



Sources of variability linked to manufacturing parameters of structural features of aeronautical composite structures

Khouloud Nagaz

► To cite this version:

Khouloud Nagaz. Sources of variability linked to manufacturing parameters of structural features of aeronautical composite structures. Materials. Université Paul Sabatier - Toulouse III, 2022. English. NNT : 2022TOU30145 . tel-03865161

HAL Id: tel-03865161

<https://theses.hal.science/tel-03865161>

Submitted on 22 Nov 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



THÈSE

**En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE**
Délivré par l'Université Toulouse 3 - Paul Sabatier

**Présentée et soutenue par
Khouloud NAGAZ**

Le 01 juillet 2022

**Sources de variabilité en lien avec les paramètres du procédé au
sein de singularités géométriques de structures composites
aéronautiques**

Ecole doctorale : **MEGEP - Mécanique, Energétique, Génie civil, Procédés**

Spécialité : **Génie mécanique, mécanique des matériaux**

Unité de recherche :
ICA - Institut Clément Ader

Thèse dirigée par
Francis COLLOMBET, Bernard DOUCHIN et Laurent CROUZEIX

Jury

Mme. Marie-Ange BUENO, Rapporteure
M. Peter DAVIES, Rapporteur
Mme Fabienne TOUCHARD, Examinatrice
M. Francis COLLOMBET, Directeur de thèse
M. Bernard DOUCHIN, Co-directeur de thèse
M. Laurent CROUZEIX, Invité
M. Yves-Henri GRUNEVOLD, Invité
M. Guillaume FERRER, Invité

Remerciements

Je veux tout d'abord remercier énormément mon directeur de thèse, M. Francis COLLOMBET, pour toute son aide et ses encouragements tout au long de cette longue route qui m'a conduit jusqu'à cette soutenance. Qu'il soit remercié pour m'avoir permis de bénéficier de l'apport scientifique, technique et matériel du projet ANR/Astrid BLAST, qu'il a piloté pendant la durée de mon travail de thèse.

J'adresse aussi un grand merci à M. Laurent CROUZEIX qui a énormément contribué à l'avancée de ces travaux grâce à la vertu créative de ses idées et des discussions toujours d'un très grand intérêt.

J'adresse également mes remerciements à M. Bernard DOUCHIN pour ses commentaires toujours critiques et pertinents, et cela toujours dans la bonne humeur quelles que soient les circonstances.

Toute ma gratitude va vers les membres du jury qui m'ont fait l'honneur de lire ce manuscrit de façon critique et attentionnée.

Je suis très reconnaissante envers mes rapporteurs, Mme Marie-Ange BUENO et M. Peter DAVIES, pour le travail, l'attention et le temps qu'ils ont bien voulu apporter dans l'analyse de ce travail de thèse.

Ma reconnaissance va aussi vers Mme Fabienne Touchard pour avoir accepté d'être membre de ce jury et m'avoir fait l'honneur de le présider.

Je voudrais exprimer mes remerciements à messieurs Yves-Henri GRUNEVOLD, de CES Works et Guillaume FERRER, de Airbus Customer Services. Leur présence dans mon jury m'honore eu égard à leur expertise reconnue dans le domaine des structures composites.

J'exprime ma gratitude envers M. Yves-Henri GRUNEVOLD pour ses conseils au travers de longues discussions sur ce sujet complexe, en y apportant toute son expertise industrielle et sa vision scientifique très avancée.

J'exprime toute ma gratitude à Mme Nathalie GLEIZES-ROCHER pour m'avoir aidé dans l'analyse expérimentale tout au long de ce travail de thèse et m'avoir accompagné pour de nombreuses démarches administratives dans ce pays qui m'a accueilli. Je lui serai toujours reconnaissante pour sa bonne humeur et sa gentillesse.

Je n'oublie évidemment pas de remercier Mme Catherine STASIULIS du Département des relations internationales de l'UPS, messieurs Abdallah BOUZID de l'ICA et Michel CAVARERO du CRITT M&C, pour leur aide précieuse sur le plan administratif et technique. Je remercie aussi M. Yves DAVILA, mon prédécesseur sur ce sujet scientifique et ingénieur de recherche sur le projet BLAST, pour son accompagnement durant les premières années de thèse et pour son aide et ses remarques pertinentes.

Je remercie bien sûr l'ensemble des doctorants et des post-doctorants de l'ICA pour l'ambiance si agréable au laboratoire.

Je n'aurai jamais pu réaliser ce doctorat sans l'aide de ma famille, de mes proches qui ont toujours su me soutenir depuis si loin. Une pensée très forte va vers mes très chers parents, mon père Mourad et ma mère Hayet, ma sœur Siwar et mes frères Wessim et Adem. Enfin, je veux remercier mes amis, au labo et à l'extérieur, qui ont été ma famille à Toulouse et avec qui j'ai passé de très bons moments.

Résumé de thèse

Mots clé : géométrie complexe, oméga lisse, composite fibres longues, variabilité, incertitude, procédé de fabrication, modélisation multi-échelle, probabiliste, éléments finis.

La thèse porte sur l'étude des origines et des effets d'un certain nombre de variabilités, au sein d'une structure composite carbone/Epoxy présentant des singularités géométriques. Le 1er chapitre est une étude bibliographique où on montre que l'étude des variabilités sur des pièces composites reste un problème ouvert qui peut être abordé de plusieurs manières et à plusieurs échelles. Il est bien établi que les propriétés d'une structure composite dépendent à la fois du choix des matières premières comme de celui du procédé de fabrication. Ces propriétés sont influencées par plusieurs types de variabilités comme la variation de la position des arrêts de pli, le désalignement des fibres et la présence de porosités. Ces trois questions sont abordées dans le cadre de cette thèse. Pour illustrer la méthodologie proposée, nous considérons comme exemple de structure avec géométrie complexe, un prototype de lisse composite inspiré d'un fuselage aéronautique. Elle constitue dans ce travail le réceptacle d'un certain nombre de contraintes de fabrication comme la présence de rayons convexes et concaves, de parois verticales, de changements d'épaisseur et d'association de plusieurs matériaux. Le 2ème chapitre est dédié à la présentation du protocole expérimental utilisé et des observations de variabilité pour l'analyse des arrêts de plis, de la porosité et de désalignement des fibres. Les observations sont réalisées tout au long du processus de fabrication, et font appel à un large éventail de techniques de mesures comme les observations optiques, micrographiques, tomographiques, etc. Chaque observation est dédiée au suivi d'un ou plusieurs types de variabilité étudiée pour des phases différentes du processus de fabrication. Le 3ème chapitre comporte l'étude mathématique des résultats d'observation et la représentation des tendances statistiques. Les données acquises sont triées en regard de leur position vis-à-vis des singularités géométriques, et analysées afin d'en proposer des modélisations mathématiques pertinentes. L'originalité du travail porte principalement sur la volonté de ne pas se limiter à des analyses statistiques, en caractérisant des tendances d'évolutions spatiales réalistes pour chaque type de variabilité observée, en lien avec la présence ou non d'une singularité géométrique à proximité. Une fois les lois mathématiques représentatives identifiées, des analyses statistiques sont menées sur les paramètres qui pilotent ces lois, principalement sous la forme de moyennes et d'écart types. Le but du 4ème chapitre est de proposer une approche numérique pour évaluer l'influence de ces variabilités sur les propriétés de la structure. On montre en quoi il est possible de créer une famille de modèles numériques représentant des pièces virtuelles (non encore fabriquées) de façon réaliste. Le réalisme recherché cible autant l'amplitude des variabilités simulées que l'évolution spatiale de cette variabilité au sein de la pièce. Les simulations numériques doivent permettre de proposer des gammes de variation des propriétés et du comportement des structures en fonction de la variabilité étudiée. Le modèle fait appel à des éléments coques 2D, pour être en capacité de mener un grand nombre de simulations basées sur des tirages aléatoires des paramètres des lois mathématiques étudiées. Enfin une autre étude numérique est proposée. Elle concerne l'effet des variabilités au sein d'une peau de fuselage composée de plis UD à partir de résultats issus d'une thèse précédente (voir Annexe 4). En conclusion, ce travail a mis en place une approche permettant l'évaluation des effets d'incertitude dans des calculs mécaniques appliqués aux structures en matériaux composites à géométrie complexe. La question abordée est de s'assurer de la robustesse d'un résultat de manière plus physique qu'en ayant recours à des facteurs de sécurité. Les perspectives de ces travaux sont ensuite présentées.

Thesis summary

Keywords: complex geometry, omega stringer, long fibre composite, variability, uncertainty, manufacturing process, multiscale modelling, probabilistic, finite element.

The thesis focuses on the study of the origins and effects of a number of variabilities within a carbon/Epoxy composite structure with geometric singularities. The first chapter is a bibliographical study in which it is shown that the study of variabilities in composite parts remains an open problem that can be approached in several ways and at several scales. For the analysis and manufacture of composite structures, there are a large number of methods, multi-scale or not, which allow the influence of variabilities to be taken into account in the study of composite structures. It is well established that the properties of a composite structure depend on both the choice of raw materials and the manufacturing process. These properties are influenced by several types of variability such as variation in the position of ply stops, fibre misalignment and the presence of porosities. These three issues are addressed in this thesis. To illustrate the proposed methodology, we consider as an example of a structure with complex geometry, a prototype of a composite stringer inspired by an aeronautical fuselage. In this work, it constitutes the receptacle of a certain number of manufacturing constraints such as the presence of convex and concave radii, vertical walls, changes in thickness and the association of several materials. The second chapter is dedicated to the presentation of the experimental protocol used and the observations of variability for the analysis of ply stops, porosity and fibre misalignment. The observations are carried out throughout the manufacturing process, using a wide range of measurement techniques such as optical, micrographic, tomographic observations, etc. Each observation is dedicated to the monitoring of one or more types of variability studied for different phases of the manufacturing process. The third chapter includes the mathematical study of the observation results and the representation of statistical trends. The acquired data are sorted according to their position with respect to geometric singularities, and analysed in order to propose relevant mathematical models. The originality of the work lies mainly in the desire to do not limit itself to statistical analyses, by characterising realistic spatial evolution trends for each type of variability observed, in relation to the presence or absence of a geometric singularity nearby. Once the representative mathematical laws have been identified, statistical analyses are carried out on the parameters driving these laws, mainly in the form of means and standard deviations. The aim of the 4th chapter is to propose a numerical approach to evaluate the influence of these variabilities on the properties of the structure. It is shown how it is possible to create a family of numerical models representing virtual parts (not yet manufactured) in a realistic way. The realism sought targets both the amplitude of the simulated variabilities and the spatial evolution of this variability within the part. The numerical simulations must make it possible to propose ranges of variation of the properties and behaviour of the structures according to the variability studied. The model uses 2D shell elements to be able to carry out a large number of simulations based on random draws of the parameters of the mathematical laws studied. Finally, another numerical study is proposed. It concerns the effect of variabilities within a fuselage skin composed of UD plies, based on results from a previous thesis (see Appendix 4). In conclusion, this work has set up an approach allowing the evaluation of uncertainty effects in mechanical calculations applied to composite structures with complex geometry. The question addressed is to ensure the robustness of a result in a more physical way than by using safety factors. The perspectives of this work are then presented.

Table of contents

General introduction	1
-----------------------------------	---

Chapter 1: Bibliography	7
--------------------------------------	---

1. Introduction to the variability study in the literature from experimental measurements to numerical validation	7
2. Variability in composites	7
3. Variability classification	9
4. Variability sources during composite structures manufacturing process	13
5. Statistical and probabilistic studies after observations.....	18
5.1. Gaussian law	20
5.2. Log-normal law	21
6. Stochastic numerical modelling	21
6.1. Monte Carlo simulation	21
6.2. Chaos polynomial method	22
6.3. Perturbation method	23
6.4. Spectral form	23
6.5. Numerical analysis: processing the results	24
7. Complex geometry structure studies with different types of manufacture	27
8. Our objective and methodology	27
9. Studied complex geometry structure in this work	28
Conclusion	31
References	32

Chapter 2: Experimental setups and protocols for variabilities observation within complex geometries: from raw material to final structure	37
---	----

1. Introduction of the general flowchart for variability observation	37
2. Stringer manufacture steps before and after curing	38
2.1. Ply cut.....	38
2.2. Lay-up process presentation, related issues and ply category definition	41
2.3. Vacuum bag preparation.....	45
2.4. Polymerisation cycle	46

3. Experimental protocol for variability observation for each manufacturing step.....	47
3.1. Prepreg measurement after cutting.....	49
3.2. Orthogonal optical shooting for ply stops and fibre waviness measurements during lay-up	50
3.3. 3D laser scan for the geometry of the real external shape after curing.....	52
3.4. Ply stops location observation by cross-section microscopy.....	54
3.4.1. Specimen preparation and microscopy observation protocol.....	54
3.4.2. Image assembly and ply stop identification in the global reference.....	56
3.5. 3D X-ray tomography for porosity observation and global cross-section view.....	61
3.6. Curing phase with temperature monitoring and resin flow analysis.....	64
Conclusion.....	65

Chapter 3: Spatial evolution of variability within a complex geometry structure:

Mathematical laws and statistical trends	67
1. Introduction and problematic statement from experimental big data to exploitation of variability into numerical modelling.....	67
2. Ply stops variability mathematical representation.....	67
2.1. Gaps between real and theoretical prepreg width generated by cut process.....	68
2.2. Ply stops location obtained during lay-up process.....	71
2.3. Ply stops location after polymerization phase.....	71
2.4. Reconstruction of left side of cross section from right side variability measurement and cut-out width.....	74
3. Fibre misalignment mathematical representation.....	75
4. Porosity distribution mathematical representation.....	79
4.1. Analytical approach for composite mechanical properties without porosity consideration.....	85
4.2. Analytical approach for composite mechanical properties with porosity consideration	86
Conclusion.....	89

Chapter 4: Numerical analysis of variability and integration of different studied variability forms in the numerical modelling	91
---	-----------

1. Introduction to insertion of variabilities in numerical modelling.....	91
2. Model presentation (objectives and methodology).....	91
2.1. Mesh creation adapted to the chosen model and to the objectives.....	91
2.1.1. Material properties used by default in the numerical modelling without variability effects.....	93
2.1.2. Definition of boundary conditions according to the studied cases.....	95
3. Ply stops variability integration.....	95
3.1. Integration of variabilities due to cut-out and lay-up process.....	95
4. Fibre misalignment integration.....	98
5. Porosity integration.....	98
6. Results and discussion of taking into account each type of variability and the total variability types.....	101
6.1. Illustration of numerical results with ply stop variability.....	101
6.1.1. Twin models of the four manufactured stringers after cooling followed by tensile loading.....	101
6.1.2. Numerical Results of stringers with random drawings after cooling.....	102
6.1.3. Numerical Results after cooling and tensile loading.....	103
6.1.4. Repetitive analysis for Monte-Carlo draws after cooling for ply stop variability	105
6.2. Illustration of numerical results with fibre misalignment variability.....	107
6.2.1. Twin models taking into account fibre misalignment variability in the case of tensile loading after cooling.....	107
6.2.2. Numerical repetitive results after thermal and/or mechanical loading.....	109
6.3. Illustration of numerical results with porosity rates variability.....	110
7. Numerical model results with three types of variability taken into account.....	113
7.1. Numerical modelling results taking into account variabilities after cooling.....	113
7.2. Numerical modelling results of influence of variabilities after mechanical loading preceded by cooling.....	115
7.3. Numerical repetitive results after thermal and/or mechanical loading taking into account all the studied variabilities.....	118
Conclusion.....	120
 General conclusions and future work	123
Appendixes	127

List of figures

Fig. 1. 1. Strategy scheme for multi-scale modelling from microscale to macroscale [1].....	8
Fig. 1. 2. Tomographic images showing the porosities in a CFRP laminate (5 plies) as a function of the treatment procedure depending on the treatment time during the cure cycle [7]	10
Fig. 1. 3. Schematic representation of objects with (a) Minimum porosity, (b) Average porosity and (c) Extended porosity level [8]	10
Fig. 1. 4. Obtained porosity images [9]	11
Fig. 1. 5. The micro-CT image slice (a) and the results of the voxel model segmentation with different methods (b, c, d and e) [10]	12
Fig. 1. 6. Variation in thickness for both types of composites with (a) 3K material and (b) 12K material [11]	13
Fig. 1. 7. Illustration of the variation of fibre volume fraction for the same type of composite [11]	14
Fig. 1. 8. Specification limits for mass properties of one prepreg [3].....	15
Fig. 1.9. Deviation between the theoretical and actual orientation of the plies per plate with (a) deviations for all plates (plates are ordered in relation to the chronological order of draping) and (b) illustration of the deviation between the orientations of the base ply and any ply [12]	16
Fig. 1. 10. Level of fibre misalignment generated by draping narrow and wide UD prepreg strips over a 100 mm diameter hemisphere [6]	17
Fig. 1. 11. Studied structure scheme [31].....	23
Fig. 1. 12. Scheme for comparison between different numerical methods [35]	25
Fig. 1. 13. Boundary state concept [37]	26
Fig. 1. 14. Presentation of the stringer with main areas of a stringer (Ω -shape) (a) and dimensions (mm) of a part's cross-section (b)	30
Fig. 1. 15. Manufactured prototype of composite fuselage from BLAST project [42] (scale 1/3)	30
Fig. 1. 16. Exploded view of the stacking sequence inside the Ω -shape stringer designed in the BLAST ANR Astrid project [42] supported by the French Ministry of Defence.....	31
Fig. 2. 1. Step flowchart of experimental work along with observations during and after manufacturing process.....	38

Fig. 2. 2. Studied stringer with coordinate system (x,y at a given z) and different areas names	40
Fig. 2. 3. Drawing and cutting tools of plies with (a) UD ply (with blue protective films) and (b) woven ply (with red protective films)	40
Fig. 2. 4. Mould (right) and counter mould (left) used for lay-up.....	41
Fig. 2. 5. Wedges used for core fixing	43
Fig. 2. 6. Comparison between difficulties in four ply categories with categories #1, #2, #3 and #4 successively along with the schematic representation on the stringer cross-section right half.....	44
Fig. 2. 7. Scheme of the vacuum bag products (not to scale) used during curing step	46
Fig. 2. 8. Thermocouples installation for temperature tracking during curing	46
Fig. 2. 9. Structure covered with vacuum bag and prepared for polymerisation in the oven...	47
Fig. 2. 10. Theoretical programmed polymerisation cycle	47
Fig. 2. 11. Example of width measurements for ply#3 (in mm) in different locations along prepreg with 119 mm as theoretical width	49
Fig. 2. 12. Example of difference between theoretical (red rectangle) and real prepreg dimensions after cut process for ply#4.....	49
Fig. 2. 13. Example of difference between theoretical (red rectangle) and real prepreg dimensions after cut stage for ply#2	50
Fig. 2. 14. Grid used in the vertical (y-direction) photographic measurement	51
Fig. 2. 15. Cut lines (left) and installed grid on the mould (right)	51
Fig. 2. 16. Camera installation for photo shooting in a clean room with (a) roof camera with focal axis set vertically, (b) computer and (c) lay-up mould	51
Fig. 2. 17. Measurement of edge ply location before polymerization using the reference grid through the example of 3 rd ply while draping before taking off top ply protector.....	52
Fig. 2. 18. Scanning device used for 3D scan, the hand-held on the left and the scanner on the right	53
Fig. 2. 19. 3D scan in progress on the scanning plan	53
Fig. 2. 20. Example of obtained 3D image for the 3 rd manufactured stringer using 3D scan ..	54
Fig. 2. 21. Extracted sections with difference measure for Y displacement	54
Fig. 2. 22. Cut-out locations for each of four stringer samples	55
Fig. 2. 23. View of samples with resin coating in preparation for microscopic observation ..	55
Fig. 2. 24. The microscope Olympus BX41M-LED used for visualisation	56
Fig. 2. 25. Definition of local coordinate system related to each image	57

Fig. 2. 26. Definition of a global coordinate system (reference frame) with (a) intersection point definition and (b) coordinate system origin definition.....	58
Fig. 2. 27. Transformation of stringer cross-section after translation of images	58
Fig. 2. 28. Comparison between theoretical (designed) and real (after polymerization) locations of ply edges of sample A1 (see Fig. 2. 22.) in the corner of top flange and web using microscopic analysis	59
Fig. 2. 29. Comparison between theoretical (designed) and real (after polymerization) locations of ply edges of sample A1 (see Fig. 2. 22.) in the corner of the bottom flange and web using microscopic analysis	60
Fig. 2. 30. Uncertainty of measurements on edge locations of plies with 100 pixels ~ 0.04 mm	60
Fig. 2. 31. Comparison of the ply#5 stop location in the four manufactured stringers in the corner between the web and the top flange areas in the same location (z)	61
Fig. 2. 32. Sample's section preparation for tomographic visualisation with (a) sample inside tomography and (b) sample glued on wooden support	63
Fig. 2. 33. Example of obtained image for the stringer's cross section (treated with Imagej). 64	
Fig. 2. 34. Real obtained polymerisation cycle recorded by the thermocouples installed between the structure and the oven	64
Fig. 3. 1. Schematic representation of gap between theoretical and real ply stops in an Ω -shape stringer cross-section.....	68
Fig. 3. 2. Measurement locations along Ω -shape stringer's length z	68
Fig. 3. 3. The gap between theoretical and cut out width of different ply categories with (a) 1 st category woven and (b) 2 nd category UD.....	69
Fig. 3. 4. Example of gap measurement between real ply locations and theoretical locations while draping using the reference grid at different locations (ply #3)	71
Fig. 3. 5. Gap between theoretical ply stop locations and after polymerization ply stop locations of different categories using microscopic observations respectively with (a) 1 st category, (b) 2 nd category, (c) 3 rd category and (d) 4 th category	72
Fig. 3. 6. Schematic representation of symmetry gap on the right and left side of an Ω -shape stringer cross-section.....	74
Fig. 3. 7. Gap of ply edge positions after polymerisation on the left and right side in an Ω -shape stringer cross-section for different plies, example of the stringer #1.....	74
Fig. 3. 8. Example of angle measurement for ply#3 using image obtained from vertical device during lay-up	76

Fig. 3. 9. Example of map for misalignment gap for ply#1 along the structure (mean gap for 4 manufactured Ω -shape stringers)	77
Fig. 3. 10. Example of map for misalignment gap for ply#11 in the top flange (mean gap for 4 manufactured Ω -shape stringers)	78
Fig. 3. 11. Schematic description by a pseudo-Gaussian surface (on right) of a localized zone of fibre orientation perturbation (on left) [8].....	79
Fig. 3. 12. Tomographic observation of a stringer's section for the first manufactured Ω -shape stringer.....	80
Fig. 3. 13. Tomographic observation of the complete stringer's section for the 2 nd manufactured Ω -shape stringer	81
Fig. 3. 14. 3D representation of voids inside an Ω -shape stringer's length equal to 10 cm.....	81
Fig. 3. 15. Zoom for tomographic image for the Ω -shape stringer's top flange	82
Fig. 3. 16. Curve of porosity rate calculation (top) and example of porosity in the top flange area (bottom)	82
Fig. 3. 17. Porosity shapes representation by tomographic images reconstruction in the corner between web and top flange with (in red ellipse) long shaped porosity and (in yellow circles) circular shaped porosity	85
Fig. 3. 18. Curve of porosity volume fraction depending on the structure thickness in the top flange area	87
Fig. 3. 19. Porosities representation (bottom) in the corner between the web and the top flange (the red circle in the top)	88
Fig. 4. 1. Numerical Ω -shape stringer cross-section with related zone numbers and symbol of each node.....	92
Fig. 4. 2. Mesh finesses in the Ω -shape stringer's numerical model.....	93
Fig. 4. 3. Schematic representation for applied mechanical loading on one side of the Ω -shape stringer with boundary conditions on the other side	95
Fig. 4. 4. Flowchart for the integration of ply stop variability in numerical modelling.....	97
Fig. 4. 5. Example of ply #12 edges after ply stop location variability integration in the numerical modelling.....	97
Fig. 4. 6. Flowchart for porosity integration in numerical modelling.....	99
Fig. 4. 7. Schematic representation for different areas considered in porosity numerical study	99
Fig. 4. 8. Schematic representation for porosity concentration in the Ω -shape stringer using finite element number.....	100

