

A block-queue based optimization method for composite structures blended with multiple templates

Jiani Zeng¹, Wenbo Wu¹, Zhengdong Huang¹ and Kuan Fan¹

¹*School of Mechanical Science and Engineering, Huazhong University of Science and Technology, P. R. China*

Abstract. Maintaining the continuity (blending) of stacking sequences is very vital when the stacking sequences of a multi-region panel is being optimized. The traditional attempt simplifies the difficulty of the blending problem by designing the whole panel based on one stacking template. However, this simplification adds an unnecessary constraint that regions with the same thickness should have the same stacking sequence. In order to overcome the drawback, this paper introduces a region block queue based method to obtain the optimal manufacturable stacking sequences of composite laminates blended from multiple stacking templates. The method discussed here groups regions into blocks, forming a block queue, and designs the entire laminate panel with multiple stacking templates, which are respectively applied to different blocks. The continuity between two adjacent blocks is guaranteed by sequentially rebuilding the template of a block from the sequences of adjacent regions within its previous block in the queue. An 18-region benchmark problem is chosen to prove the effectiveness of the proposed method, and the solution obtained is 0.27kg lighter than the traditional template-based method. The reduction of mass reveals that the use of multiple templates does contribute to searching for lighter solution.

1 Introduction

In spite of high manufacturing cost, fiber-reinforced composites are widely used in aerospace due to their superior mechanical performance such as high strength and stiffness [1-3]. The optimization of composite structures attracts many researchers' attention [4]. The design variables include number of plies, ply orientations, stacking sequences in the composite laminates, while the goal is usually minimizing the weight without buckling failure [5] or maximizing the critical buckling load with the number of plies fixed [6,7]. For ease of manufacture, value of the fiber orientation in each ply is restricted to some discrete sets of angles. The discrete nature of laminate ply thicknesses and fiber orientation angles makes the genetic algorithm (GA) to be one of the most popular methods for optimizing the stacking sequence of composite laminates [8,9].

In engineering practices, the different regions in a multi-region composite structure are often subjected to different local loads and design constraints [5,10]. Optimizing each region locally will result in the incompatibility in stacking sequences of adjacent regions. In order to ensure the compatibility of stacking sequences, the blending constraint is introduced. In a blending design, the plies of the thinner one should be a subset of the plies of the thicker one between adjacent regions [11]. Therefore, all regions in the multi-region panel need to be optimized together, instead of being separately optimized [12-14]. The goal of the blending optimization problem is

to minimize the weight of the structure with no buckling failure occurring in any regions and all regions satisfying the blending constraints [15].

The two most commonly used kinds of methods to blend the multi-region panel are the template-based method and the bi-level method. In the template-based method, the stacking sequence of each region in the panel is determined by the same stacking sequence template [16-18]. In the template, each thickness corresponds to a specific stacking sequence, and each thinner sequence is involved in all the other thicker sequences. The particular structure of template makes the blending constraints automatically satisfied [19]. In the bi-level method, the ply thicknesses and the stacking sequences of regions are obtained respectively in two levels [20,21]. At the top level, the number of plies are optimized to minimize the whole weight with buckling constraints applied. And at the bottom level, stacking sequences of all regions are optimized to minimize the difference between the values of intermediate variable values calculated from the current sequences and those calculated from the top bottom [22,23]. Both blending methods have their own drawbacks. An unnecessary constraint that regions with the same thickness have the same thickness is introduced in the template-based method, while the operator which rounds the ply thicknesses calculated at the top level to integers introduces a design gap in the bi-level method [24].

The present work intends to propose a new method to blend the multi-region panel with more than one

templates. As multiple blending templates are used, the proposed method expands the design space compared to template-based method. Besides, no design gap is introduced. In the proposed method, regions in the design panel are grouped into blocks, forming a block queue. The entire laminate panel is designed with multiple stacking templates, which are respectively applied to regions in different blocks. The continuity between two adjacent blocks is guaranteed by sequentially rebuilding the template of a block from the sequences of adjacent regions within its previous block in the queue. Since this extension does not involve any complicated process for adapting the optimization algorithm, the proposed approach shows its good applicability on the design of common composite structures.

