obtained while optimising the matching stacking sequence. Additionally, is the vector of fibre angles used to describe the guide laminates and is the vector used to store the ply drop locations.

For the **LPMatch** strategy, the general representation of a stacking sequence retrieval problem with a N-ply guide laminate and *Ndrop* ply-drops can be represented as follows:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *θ1* | *θ2* | *…* | *θN/2* | *1* | *2* | *…* | *Ndrop/2* |

Ply-drops in descending ordered from the guide

Fibre angles of the thickest Laminate (the guide). Ordered from the top layer to the plane of symmetry

**3-Laminate Numerical Implementation**

An numerical input code of the above example is available in the Learn2use\_Example0.m file and also shown in Figure 3. The laminations parameters for the 3 laminates are given in a (12X3) matrix format in which each column contains the **LP2Match***p.*The parameter Objectives.IndexLP indicates that only lamination parameters 1 and 3 of [LP2Match] (represented in Bold) are taken into account during the optimisation. In other words, all other LPs will have no influence on the optimisation run. The Objectives.Table contains the bear minimum of information the user must provide. Other options, including the enforcement of manufacturing constraints and options for the GA, are also available. Before going into further details about these options, let us run Learn2use\_Example0.m.

After full completion of the Learn2use\_Example0.m run you should get an output similar to Figure 4 (note that the final solution will likely be different). During the run, the GA displays the mean and best fitness value of the population. Once terminated, the self-explanatory results of the search are all saved into the structure output\_Match. The stacking sequence and lamination parameters obtained for each laminate are detailed in the table field of output\_Match and is accessible by typing output\_Match.Table in the command window.

% A 3-Laminate Example - User Input

Lp2Match = [

% LP2Match1 LP2Match2 LP2Match3

**0.1821 0.2102 0.3000** % V1A

-0.3643 -0.2871 -0.1732 % V2A

**0.0667 0.1539 0.1000** % V3A

-0.1155 -0.1332 -0.1732 % V4A

0.0000 0.0000 0.0000 % V1B

0.0000 0.0000 0.0000 % V2B

0.0000 0.0000 0.0000 % V3B

0.0000 0.0000 0.0000 % V4B

0.1699 0.1227 -0.0741 % V1D

-0.2584 -0.1579 0.1131 % V2D

0.2261 0.3518 0.4120 % V3D

-0.3177 -0.2444 -0.3811]; % V4D

Objectives.IndexLP = [1 3];

Objectives.Table = [{'Laminate Index'} {'Nplies'} { ' LP2Match ' } ;

{1} {20} {Lp2Match(:,1)} ;

{2} {18} {Lp2Match(:,2)} ;

{3} {12} {Lp2Match(:,3)} ; ];

Figure - Numerical input for the 3-laminate example in SSR

output\_Match =

ply\_angle: [10 -90 65 50 60 -5 40 -15 40 10] % Guide fibre angles [θ1 θ2 … θ10]

DropsIndexes: {[4] [7 10 2]} % PlyDrops [Γ1 Γ2 Γ3 Γ4]

feasible: 1 % Manufacture Feasiblility Boolean

Table: {4x6 cell} % Summarise the Results

xOpt: [20 0 31 28 30 17 26 15 26 20 4 7 10 2] % Coded Optimal Solution

fval: 0.020909 % Final Fitness value

output\_Match.Table=

'Lam #' 'Nplies' 'Ply Angles' 'Lam. Param.' 'Error %' 'Error Norm'

[ 1] [ 20] [1x20 double] [12x1 double] [ 31.468] [ 0.019727]

[ 2] [ 18] [1x18 double] [12x1 double] [ 4.6984] [ 0.0060603]

[ 3] [ 12] [1x12 double] [12x1 double] [ 2.4554] [ 0.003783]



Figure - Output figure to Learn2use\_Example0.m

As previously mentioned, in addition to the objective, the user can also specify manufacturing constraints and search options.

Constraints

In Learn2use\_Example0.m, the constraints are by default set to:

% Constraints.Vector = [Damtol Rule10percent Disorientation Contiguity DiscreteAngle InernalContinuity Covering];

Constraints.Vector = [false false false false true false false];

Constraints.DeltaAngle = 5;

Constraints.ply\_t = 0.000127; % ply thickness

Constraints.Vector contains Boolean values which determine which manufacturing constraints will be active during the stacking sequence search. The different manufacturing constraints generally encountered in aerospace [4] are summarised below:

1. Symmetry: Not an option. Symmetry is always enforced in the current version of SSR.

2. Balance: Not an option. The current version does not enforce balanced laminate.

3. Damtol: Damage Tolerance, ±45° degree plies are used on the upper/lower laminate surface.

4. Rule10percent: A minimum of 10% of plies in each of the 0, ±45 and 90° is enforced.

5. Disorientation: The change of angles between two consecutive plies should not exceed 45°.

6. Contiguity: The change of angles between two consecutive plies should not be below 5°.

7. DiscreteAngle: Discrete fibre angles are used (values are set by DeltaAngle).

8. InernalContinuity: 1 ply must be kept spanning the entire structure every 3 plies.

9. Covering: Covering plies on the lower /upper surfaces of the laminate are not dropped.

The Genetic Algorithm Search Options

GAoptions.Npop = 100; % Population size

GAoptions.Ngen = 200; % Number of generations

GAoptions.NgenMin = 200; % Minimum number of generation calculated

GAoptions.Elitism = 0.1; % Percentage of elite passing to the next Gen.