Fig. 4. 9. Elastic woven composite component along the transverse direction x with voids consideration	100
Fig. 4. 10. Twin model results in the case of vertical displacement fields (Y direction), after cooling followed by tensile loading for the 4 manufactured Ω -shape stringers.....	101
Fig. 4. 11. Curve of y-axis displacements of edge nodes along stringer length for the case of models with and without ply stop variability after cooling.....	102
Fig. 4. 12. Curve of Strain ε_{11} of edge nodes along stringer length for the case of models with and without ply stop variability after cooling	102
Fig. 4. 13. Comparison between models without and with variability (random draw from input data from the 4 stringers) in the case of (a) displacement in y-direction and (b) shear strain fields after tension preceded by cooling.....	104
Fig. 4. 14. Curve of maximum strain in function of maximum gap between real and theoretical ply edge locations (red point indicates the value in theoretical case without variability) after cooling.....	106
Fig. 4. 15. Tsai-Hill criterion along the Ω -shape stringer length (see complementary scheme) in different calculations of variability (Var 1 to 6) after cooling	106
Fig. 4. 16. Comparison between displacements in the horizontal direction for numerical models without (left) and with (right) fibre misalignment consideration in the case of mechanical loading after cooling	107
Fig. 4. 17. Schematic representation for the path used for results analysis	108
Fig. 4. 18. Curves for x-direction displacement U1 along the transverse direction for cases with and without fibre misalignment variability consideration.....	108
Fig. 4. 19. Comparison between Tsai criteria for numerical models without (left) and with (right) fibre misalignment consideration in the case of mechanical loading after cooling	109
Fig. 4. 20. Repetitive calculation of horizontal displacement U1(z=0) with different random draws (1 to 9) for fibre misalignment variability analysis in the case of mechanical loading after cooling.....	110
Fig. 4. 21. Comparison between numerical modelling without (left) and with (right) porosity variability consideration for mechanical response after cooling.....	111
Fig. 4. 22. Curves for displacement U1 along the transverse direction for cases with (orange curve) and without (blue curve) porosity variability consideration for mechanical response after cooling with results location scheme in the stringer length.....	112

Fig. 4. 23. Curves for Tsai-Hill criterion along the transverse direction for cases with (orange curve) (one random draw) and without (blue curve) porosity variability consideration for mechanical response after cooling with results location scheme in the stringer length.....	112
Fig. 4. 24. Comparison between numerical modelling of horizontal displacement U1 without (left) and with (right) total studied variabilities consideration after cooling	113
Fig. 4. 25. Curves for displacement U1 along the transverse direction for cases with (orange curve) and without (blue curve) variability consideration after cooling with scheme for result location in the stringer length.....	114
Fig. 4. 26. Comparison between numerical modelling of Tsai-Hill criteria without (left) and with (right) total studied variabilities consideration after cooling	114
Fig. 4. 27. Curves for Tsai-Hill criterion along the stringer length for cases with (orange curve) and without (blue curve) variability consideration after cooling with scheme for results location	115
Fig. 4. 28. Comparison between numerical modelling of horizontal displacement U1 without (left) and with (right) total studied variabilities consideration for mechanical loading after cooling.....	116
Fig. 4. 29. Curves for x-direction displacement U1 along the transverse direction for cases with (orange curve) (one random draw) and without (blue curve) variabilities consideration after mechanical loading after cooling with scheme for result location.....	117
Fig. 4. 30. Comparison between numerical modelling of Tsai-Hill criterion without (left) and with (right) total studied variabilities consideration for mechanical response after cooling .	117
Fig. 4. 31. Curves for Tsai-Hill criterion along the transverse direction for cases with (orange curve) and without (blue curve) variabilities consideration for mechanical loading after cooling	118
Fig. 4. 32. Repetitive calculation of horizontal displacement U1 ($z=0$) with different random draws (Var from 1 to 5) for variabilities analysis in the case of mechanical loading after cooling.....	119
Fig. 4. 33. Repetitive calculation of Tsai-Hill criterion along Ω -shape stringer's length with different random draws (Var from 1 to 7) for variabilities analysis in the case of mechanical loading after cooling.....	119

List of tables

Table 2.1. Definition of plies in the stringer by ply number, ply orientation, prepreg width and prepreg type.....	39
Table 2.2. Ply number and their edge locations in the left half section of the stringer.....	42
Table 2.3. Representation of ply categories with (in red) woven plies and (in green) UD plies	45
Table 3.1. Standard deviation and mean for coefficients of polynomials after cut-out process obtained from the 4-stringer analysis	70
Table 3.2. Standard deviations and means for polynomial coefficients after polymerization process obtained from 4 Ω -shape stringers	73
Table 3.3. Examples for waviness classification [6].....	75
Table 3.4. Example of angle gap measurements for different woven plies in different locations (mean gap for 4 manufactured stringers) in z direction	76
Table 3.5. Example of angle gap measurements in z direction for different UD plies in the top flange area (mean gap for 4 manufactured Ω -shape stringers)	77
Table 3.6. Voids rates in different sections of an Ω -shape stringer's length in the top flange area	84
Table 4. 1. Zone definitions with specifications and theoretical ply stops included in each cross-section area	92
Table 4. 2. Material theoretical properties and failure characteristics used in the numerical models	94

General introduction

Composite materials have excellent mechanical properties, combining high strength and stiffness with low density, allowing for low energy consumption (may be specifically low fuel consumption in transportation industry). Other advantages, compared to conventional materials, are the durability and the favourable fatigue properties. In addition, the design and the manufacture of composite structures allow to obtain good mechanical performance with a minimum of material. Composite structures are widely used in the aerospace, leisure and energy industries.

However, the use of composites introduces difficulties. The cost of manufacturing is high. Another major obstacle is the uncertain reliability. It is known that variability and uncertainties exist in all manufacturing processes, but composite materials generate a higher level of variability than conventional materials. The variability of macroscopic performance is directly related to uncertain parameters, such as material, physical and geometric properties. While the concepts of error and uncertainty have become established in the experimental field, calculations generally provide only one result that is considered certain. However, small variations affect the parameters from one part to another, and the real loading conditions are rarely perfectly known. Ensuring the safety of design, or improving the identification of models from tests, therefore implies integrating the notion of uncertainty in one way or another.

The quantification of their uncertainties and their consequences are important. Then, the development of methods to propagate the variabilities on the uncertain parameters and quantifying them is necessary. Considering that experimental tests are costly in time and material, stochastic and mathematical mechanics and along with numerical methods are to be combined.

Currently, many methods have been proposed and developed to study non-deterministic problems. Among them, probabilistic approaches require knowledge of the probability law of the input variables and allow the output variables to be expressed in terms of statistical quantities such as the mean, standard deviation or distribution. These methods are generally based on non-realistic input data, or data taken from non-industrial composite structures. Also, few studies present continuous developments in surface or volume properties of parts, based on real data.

In this context, this thesis proposes a methodology of variability study from raw material to numerical representation, within a composite omega-shape structure that presents complex geometry zones and is inspired from the composite fuselage Ω -shape stringer.

The thesis is constituted by four chapters and ends up with a general conclusion and three appendixes.

In the 1st chapter, the reader finds a complete description of the context of this work. In particular, the interaction between composite materials and uncertainties is detailed, as well as the way these uncertainties influence the design of the composite structures. These descriptions are then used to guide the literature review on the transport of uncertainty, its forms, its sources and the different methods used to represent it. It shows that the study of variabilities in composite parts remains an open problem that can be approached in several ways and at several scales. Indeed, there is no general methodology defining each step associated with the establishment of the composite solution. The unique definition of variability remains subject to interpretation by researchers or designers.

For the analysis and manufacture of composite structures, there are a large number of methods, multi-scale or not, allowing the influence of variability to be taken into account in the study of composite structures. But among all these methods, few concern the consideration of the continuous variation of properties within a composite part, or the influence of the presence of localised defects.

For the study of variability, the work in the literature focuses, on the one hand, on flat structures without taking into account the situation corresponding to geometrical complexities, and on the other hand, on data which are not necessarily derived from physical observation. It is well established that the properties of a composite structure depend on both the choice of raw materials and the manufacturing process. These properties are influenced by several types of variability such as variation in the position of ply stops, fibre misalignment and the presence of porosities. These three issues are addressed in this thesis.

It has been chosen to gather realistic input data to feed probabilistic models. Mathematical laws adapted to the description of the spatial variations of the variability studied are proposed. The studies are carried out on the basis of magnitudes derived from statistical investigations concerning the identification of the parameters of the mathematical laws chosen beforehand through physical observation.

To illustrate the proposed methodology, we consider as an example of a structure with complex geometry, a prototype of a composite Ω -shape stringer inspired by an aeronautical fuselage. In this work, it constitutes the receptacle of a certain number of manufacturing constraints such as the presence of convex and concave radii, vertical walls, changes in thickness and the association of several materials. In this work, an omega-shaped structure is examined, consisting of 20 plies from preforms with unidirectional and woven reinforcements. The structure is manufactured using a dedicated mould, including a core for internal geometry control, and cured in a single stage in an oven.

The 2nd chapter is dedicated to the presentation of the experimental protocol used and the variability observations for the analysis of ply stops, porosity and fibre misalignment. The observations are carried out throughout the manufacturing process, using a wide range of measurement techniques such as optical, micrographic, tomographic observations etc. Each observation is dedicated to the monitoring of one or more types of variability studied for different phases of the manufacturing process.

The different manufacturing and observation steps are explained starting with the observations of the actual dimensions of the prepreg preforms, via an optical technique using a scanner. Then during the draping phase, each laid ply is optically observed to determine the position and shape of the ply ends. Local measurements of the actual orientation of the fibre strands before curing are taken. The structure is then cured in an oven under a vacuum cover with a plateau at 120°C for two hours. Once the structure is consolidated, it is scanned in 3D to identify its real geometry, which is different from the theoretical geometry due to the presence of internal stresses. The fabricated part is then cut out for observation. Microscopic and tomographic analyses are performed to determine the variations of the studied parameters, mainly the positions of the ply edges, the size and the extent of the porosities that appeared during the consolidation phase.

The 3rd chapter includes the mathematical study of the observation results and the representation of statistical trends. The data acquired are sorted according to their position with respect to geometric singularities and analysed in order to propose relevant mathematical models. The originality of the work lies mainly in the desire not to limit itself to statistical analyses, by characterising realistic spatial evolution trends for each type of variability observed, in relation to the presence or absence of a geometric singularity nearby. Once the representative mathematical laws have been identified, statistical analyses are carried out on the parameters driving these laws, mainly in the form of means and standard deviations.

The spatial evolution of the position of the ply stops is modelled by polynomials of degree 5, sufficient to describe the undulations observed over a Ω -shape stringer length of about 400 mm. The 6 parameters describing these polynomials are analysed from a statistical point of view, by separating the measurements into four families of ply (categorised based on the manual lay-up specifications for each ply), determined according to characteristics such as strand orientation or position with respect to a geometrical feature. The modelling of the spatial evolution of the fibre orientations is based on previous work and is represented by a finite sum of deformed zones, using a Gaussian surface centred on the centre of the perturbations. The amplitudes of the perturbations are statistically studied with an adaptation to the characteristics of the structure studied from the observations made. Finally, the porosities appearing during the consolidation phase are evaluated in terms of position and extent in the horizontal plane and represented by statistical laws conducted according to their location in the cross-section of the studied stringer.

The aim of the 4th chapter is to propose a numerical approach to evaluate the influence of these variabilities on the properties of the structure. It is shown how it is possible to create a family of numerical models representing numerical twins of fabricated structures and then virtual parts (not yet manufactured) in a realistic way. The realism sought targets both the amplitude of the simulated variabilities and the spatial evolution of this variability within the part. Indeed, a realistic model cannot propose discontinuous evolutions of certain properties from one finished element to another if the property studied cannot present this type of jump within the real structure.

Among the objectives of the work, it is explained how the numerical model should not only provide a numerical twin of the real manufactured and observed parts, but how it should also allow to extrapolate the information, obtained from the observation of a finite number of real structures, to extrapolate an infinite number of virtual composite structures. It is also a question of extrapolating the results to lengths greater than those observed during the experimental part. The numerical simulations should make it possible to propose ranges of variation in the properties and behaviour of the structures according to the variability studied.

The set of mathematical models from chapter 3, proposed for the representation of the spatial evolution within the part respectively of the ply stops, the fibre orientations and the porosity rate, is used for the creation of a Finite Element model taking into account local variations of properties. The model uses 2D shell elements, to be able to run a large number of simulations based on random draws of the parameters of the mathematical laws studied. In this proposed model, the values of the material and geometric parameters change not only from one mesh to

another, but also for the same mesh, for each of the 20 plies constituting the shell element of the composite structure.

In general conclusion, it is shown that this work has set up an approach allowing the evaluation of uncertainty effects in mechanical calculations applied to composite structures with complex geometry. The question addressed is to ensure the robustness of a result in a more physical way than by using safety factors. The perspectives of this work are then presented.

Chapter 1: Bibliography

1. Introduction to the variability study in the literature from experimental measurements to numerical validation

Today, the aerospace industry relies heavily on the use of advanced composite materials to reduce structural weight and maintenance costs associated with corrosion and fatigue.

However, composite materials have a high variability in their mechanical properties compared to conventional engineered materials. The variability of the semi-finished products, the manufacturing conditions and the chosen geometries of the composite structure can be the origin of the material variability.

The design of these aeronautical structures depends on structural performance prediction based mainly on numerical finite element modelling. But the theoretical numerical model will differ from the real structure because of the variabilities, which can affect its final properties. The numerical consideration of the numerous variabilities of composite structures is a challenge to enable the optimisation of this type of structure in the long term, while improving reliability and air safety.

In order to model these variabilities, for example by inserting them in the form of statistical mathematical laws, it is necessary to use appropriate numerical modelling. The elaboration of these specific numerical models is based on the physical observation of the study structure in order to determine the different existing sources of variability. Then, based on the literature, the trends found are represented by statistical laws, in order to choose an appropriate numerical method to represent the outputs of our system.

2. Variability in composites

The term composite designates a material composed of several components at different scales, whose combination presents a significant advantage compared to a homogeneous situation. Despite these widely proven advantages, mastering the perfect relative position of these components, during all steps of the manufacturing process, is difficult or even impossible. This is due to the nature of the composite material, and results in variabilities in composite structures at different scales. Indeed, Composites are tailor-made materials whose rigidity or strength can be adapted to the loads encountered. Faced with the diversity of fibre and matrix choices, fibre ratio values and possible stacking for laminates, making relevant choices without a predictive

model appears impossible. As a result, multi-scale modelling (cf. Fig. 1. 1) of composites has become a very relevant topic in composite science [1].

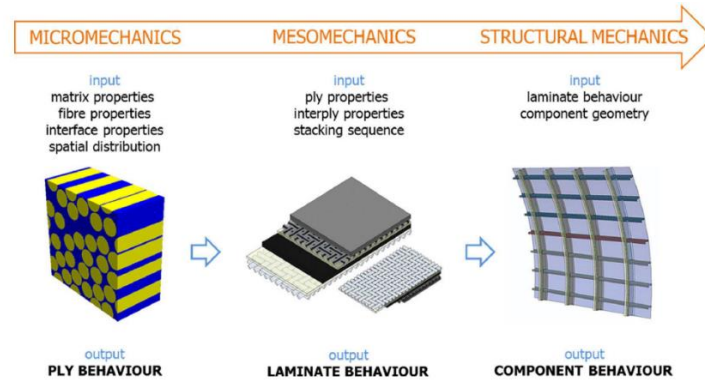


Fig. 1. 1. Strategy scheme for multi-scale modelling from microscale to macroscale [1]

For studies dealing with variabilities in composite materials, it is important to first refer to the proper definitions of variability and uncertainty terms which are the vocabulary used by authors of reviews dedicated to such issues related to the composite materials and structures [2] [3] [4].

The definition of these notions remains an open problem and differs from one author to another [5]. This field of investigation regarding the definitions is clearly a mathematical issue in itself whatever the associated mechanical applications. In chapter 1 of [5], an author proposes a review providing clear definitions of uncertainty, involving variability, particularly in the field of mechanics of materials and structures. This author in [5] provides two distinct definitions of uncertainties; between the random and epistemic ones. Variabilities of material properties belong to random uncertainties, which require probabilistic analysis but can also be considered as epistemic uncertainties because of a lack of information.

This lack of knowledge could be caused by the constraint of the discrete observation along with a composite structure. Indeed, continuous observations and measurements, addressing the variable data from one point to another through the structure, is nearly impossible to be carried out in a continuous manner perform, but in a discrete one. On one side hand, this leads to uncertainty issues in identifying the evolution laws of the studied data due to the lack of information. On the other side, the lack of information is due to the difference of property variabilities from one structure to another one, despite having the same geometric characteristics and the same manufacturing process. The author in [5] explains that uncertainty problems are generally treated using probabilistic analysis with different mathematical methods related to the type of uncertainty.

The reviews in the literature underline the importance of studying variabilities, in order to take them into account, including their influence in numerical modelling. These reviews point out that numerical consideration can be achieved by studying the variabilities and sources associated with all scales and at all steps of the manufacturing process in [2] and more particularly for long fibre reinforced preregs with a thermosetting matrix. The listed works start from the different variability sources, including those caused by the manufacturing process and their influence on structural behaviour, to the analysis of these uncertainties using statistical and probabilistic tools [6].

3. Variability classification

Authors in literature classify variability in different kinds, according to their type, source, effect on the structure's behaviour etc... The first classification concerns different types of variability such as fibre misalignment, thickness, material properties etc. The second one related to variability sources.

Let's start with porosities which is the subject of many works in the literature. For example, in [7], authors study porosity variability using tomographic imaging methodologies for composites and the different parameters that can be improved to obtain a better resolution.

They also give examples of tomographic observations (cf. Fig. 1. 2) to see plies, porosities and cracks in a composite structure based on an imaging strategy (depending on the observation objective). Void content decreases when the processing time increases.

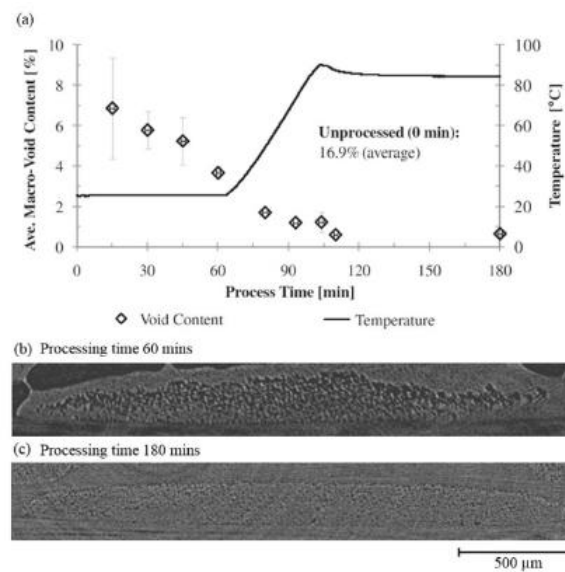


Fig. 1. 2. Tomographic images showing the porosities in a CFRP laminate (5 plies) as a function of the treatment procedure depending on the treatment time during the cure cycle [7]

In order to evaluate porosity effect on the composite behaviour, another work [8] presents a methodology that uses tomographic observations to develop a numerical model of the composite according to the size of porosities (cf. Fig. 1. 3) (minimum porosities, medium porosities and extended porosities). To validate the used method, mechanical loadings were applied. Results show that there is a good correlation between numerical modelling and mechanical results [8]. The specimens used in the experimental tests were CFRP specimens made from EHkF 420-HTA unidirectional prepregs with a resin mass fraction of 35%. The obtained numerical and experimental results show that when increasing porosity rates (size), transverse stiffness and strength decrease.

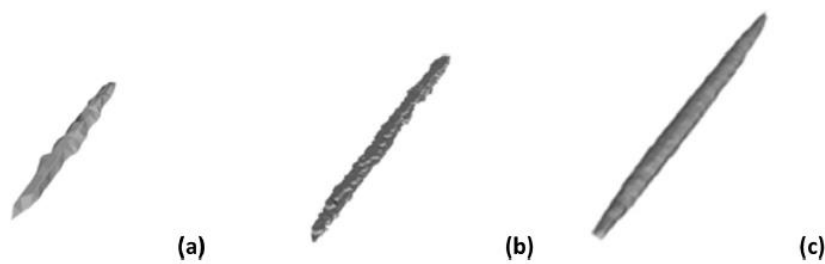


Fig. 1. 3. Schematic representation of objects with (a) Minimum porosity, (b) Average porosity and (c) Extended porosity level [8]

In relation with the curing cycle, works in the literature investigate the effect of porosities on the mechanical behaviour of composites as a function of curing cycle (autoclave) such as in [9]. The determination of porosities is realised using tomographic images (cf. Fig. 1. 4) on composite specimens made from 16 and 32 ply prepreg unidirectional composite laminates (unidirectional HTA 24 k carbon fibre and EHkF 420 epoxy resin). The dimensions of the samples for tomographic observation are: 13.45 mm x 9.46 mm x 2.75 mm. Results show that the presence of pores reduces all matrix-dominated material properties of UD CFRP material. The reduction in strength is greater than the reduction in the elastic properties as at high low load-levels the pores interact with damage mechanisms accelerating the fracture process [9].

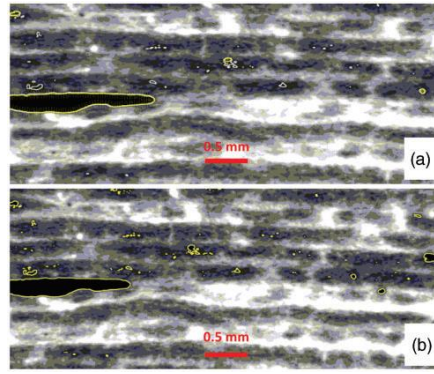


Fig. 1. 4. Obtained porosity images [9]

For fibre orientation variability, there are different methods of observation. Some papers use X-ray tomography method to evaluate fibre orientation such as in [10]. The sample studied is made from a 3D woven carbon/epoxy composite with a cross-sectional area of 2.7 mm^2 that has been cut into a plate along the fibre direction. The composite is reinforced with Toho Tenax carbon fibres and has a fibre volume fraction of 51.1%. From the X-ray images (cf. Fig. 1. 5), an analysis was made of the degree of anisotropy and the orientation of the fibres. These results were implemented in a numerical model. Results show that the uncertainty of the used methodology depend on the voxel size and the size of integration. Also, gradual transition of the orientation between orthogonal directions has been noticed, which can introduce a fraction of non-physical orientations [10].

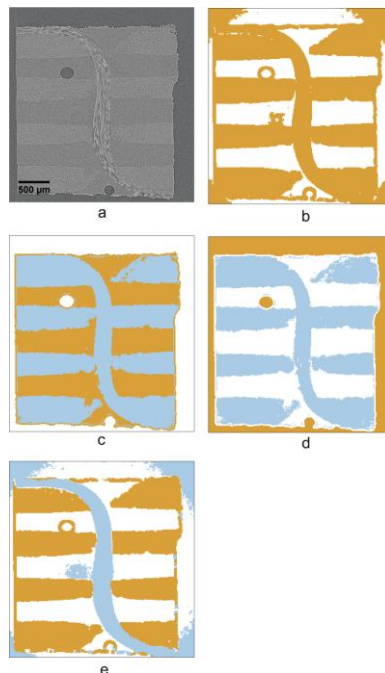


Fig. 1. 5. The micro-CT image slice (a) and the results of the voxel model segmentation with different methods (b, c, d and e) [10]

The authors in [11] measure fibre orientation through image analysis techniques of the composite plies during the lay-up phase within a composite plate. Image analysis results show that variations in the mean ply orientations are between -0.5 and 0.5° with standard deviations ranging between 0.34 and 0.41° . The obtained set of data was used for numerical validation.