2 Block-queue based laminate blending

To make the panel easily manufacturable, it is desirable to maintain the continuity between adjacent regions in the stacking sequence optimization. In order to overcome the limitation of the template based methods, in which the regions of the same thickness must have the same stacking sequence, this paper presents a block queue based method to reduce the loss of freedom. Considering that the stacking sequence table (SST) contains more information on the ply-drop order in contrast with the guide, SST is chosen as the template type. In the method, a pre-processing process is adopted to divide the entire design panel into several blocks according to certain rules, resulting in a numbered block queue; the process will be discussed in detail in Section 2.1. Then, a genetic algorithm is implemented to guide the exploration of the search space toward the best performing individuals, where performance is measured by the total weight of panel in the individual design.

2.1 Block division rules

In order to overcome the limitation of the traditional single template methods in the design capability and flexibility, this paper attempts to improve the design freedom by dividing the panel under design into several blocks and designing each block with its own SST template. To ensure the compatibility between the multiple templates in the design process, this paper requires that the following division rules should be satisfied when a panel is divided into multiple blocks:

1. The number of blocks adjacent to each block cannot exceed two.
2. The number of blocks with only one adjacent block must be two.

In Figure 1, the 13-region plate is divided into three blocks. Block 1 consists of regions 1, 2, 3 and 4, Block 2 consists of Regions 5, 6, 7, 8, 9, 10 and 11, and Block 3 consists of Regions 12 and 13. There are only one block adjacent to Block 1, two blocks adjacent to Block 2, and only one block adjacent to Block 3, which is consistent with rule 1. Both Block 1 and Block 2 have a unique

adjacent block, which is consistent with rule 2. Therefore, the block division scheme in Figure 1 is feasible.

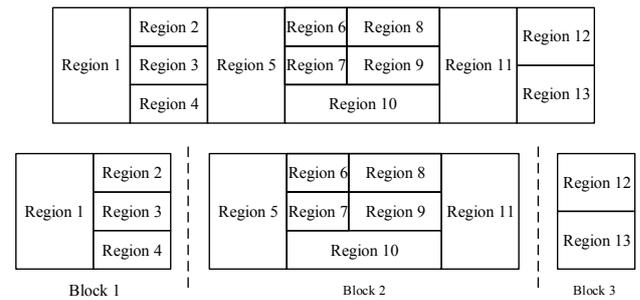


Figure 1. A feasible partitioning scheme.

Moreover, it is worth mentioning that the appropriate partitioning scheme can greatly reduce unnecessary dependencies among regions in the design process of stacking sequences. In the design procedure of Regions 12 and 13 in Block 3, only the stacking sequence of adjacent Region 11 in Block 2 needs to be considered, while sequences of other regions in Block 2 and regions in Block 1 do not have any effect on the design process.

2.2 Block division rules

According to a feasible scheme satisfying the division rules mentioned in Section 2.1, the whole panel will be divided into several blocks forming a block queue, in which the two blocks with only one adjacent block as the queue head and tail respectively and each of other blocks is spatially adjacent to the two blocks located before and after its queue position. Taking the partitioning scheme shown in Figure 1 as an example, the detailed design process of stacking sequences is as followed. Firstly, design Block 1 with one individual SST, obtaining the stacking sequences of all regions in the block. Since the stacking sequences of regions in the first block are taken from the same SST, regions in Block 1 satisfies the blending constraints. Take the first block as the previous block and remove it from the block queue. Secondly, choose the smallest numbered block from the block queue as the current block and remove it from the block queue, too. Thirdly, rebuild n_{reb} new SSTs containing all the stacking sequences of the adjacent regions in the previous block as current SSTs. The detailed rebuilding method will be discussed later in Section 2.3. Fourthly, from the n_{reb} current SSTs, the one that makes the current block have the lightest weight is selected, and the corresponding sequences are selected as sequences of regions in the current block. Assuming that the previous block is Block 1 and the current block is Block 2, since the stacking sequences of adjacent regions in Block 1 and all regions in Block 2 are taken from the same SST, regions in Block 1 and Block 2 satisfies the blending constraint. Repeat the above procedure until all blocks have been designed. It can be seen that the multi-SST blending of the entire structure has been achieved by processing blocks in the order of the block queue.