GAoptions.Plot = true; % Plot Boolean

GAoptions.Method = 'LPMatch'; % Search method used

**Remarks**

The above information should be sufficient to get you started with the first “LPMatch” strategy. Note that if the final solution found does not satisfy all the manufacturing constraints,

output\_Match.feasible will be equal to 0.

The user should also be aware that GA are meta-heuristic evolutionary algorithms which are prone to the curse of dimensionality. That is, computational time will drastically increase with the number of design variables. The chance of finding the true optimal solution will also reduce as the number of variables increases. It is strongly recommended to repeat Searches before making conclusions.

## Second Approach: SST - Stacking Sequence Table

With this approach both the LPs and Number of plies are used as design variables. While this strategy can be useful, I would generally recommend to use the first approach if possible. That is, the addition of thickness as design variable increases the problem complexity and therefore the chances of finding good solutions.

For sake of clarity, the 3-laminate example presented above is re-used in this section. The file Learn2use\_Example1.m contains the ready-to-run example. From the user point of view, the objective definition remains identical to the one illustrate in Figure 3. Most changes happen in the code itself and are not noticeable by the user. The only necessary changes are the following:

alpha: Scalar value between [0,1]. Because thickness is now a design variable, the total “weight” of the structure also come into play during the fitness calculation in order to avoid excessive increase in the number of plies. The value of alpha relates to the prioritisation of the search. alpha=0 referes to a pure weight LP matching while alpha=1 will lead to a matching strategy in which thickness increase will be strongly penalised.

ORDERED: Boolean value. If set to TRUE, it ensures that the final solution will have the same trend in thickness than the initially supplied Nplies values. This option is recommended for structure in which the thickness have to follow a given trend (e.g. monolithically decreasing).

In order to restrict the increased design space due to the addition of thickness as a variable, an initial number of plies is still required as a starting point. The SST strategy will look for solutions close to the initial number of plies supplied by the user up to 160% times the original Nply value.

**3-Laminate Numerical Implementation**

Figure 4 - Output figure to Learn2use\_Example0.m

output\_Match =

Nplies: [42 40 18] % Final number of plies

ply\_angle: [-10 5 10 40 -15 40 -90 -70 -35 80 40 50 0 30 85 60 20 0 60 -10 -45] % Guide fibre angles [θ1 θ2 … θ10]

DropsIndexes: [12 13 10 15 16 11 17 14 5 9 2 8] % PlyDrops [Γ1 Γ2 Γ3 Γ4]

feasible: 1 % Manufacture Feasiblility Boolean

Table: {4x7 cell} % Summarise the Results

xOpt: [39 41 17 16 19 20 26 15 26 0 4 11 34 26 28 18 24 35 30 22 18 30 16 9 16 16 30 10 29 12 13 10 15 16 10 10 16 11 17 14 17 13 5 9 2 12 13 5 8 11 9 9 12 10 8] % Coded Optimal Solution

fval: 0.025275

output\_Match.Table =

'Lam #' 'Nplies Ori' 'Nplies SST' 'Ply Angles' 'Lam. Param.' 'Error %' 'Error Norm'

[ 1] [ 20] [ 42] [1x42 double] [12x1 double] [ 172.85] [ 0.018177]

[ 2] [ 18] [ 40] [1x40 double] [12x1 double] [ 165.93] [ 0.019217]

[ 3] [ 12] [ 18] [1x18 double] [12x1 double] [ 160.17] [ 0.0042777]

The difference between the original number of plies ('Nplies Ori') and the final number of plies ('Nplies SST') is shown in output\_Match.Table. Depending on the value of alpha used, the increase in thickness can be significant (i.e. if alpha=0). Note that, as a result, the coded genotype (xOpt) will include more variables which, in turn, will hinder the performance of the algorithm to find a good solution.

**References**

[1] Ghiasi H, Fayazbakhsh K, Pasini D, Lessard L. Optimum stacking sequence design of composite materials Part II: Variable stiffness design. Composite Structures. 2010;93:1-13.

[2] Ghiasi H, Pasini D, Lessard L. Optimum stacking sequence design of composite materials Part I: Constant stiffness design. Composite Structures. 2009;90.

[3] Vincenti A, Vannucci P, Ahmadian MR. Optimization of laminated composites by using genetic algorithm and the polar description of plane anisotropy. Mechanics of Advanced Materials and Structures. 2013;20:242-55.

[4] Irisarri F-X, Lasseigne A, Leroy F-H, Le Riche R. Optimal design of laminated composite structures with ply drops using stacking sequence tables. Composite Structures. 2014;107:559-69.

[5] Adams DB, Watson LT, Gürdal Z, Anderson-Cook CM. Genetic algorithm optimization and blending of composite laminates by locally reducing laminate thickness. Advances in Engineering Software. 2004;35:35-43.

[6] Peeters DM, Hesse S, Abdalla MM. Stacking sequence optimisation of variable stiffness laminates with manufacturing constraints. Composite Structures. 2015;125:596-604.

[7] Dillinger J, Klimmek T, Abdalla M, Gürdal Z. Stiffness optimization of composite wings with aeroelastic constraints. Journal of aircraft. 2013;50:1159-68.

[8] Liu D, Toroporov VV, Querin OM, Barton DC. Bilevel optimization of blended composite wing panels. Journal of aircraft. 2011;48:107-18.

[9] IJsselmuiden ST, Abdalla MM, Seresta O, Gürdal Z. Multi-step blended stacking sequence design of panel assemblies with buckling constraints. Composites Part B: Engineering. 2009;40:329-36.

[10] Liu S, Hou Y, Sun X, Zhang Y. A two-step optimization scheme for maximum stiffness design of laminated plates based on lamination parameters. Composite Structures. 2012;94:3529-37.