As for ply thickness variability, an example of paper in literature investigates variability within two types of composites [12] based on the same composition but with different fibre thicknesses (3K: A fabric has 3000 filaments per yarn and 12K: A fabric has 12000 filaments per yarn). The material used in this work is made from HexPly1 M10.1 prepregs with T700S carbon fibres. The prepregs, which are stored in a freezer at -18°C , are cut into $300\text{ mm} \times 300\text{ mm}$ (sheet). The polymerisation cycle lasts 2 hours with a maximum temperature of 120°C . The samples for microscopic observations are prepared using an automatic polishing machine. The measured parameters are thickness of the samples, size and shape of the fibres in a cross section, volume fraction of the fibres, alignment defects, etc. in order to quantify their effect on the behaviour of the two composite materials. From the results found by authors (cf. Fig. 1. 7.) [12], mathematical distribution models are calculated to be used in a numerical model (cf. Fig. 1. 6). The variability of measured geometrical values is higher for 12K material than for 3K material and there is a significant variation in the average fibre volume fraction among different types using two different methods for the area measurement.

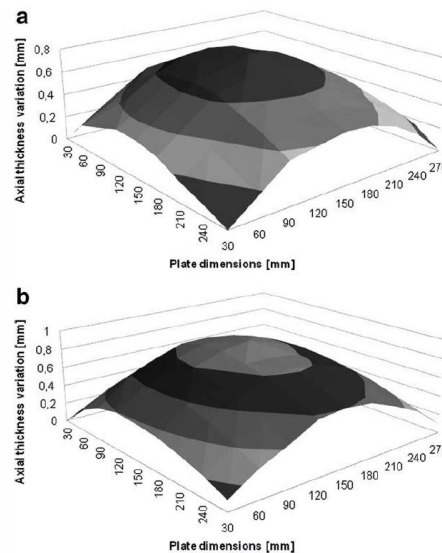


Fig. 1. 6. Variation in thickness for both types of composites with (a) 3K material and (b) 12K material [12]

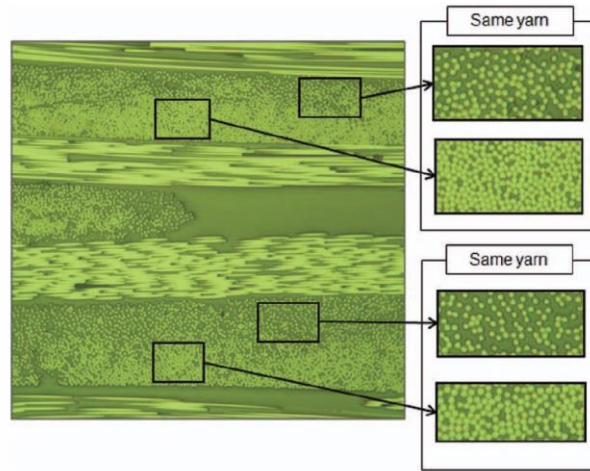


Fig. 1. 7. Illustration of the variation of fibre volume fraction for the same type of composite
[12]

There are other forms of variability in composite structures such as plies drop off, elastic properties, etc. that have different sources, from structure design to curing conditions.

4. Variability sources during composite structures manufacturing process

There are more than 130 types of defects and more than 60 sources of variability for a composite material in relation to its manufacturing process according to [3]. Sources are classified as follows [3]:

- Variability in the prepreg upon receipt;
- Variability due to lay-up and stacking;
- Variability due to solidification and curing of the resin;
- Variability due to residual stresses and thermal distortion.

The sources of variability in mechanical systems composed of composite structures are diverse and numerous. They can be classified also into two main categories respectively, those related to all the steps of the manufacturing process, leading to inter variability and those related to the environmental parameters and the intrinsic nature of the systems under study, leading to intra variability. The overall variability includes both intra and inter variability.

For composite structures, the sources of variability can be associated with parametric uncertainties, i.e., material properties (moduli of elasticity, Poisson's ratio, shear moduli, densities, etc.) and physical parameters (thicknesses, fibre orientations, cross-sectional areas,

loadings, etc.). These parametric uncertainties are intrinsic to the behaviour of the structure studied and are therefore observed experimentally as well as numerically.

Starting with the manufacturing process as a main source of variability, in [2], a state of the art on experimental methods and stochastic simulations and the influence of variabilities on the main stages of composite manufacturing (draping, compaction, curing) is presented as:

- Variabilities in the prepreg (shape, fibre distribution, alignment ...);
- Variability during lay-up and shear caused by misalignment;
- Variability during consolidation (liquid composite moulding process): porosities, resin pockets ... which can affect the anisotropy coefficient of the composite as well as the fibre volume fraction and the porosity distribution;
- Curing cycle variability, cycle parameters and environmental conditions that can lead to residual stresses, porosities and sometimes thermal degradation due to temperature variability.

In [13], the author represents variability forms in relation with their sources. Variability sources are different and due to each step of manufacturing process, including the initial state of prepregs. The known sources are respectively, the variation in mass per unit area (unidirectional prepreg), the fibre misalignment, ply thickness, autoclave polymerization, porosity rate. During the autoclave polymerization of unidirectional prepregs, a number of parameters must be studied, respectively, the variation of the mass per unit area of the prepregs, the difference in orientation between the real and the theoretical case of the hand-placed plies in the lamination, the differences between the lay-up conditions during the autoclave cure cycle, the thickness of the plies and the total thickness of the cured plate [13].

In raw material, at the prepreg stage, the variability of fibre misalignment is addressed in two ways in [12], by direct measurements of fibre misalignments in the as-delivered prepregs, and by inference from measurements of the tensile load response of the cured prepreg. A significant level of fibre misalignment is detectable in the as-delivered prepreg, mainly in the form of in-plane wrinkles, although macroscopic out-of-plane wrinkles and more localised, small-scale out-of-plane wrinkles can also be detected.

From structural design as a source itself [3], it was examined how choices made during component design can influence the properties of composite materials. In [3], the aim was to demonstrate that the micro and meso structural features directly attributable to design decisions

must be rigorously identified. The influence of these features on composite properties must be explicitly included in the determination of allowable properties.

Also, in [3], the author shows that variabilities are inherent in the prepreg upon receipt. He estimates that variabilities in the prepreg can affect the local surface masses ($\sim 7\%$), on the resin mass rates ($\sim 6\%$) and on fibre orientations ($\sim 4^\circ$). The fibre orientation variability is relatively high due to packaging conditions for storage (generally large rolls) (cf. Fig. 1. 8.).

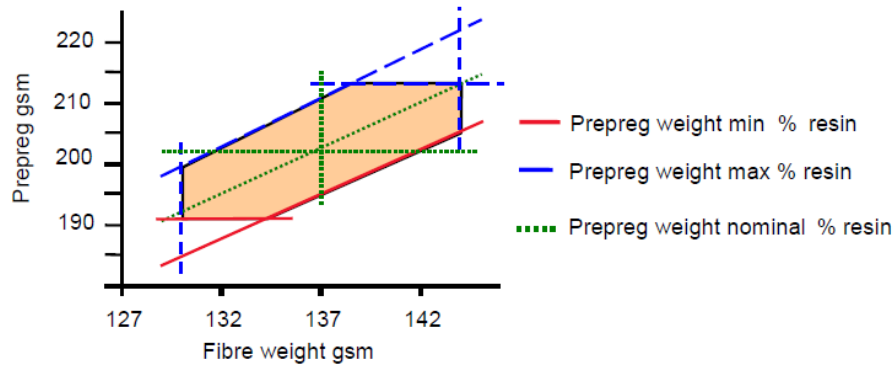


Fig. 1. 8. Specification limits for mass properties of one prepreg [3]

Many authors have studied the mechanical properties of textile materials at the microscopic (fibre/matrix delamination), mesoscale (fibre bundle transverse cracking or matrix cracking) and macroscopic (fibre bundle boundary delamination) scales and numerically. The material data introduced in the mathematical models must be reliable in order to obtain solutions as close as possible to the real physical phenomenon. The importance of the internal geometry of the structure is crucial when analysing the behaviour of the material under loading conditions. The dispersion of the geometrical structure in textile composites can also lead to a dispersion of the physical properties. Therefore, knowledge of the correlation between the source of variability and the resulting behaviour is essential to control the manufacturing process and consequently the final material properties.

Important characteristics such as mechanical elastic properties, formability, permeability, initial failure and damage progression are very sensitive to the variability of the internal geometric structure. Many authors [7-10] have found a direct relationship between fabric structure and material permeability values. For example, authors in [14] identified layer nesting as the main source of variation in permeability values for fabric-based composites. In relation to other properties, authors in [15] have demonstrated significant variations in formability due to the effect of yarn misalignment and local unit cell size in pre-impregnated woven textiles [12].

Chapter 1

Manual lay-up process has a significant effect the presence of variabilities in composite structures. Hand draping is a source of uncertainty when placing the ply in the composite laminate. There are deviations in ply placement from the theoretical designed orientation.

In [13], concerning the draping of medium-sized flat sheets (600 x 300 mm²) with quasi-isotropic lamination, the values found for the deviations between the theoretical fibre direction and the actual orientation measured during a manual draping operation indicate that this is a random variability, with deviations varying between -1° and +1.5° (cf. Fig. 1.9.). This operation remains highly dependent on the operator and the manufacturing conditions.

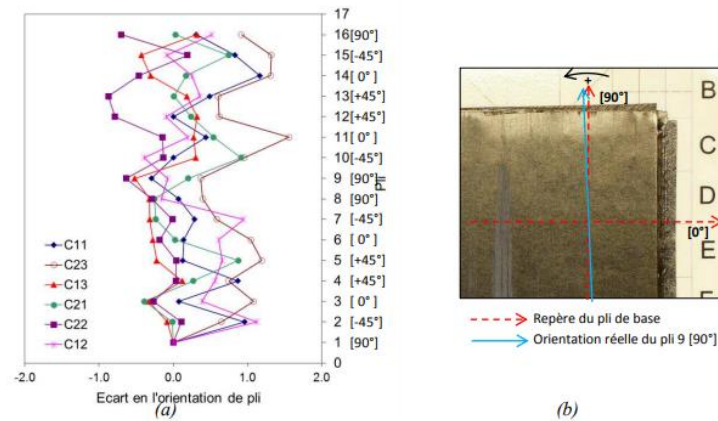


Fig. 1.9. Deviation between the theoretical and actual orientation of the plies per plate with (a) deviations for all plates (plates are ordered in relation to the chronological order of draping) and (b) illustration of the deviation between the orientations of the base ply and any ply [13]

The basic assumption in the design of composite components is that the properties of laminates in measured current areas are a reasonable representation of the properties of the actual components obtained. It is not clear that this is the case when considering complex components with significant geometric singularities.

Authors in [6] analysed the orientation of the fibres in a corner, where they found that each layer has a misalignment of the fibres, which increases when the point of observation is moved from the outside to the inside of the structure (cf. Fig. 1. 10.).

Fig. 1. 10. shows the level of waviness developed when laying a set of strips (~ 1.5 mm) of UD prepreg on the surface of a 100 mm diameter hemisphere and also illustrates the result of using much wider strips of UD prepreg following the same procedure.

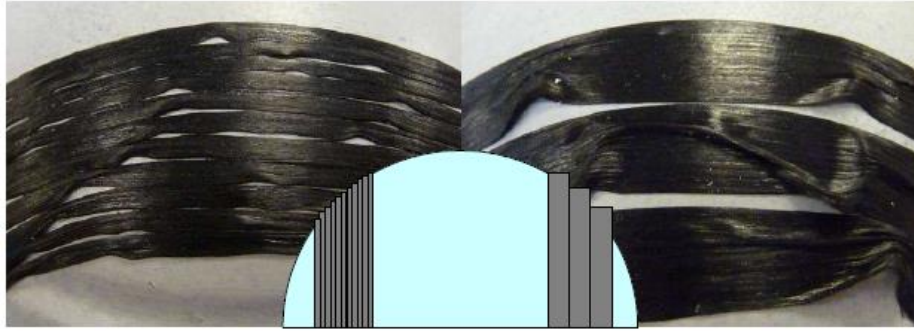


Fig. 1.10. Level of fibre misalignment generated by draping narrow and wide UD prepreg strips over a 100 mm diameter hemisphere [6]

The manual procedure of positioning the layers in the mould introduces additional uncertainty, as the accuracy is limited. Also, the thickness of the structure is not known precisely, especially when the layers overlap [6].

Most processes do not have the ability to guarantee the precise position and orientation of the fibre reinforcement. For some processes, the volume fraction of the fibres is also not constant in the volume of the component [16]. The crimp factor is another geometric parameter that determines the homogenised stiffness characteristics of a textile composite material. It is a measure of the crimp of the yarn through the thickness of the panel. A general trend is that the equivalent modulus of a woven composite increases as the crimp decreases [16].

The main factor that affects variability is the manual procedure of positioning the layers in the mould according to in [17], where the thickness of the structure is not precisely known, especially when layers overlap each other. For example, in [18], variability sources are analysed at the prepreg scale where the forming/draping stage causes significant shear deformations. The effect of variability on material parameters is also mentioned in [3] using a graph of dependency of in-plane material parameters on the orientation of the major fibre axis. Fibre orientation uncertainties after curing can cause significant variations in the outcome of the forming of woven composites, with variation coefficients from one point to another and with an average wrinkling strain in the range of 10–20 %. In [3], the author analyses the variation of the elastic orthotropic stiffness constants for different orientations of a uniaxial reinforced glass fibre composite lamina with respect to the applied uniaxial tensile load. The resulting graph exhibits a significant decrease of stiffness with increasing misalignment of the fibre, the variation of Young moduli for different alignments of the fibre orientations for the loading direction and demonstrates that the equivalent material stiffness depends strongly on the fibre placement. The heterogeneity of composites and the influence of the microstructure are manifested in different

forms, for example with the effect of the realignment of the fibres. Indeed, at the time of shaping, these fibres can be slightly wavy, and become taut when a tensile load is applied, thus in turn leading to a non-linear behaviour in a domain where it should have been purely elastic. As another example, the role of the interfaces (fibres/matrix, or between plies) is important [19], its effects on the behaviour and the damage initiation are essential, and interfacial properties are extremely sensitive to the elaboration conditions [20].

According to [17] and [19], variabilities during the polymerization cycle (cycle parameters and environmental conditions) can cause residual stresses, porosities and sometimes, thermal degradation due to temperature variability. While During curing, issues can cause the formation of dry spots and voids, extended impregnation cycles, uneven impregnation and resin-rich pockets can develop [21]. These variabilities can affect the Poisson's ratio of the composite as well as the volume fraction of the fibres and the distribution of porosities. In [22], it has been proven also that the consolidation of a prepreg layup to a target thickness is critical to achieve the required fibre volume fraction and dimensions of a composite part. Experiments on thick composite plates presented in this paper show that different processing conditions lead to different levels of compaction and variability in the thickness.

The thicknesses of the plate and the plies are the most affected during the polymerisation, notably by the non-homogeneity of the resin flow. In [13], the dispersion of ply thickness is quite significant despite the fact that the ideal case to limit variability was considered: flat plates (therefore without geometric accidents), draped manually with great care, and cured in an autoclave, with control according to core temperatures, and constant control of void values and external pressure. In the case of ply thickness measurements, the coefficient of variation is 13% for the four samples analysed. In addition, the coefficient of variation of the total plate thickness is about 2%. This variation was illustrated through a micrographic image of the cross-section of the sample, where the variation of the same ply is significant along the width of the sample [13]. The manufacturing process also affects the total plate thickness, which in the ideal case is taken as the sum of the ply thicknesses. In reality, the thickness of the plate or sample used is not uniform, as explained in [13], where the variabilities in plate thickness for a common area can be as high as $\pm 10\%$, i.e., well over one ply thickness for the 16 plies used.

5. Statistical and probabilistic studies after observations

After observing and classifying the sources of variability that can be found, and after obtaining experimental results, the next step will be to process and analyse these experimental data, which

will represent the uncertain input parameters of numerical models with variabilities. These parameters are represented by variables or random fields and follow statistical laws that should reflect observed data.

Let's start with the statistical variability definition in the literature. This term has many significations related to each field. The statistical variability (also called spread or dispersion) definition is presented in the first section. It generally refers to the way in which a data set is spread. Variability makes it possible to describe the variation in data sets and to use statistics to compare the data to other data sets. The parameters adopted to describe variability are generally, the range, the interquartile range, the variance and the standard deviation.

The range is the amount between the smallest and largest parameter in the set. The range can be found by subtracting the smallest number from the largest. The interquartile range is almost identical to the range. Only, instead of giving the range for the data set, it gives the amount for the "half". This is sometimes more useful than the range because it shows where most of the values are. The formula is $IQR = Q3 - Q1$, where $Q3$ is the third quartile and $Q1$ is the first quartile. The variance of a data set gives an approximate idea of the fluctuation of the data. When the variance is relatively small the data set is clustered and a large variance implies that the values are more dispersed. The variance is rarely useful except for calculating the standard deviation. The standard deviation indicates how clustered the data are around the mean. A small standard deviation indicates that the data are clustered, so we will also have a higher curve; a large standard deviation indicates that the data are more scattered.

Many studies represent variabilities from a statistical probabilistic point of view, in for plane or complex composite structures. For example, in [21], an analysis of fuselage type composite structure was carried out to simulate by calculation the uncertain behaviour of the composite structures. In [17], an attempt is proposed to quantify the uncertainty of FRP composites summarising the different stochastic modelling approaches proposed in the literature. The sources of uncertainty inevitably lead to the presence of various defects in the fibres, matrix, and fibre-matrix interfaces, and they result in dispersions and reductions in mechanical properties. Manufacturing defects in composites and their influences on mechanical properties of materials have been intensively studied, but quantitative studies for uncertainties arising from manufacturing defects are still lacking. In [23], variabilities were studied in composite plies and fibre orientation for a carbon fibre composite plate, where this aspect was modelled using mathematical models based on the observation of the material through the composite part. In this same context in [23], a Finite Element-based model was performed to estimate stiffness

properties of unidirectional composite laminas. This was done while accounting for geometric and material property uncertainties at micro, meso and laminate scales, all within a probabilistic framework. In [24], the available property estimation/homogenisation tools were presented by dividing them into two categories respectively, analytical methods constrained by configuration assumptions, and numerical homogenisation using Finite Element Analysis.

The study of the listed variability effects is represented by mathematical probabilistic evaluations, as affirmed in [25], where they are often conducted by numerical modelling. In [16] for example, probabilistic methods are used to describe scattering in properties where probability distribution functions can be established for all uncertain parameters, taking into account the correlation between different parameters such as in [17]. The result of the analysis can be interpreted in a statistical sense, and the probability of every output quantity depends on the input probabilities and their correlations. It is important that all these inputs are based on experimental observations to allow a statistical interpretation of the results (in [23]). Unfortunately, this fact is often neglected by many scientists.

Various mathematical tools can be found in the literature to represent the effect of uncertainty and variability on finite element models of mechanical systems. However, there is sometimes a lack of practical data on the mechanical systems of real materials to generate reliable and useful data. The author [16] proposes a reliable approach to real material variability modelling by the collection of experimental data following specific experimental approaches in the first step, then the second step is the stochastic multi-scale modelling of the architecture identified experimentally in step 1 and finally by the construction of the virtual specimens in the numerical modelling software. The method presented here is simple and efficient in the case of a plate geometry, but in the presence of geometric singularities, it becomes complicated.

Many forms of statistical laws can be found in this context, of which we will mention the most commonly used here after.

5.1. Gaussian law

The Gaussian distribution is the most commonly used to express the distribution of physical parameters (Eq. 1.1). The probability density of the normal distribution is given by:

$$f(X) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(X-\mu)^2}{2\sigma^2}\right) \quad \text{Eq. 1.1}$$

where μ is the mean and σ is the standard deviation.

This is the simplest and most commonly used form when an exponential distribution of experimental data is available.

5.2. Log-normal law

The probability density of the lognormal distribution is represented by Eq. 1.2:

$$f(X) = \frac{1}{X \sigma \sqrt{2\pi}} \exp \left(-\frac{(\ln(X)-\mu)^2}{2\sigma^2} \right) \quad \text{Eq. 1.2}$$

where μ and σ are the mean and standard deviation of the variable X , respectively.

This form is generally used when the variable under study is the result of the multiplication of a large number of small independent factors.

Statistical representations in the literature are different and numerous but they can be limited in many cases (related to physical reality) especially when no experimental work is conducted. The work in [26], for example, presents a comparison of the models obtained with the results of the Monte Carlo simulation. Good agreement was obtained, especially in the case of small strains. The proposed method is limited in the case of large deformations by making comparisons with deterministic results. The use of these different forms depends on different factors; boundary conditions, nature of loading, shape and type of specimen ... But these laws may be insufficient to relate to physical realities, so other laws of spatial evolution must be sought. There are some works in the literature such as those of [11], where the authors represent the observed experimental data of the orientations of a ply with a mathematical model using identification and optimisation. The study of reinforcement orientation variations uses meta-models to identify and control a continuous variation across the composite ply by direct measurement of angles through image analysis technics.

6. Stochastic numerical modelling

After studying the uncertain parameters by random variables or fields, probabilistic analysis requires the choice of a stochastic method to propagate the uncertainties and thus evaluate the response of the probabilistic mechanical model. There are many methods for introducing variabilities into stochastic models in the literature. Some examples are given below.

6.1. Monte Carlo simulation

Monte Carlo simulation (MCS) is a simple and reliable stochastic method for studying the variability of a model in probabilistic analysis [27], as it provides confidence intervals on the calculated probability. It consists of evaluating the expectation (and variance) of a random

variable by generating a large number of samples that follow the same probability distribution as the random variable. The MSC method requires a large number of draws. The model response is calculated for each draw and the population of results is then processed to evaluate the statistical quantities. It is observed that convergence depends on the number of draws. In order to obtain converged results, the number of draws must be quite large, usually more than 10000. The author in [27] has shown that for an accurate estimate to within 5% of a 10-k probability, one requires about $4 \times 10^k + 2$ points in the sample for the classical Monte Carlo method. In the literature, the Monte Carlo method is a basic method for studying stochastic problems. For example, the CGSM (Certain Generalized Stresses Method) developed in [19], where the Monte Carlo method is performed on a meta-model. The meta-model consists in decreasing the number of input variables after a sensitivity study with a certain shift of the physical reality [28]. For our study, we expect to have a few probabilistic variables per ply and per mesh, i.e., several hundred variables on a single structure. Classical Monte Carlo methods seem to us to be suitable.

6.2. Chaos polynomial method

Polynomial Chaos (PC), is a sampling-based method for determining the evolution of uncertainty in a system when there is probabilistic uncertainty in the system parameters. The Polynomial Chaos decomposition extends to the general case of independent random variables with known probability densities [29]. The polynomial chaos is based on the following form (cf. Eq. 1.3):

$$Y = \sum_{j=0}^{\infty} a_j \Psi_j \quad \text{Eq. 1.3}$$

where $\{\Psi_j, j \in N\}$ denotes a set of random variables forming the basis and $\{a_j, j \in N\}$ are the "coordinates" of the random variable Y in this database.

The polynomial Chaos method is widely used due to its simplicity of implementation, and it allows to integrate a large number of random parameters [30]. For example, in [31], an analysis of the random parameters within a mechanical component of a CNES launcher (cf. Fig. 1. 11.) was carried out by considering that the data likely to have a random effect are: the thickness of the folds of the composite (26 values), the deviation of the orientation of the folds (26 values), the vertical misalignment of the rivets (52 values), and the coefficients of the stress-strain relationship of the carbon.