Figure 2 shows a stacking sequence design process of a 4-region panel with the block queue based method

presented in this paper. Since SST2 includes Seq 1, Seq 2 and Seq 3, Block 1 (Region 1) and Block 2 (Region 2 and Region 3) satisfy the blending constraint. Similarly, SST3 includes Seq 2, Seq 3 and Seq 4. Therefore, all the laminates in the whole panel are blended. It can be seen from the cross-section 1-2-4 that although Regions 1 and 4 have the same thickness, they have different stacking sequences.

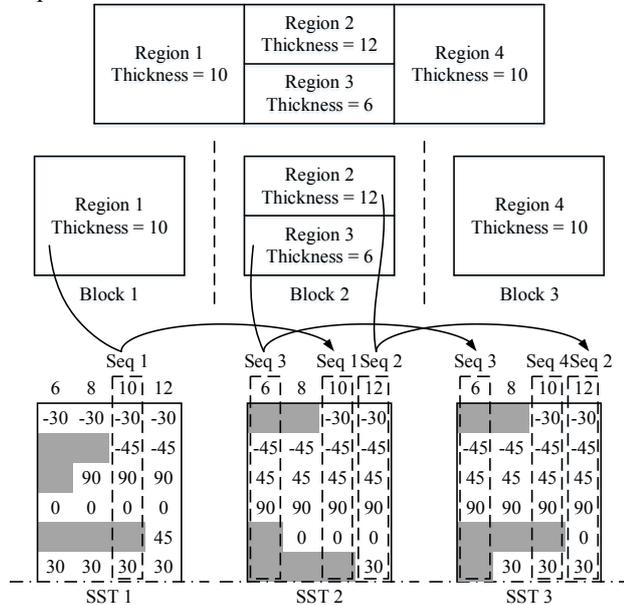


Figure 2. Stacking sequences and their blending principle in the scheme of multiple SSTs.

2.3 Rebuilding process of a new SST

The initialization process of SST is described in detail by Irisarri et al. [16]. Here we focus on the rebuilding of a new SST containing several given sequences, which serve as the input data of the rebuilding process. Since the rebuilding process guarantees the compatibility of multiple SST templates, it has played a critical role in the block queue based blending design.

The input stacking sequences are arranged from small to large in thickness recorded as matrix $SEQ = [seq_1 \ seq_2 \ seq_3 \ \dots \ seq_k]$, (the sequence of region i is recorded as seq_i), and their corresponding thicknesses are recorded as vector $N = [n_1 \ n_2 \ n_3 \ \dots \ n_k]$, (the thickness of region i is recorded as n_i). The rebuilding process of a new SST is composed of multiple rebuilding procedures, each $[seq_i \ seq_{i+1}]$ of which correspond to one of the ply number intervals $[n_i, n_{i+1}]$. The corresponding algorithm is summarized in Algorithm 1.

Algorithm 1: The rebuilding procedure for one ply number interval $[n_i, n_{i+1}]$

- (1) Preprocessing. Once the sequence seq_i (or seq_{i+1}) violate the balance requirement, one ply is added (or remove) to the sequence to form a balanced sequence.
- (2) Enumeration. Plies are added from seq_i one by one.

For each added ply, feasible ply angles and corresponding positions are enumerated under the constraints of the symmetry and balance guidelines.

- (3) Random choice. Roulette wheel selection is adopted to determine the ply angle and position of the added ply.
- (4) Repeat. Repeat steps (2) and (3) until n_{i+1} is reached.