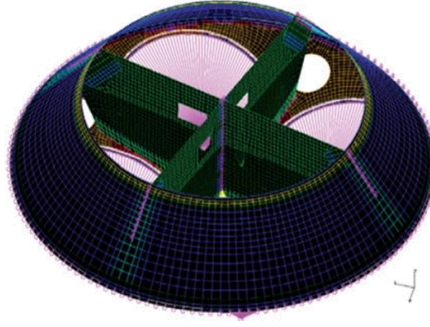


Fig. 1. 11. Studied structure scheme [31]

6.3. Perturbation method

The perturbation method [32] is a stochastic method based on the Taylor series development of uncertain parameters. The perturbed quantity Y (cf. Eq. 1.4) (stiffness matrix, loading or displacement...) can be expressed in terms of the uncertain parameters X_i with $i = 1, 2, \dots, n_v$:

$$Y = Y_0 + \sum_{i=1}^{n_v} \left. \frac{\partial Y}{\partial X_i} \right|_{X=X_0} (X_i - X_{0,i}) + \frac{1}{2} \sum_{i=1}^{n_v} \sum_{j=1}^{n_v} \left. \frac{\partial^2 Y}{\partial X_i \partial X_j} \right|_{X=X_0} (X_i - X_{0,i})(X_j - X_{0,j}) + \dots$$

Eq. 1.4

where n_v is the number of variables and the subscript "0" indicates the nominal values. In this expression, Y is the sum of the nominal value and the perturbed terms involving the increasing order sensitivities [32]. This method is used when the random variables are considered independent [33]. The perturbation method can be combined with the use of finite element software, as long as the software has the capacity to calculate first order sensitivities. However, when the level of input variability is high, or when the behaviour of the system is highly non-linear (plasticity, impact), the first order perturbation method is no longer robust.

6.4. Spectral form

The stochastic field is represented as a sum of trigonometric functions with varying phase angles and amplitudes as follows (cf. Eq. 1.5):

$$\hat{f}^{(i)} = \sum_{n=0}^{N-1} A_n \cos(K_n x + \varphi_n^{(i)}) \quad \text{Eq. 1.5}$$

with $K_n = n \Delta K = n \frac{K_u}{N}$; $n = 0, 1, \dots, N$.

A_n coefficients are defined by:

$$A_0 = 0, A_n = \sqrt{2S_{f_0}(K_n)\Delta K}, \quad n = 1, 2, \dots, N - 1$$

The spectral method is generally used when a small number of random variables are available, but in the case of large variabilities.

6.5. Numerical analysis: processing the results

6.5.1. Numerical study

The work in [34] investigates numerical tools used to determine internal geometric imperfections during the manufacture of composites, these imperfections are given in the form of 4 categories: Fibre misalignment, Porosity, Delamination and Residual stress.

The authors gave examples of these tools (exploitation of tomographic images to develop a numerical composite model to determine the effect of fibre misalignment (woven composite) on the local elastic behaviour of the composite. The comparison of different numerical methods conducts to conclude that every variability situation should be studied using specific numerical model.

Another work [35] compares different numerical methods for simulating uncertainties during the manufacture of unidirectional composites (cf. Fig 1.12).

These numerical methods predict the behaviour of the composite during manufacture using characteristic functions of the composite.

PAM-FORM	• The modelling is based on the assignment of characteristic curves for specific deformation mechanisms
Aniform	• based on an implicit time integration scheme
LS-Dyna	• Manufacturing simulation based on an explicit time integration scheme
Abaqus	• Enable the creation of a single global material model

Fig. 1. 12. Scheme for comparison between different numerical methods [35]

Another work from literature [36] gives a classification for numerical methods used to represent out-of-plane fibre misalignment variability. The authors affirm that since the manufacturing processes of composite materials are very different, a variety of waviness types may occur. Potential root causes are described and, based on that, a generic classification scheme was developed to differentiate between types of waviness. Among others, process parameters (e.g.,

temperature, pressure and deformation rate), the selection of the fibre and tooling material and its properties, as well as the complexity of the geometry of the final component are the main parameters that influence the occurrence of fibre waviness. A set of numerical methods is given taking into account all these specifications.

6.5.2. Reliability study

Reliability methods are methods by which one seeks to calculate a probability of failure associated with a criterion, represented by a limit state function whose arguments depend on the result of a finite element calculation [31].

There are a large number of reliability methods in the literature, of which we will mention some examples: direct Monte Carlo method, FORM/SORM methods, Kriging method.

The starting point of all methods is a performance function, which gives the relationship between the chosen performance and the model inputs (cf. Eq. 1.6):

$$g = g(x, \bar{x}) = \bar{y} - y(x, \bar{x}) \quad \text{Eq. 1.6}$$

where x is the vector of deterministic variables, $\bar{x}$ the vector of the stochastic variable, y the performance variable and $\bar{y}$ the corresponding limit level. If a single element of $\bar{x}$ varies stochastically, the performance variable y will vary in the same way, and g can be given as a probability distribution function [37].

The above equation can only be expressed analytically for simple problems. In general, it is given numerically, as in industrial applications, where the finite element method is widely used in structural analysis. The performance function gives the state (failing/not failing) (cf. Fig. 1. 13.) of the structure, as the reliability can be defined as the area of the probability function of y below the level $\bar{y}$.

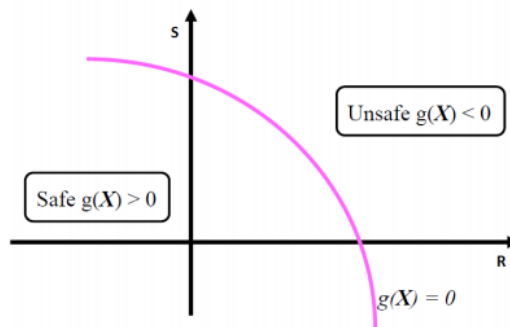


Fig. 1. 13. Boundary state concept [37]

The simplest and most intuitive method is the Monte Carlo method, which involves calculating the equation for a large number of combinations of $\bar{x}$. The combinations, called "trials", are randomly sampled from the probability distribution of each $\bar{x}_i$, using the standard random generator functions implemented on any modern computer. The method is useful because of its ease of implementation and its accuracy depending mainly on the number of trials one can afford, always considering the significant time consumed.

The FORM method (First Order Reliability Methods) can be represented geometrically on a multidimensional Cartesian coordinate system as minimum distance search procedures. The coordinate system consists of all stochastic variables $\bar{x}$. In all FORM methods, it is assumed that each stochastic variable is described by a normal linear probability function and is statistically uncorrelated from all others. Under such assumptions, it becomes useful to transform all stochastic variables into standard normal distributions [37]. The Second Order Reliability Methods (SORM) assumes that each stochastic variable is described by a quadratic probability function.

Engineering problems involve increasingly complex computer codes and the assessment of the probability of failure can be very computationally intensive. Meta-models are used to reduce these computation times.

Kriging is based on the idea that the probability function $g(x)$ is seen as the realisation of a stochastic field. The first step of Kriging is to define this stochastic field with its parameters according to a design of experiments. Then, Kriging is a stochastic spatial interpolation method that predicts the value of a natural phenomenon at unsampled sites by an unbiased, minimum variance linear combination of observations of the phenomenon at neighbouring sites.

The Monte Carlo method has proven to be the simplest to implement and the most intuitive to understand. The accuracy of the results depends mainly on the number of trials the designer can afford. FORM/SORM methods are more interesting as they have a faster performance, and can provide sensitivity information about the stochastic variability of the input variables [38].

The above stochastic methods have advantages and disadvantages. The direct MSC method is robust but requires a large number of draws, which makes it difficult to use for large problems. The perturbation method is more efficient in terms of computation time, but it is limited to problems with low input variability, otherwise it is no longer robust. Furthermore, sensitivity analysis on systems involving a large number of random variables can quickly become computationally expensive. The spectral stochastic method is interesting when the variables are

of Gaussian type but the computational cost increases exponentially with the number of random variables.

7. Complex geometry structure studies with different types of manufacture

There are few examples in the literature that combine the study of complex structures at a representative scale with the estimation of variabilities and their effects on the overall behaviour. In [20], an example of studying more or less complex geometry parts in from a probabilistic point of view is given, where random bending has been applied on a virtual hat-shape composite structure in the context of a multiscale analysis. This example of composite structure was considered numerically where the variations of the maximum bending value were studied according to the values of the material properties. As another example in [39], a multi-objective probabilistic design is presented for reliability and robustness analysis of composite structures on a mono-omega-stringer stiffened panel. The studied approach uses a model of the composite structure while accounting for uncertainties in material properties. The inputs of the model are the omega-stringer geometry and the mechanical properties of the composite material. The mono-stringer length and width as well as the skin and omega-stringer thickness and lay-up are were kept fixed in this study. The stringer foot, the flange and the height were considered as variables along with Young's and shear moduli. No experimental work was conducted. In [40], spring-in behaviour is analysed as a function of time on L-shaped composites. A statistical investigation on manufactured composite samples was conducted for 27 designs of experiments concerning different autoclave cooling rates.

Moreover in [41], an analysis was carried out on the hat stringer-stiffened flat panels to assess their buckling and post-buckling responses when exposed to axial compression. The variability due to the manufacturing process and conditions was not taken into account. The results showed that the hat-stringer flat-panel had a large load capacity after initial buckling in [41]. The overall objective of the work in [41] was to develop cost-effective and robust methods to study the variability of the static or dynamic response of finite element modelled composite structures. The authors took into account uncertain material (moduli of elasticity, Poisson's ratio, density...) and physical (thickness and orientation of fibres) properties on in two academic examples of an ordinary composite plate [11].

8. Our objective and methodology

The objective of the present work is to propose a methodology to take into account, according to a detailed manufacturing process, variabilities in mesoscale including type, dimensions and

positions of plies, and to propose a numerical approach to evaluate the influence of these variabilities on the properties of the structure. Our ambition is to create a family of numerical models representing virtual but realistic parts, with similar complex geometry zones. This realism is due to the amplitude of the simulated variabilities but also the spatial evolution of this variability within the part. Indeed, a realistic model cannot propose discontinuous evolutions of certain properties from one finite element to the next. For example, the rigidity level of the prepreg prevents significant orientation variation over a small distance. Our model must not only provide a numerical twin of real observed parts, but must also extrapolate the information obtained by observing a finite number of real structures to provide an infinite number of future composite structures and over lengths greater than those observed. The goal of our numerical simulations of a large number of realistic parts is to be able to propose variation ranges of the properties and behaviour of the structures according to the variability studied. One of the consequences of such studies could be to refine the knockdown factors of the mechanical properties used in the design of industrial composite structures. Another consequence of this study could be the ability to demonstrate that it is possible, depending on the part, to tolerate a certain amplitude of manufacturing variability without the need for blind knockdown factors.

To be able to gather these realistic input data for the probabilistic models, it is, therefore, necessary to choose mathematical laws describing the spatial variations of the studied variability. In addition to extrapolating parts to be realistic as concerns numerical twins of the observed parts, it is necessary to carry out statistical studies from the analysed quantities, by choosing to perform statistical investigations on the parameters of the mathematical laws identified beforehand. The proposed approach has to be able to use results taken from a few points and extrapolate them to predict variabilities in the entire studied structure, and allow extending future structures with realistic uncertainties. For example, a fuselage contains a family of stringers, which underlines the interest of the variability representation not only within a single part but also for a family of parts. The difficulties that we wish to encounter by this study are: small radii, plies drop-off, different materials, etc. We chose to study a complex geometry structure inspired from aeronautical part that contains the difficulties to be faced in a real industrial part. So, the studied part is not the aim of the study, but an object that contains the difficulties that are themselves the aim of the study.

9. Studied complex geometry structure in this work

The studied part is a composite structure with geometric singularities, presented in Fig. 1. 14. . It is a research Ω -shape stringer of a composite fuselage for aircraft designed within the BLAST

project [42]. It is a representative model inspired by a real composite aeronautical fuselage stringer.

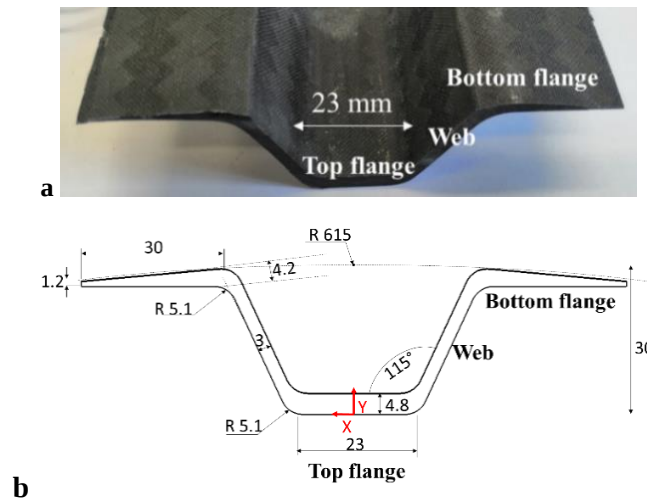


Fig. 1. 14. Presentation of the stringer with main areas of a stringer (Ω -shape) (a) and dimensions (mm) of a part's cross-section (b)

The stringer is taken from an inspired fuselage from a fuselage section manufactured during the ANR Astrid Blast project (cf. Fig. 1. 15.), consisting of 20 stringers, 4 frames and a cylindrical skin.



Fig. 1. 15. Manufactured prototype of composite fuselage from BLAST project [42] (scale 1/3)

A closer look at these components shows the presence of numerous geometric peculiarities such as radius (concave and convex), thickness and material variations on the stringer, and for the frame, changes in thickness, 90° angles, but above all a complex junction zone containing double curvatures.

This studied structure involves industrial difficulties to be faced during the manufacturing process. It is composed of 20 plies, respectively the UD and woven plies of carbon/Epoxy noted above. These raw materials are not dedicated to aeronautical parts but for ground transportation,

and were chosen because they are easier to cure (at 120°C) and cheaper for similar mechanical properties. The stringer is composed mainly of three types of area. These are respectively, a top flange which is the thickest zone, a web and a bottom flange (cf. Fig. 1. 14.). The web contains mainly woven plies since it is under shear stress. The top flange is composed of UD and woven plies since it is under flexural loads in the aeronautical fuselage.

The section of the stringer has the form of an “omega” (cf. Fig. 1. 14.), with different thicknesses in each zone. The top flange’s theoretical thickness is 4.8 mm while it is 3 mm on the web. This thickness variation causes ply drops in the corner between the two areas. The bottom flange has a variable thickness, which starts with 3 mm and ends with 1.2 mm. The thickness at the end of the bottom flange needs to be small due to the form of the skin (circular shape) and to reduce stress concentration in that area. The fabricated sample of stringer is 380 mm long, which is the length that can give enough information about variabilities in a representative sample (chosen according to the required sample sizes). The total width of the stringer’s section is 100 mm. Both the thickness and the nature of the plies is differed from one zone to another. The 5.1 mm radius is the smallest radius allowing the correct lay-up of the structure.

Theoretical dimensions could not be totally obtained in the real manufactured object due to the cut-out process, ply preparation (they can be relatively thin) and type of ply (UD or woven). UD plies can take their places in the mould much more easily contrary to woven ones that can be dilated due to hand force. There exists the possibility to drop some inclusions (such as hair, dust, a small part of protector ply, etc.) and during the polymerisation cycle [6], plies can be shifted from their initial position due to applied pressure and residual stress. Ply thicknesses can also be affected by residual stresses that can change the geometric form of the stringer after curing.

The exploded view of the lay-up (cf. Fig. 1. 16.) shows the estimated locations of plies in the stringer section where ply #1 is the first in the lay-up to take the mould’s shape. Plies #2 and #19 are different from other plies because they have edges in more than one area of the stringer. As shown in fig.3, ply #2 for example has an edge in the top flange, in the web and the bottom flange. Also ply #5, ply #7, ply #14 and ply #16 have edges in more than one area in the web and the bottom flange.

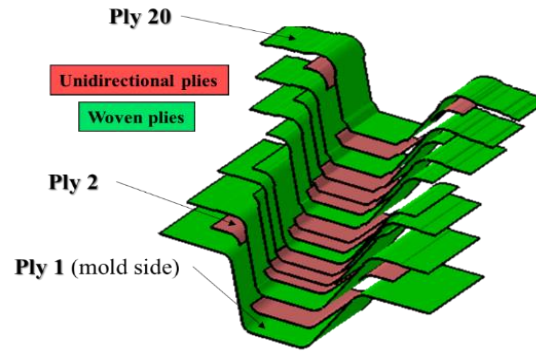


Fig. 1. 16. Exploded view of the stacking sequence inside the Ω -shape stringer designed in the BLAST ANR Astrid project [42] supported by the French Ministry of Defence

In this study, the goal is to compare between the theoretical designed Ω -shape stringer and the real one obtained after the manufacturing process, due to its effect on the structural behaviour detailed previously [16].

Previous descriptions have shown how complicated the stringer's structure is. Our ambition is to extrapolate the information obtained on a family of stringers to be able to use them in other composite structures with similar types of singularities. To achieve this goal, different steps are required during and after the manufacturing process, which are described in the sections here after.

Conclusion

The literature review demonstrates that there is no common methodology for the study of variabilities in composite structures regardless of the evidence that composite materials present a dispersion in their properties. According to this, a methodology is chosen to represent different forms of variability in a specific composite part with complex geometry. This methodology is explained in the next chapter with different observations to be done in order to achieve the work objective.

References

- [1] Lilli M, Rossi E, Tirillò J, Sarasini F, di Fausto L, Valente T, et al. Quantitative multi-scale characterization of single basalt fibres: Insights into strength loss mechanisms after thermal conditioning. *Materials Science and Engineering A*. 2020;797. <https://doi.org/10.1016/j.msea.2020.139963>.
- [2] Mesogitis TS, Skordos AA, Long AC. Uncertainty in the manufacturing of fibrous thermosetting composites: A review. *Composites Part A: Applied Science and Manufacturing*. 2014; 57:67–75. <https://doi.org/10.1016/j.compositesa.2013.11.004>.
- [3] Potter KD. Understanding the origins of defects and variability in composites manufacture. ICCM17. Edinburgh: 2009.
- [4] Zhou XY, Qian SY, Wang NW, Xiong W, Wu WQ. A review on stochastic multiscale analysis for FRP composite structures. *Composite Structures*. 2022;284. <https://doi.org/10.1016/j.compstruct.2021.115132>.
- [5] Gogu C. Mechanical Engineering under Uncertainties. ISBN: 9781789450101, ISTE & Wiley. 2021.
- [6] Potter K, Khan B, Wisnom M, Bell T, Stevens J. Variability, fibre waviness and misalignment in the determination of the properties of composite materials and structures. *Composites Part A: Applied Science and Manufacturing*. 2008;39:1343–54. <https://doi.org/10.1016/j.compositesa.2008.04.016>.
- [7] Garcea SC, Wang Y, Withers PJ. X-ray computed tomography of polymer composites. *Composites Science and Technology*. 2018;156:305–19. <https://doi.org/10.1016/j.compscitech.2017.10.023>.
- [8] Tserpes KI, Stamopoulos AG, Pantelakis SG. A numerical methodology for simulating the mechanical behavior of CFRP laminates containing pores using X-ray computed tomography data. *Composites Part B: Engineering*. 2016;102:122–33. <https://doi.org/10.1016/j.compositesb.2016.07.019>.
- [9] Stamopoulos AG, Tserpes KI, Prucha P, Vavrik D. Evaluation of porosity effects on the mechanical properties of carbon fiber-reinforced plastic unidirectional laminates by X-ray computed tomography and mechanical testing. *Journal of Composite Materials*. 2016;50:2087–98. <https://doi.org/10.1177/0021998315602049>.

- [10] Straumit I, Lomov S v., Wevers M. Quantification of the internal structure and automatic generation of voxel models of textile composites from X-ray computed tomography data. *Composites Part A: Applied Science and Manufacturing* 2015;69:150–8. <https://doi.org/10.1016/j.compositesa.2014.11.016>.
- [11] Davila Y, Crouzeix L, Douchin B, Collombet F, Grunevald Y-H. Identification and Modelling of the In-Plane Reinforcement Orientation Variations in a CFRP Laminate Produced by Manual Lay-Up. *Applied Composite Materials*. 2018;25. <https://doi.org/10.1007/s10443-017-9642-4>.
- [12] Olave M, Vanaerschot A, Lomov S v., Vandepitte D. Internal geometry variability of two woven composites and related variability of the stiffness. *Polymer Composites*. 2012;33:1335–50. <https://doi.org/10.1002/pc.22260>.
- [13] Davila. Y. Etude multi-échelle du couplage matériau-procédé pour l'identification et la modélisation des variabilités au sein d'une structure composite. Université Toulouse 3 Paul Sabatier. 2015.
- [14] Bodaghi M, Lomov S v., Simacek P, Correia NC, Advani SG. On the variability of permeability induced by reinforcement distortions and dual scale flow in liquid composite moulding: A review. *Composites Part A: Applied Science and Manufacturing*. 2019;120:188–210. <https://doi.org/10.1016/j.compositesa.2019.03.004>.
- [15] Skordos AA, Sutcliffe MPF. Stochastic simulation of woven composites forming. *Composites Science and Technology*. 2008;68:283–96. <https://doi.org/10.1016/j.compscitech.2007.01.035>.
- [16] Vanaerschot A, Lomov S, Moens D, Vandepitte D, Leuven KU. Variability in composite materials properties. *Applied Mechanics and Materials*, 807, 23–33. <https://doi.org/10.4028/www.scientific.net/amm.807.23>.
- [17] Zhou XY, Qian SY, Wang NW, Xiong W, Wu WQ. A review on stochastic multiscale analysis for FRP composite structures. *Composite Structures*. 2022;284. <https://doi.org/10.1016/j.compstruct.2021.115132>.
- [18] Potter K, Langer C, Hodgkiss B, Lamb S. Sources of variability in uncured aerospace grade unidirectional carbon fibre epoxy preimpregnate. *Composites Part A: Applied Science and Manufacturing* 2007;38:905–16. <https://doi.org/10.1016/j.compositesa.2006.07.010>.

- [19] Yin Q, Druesne F, Lardeur P. The Certain Generalized Stresses Method for static analysis of multilayered composite plates with variability of material and physical properties. *Composite Structures*. 2016;140:360–8. <https://doi.org/10.1016/j.compstruct.2015.12.070>.
- [20] Rollet Y, Bonnet M, Carrère N, Leroy FH, Maire JF. Improving the reliability of material databases using multiscale approaches. *Composites Science and Technology*. 2009;69:73–80. <https://doi.org/10.1016/j.compscitech.2007.10.049>.
- [21] Shiao MC, Chamis CC. Probabilistic evaluation of fuselage-type composite structures. *Probabilistic Engineering Mechanics* 14. (1999) 179±187.
- [22] Matveev MY, Belnoue JPH, Nixon-Pearson OJ, Ivanov DS, Long AC, Hallett SR, et al. A numerical study of variability in the manufacturing process of thick composite parts. *Composite Structures*. 2019;208:23–32. <https://doi.org/10.1016/j.compstruct.2018.09.092>.
- [23] Davila Y, Crouzeix L, Douchin B, Collombet F, Grunevald Y-H. Spatial Evolution of the Thickness Variations over a CFRP Laminated Structure *Applied Composite Materials*. 2017;24. <https://doi.org/10.1007/s10443-016-9573-5>.
- [24] Omairey SL, Dunning PD, Sriramula S. Multiscale surrogate-based framework for reliability analysis of unidirectional FRP composites. *Composites Part B: Engineering*. 2019;173. <https://doi.org/10.1016/j.compositesb.2019.106925>.
- [25] Sriramula S, Chryssanthopoulos MK. Quantification of uncertainty modelling in stochastic analysis of FRP composites. *Composites Part A: Applied Science and Manufacturing*. 2009;40:1673–84. <https://doi.org/10.1016/j.compositesa.2009.08.020>.
- [26] Lekou DJ, Philippidis TP. Mechanical property variability in FRP laminates and its effect on failure prediction. *Composites Part B: Engineering*. 2008;39:1247–56. <https://doi.org/10.1016/j.compositesb.2008.01.004>.
- [27] *Stochastic Finite Elements: A Spectral Approach*. Rocher G. Ghanem. Pol D. Spanos. Springer-Verlag. 1991.
- [28] Mathilde Grandjacques. Analyse de sensibilité pour des modèles stochastiques à entrées dépendantes : application en énergétique du bâtiment. *Energie électrique*. Université Grenoble Alpes. 2015.
- [29] Christos C. Chamis. *Probabilistic Composite Design*. American Society for Testing and Materials. 1997.

- [30] Sudret B, Berveiller M, Lemaire M. Eléments finis stochastiques en élasticité linéaire. *Comptes Rendus – Mécanique*. 2004;332:531–7. <https://doi.org/10.1016/j.crme.2004.02.024>.
- [31] Lemaire M. Approche pratique des éléments finis stochastiques Variables aléatoires. *Mécanique, Fonctions et composants mécaniques*. 2015.
- [32] Prise en compte de la variabilité dans les calculs par éléments finis des structures composites en régime statique ou vibratoire. Qi Yin. Université de Technologie de Compiègne. 2016.
- [33] Chen X, Qiu Z. A novel uncertainty analysis method for composite structures with mixed uncertainties including random and interval variables. *Composite Structures*. 2018;184:400–10. <https://doi.org/10.1016/j.compstruct.2017.09.068>.
- [34] Stamopoulos AG, Ilio A di. On the Predictive Tools for Assessing the Effect of Manufacturing Defects on the Mechanical Properties of Composite Materials. *Procedia CIRP*, vol. 79, Elsevier B.V.; 2019, p. 563–7. <https://doi.org/10.1016/j.procir.2019.02.108>.
- [35] Dörr D, Brymerski W, Ropers S, Leutz D, Joppich T, Kärger L, et al. A Benchmark Study of Finite Element Codes for Forming Simulation of Thermoplastic UD-Tapes. *Procedia CIRP*, vol. 66, Elsevier B.V.; 2017, p. 101–6. <https://doi.org/10.1016/j.procir.2017.03.223>.
- [36] Thor M, Sause MGR, Hinterhölzl RM. Mechanisms of origin and classification of out-of-plane fiber waviness in composite materials — A review. *Journal of Composites Science* 2020;4. <https://doi.org/10.3390/jcs4030130>.
- [37] di Sciuva M, Lomario D. A comparison between Monte Carlo and FORMs in calculating the reliability of a composite structure. *Composite Structures* 59 (2003) 155–162.
- [38] Yin Q, Druesne F, Lardeur P. The Certain Generalized Stresses Method for static analysis of multilayered composite plates with variability of material and physical properties. *Composite Structures* 2016;140:360–8. <https://doi.org/10.1016/j.compstruct.2015.12.070>.
- [39] Farokhi H, Bacarreza O, Aliabadi MHF. Probabilistic optimisation of mono-stringer composite stiffened panels in post-buckling regime. *Structural and Multidisciplinary Optimization*. 2020. <https://doi.org/10.1007/s00158-020-02565-9>.
- [40] Pereira GC, LeBoulluec P, Lu WT, Yoshida MI, Alves AP, Ávila AF. Spring-back behavior on L-shaped composite structures: A statistical analysis of angular recovery as a

function of time and residual cure. *Composites Part A: Applied Science and Manufacturing* 2019;124. <https://doi.org/10.1016/j.compositesa.2019.105491>.

[41] Mo Y, Ge D, He B. Experiment and optimization of the hat-stringer-stiffened composite panels under axial compression. *Composites Part B: Engineering* 2016;84:285–93. <https://doi.org/10.1016/j.compositesb.2015.08.039>.

[42] Collombet F, Paccou T, Barbarin Y, Crouzeix L, Davila Y, Grunevald Y-H, et al. Vulnerability of Composite Structure Fuselage under Internal Blast Effects: MITE Approach, Melbourne (Australia) : Twenty-Second International Conference on Composite Materials (ICCM22); 2019.

Chapter 2: Experimental setups and protocols for variabilities observation within complex geometries: from raw material to final structure

1. Introduction of the general flowchart for variability observation

The objectives of this study are to focus on the real ply stop, fibre orientation and porosity rates variability within composite structures that present a certain number of implementation difficulties related to the geometry of the part (presence of singularities) and a manufacturing method that generates variability. The difficulties that we wish to address in this study for a manufactured stringer concern the presence of simple concave and convex curvatures of reduced radii, irregular ply stops, thickness variations, a juxtaposition of plies in the context of manual manufacturing without external assistance to the operator as concerns the precise positioning the prepregs inside the mould.

The manufacturing process of the stringers is close to an industrial one, which leads to problems and variabilities presented in an industrial project. The steps described in this section are mainly the same for the manufacture of each stringer. As mentioned in Fig. 2.1, every single step of manufacture is preceded by a type of observation and analysis. Ply dimensions must be measured after the cut-out process to obtain variabilities related to the cut-out process described in detail after this section. After ply cut, and during the lay-up process using a mould, ply edge locations and fibre misalignment are observed using a process presented hereafter. This process was adopted in earlier works too on regular composite structures in [1]. When the stringer is ready, it must be cured according to the polymerisation cycle, but in this step, object observation is delicate, only temperature evolution was chosen to be retained. Finally, after the polymerization process, two different types of analysis are considered respectively, microscopic and tomographic. The tomographic analysis is not a destructive process as microscopic analysis, but in our case, it provides central information about porosity distribution along the stringer. A 3D scan is realised after curing in order to compare the real stringer to the designed one.

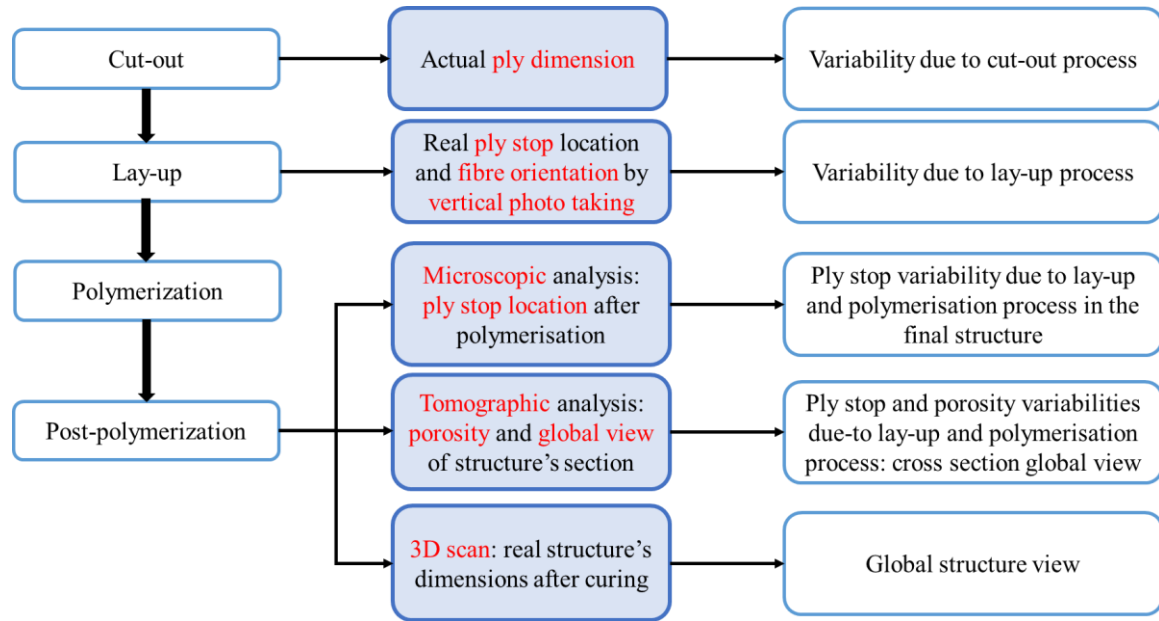


Fig. 2. 1. Step flowchart of experimental work along with observations during and after manufacturing process

2. Stringer manufacture steps before and after curing

This section explains the manufacture steps of the stringers (see Appendix 1 for experimental protocol). In total, four stringers have been manufactured in order to obtain sufficient data to be used in the statistic representation and analysis (presented in the third chapter).

2.1. Ply cut

The first step of manufacture is manual plies cut-out using simple tools. Prepreg rolls types are from Hexcel raw materials, respectively, M79/34%/UD300/CHS for UD and M79/42%/285T2/CHS for woven. The omega shaped stringer with 20 plies laid up in the two different reinforcements.

Table 2.1 presents the number of plies, their orientation, their dimensions and the type of prepreg associated with every ply. The reference frame was chosen in the middle of the top flange, where the x-axis is horizontal (tangent to the first ply) and the y-axis is the perpendicular in direction of the layup (cf. Fig. 2. 2.).

Ply number	Ply orientation	Ply width (mm)	Type of prepreg
2	0	26+2x13	UD
3	± 45	119	woven
4	0	26	UD
5	± 45	2x27	woven
6	0	26	UD
7	± 45	2x42	woven
8	0	26	UD
9	± 45	90	woven
10	0	26	UD
11	0	26	UD
12	± 45	90	woven
13	0	26	UD
14	± 45	2x42	woven
15	0	26	UD
16	± 45	2x29	woven
17	0	26	UD
18	± 45	116	woven
19	0	26+2x13	UD
20	± 45	124	woven

Table 2.1. Definition of plies in the stringer by ply number, ply orientation, prepreg width and prepreg type

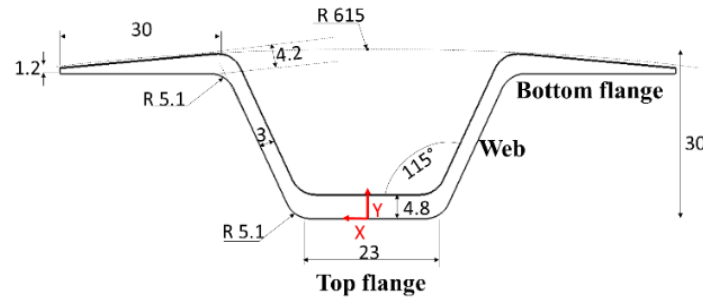


Fig. 2. 2. Studied stringer with coordinate system (x,y at a given z) and different areas names

Prepreg rolls are stored in a cold storage room at $-18\text{ }^{\circ}\text{C}$ temperature and placed at room temperature for 12 to 24 hours before cut process. Ply cut out is a manual process (to create variabilities required for study) using simple tools (ruler, felt pens, square etc.) (cf. Fig. 2. 3.). Rulers and squares can exhibit alignment and measurement variability due to their frequent use and to the operator. The prepreg is cut first by marking the edges using a permanent pen following a ruler. The marking on the roll depends on the user. The dimensions are therefore not exact. There is always measurement variability due to the operator and even to the section of the pen used. Manual cut can cause variability and eventually small undulations ($\pm 2\text{ mm}$). During this work, 4 different operators participated in the manufacturing process.

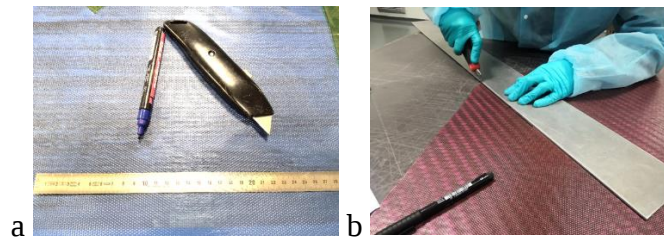


Fig. 2. 3. Drawing and cutting tools of plies with (a) UD ply (with blue protective films) and (b) woven ply (with red protective films)

After the cut-out of all plies, the mould and the counter mould (cf. Fig. 2. 4.) used for lay-up and curing must be covered with a demoulding wax (Cirex Si 041WB) that aids in the structure's removal after curing. The counter mould is a core (see Appendix 2) used during compacting and curing to insure our structure's thickness in the top flange and web areas.

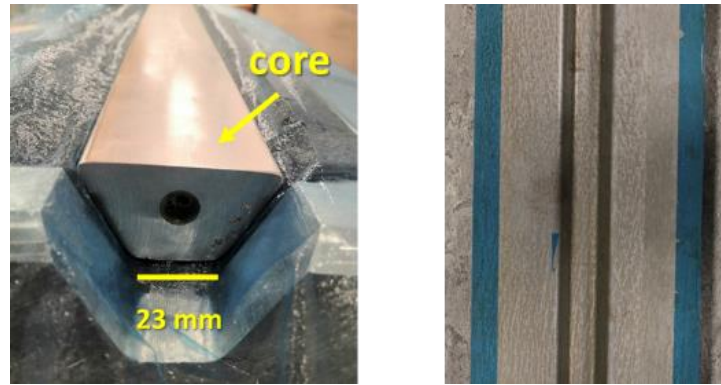


Fig. 2. 4. Mould (right) and counter mould (left) used for lay-up

2.2. Lay-up process presentation, related issues and ply category definition

After cut-out, plies are draped according to their positions relative to the mould as in Table 2.2. Positions of plies depend on the operator from which the variability arises. The compaction is done after the installation of the 1st ply (to take the shape of the mould) then after every 4 plies for about 20 minutes with a vacuum level of approximately 700 mbar. The workpiece is placed under a vacuum bag, separated from the drainage felt by a Teflon separator and a layer of tear cloth. The core is required during compaction and polymerization to obtain the desired thickness. The core also provides the shape of the stringer, which was fixed using two wedges in the stringer's ends (cf. Fig. 2. 5.).

According to Table 2.2, each type of ply has edge location in different stringer's areas. There are plies in the top flange, other in the web and others in the bottom flange. The empty cases in the table means that the considered ply has no edge location in the selected area. The reference frame for measurements is considered in the centre of the stringer cross-section, on the top flange.

Ply number	Theoretical edge locations in the coordinate system (mm)					
	Top flange		Web		Bottom flange	
	X	Y	X	Y	X	Y
1					49.7	26.5
2	12.2	2.6	19.2	23.1	33.6	26.6
3					44.4	26.7
4	12.2	3.31				
5			12.2	3.3	24.7	26.7
6	12.1	3.7				
7			12.1	3.7	39.9	27
8	12	4.21				
9					30.1	27.4
10	11.9	4.9				
11	11.9	5.3				
12					30.5	27.5
13	11.9	5.9				
14			12.01	6.2	40.5	27
15	12	6.24				
16			12.01	6.2	27.6	28.3
17	11.9	6.6				
18					43.7	27
19	11.9	7.3	29.37	28.6	21.9	23.3
20					47.6	26.8

Table 2.2. Ply number and their edge locations in the left half section of the stringer

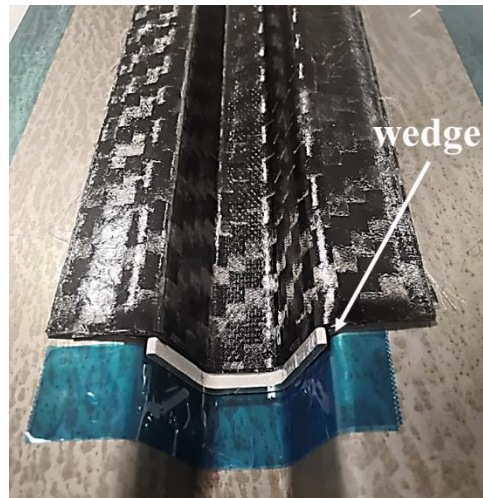


Fig. 2. 5. Wedges used for core fixing

The gap between theoretical and real ply stop locations is different from one ply to another due to its reinforcement nature (UD or woven) on one hand and, on the other hand to its position in the mould. This leads to different manual lay-up difficulties. So according to the difficulty faced during lay-up, plies are divided into different categories for further analysis. This categorising helps in plies stop locations variability analysis. The choice of categories numeration is a priori considering variability difficulties to be found. This classification is not absurd and depends on the chosen criterion.

The 1st category contains woven plies that cover all the stringer's width. Their edges are located in the stringer's bottom flange. This type of plies is difficult to shape in the mould, due to its large width and complex geometry, where it should be placed inside the concave shape and in the right location, leading to two concave shapes and two convex shapes for each ply.

The 2nd category groups woven plies that cover the zone between the web and bottom flange. The placement of this type of plies is delicate because it's situated on a vertical wall with ply junctions on one side and drop off plies on the other side.

The 3rd category includes UD plies that are situated in the stringer's bottom flange. Their stops are designed to be placed in the corner between the bottom flange and the web. These plies have a small width (around 13 mm) and are located in convex zones without visual reference for the operator. So, this category of ply is the most delicate to install, leading to the approximate location. Besides, this category contains a small number of plies (2 plies per side over 20) in each stringer, so it is mandatory for statistical analysis to manufacture several parts. In this

study, four stringers are manufactured thanks to the compromise between statistical needs (see paragraph 3) and the manufacturing cost.

UD plies located in the top flange belong to the 4th category, where their stops are between the top flange and the web. Plies in this category are adjacent to plies from the 2nd category, so either voids or intersections of two plies can be obtained. This type of category shows a delicate positioning of this type of plies, especially in corners. Also, their location in the bottom of the mould makes its lay-up even more difficult. Although the visual symmetry is provided for the operator, the difficulty remains to install plies in the bottom of the mould.

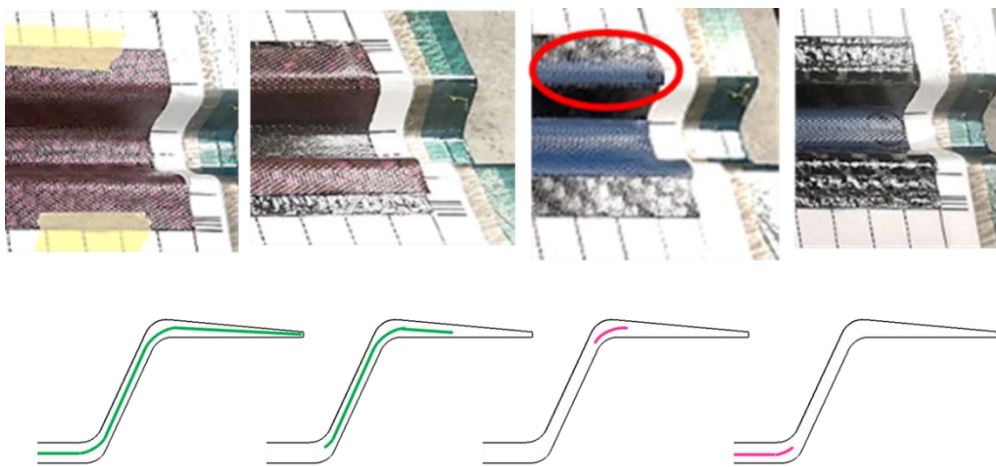


Fig. 2. 6. Comparison between difficulties in four ply categories with categories #1, #2, #3 and #4 successively along with the schematic representation on the stringer cross-section right half

Fig. 2. 6. resumes and compares the four chosen categories and shows the difficulty of the installation of each ply type in the mould. Table 2.3 also resumes each ply category with its specifications. The notations of subscript are respectively, b.flange for “bottom flange” and t.flange for “top flange”. For the 4th category, there is no need to use subscript notation because only the right half section is observed. We have to note that the stringers areas notations were chosen according to the stringer placement in the fuselage, which is upside down the lay-up direction.

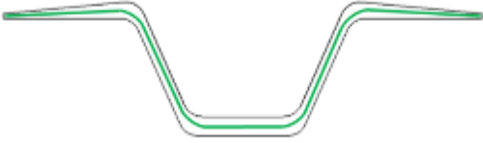
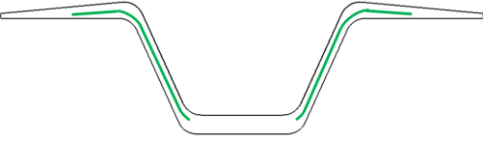
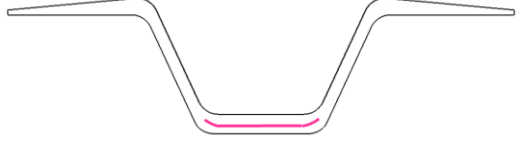
Category	Ply number	Ply orientation	Ply characteristics
1	1, 3, 9, 12, 18, 20	45 °	cover all the stringer's width
			
2	5, 7, 14, 16	45 °	in the web and the bottom flange
			
3	2 _{b.flange} , 19 _{b.flange}	0 °	in the bottom flange
			
4	2 _{t.flange} , 4, 6, 8, 10, 13, 15, 17, 19 _{t.flange}	0 °	in the top flange
			

Table 2.3. Representation of ply categories with (in red) woven plies and (in green) UD plies (according to Table 2.2)

2.3. Vacuum bag preparation

After lay-up, a vacuum bag is needed for polymerisation. It is composed from a peel ply for easy peeling on the inner side only. The peel ply would also have the function of promoting the adhesion of the stringer to the skin, if the stringer were to be used later in the manufacture of a fuselage. We have therefore chosen to keep it here to be representative. The vacuum bag also contains a perforated release film designed to allow outgassing during the curing, a bleeder fabric designed to allow airflow throughout the vacuum bagging process as well as bleed out an excess resin in a composite part and a not perforated release film used as a parting film. All

these are covered by the breather fabric, that serves to provide an air path from the laminate to the vacuum connector.

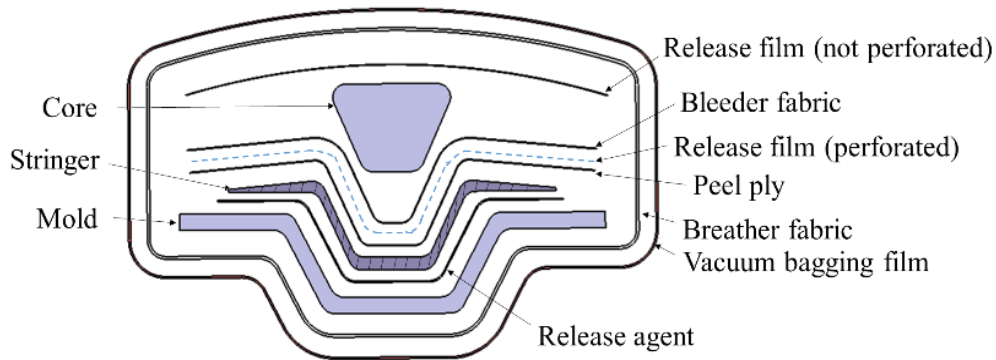


Fig. 2. 7. Scheme of the vacuum bag products (not to scale) used during curing step

Materials used to cover the omega-shape structure while polymerization are chosen according to the stringer's complex geometry (cf. Fig. 2. 7.). As shown, the breather fabric that is responsible for the application of void on all the structure is completely covering the mould. This can reduce pressure on the web and top flange compared to the bottom flange. After the vacuum port and thermocouples' installation (cf. Fig. 2. 8.), the structure is ready to be cured.



Fig. 2. 8. Thermocouples installation for temperature tracking during curing

2.4. Polymerisation cycle

The structure was prepared for the polymerization process (cf. Fig. 2. 9.) using an oven (H529-XXL08). The polymerization cycle begins with an ambient temperature that increases to 120 °C and remains constant for approximately 3 hours. Eventually, the temperature begins to drop

until it returns to room temperature of around 20 °C. The climb speed of temperature is about 2°C/min and the void pressure is up to 700 mbar along the cycle (cf. Fig. 2. 10.).