Figure 3 shows one possible result of the rebuilding process of a new SST from a 10-ply laminate and a 16-ply laminate. In the preprocessing process, a new sequence respectively with 8 plies can be derived from the 10-ply laminate according to the balance requirement. Then, a new SST is rebuilt by running the rebuilding procedure for ply number intervals $[10, 16]$.

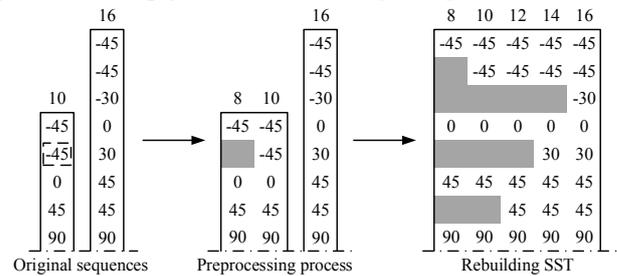


Figure 3. A new SST rebuilding process from two sequences.

3 Genetic algorithm

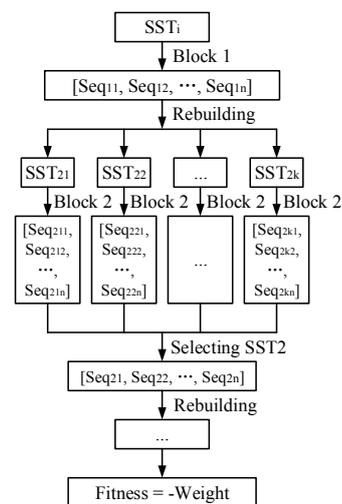


Figure 4. The detailed diagram for evaluating the fitness function.

A block queue based genetic algorithm is used to search for the stacking sequence design scheme which owns a lighter weight while satisfying loading constraints [25,26]. Figure 4 shows the detailed calculation process of the individual fitness value. In the process, each one SST i corresponds to an individual i of the population. First of all, stacking sequence for each region in the Block 1 is

	90 60 0 0]s	
17	[60 -60 60 -60 60 -60 -60 60 0]s	18
18	[75 60 -60 60 -60 60 -60 -75 -60 60 0]s	22

In order to compare the results with single template based method more intuitively, both symmetry and balance guidelines are enforced. The block division scheme in Figure 6 is used, and the value of n_{reb} is set to 30. The lightest solution (Solution 0) found in the present work weighs 28.58kg, which contains 0.27kg less weight compared to the SST method. The detailed stacking sequences of Solution 0 are given in Table 2 respectively. The result comparisons are listed in Table 3. The execution time for a single run in a workstation with two Intel(R) Xeon(R) X5680 (3.33 GHz) processors is around 53min in this case.

Table 3. Result comparison.

Region	Solution 0		Irisarri et al. [16]	
	m(kg)	λ_{cr}	m(kg)	λ_{cr}
1	2.85	1.055	2.85	1.072
2	2.51	1.172	2.51	1.159
3	1.02	1.218	1.02	1.205
4	0.84	1.038	0.84	0.965
5	0.74	1.202	0.84	1.593
6	1.02	1.095	1.02	1.083
7	0.84	1.005	0.84	0.934
8	1.12	1.011	1.21	1.182
9	3.35	1.117	3.18	0.983
10	3.01	1.022	3.18	1.256
11	2.51	1.016	2.51	0.998
12	2.43	1.065	2.51	1.154
13	1.02	1.144	1.02	1.134
14	0.84	1.099	0.84	1.023
15	1.12	1.002	1.21	1.146
16	1.49	1.201	1.40	0.968
17	0.84	1.017	0.84	0.948
18	1.02	1.069	1.02	0.982
Total	28.58	1.002	28.85	0.934

5 Results and discussion

In this paper, a block queue based optimization method has been proposed for the stacking sequence blending design of composite structures with multiple templates. A block consists of one or more regions that are blended with one template. Although there are certain dependence relations between the templates for two adjacent blocks, each block has its own template here. It is just this feature that makes the proposed method with more design capability and flexibility than traditional single template based methods. To establish the block queue based blending method, this paper has developed a block template rebuilding approach from the stacking sequences of the adjacent regions in the previous block. The approach constructs a SST under the constraints that several given sequences should be included. The proposed method is applied to the 18-region horseshoe benchmark problem and the results show its effectiveness in obtaining a lighter design by using more than one blending template.