Fig. 2. 9. Structure covered with vacuum bag and prepared for polymerisation in the oven

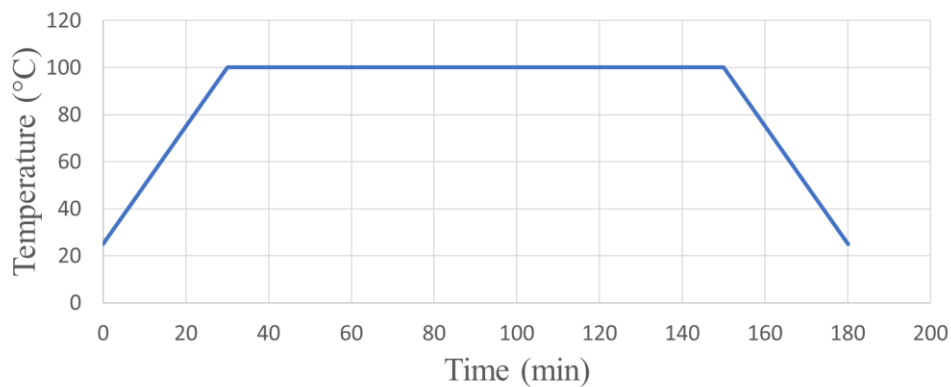


Fig. 2. 10. Theoretical programmed polymerisation cycle

3. Experimental protocol for variability observation for each manufacturing step

Previous descriptions have shown how complicated the stringer's structure is. Our ambition is to extrapolate the information obtained on a family of stringers to be able to use them in other composite structures with similar types of singularities. To achieve this goal, different observation steps are required during and after the manufacturing process, which are described in the section hereafter.

Chapter 2

Observations are performed during each manufacturing step, including raw material dimension observation before lay-up, observation during lay-up process, and after polymerisation. This section explains these steps with their relation to variability analysis for each type.

3.1. Prepreg measurement after cutting

Raw material observation [2] is required to identify gaps between theoretical and real width of all prepreg types in different positions.

Therefore, each cut prepreg is scanned using a simple office scanning machine (Toshiba e.studio 3518a with a resolution up to 600 x 600 dpi). Ply widths are calculated in each location along the prepreg. We chose to use this type of scan due to its simplicity and to reduce the processing time. In addition, by keeping the prepreg flat, this simple technique avoids deformations of the plies during measurement that would distort the results. The measurements are relatively simple, and are done directly in image analysis software (Imagej). The accuracy of the measurements provided was verified by applying the same procedure to objects of known dimensions, and the accuracy was thus assessed as less than 2/10 of a mm.

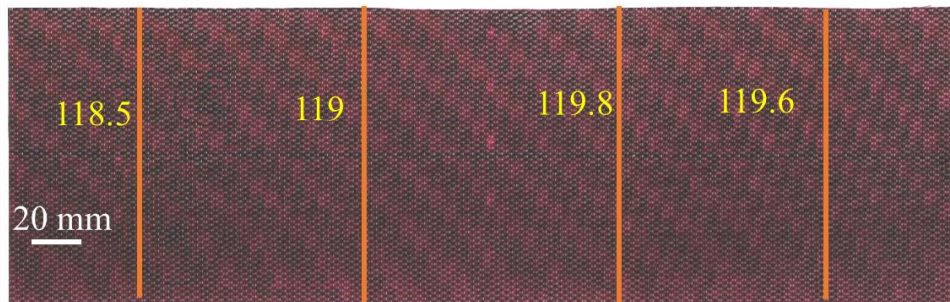


Fig. 2. 11. Example of width measurements for ply#3 (in mm) in different locations along prepreg with 119 mm as theoretical width

An example of a scan of 3rd ply of the first stringer (cf. Fig. 2. 11.) is presented using the scanning machine.



Fig. 2. 12. Example of difference between theoretical (red rectangle) and real prepreg dimensions after cut process for ply#4

Some cut prepregs present waves along the length, as example ply 4 in Fig. 2.12 UD prepregs are easier to cut, but in our case, they present small width. Fig. 2. 13. shows an example of UD prepreg cut for the 2nd ply. As we can see, for the small part (13 mm as width) it is difficult to obtain the exact dimensions.



Fig. 2. 13. Example of difference between theoretical (red rectangle) and real prepreg dimensions after cut stage for ply#2

3.2. Orthogonal optical shooting for ply stops and fibre waviness measurements during lay-up

During the layup process, we have used a grid as a tool that helps by providing reference directions, offering known distances at as many points as possible to be able to correct distortions, etc. for observation of each (cf. Fig. 2. 14) installed on the mould. This process was used in [3] for fibre orientation measurements. The purpose of the grid is to illustrate the measurements linked to the location of a ply extremity and fibre orientation. It allows each ply to be identified in relation to its theoretical location and theoretical orientation. The distance between a vertical axis and another is 2 cm (vertical location of the plies and orientation). The horizontal axis indicates the edge of ply location. The squares in the corners indicate the 45 ° and - 45 ° orientations. The grid (29.7 cm x 42 cm) is printed on an A3 sheet (29.7 cm x 42 cm). Dimensions are checked after printing, and considered accurate to +/- 0.2 mm. The grid was plasticized to protect it.

To install the grid on the mould, we need to cut out first the central part (the stringer's area), then we tried to glue the grid using double-sided tape. The first attempts to glue the grid in one part to the mould were difficult, misplaced and distorted, so we cut the grid into four pieces to facilitate the operation (cf. Fig. 2. 15.). Gluing and cutting by hand can still present measurement and positioning errors (≈ 0.5 mm) which cause the presence of variability in the reference grid compared to the theoretical state.

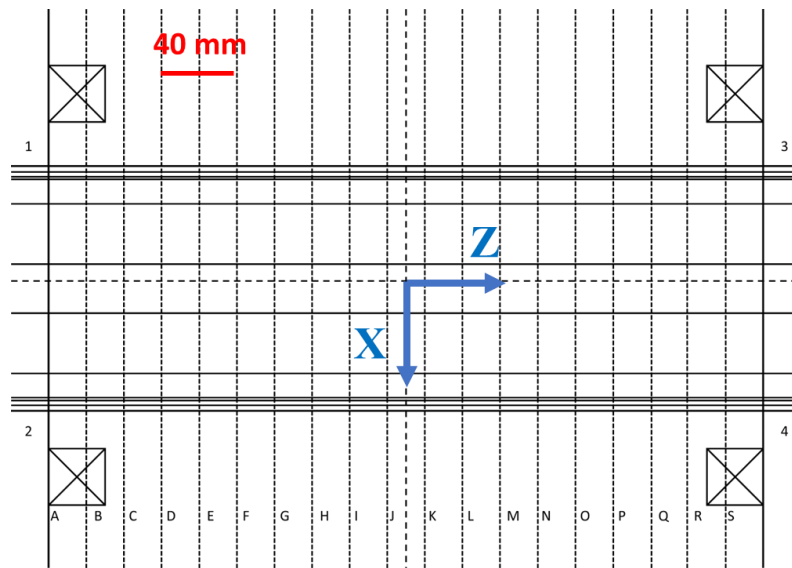


Fig. 2. 14. Grid used in the vertical (y-direction) photographic measurement

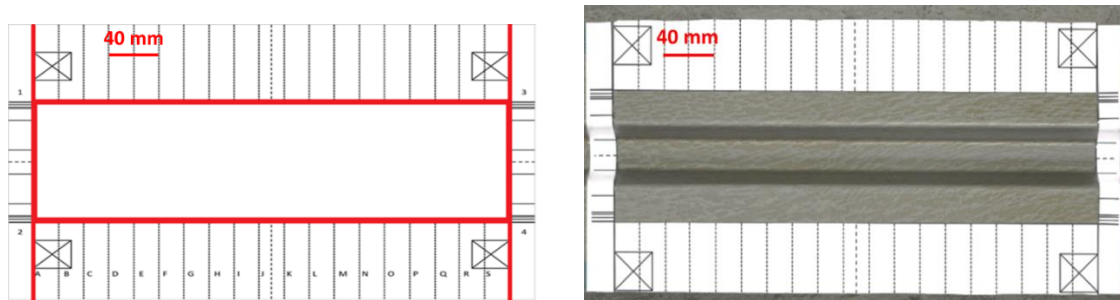


Fig. 2. 15. Cut lines (left) and installed grid on the mould (right)

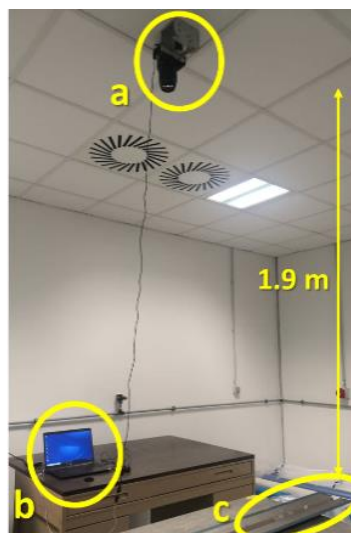


Fig. 2. 16. Camera installation for photo shooting in a clean room with (a) roof camera with focal axis set vertically, (b) computer and (c) lay-up mould

Photos were taken for each ply during layup (example 3rd ply in Fig. 2. 17.) to determine its location. The variabilities in location and orientation, coming from the draping operation, are measured by taking the grid as a reference. A minimum number of 4 photos should be taken after the draping of each ply (draped ply with protectors, ply without protectors, focus on the top flange and focus on the bottom flange). A device (cf. Fig. 2. 16.) is installed in the lay-up room to allow taking pictures of the stringer during lay-up. This device is made of a camera (Canon EOS 550D with a 100 mm lens and 18-megapixel APS-C CMOS sensor and about 0.1 mm uncertainty) which is fixed above the workspace at a distance of 1.9 m by taking care of the light and photo-taking conditions. The camera is placed perpendicularly over the draping table while making the necessary adjustments to obtain better photos. The vertical camera axis is ensured by observing the capture of the photo. The mould must keep a well-defined location in relation to the camera by using schemes placed on the mould support table. The camera is associated with a computer to make the necessary adjustments remotely and to save the photos as well as prevent the camera from moving while making necessary settings.

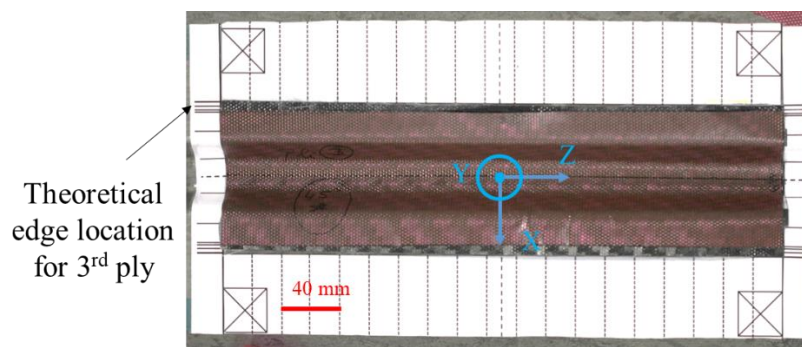


Fig. 2. 17. Measurement of edge ply location before polymerization using the reference grid through the example of 3rd ply while draping before taking off top ply protector

3.3. 3D laser scan for the geometry of the real external shape after curing

After polymerisation, and before the structure cut, we've scanned the stringer using 3D scanning laser device presented in Fig. 2. 18.

3D scanning measurement may be used to observe model deformation defined as a change in the shape of an analysed structure. Measurement scanners are used to capture the object shape and to generate a virtual model. The irregularities in 3D data scans make it necessary to use additional software for data processing. A 3D scanner measurement produces a set of 3D points with high density from the object surfaces in the form of a range image. The depth is recovered

by triangulation. Next, the data are transformed into a geometrical virtual model of the object. The data from 3D scanning have huge density and are independent from the complex shapes of the measured structures. Simplification algorithms of 3D scanner data processing are now being widely developed [5].



Fig. 2. 18. Scanning device used for 3D scan, the hand-held on the left and the scanner on the right

After device installation (cf. Fig. 2. 18.), the scanning area is chosen according to the structure's dimensions and to the scanning plan installation. We start with structure first scan using the hand-held until obtaining the complete shape (cf. Fig. 2. 19.). This step can be repeated several times in order to obtain coherent complete scan. The second step is the plan definition which will define the flat base for the 3D model. The process uncertainty is evaluated at about 0.1 mm.

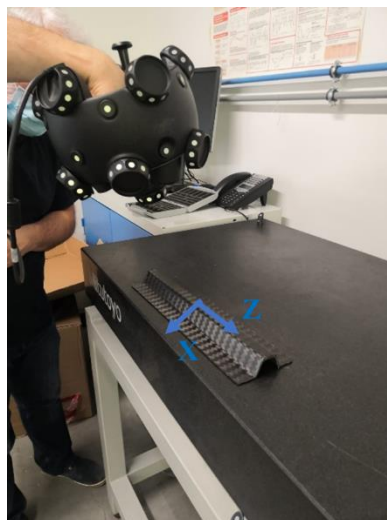


Fig. 2. 19. 3D scan in progress on the scanning plan

The obtained model is presented in Fig. 2. 20.

Taking two sections of the stringer at the two ends (cf. Fig. 2.21), with distance between them about 200 mm, we notice a difference in displacement between them. The structure has been twisted after polymerization process. This torsion corresponds to a delta displacement in the vertical direction y of $Y_A - Y_B = 0.5$ mm.

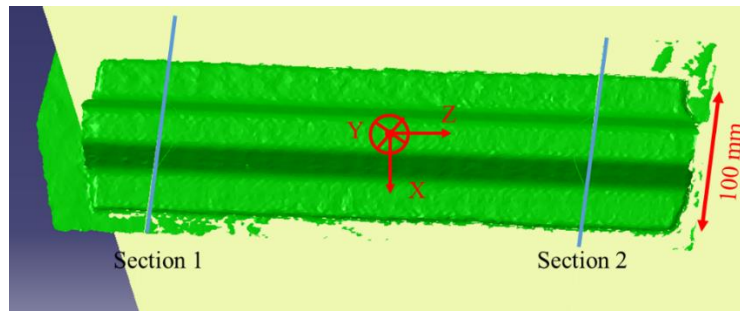


Fig. 2. 20. Example of obtained 3D image for the 3rd manufactured stringer using 3D scan

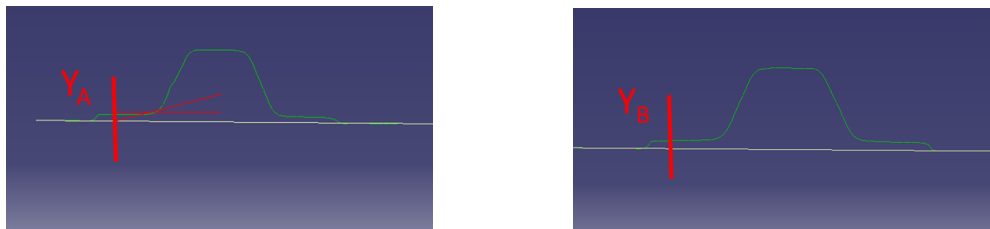


Fig. 2. 21. Extracted sections with difference measure for Y displacement

3.4. Ply stops location observation by cross-section microscopy

3.4.1. Specimen preparation and microscopy observation protocol

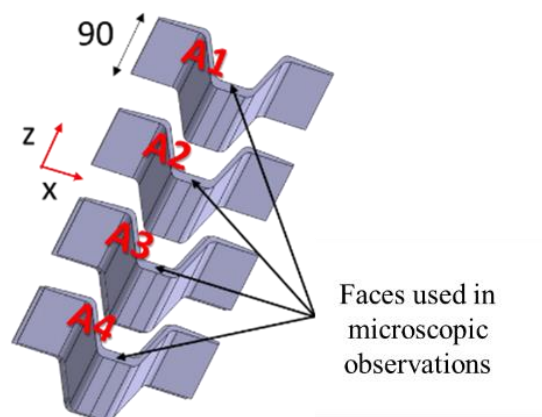


Fig. 2. 22. Cut-out locations for each of four stringer samples

Microscopic observation is a destructive process where the sample must be cut into observation specimens.

As we can see in Fig. 2. 17., the hand lay-up process is not perfect. Manual draping can create different types of variabilities (orientation, ply edge positions. dimensions etc.). In this study as mentioned, our choice was to analyse ply stop location variability. The observation cannot be continuous over the stringer's length, that's why the structure is investigated discretely. Therefore, samples were cut into 4 sections for microscopic observations (cf. Fig. 2. 22.). This choice allows to observe the chosen stringer's length (380 mm), and to extrapolate observations into statistical representation as explained in the sections after. The choice was to observe the right half side of each section related to each sample. This choice minimises the number of observations since assuming that variabilities on the left side are identified through overlapping variabilities found on the left side with variabilities due to the cut-out process. Some 48 observations have been done (i.e. 16 samples, cut into 3 areas per sample).

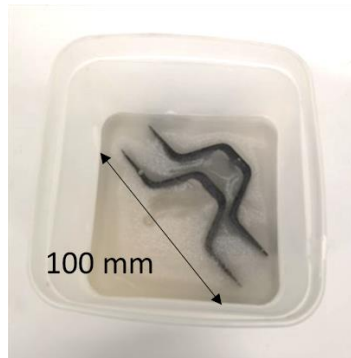


Fig. 2. 23. View of samples with resin coating in preparation for microscopic observation

For microscopic observations, samples are prepared following a specific protocol. First, samples are cut according to the cutting plane using a diamond-sawing machine with a 3 mm blade. Then after deburring them, they are coated in resin (KMU powder and Catalyst) (cf. Fig. 2. 23.). To obtain clear views, samples coated in resin are polished using a METASERV 2000 polishing machine. Since the samples are large (≈ 10 cm), the hand polishing machine was used.

Polishing was done in 6 steps, with abrasive papers ranging from 200 to 1200 and then a switch to diamond solutions from 3 μm to 1 μm , with polishing times adapted to the particularity of the samples, of 5 minutes per step.

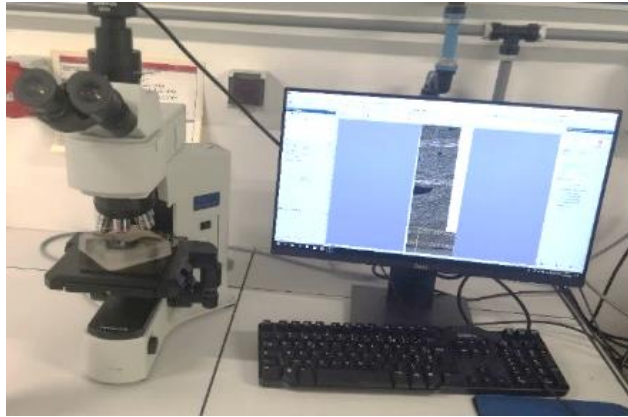


Fig. 2. 24. The microscope Olympus BX41M-LED used for visualisation

The used microscope is an “Olympus BX41M-LED” with the type of camera “Olympus SC50 U-CMAD3” (cf. Fig. 2. 24.). The attached software is “Stream Olympus” which is used for scanning the sample sections. The software’s interface is limited by 50000 pixels both in image’s length and width (about 23 mm) and since the dimensions of the sample section are large, it is impossible to visualize all of it in one image. Therefore, the section is divided into 3 areas with a first area that contains the top flange, the second area that contains the web and the last area containing the bottom flange. For each part, a local frame is created by the interface of microscope software (cf. Fig. 2. 25.). So, before being able to correctly analyse the positions of each ply stop in a global frame linked to the stringer, it will be necessary to reassemble the images and introduce a frame adapted to our study. The first step is to create the bond between different frames in the global reference frame. The analysis method used is associated with the three steps.

We have to note that This work corresponds to the lockdown period, which prohibited access to the laboratory and therefore to correlation software that could have done this work automatically. It was therefore necessary to devise a solution based on simple digital tools.

3.4.2. Image assembly and ply stop identification in the global reference

The first step is to identify two special points (easy to identify, special location) in the common area between each image to obtain a transfer relation of the point coordinates from one image to another (in Fig. 2. 25., the couples of points A1, A2 and B1, B2 for example). These points are chosen to be distant from each other in the overlap area to limit the impact of repositioning errors. To check that the different images have the same scale, it was necessary to check the equidistance between each couple of points in the initial pictures. The following stage is the

determination of the angle of rotation between image #2 and image #1 and the angle of rotation between image #3 and image #2, by comparing the direction of the vector pairs on each image. This method allows to obtain rotation angles with observation uncertainty of about 1° . The definition of the global reference's directions is necessary to obtain specific locations. The assumption followed here was to define the x-direction of the global reference frame tangent to the upper skin of the top flange, located against the mould during manufacturing. After that, pictures are rotated in the horizontal position of the global coordinate system and images #2 and #3 are translated to rebuild the entire stringer's section. The translation uncertainty is about 0.1 mm due to errors made while taking information. The aim is to obtain the coordinates of all the nodes in images #2 and #3 in the frame of image #1.

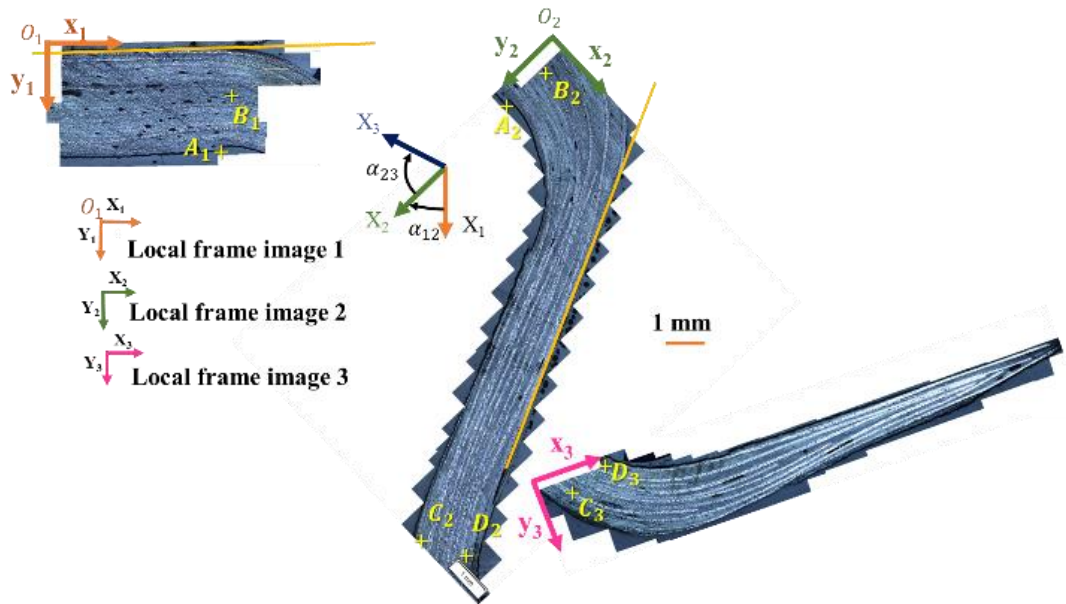


Fig. 2. 25. Definition of local coordinate system related to each image

The following stage is therefore to identify the origin of the global coordinate system. Since the exact location of the global frame origin is not known in microscopic images, tangent lines to the top flange and the web are drawn for each section. Then, after the calculation of equations of two tangent lines (thanks to the local references of the graphic interface), the coordinates of their point of intersection I is identified (which is also known in the theoretical model). The distance between this point and the chosen origin is given by the theoretical geometry of the mould, which is here 11.5 mm (the real mould geometry was verified). This allows determining the coordinates of the origin O (cf. Fig. 2. 26.).

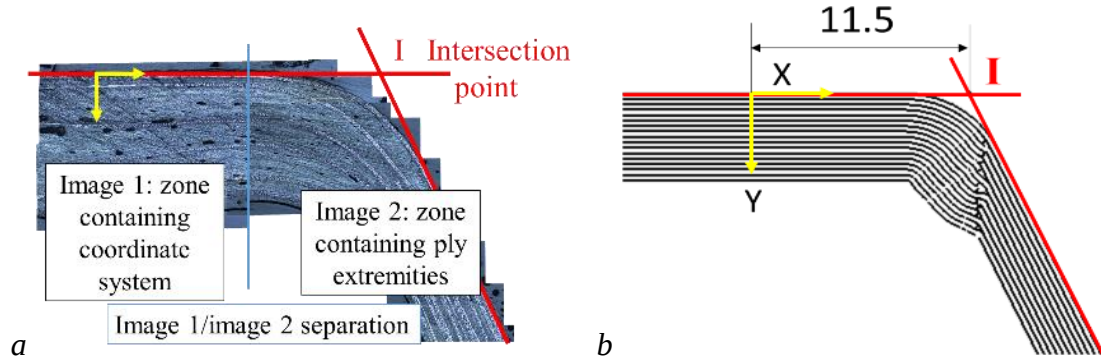


Fig. 2. 26. Definition of a global coordinate system (reference frame) with (a) intersection point definition and (b) coordinate system origin definition

At this stage, the image of the complete stringer's section can finally be reconstructed in a global reference frame that can be used for measurements (cf. Fig. 2. 27).

The last step is to determine the coordinates of ply stop locations in the global frame. We have chosen not to work on the reconstructed global image as shown in Fig. 2. 27, but to identify the ply stops in the local frame of each image, and to transfer these coordinates to the global frame using a transfer function from the previous steps. This choice is linked to the fact that the global image is too heavy to be easily manipulated. The transfer function is the sum of a rotation and a translation. The parameters come from steps #1 and #2 of the process (see Eq 2.1). P1 is the initial position of the studied point in the local frame of the image. P2 is the location of the same point in the global frame of the stringer. R and T are respectively, the rotational matrix from step #1 and the translation vector from the second step of the process.

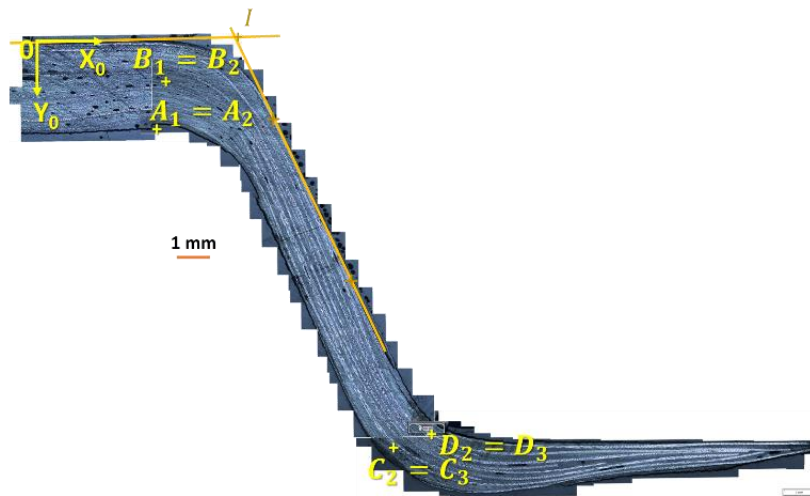


Fig. 2. 27. Transformation of stringer cross-section after translation of images

The transformation equation is a rotation plus a translation as in Eq 2.1.

$$P_2 = RP_1 + T \quad \text{Eq 2.1}$$

where R and T are respectively, the rotation matrix and the translation vector along with P_2 the final position and P_1 the initial position.

After these steps, the location of ply edges in the global reference system are identified, to compare them with theoretical ones.

The obtained images from the microscope for observed zones introduce the presence of porosities, ply edges in different parts of the stringer's sections, different ply thicknesses etc... From these observations, several types of variability could be studied, but the point of interest is ply edge locations. A comparison between a lay-up theoretical model and the images obtained is realised based on coordinates of each ply location in a global reference system. For example, Fig. 2. 28. and Fig. 2. 29. show the gap between theoretical ply edge locations and obtained ones after polymerization, in the corner between the top flange and web (for sample A1) and the bottom flange for the same sample.

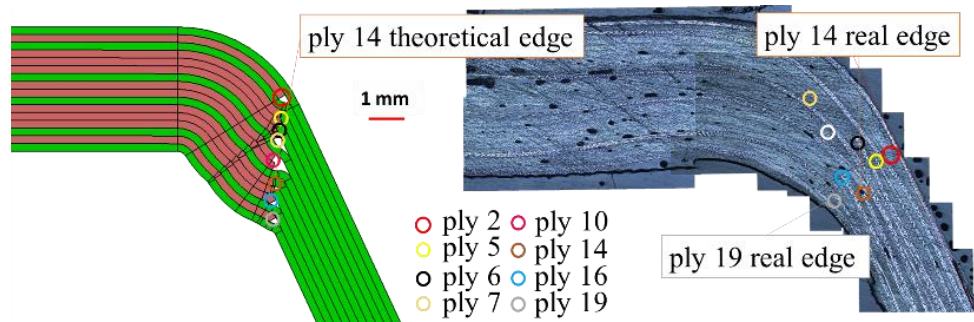


Fig. 2. 28. Comparison between theoretical (designed) and real (after polymerization) locations of ply edges of sample A1 (see Fig. 2. 22.) in the corner of top flange and web using microscopic analysis

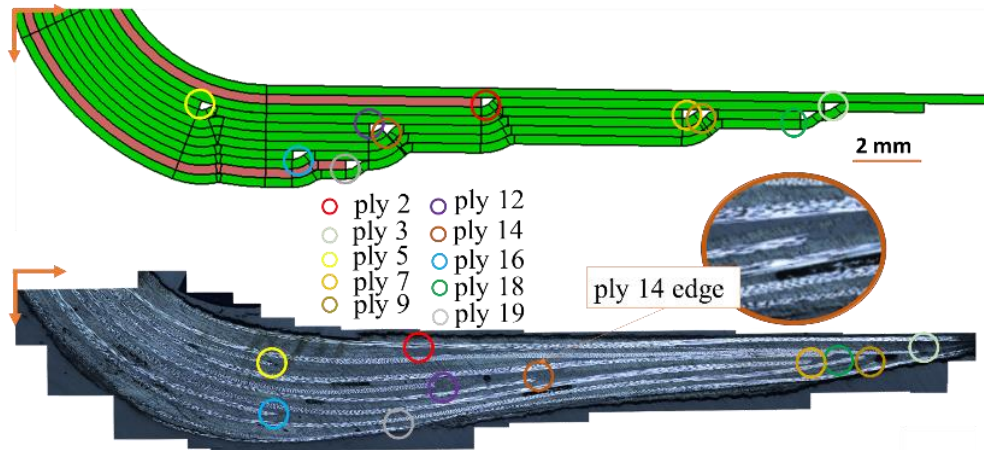


Fig. 2. 29. Comparison between theoretical (designed) and real (after polymerization) locations of ply edges of sample A1 (see Fig. 2. 22.) in the corner of the bottom flange and web using microscopic analysis

The measurement of ply edge's locations is not 100 % certain because ply edge is not a single point, so there is always an uncertainty evaluated to 0.04 mm for each location (cf. Fig. 2. 30.).

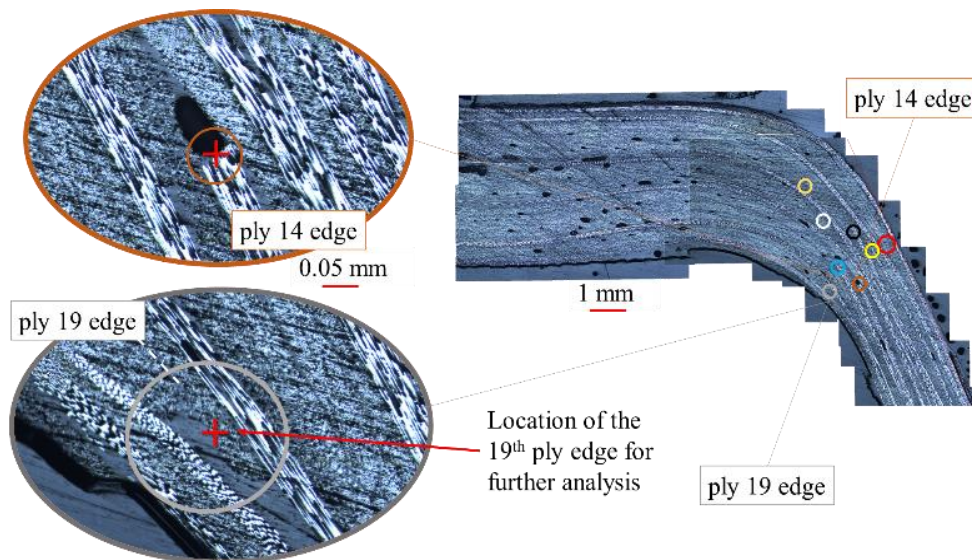


Fig. 2. 30. Uncertainty of measurements on edge locations of plies with 100 pixels ~ 0.04 mm

These different steps are applied on a finite number of stringers. The utility of these measurements is to create mathematical approaches that will be used in numerical models, to represent ply stop variability and eventually porosity rates.

Microscopic observations of the four stringers show the difference between each structure despite the fact that they were manufactured using the same process. Here after an example of

comparison in the coin between the top flange and the web for the ply#5 stop location (cf. Fig. 2. 31.).

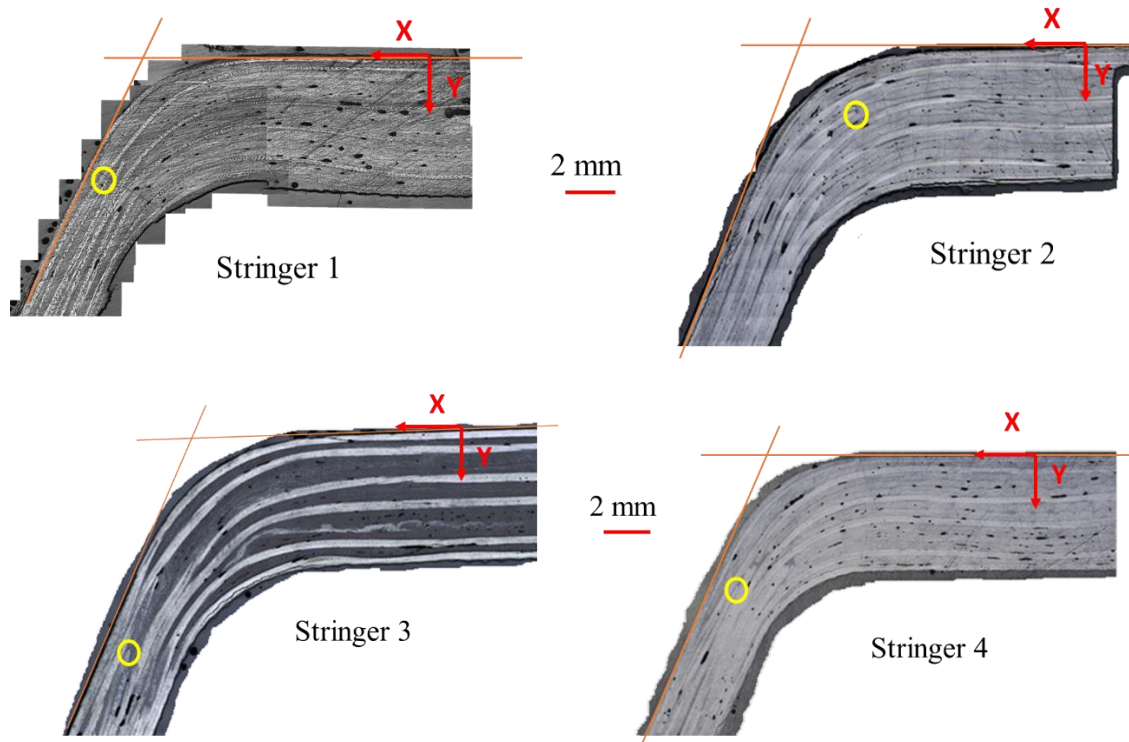


Fig. 2. 31. Comparison of the ply#5 stop location in the four manufactured stringers in the corner between the web and the top flange areas in the same location (z)

The distance between the reference frame origin and the ply #5 stop location is for each stringer respectively, 5.94 mm for the first stringer, 4.14 mm for the second stringer, 7.52 mm for the third stringer and 6.1 mm for the fourth stringer.

For conclusion, although the same manufacturing process used for all the structures, there is differences between each one. One of our goals is to identify this difference and predict it for the future structures.

3.5. 3D X-ray tomography for porosity observation and global cross-section view

The following sections outline the analytic methods used to measure porosity concentration using various techniques.

Destructive methods for the measurement of porosity are commonly conducted in the literature, however, non-destructive techniques were employed to gain further confidence in

measurements [4]. Also, for us, we wanted an assessment that was as local as possible. Destructive methods such as dissolution only provide an average value in the volume studied.

X-ray Computed Tomography, or CT, scans were conducted on pre-impact specimens. The analytic method for calculating porosity from CT scans is identical to the method outlined in equation here after. The Defect Analysis feature was used to create a threshold corresponding to voxel intensity. All voxels with an intensity below a threshold were summed and divided by the total number of voxels (cf. Eq. 2.2):

$$\%V_v = \frac{\sum Voxel_{I>T}}{\sum Voxel} \quad \text{Eq. 2.2}$$

In situations where destructive testing is not suitable for porosity determination, the experimenter must turn to non-destructive testing to determine porosity concentration. X-ray computed tomography uses scattered X-rays to reconstruct a 3-dimensional model of the sample. Commercial software is available to calculate and perform defect analysis by thresholding voxels [4].

As quality verification, porosity concentration was measured using X-ray computed tomography (CT) scanning. This allows the experimenter to estimate porosity without damaging and compromising the integrity of the material.

The X-ray output from the X-ray source is not mono-energetic. Different materials and thicknesses of those materials were used to filter out lower energy spectrums to get more useful X-rays and reduce scatter (lower energy X-rays that are not strong enough to penetrate the part can only add scatter). Frame averaging is to reduce static noise; any imperfections in the scintillator or the detector pixels can be averaged out by using frame averaging [4]. The number of projections (samples) is the number of images taken through the 360-degree rotation. The more projections, the more accurate the 3-dimensional representation. This number is generally governed by the Nyquist sampling criteria, but under sampling can be used in some cases where detectability is more important than accuracy and time is critical. Keeping the number of projections high reduces artifacts, producing a more realistic 3-dimensional image.

The Defect Analysis Toolbox works by setting a threshold that distinguishes voids from the material. All voxels below the threshold intensity were summed and divided by the total number of voxels [4].

For our case, the parameters used to visualise the studied structure are presented here after. X-Ray 3D tomography images were taken using a Micro-Tomography EasyTom 130 machine, manufactured by RX Solutions, France. The sample was glued on a wooden support and placed on the plate of tomography and fixed with hot glue (cf. Fig. 2. 32.). The source has a voxel size of $18\text{ }\mu\text{m}$. Each specimen was scanned through a 360° rotation using a Varian PaxScan 1313DX imager to capture layer-by-layer 2D X-ray images used for full-scale 3D reconstruction. RX Solutions X-Act 2.0 software was used for 3D reconstruction and post-processing. Due to the samples size and dimensions, the maximum possible and workable resolution was $38,2\text{ }\mu\text{m}$. The source-object distances (sod) and source-detector distances (sdd) were 208.85 mm and 692.87 mm respectively, which determine the magnification (sdd/sod) at 3,31. The X-ray voltage and current were set to 80 kV and 110 mA respectively. Every sample was scanned for 4 hours with shift x2 and 6 pass acquisition (and about 2 hours for reconstruction and post-processing). The obtained images are presented in the next section.

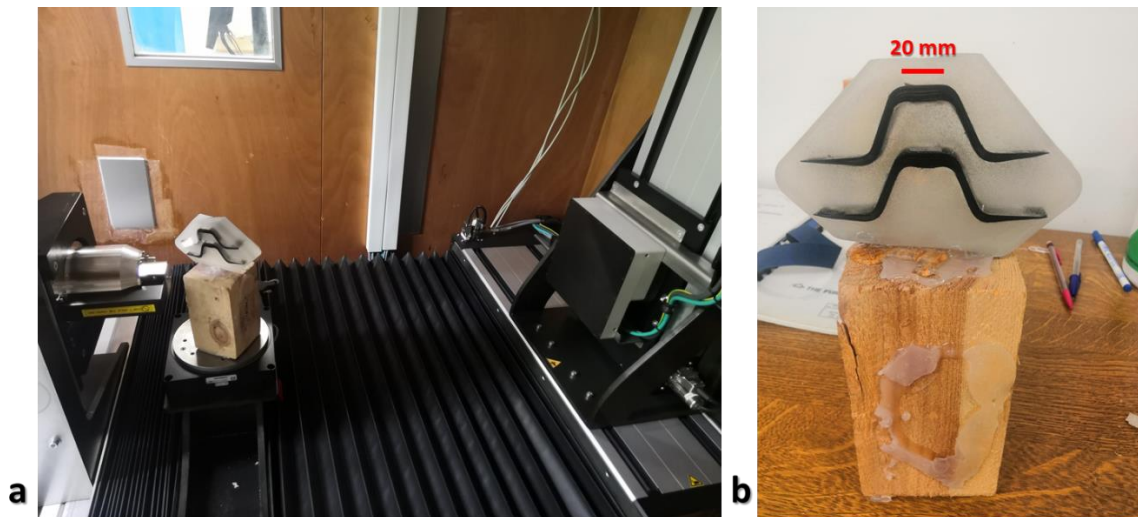


Fig. 2. 32. Sample's section preparation for tomographic visualisation with (a) sample inside tomography and (b) sample glued on wooden support

We present here after examples of obtained images in a cross section of the composite stringer. The image shows mainly porosity distribution all over the section.

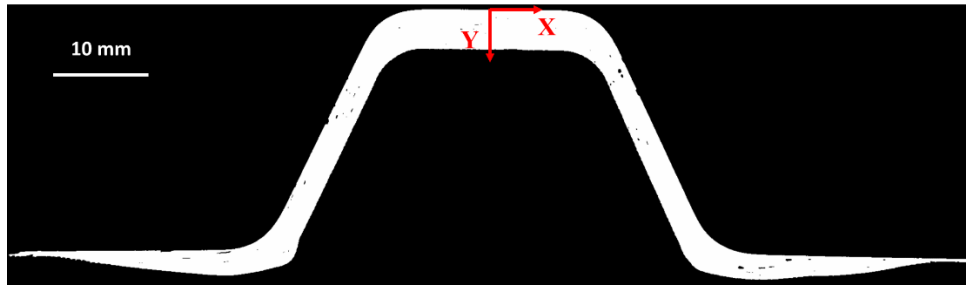


Fig. 2. 33. Example of obtained image for the stringer's cross section (treated with Imagej)

3.6. Curing phase with temperature monitoring and resin flow analysis

During polymerization cycle, although the possibility to observe the evolution of the structure (deformation, variabilities ...) along temperature using Bragg grating optical fibres for example, our choice was to save only the temperature evolution along cycle time.

Temperature tracking along the polymerisation cycle gave the graph represented in the Fig. 2. 34. As we can see, the cycle is slightly different from programmed cycle due to environment conditions and the oven itself. The chosen maximum temperature was attended.

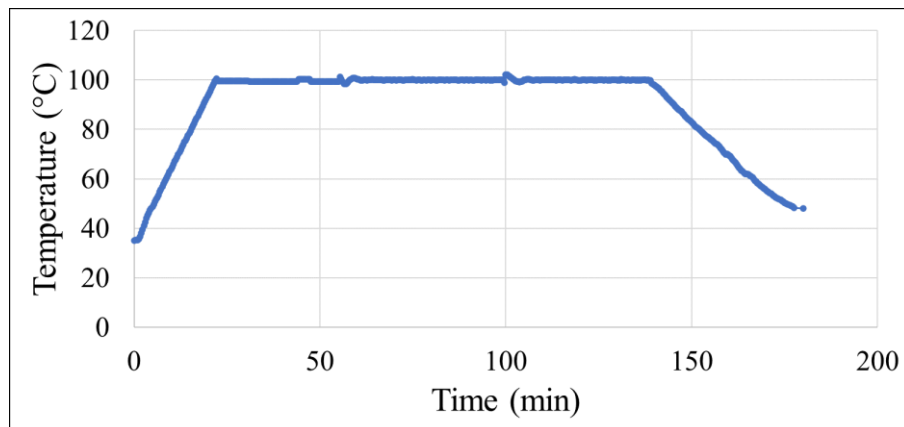


Fig. 2. 34. Real obtained polymerisation cycle recorded by the thermocouples installed between the structure and the oven

It is possible also to quantify the resin flow by measuring the mass of the vacuum bag products before and after curing.

The cure process involves the application of heat and pressure. The applied heat increases the temperature in the composite, resulting in changes in the molecular structure of the resin and,

correspondingly, in resin viscosity. When the resin viscosity has become sufficiently low, a pressure is applied to the system squeezing resin from the composite into the bleeder [6].

The excess of resin is absorbed mostly by the bleeder fabric. The resin flow rate has an effect on the presence of porosities in the structure. This effect is not considered in this study.

Conclusion

The experimental protocol used and the variability observations for the analysis of ply stops, porosity and fibre misalignment were presented in this chapter. The observations were carried out throughout the manufacturing process, using a wide range of measurement techniques such as optical, micrographic, tomographic observations, etc. after the structure's polymerization, it was scanned in 3D to identify its real geometry, which is different from the theoretical geometry due to the presence of internal stresses. The fabricated part is then cut out for observation. Microscopic and tomographic analyses were conducted to determine the variations of the studied parameters, mainly the ply stop locations, the fibre misalignment and the size and the extent of the porosities. All the observations done during every manufacturing step serve to obtain a lot of data sets from raw material to complete structure analysis. In the lay-up phase (see Appendix 3), a minimum number of 384 measure was taken using microscopic observation. The observation and measurement time was up to 2 hours per cross-section. Tomographic observation takes about 6 hours per sample. The objective for the next chapter is how to illustrate these data sets by choosing an analysis strategy to put in place. Also, obtained data are punctual with need for interpolation using mathematical tendencies related to each variability type, plies stop locations, fibre misalignment and porosity rates.

References

- [1] Davila Y, Crouzeix L, Douchin B, Collombet F, Grunevald Y-H, Rocher N. Variability in Monolithic Composite Parts: From Data Collection to FE Analysis, 2018, p. 75–97. https://doi.org/10.1007/978-981-13-2417-8_3.
- [2] Olave M, Vanaerschot A, Lomov S v., Vandepitte D. Internal geometry variability of two woven composites and related variability of the stiffness. *Polymer Composites* 2012 ;33 :1335–50. <https://doi.org/10.1002/pc.22260>.
- [3] Davila Y, Crouzeix L, Douchin B, Collombet F, Grunevald Y-H. Identification and Modelling of the In-Plane Reinforcement Orientation Variations in a CFRP Laminate Produced by Manual Lay-Up. *Applied Composite Materials* 2018;25. <https://doi.org/10.1007/s10443-017-9642-4>.
- [4] Angeles L. Mechanistic Effects of Porosity on Structural Composite Materials. UCLA Electronic Theses and Dissertations. 2014.
- [5] Bondyra A, Chwał M, Pastuszak PD, Stawiarski A. Analysis of composite structure deformations based on 3d laser scanner measurements. *Composite theory and practice*. 14: 1 (2014) 38-42.
- [6] Springer GS. Resin Flow During the Cure of Fiber Reinforced Composites. *Composite Materials*, Vol. 16 (September 1982), p 400.

Chapter 3: Spatial evolution of variability within a complex geometry structure: Mathematical laws and statistical trends

1. Introduction and problematic statement from experimental big data to exploitation of variability into numerical modelling

Starting from different manufacture steps in the previous chapter, we can extrapolate many forms of data related to each chosen type of variability to study, ply stop location, fibre misalignment and porosity rates. The main source of these types of variability comes from manufacturing process of the omega shaped stringer especially since it is chosen to be hand made. In this third chapter, we are going to analyse the observations realised during the manufacturing process to transform them into mathematical and statistical tendencies in order to use them later to create our numerical modelling that takes into account different types of variabilities. We are going to start with presenting analysis related to ply stop variability, then those related to fibre orientation and finally porosity rate uncertainty. All the gathered data will be transferred into mathematical tendencies, as polynomial for ply stop variability, based on a model from literature for fibre misalignment and normal gaussian representation for void distribution in the stringer's section and length depending on the porosity shape.