Acknowledgement

This project is supported by National Natural Science Foundation of China (Grant nos. #51475186 and #51775202).

References

1. H. Ghiasi, D. Pasini, L. Lessard. *Compos Struct* **90**, 1-11 (2009)
2. H. Ghiasi, K. Fayazbakhsh, D. Pasini, L. Lessard. *Compos Struct* **93**, 1-13 (2010)
3. JN. Reddy. *Mechanics of Laminated Composite Plates and Shells: Theory and Analysis* (CRC Press, 2004)
4. A. Catapano, M. Montemurro. *Compos Struct* **118**, 677-90 (2014)
5. G. Soremekun, Z. Gürdal, C. Kassapoglou. *Compos Struct* **56**, 53-62 (2002)
6. L. Lanzi, V. Giavotto. *Compos Struct* **73**, 208-20 (2006)
7. J. Dillinger, T. Klimmek, MM. Abdalla, Z. Gürdal. *J AIRCRAFT* **50**, 1159-68 (2013)
8. A. Todoroki, T. Ishikawa. *Compos Struct* **64**, 349-57 (2004)
9. M. Abouhamze, M. Shakeri. *Compos Struct* **81**, 253-63 (2007)
10. BP. Kristinsdottir, ZB. Zabinsky, ME. Tuttle, S. Neogi. *Compos Struct* **51**, 93-102 (2001)
11. DB. Adams, LT. Watson, Z. Gürdal. *Mech Adv Mater Struct* **10**, 183-203 (2003)
12. Z. Jing, Q. Sun, VV. Silberschmidt. *Compos Struct* **141**, 240-52 (2016)
13. Z. Jing, X. Fan, Q. Sun. *Compos Part B-eng* **69**, 181-90 (2015).
14. J. Zhao, S. Qin, VV. Silberschmidt. *Compos Struct* **154**, 106-28 (2016).
15. DB. Adams, LT. Watson, Z. Gürdal. *Adv Eng Software* **35**, 35-43 (2004).
16. FX. Irisarri, A. Lasseigen, FH. Leroy, R. Le Riche. *Compos Struct* **107**, 559-69 (2014).
17. J. Yang, B. Song, X. Zhong. *Compos Struct* **135**, 30-37 (2016).
18. H. Fan, H. Wang, X. Chen. *Compos Struct* **136**, 650-61 (2016).
19. O. Seresta, MM. Abdalla, Z. Gürdal. In: *Proceedings of 50th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference*, Palm Springs, California, USA; 4-7 (2009).
20. D. Liu, VV. Toropove, OM. Querin, DC. Barton. *J Aircraft* **48**, 107-18 (2011).
21. YM. Meddaikar, FX. Irisarri, MM. Abdalla. *Struct Multidiscip O* **55**, 535-46 (2016).
22. M. Montemurro, A. Vincenti, P. Vannucci. *J Optimiz Theory App* **155**, 24-53 (2012).
23. S. Zein, B. Colson, S. Grihon. *Struct Multi Optim* **45**, 669-80 (2012).

24. S. Zein, V. Madahavan, D. Dumas. *Compos Struct* **141**, 32-38 (2016).
25. S. Mashohor, JR. Evans, T. Arslan. In: *Proceedings of 47th IEEE Congress on Evolutionary Computation, Edinburgh, Scotland, UK*; 974-78 (2005).
26. A. Lipowski, D. Lipowska. *Physica A* **391**, 2193-96 (2012).