2. Ply stops variability mathematical representation

Following from the manufacturing process presented earlier, various stringers were manufactured. This is essential, due to a lack of information for statistical study if only one single stringer was studied. For one stringer, considering ply stop variability analysis after polymerisation process, statistical tendencies for each ply category are based on respectively, 6 measurements for the 1st category, 8 measurements for the 2nd category, 4 measurements for the 3rd category and 10 measurements for the 4th category, all per cross-section per stringer (see previous chapter for categories definition). These measurements, especially those from the 2nd category, are not sufficient to have an adequate statistical representation for gaps. So, to enrich statistical tendencies and to obtain more recurrent results related to a finite number of composite structures, four stringers have been manufactured, which lead to obtaining from 16 sets of data for the least populated category (3rd) up to 40 sets of data for the most populated one. This choice is a compromise between statistical needs and manufacturing complexity.

In this section, the goal is to identify the difference between the theoretical positions of the end of each ply and the real position obtained after the manufacturing process, due to its effect on the structural behaviour detailed previously (cf. Fig. 3.1).

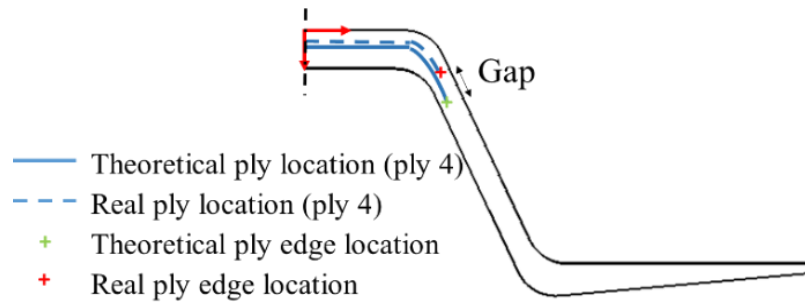


Fig. 3. 1. Schematic representation of gap between theoretical and real ply stops in an Ω -shape stringer cross-section

For further analysis where for each stringer, where discrete measurements are taken, 4 essential locations are considered according to Fig. 3. 2. This choice was due to its simplicity for information saving and to statistical needs for variability representation. The origin is considered in the structure's centre and observation locations are symmetric according to this origin.

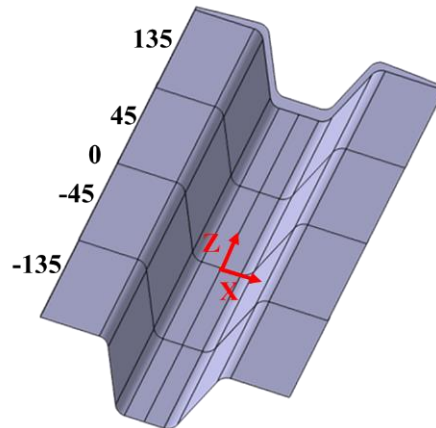


Fig. 3. 2. Measurement locations along Ω -shape stringer's length z

2.1. Gaps between real and theoretical prepreg width generated by cut process

Cut defects are due to the imperfection of the equipment used (ruler set square not perfectly straight), and to the prepreg nature itself (UD or woven), which causes millimetric deviations from the theoretical width of each ply.

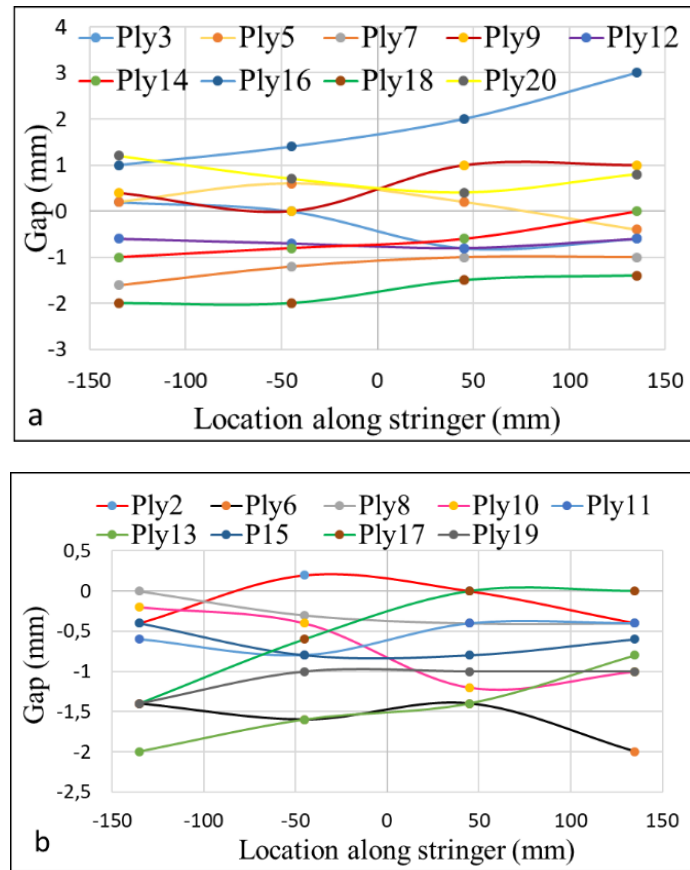


Fig. 3. 3. The gap between theoretical and cut out width of different ply categories with (a) 1st category woven and (b) 2nd category UD

Fig. 3. 3. represents results obtained from Ω -shape stringers to show differences of gaps between each category of plies. The reference frame for measurements is considered in the centre of the prepreg. According to these results, there is variability due to cut out in almost all plies but with different levels (depending on reinforcement type, UD or woven, width and cutting conditions).

Observations show waviness in the shape of the ply cut at long wavelengths (decimetre). To be able to represent these undulations mathematically, as implemented in many works in the literature [1] [2], the selected mathematical model is polynomial. Because the number of measurement points along the stringer is 4, especially for micrographic observations, we have oriented ourselves towards polynomials of degree 5, which allow both an interpolation of the data between the 4 measurement points along the stringer, and extrapolation of the data for longer stringers. Indeed, for long stringers, a repetition of the observed pattern over 380 mm is foreseen, a length considered as representative of the observed variabilities. To guarantee the derivability of the interpolated pattern, assumptions of horizontal tangents on the stringer's

extremities have been applied. This choice is a compromise between the accuracy of the interpolation on a few mm preceding the extremities and the numerical displayed ambitions.

The mathematical model selected to represent the spatial evolution of the gaps $C(z)$ along the stringer compared to the expected nominal value is a polynomial represented in Eq.3.1:

$$C(z) = \sum_{i=0}^5 c_i z^i \quad \text{Eq. 3.1}$$

where $(c_0, c_1, c_2, c_3, c_4 \text{ and } c_5)$ are identified from experimental analysis after cutting, c_0 being the average deviation at the centre (mm), c_1 being the average inclination while c_2 (mm^{-1}), c_3 (mm^{-2}), c_4 (mm^{-3}) and c_5 (mm^{-4}) define the inflection points. The coordinate $z = 0$ corresponds to the stringer geometric centre. The choice here was to divide plies into 2 different categories for further analysis according to their nature, 1st category for woven plies and 2nd category for UD plies. Table 3.1 represents means and standard deviations for each polynomial parameter after cut process for each ply category (Woven and UD).

	Coefficient	c_5	c_4	c_3	c_2	c_1	c_0
1 st category	Standard deviation	1.4E-11	1E-09	5.4E-07	3.9E -05	0.007	1.033
	Mean	-2.5E-12	-3E-10	5.9E-08	1.3E -05	0.001	-0.197
2 nd category	Standard deviation	1.0E-11	1E-09	3.5E-07	3.6E-05	0.004	0.58
	Mean	-8E-13	6E-10	2.7E-08	-2.2E-05	-5E-05	-0.58

Table 3.1. Standard deviation and mean for coefficients of polynomials after cut-out process obtained from the 4-stringer analysis

Coefficients are identified through the 4 crossing points of polynomial representation plus two tangent lines applied on the left and right sides of each gap.

For plies of the 1st category (Fig. 3. 3.), gaps are between -2 mm and 1 mm which is thus natural since the cut is a manual process. But compared to the 2nd category (between -1.6 and 3 mm), gaps related to the 1st category are less important, due to the form of plies in each category (the 1st one corresponds to large woven plies and the 2nd one to UD ones).

Taking c_0 as variability indicator that represents the average deviation at the centre, observation results show that woven plies represent less variation mean after cut-out step with c_0 parameter mean equal to -0.197 mm while being about 0.58 mm for UD plies, but standard deviation for

woven preregs is higher than the one associated to UD preregs, which indicates larger variation ranges. These results are due to prepreg specifications and dimensions.

2.2. Ply stops location obtained during lay-up process

Due to the manual process of draping, there is a gap between the real and theoretical location of plies edges. This gap is different according to each category of ply. Here the obtained results from stringer 1 are represented as an example (cf. Fig. 3. 4.). The reference frame for measurements ($z=0$) is considered in the centre of the stringer length.

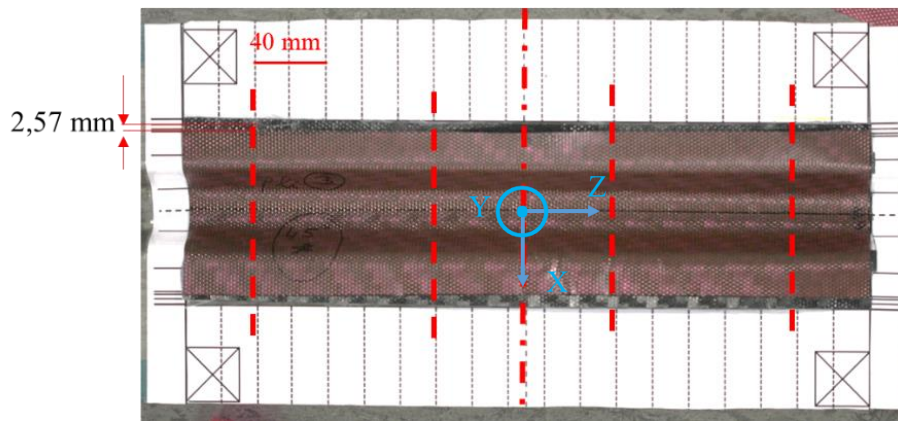


Fig. 3. 4. Example of gap measurement between real ply locations and theoretical locations while draping using the reference grid at different locations (ply #3)

For this sample in this position, the gap of the real position of ply #3 (theoretical width 119 mm) compared to the theoretical one is due to several factors as the cutting gap of 29.5 % (0.5 mm), layup gap of 67.8 % (2.5 mm). The remaining gap (2.7 % $\approx$ 0.07 mm) could be due to slippage of the ply during curing or to measurement errors (see § 2.3). After measures were made on different plies, the gap due to cutting was between 16 % and 30 % while the gap due to lay up was between 20 % and 65 % depending on the nature of ply. The remaining gap, which could be due to the curing process, is between 0 % and 15 % and is neglected compared to other sources.

2.3. Ply stops location after polymerization phase

Obtained graphs after observation are represented in Fig. 3. 5. for each category.

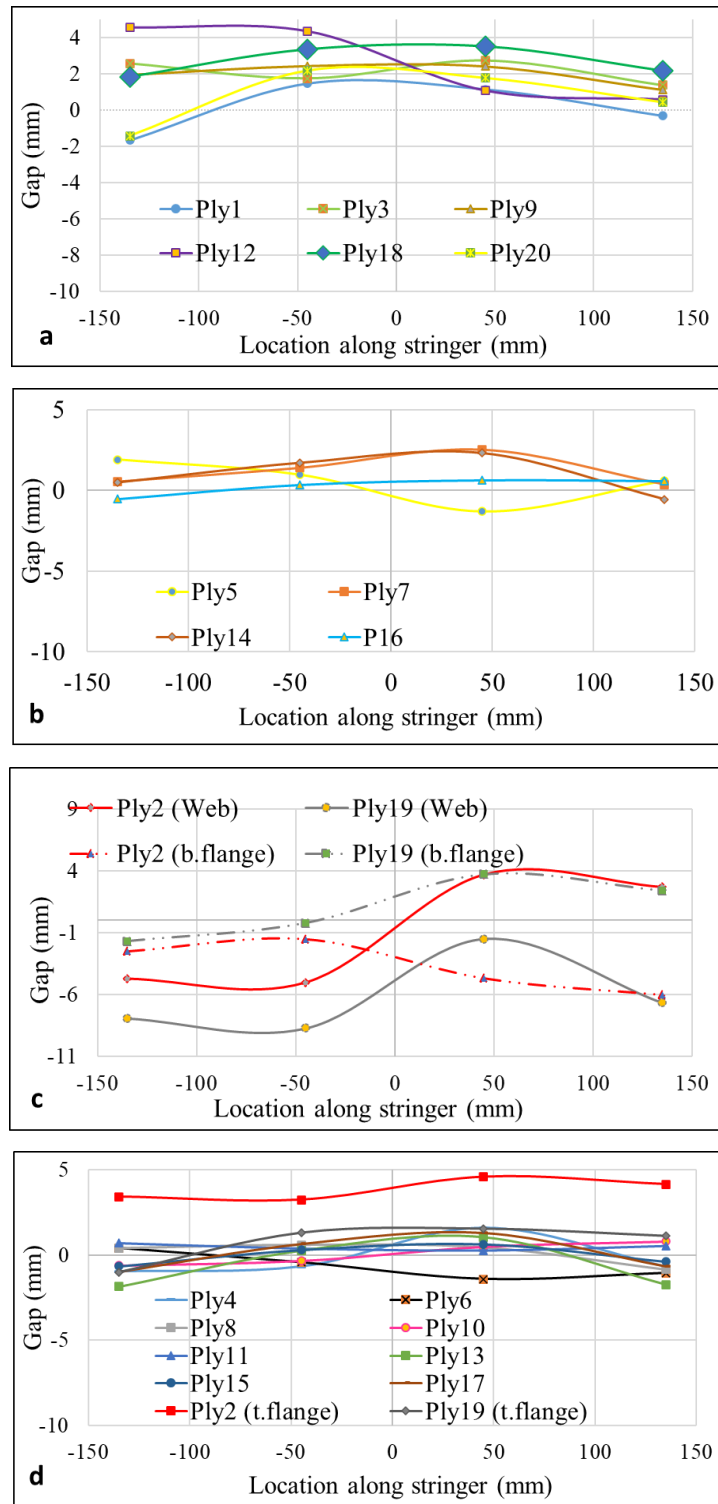


Fig. 3. 5. Gap between theoretical ply stop locations and after polymerization ply stop locations of different categories using microscopic observations respectively with (a) 1st category, (b) 2nd category, (c) 3rd category and (d) 4th category

Chapter 3

Following the same approach used after the cut-out process, 5th order polynomial trends were identified as in Eq. 3.2, but this time, we have to consider 4 types of ply categories.

$$A(z) = \sum_{i=0}^5 a_i z^i \quad \text{Eq. 3.2}$$

where $(a_0, a_1, a_2, a_3, a_4, a_5)$ are identified from experimental results after polymerisation, a_0 being the average deviation at the centre (mm), a_1 being the average inclination while a_2 (mm^{-1}), a_3 (mm^{-2}), a_4 (mm^{-3}) and a_5 (mm^{-4}) define the inflection points.

After obtaining results from four manufactured Ω -shape stringers, polynomial parameters were assembled into one set of toolbox data used to feed numerical modelling, via the means and standard deviations related to the four parts (Table 3.2).

	Coefficient	a_5	a_4	a_3	a_2	a_1	a_0
Category 1	Standard deviation	6E-12	2E-09	5E-07	8E-05	0.0158	0.62
	Mean	4E-12	8E-09	1E-07	-7E-05	-0.009	2.7
Category 2	Standard deviation	4 E-11	2E-09	1E-06	9.2E-05	0.017	1.6
	Mean	2E-11	1E-09	-9E-07	-6.3E-05	0.007	1.69
Category 3	Standard deviation	1E-10	1E-09	3E-06	6E-05	0.046	2.92
	Mean	7E-11	4E-09	-2E-06	-0.0001	0.028	-0.48
Category 4	Standard deviation	2E-11	2E-09	7E-07	8.9E-05	0.008	0.8
	Mean	1E-11	1E-09	-3E-07	-5.4E-05	0.003	0.38

Table 3.2. Standard deviations and means for polynomial coefficients after polymerization process obtained from 4 Ω -shape stringers

If we consider a_0 as a variability indicator (average deviation's mean), the maximum obtained value from the stringers' observation is about 2.7 mm corresponding to the 1st category of ply. Prepregs belonging to this category are difficult to lay-up since they cover all the stringer's width passing by 4 corners (2 concave and 2 convex) and by the mould bottom. These specifications conduct to obtain the maximum variability in the 4 types of categories. The 4th

category has the minimum value of a_0 with 0.38 since the operator has a visual aid while lay-up (symmetry as explained in the previous chapter).

We have to note the following remark concerning the section symmetry in paragraph §2.4.

2.4. Reconstruction of left side of cross section from right side variability measurement and cut-out width

The analysis above was only performed on the right half of the stringer's section, but the real symmetry between the two sides is not perfect. The mould is considered geometrically perfect. In the case of the perfect width of plies, the gap on the right side should be equal to the gap on the left side of the stringer's section because if a ply slides outwards to the right, it slides inwards to the left by the same amount. However, here, since we have an imperfection in the cut-out process of plies, the gap between the right side and left side plies edge is different (cf. Fig. 3. 6.). It is the sum of the gap due to the cut-out process and the gap in the polymerisation process. The gap between the left and right sides of a stringer is not the same, due to the cut gap in the first place and occasionally to the expansion of some plies while lay-up or polymerisation process (cf. Fig. 3. 7.).

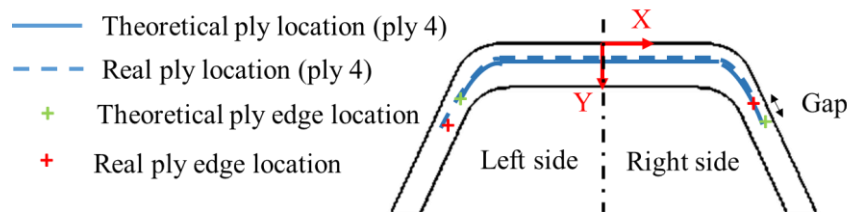


Fig. 3. 6. Schematic representation of symmetry gap on the right and left side of an Ω -shape stringer cross-section

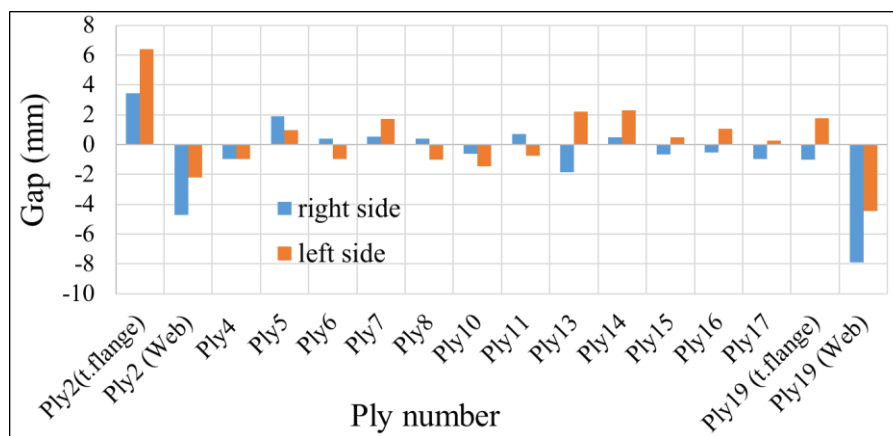


Fig. 3. 7. Gap of ply edge positions after polymerisation on the left and right side in an Ω -shape stringer cross-section for different plies, example of the stringer #1

In conclusion, the gap between theoretical modelled ply locations and real ones can be important for some plies. The global gap obtained after the manufacturing process is divided into three different sources respectively, prepreg cut-out process which represents about 30 % of the gap, lay-up process with 45 % of gap and polymerisation process with 15 % of gap. Note that these percentages depend on the type of plies, their location and even environmental conditions while manufacturing.

3. Fibre misalignment mathematical representation

For the past years, a large volume of analytical studies has been carried out by many researchers to analyse the effect of waviness in the response of composite structures. Analytical procedures were developed to capture local and overall mechanical behaviour of the laminate containing fibre waviness, evaluating its effect on key structural properties such as stiffness and strength. The development of those methods aims to present easy-to-implement solutions for the waviness problem, with high computational efficiency and sufficient accuracy [3].

Variations in fibre orientation have an important impact on the composite structure performance [4]. These variations are the result of several factors present during the different stages of the structure manufacturing, resulting in global and local misalignments. A global misalignment happens when the mean orientation of the ply or fibre bundles differs from the intended ply orientation. Local misalignments are produced by the periodic undulations present in the prepregs and dry preforms [2], as well as the preform manipulation to give shape to the composite structure [5].

In the literature, fibre misalignment is presented using many forms depending on several factors resumed in the table here after [6].

Example of Fiber Waviness	Number and Distribution (Single, Stochastic or In-Phase Distributed)	Through-Thickness Wave Form (Uniform vs. Graded)	Phase Characteristics of the Wave Form (Iso-Phase, Random-Phase)	Visibility (Embedded, Hump, Indention, Wave)	Dimensional Characteristics (2D, 3D)	Position (Centered, Outer Plies, Whole Laminate)	Phase Characteristics (Microscopic, Macroscopic)	Level of Influence (Material, Structure)	Geometric Position (Flat or Slightly Curved Areas, Complex Geometries)
	Single	Uniform	Iso-phase	Wave	2D	Whole laminate	Microscopic	Structural	Flat
	Single	Graded	Iso-phase	Embedded	2D	Whole laminate	Microscopic	Material	Flat
	Stochastic distributed	Graded	Random-phase	Embedded	3D	Whole laminate	Microscopic	Material	T-joint
	Single	Graded	Iso-phase	Hump	2D	Whole laminate	Macroscopic	Material	Flat

Table 3.3. Examples for waviness classification [6]

Chapter 3

As explained, in this work, we measured fibre misalignment using the vertical device installed during lay-up process in order to generate a mathematical representation related to the structure's complex geometry. We chose to observe the total misalignment obtained after cut and lay-up process. Obtained images can show us the gap between the real and theoretical fibre orientation for each draped ply. Here after (Fig. 3. 8.) an example of angle measurement in one location for ply #3. The angle measurement was realised using Imagej software with uncertainty evaluated at about 0.1° .

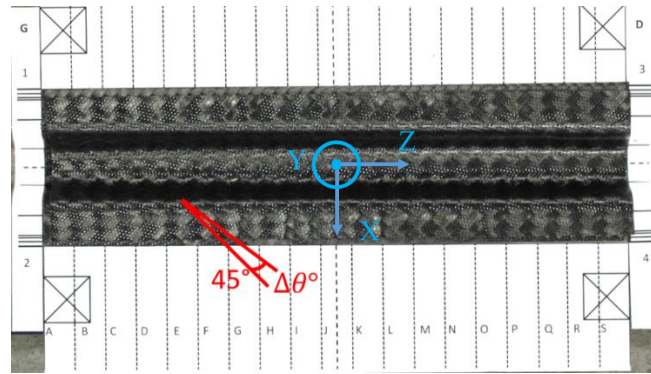


Fig. 3. 8. Example of angle measurement for ply#3 using image obtained from vertical device during lay-up

The reference angle used for gap measure is based on the preinstalled grid ($45^\circ/-45^\circ$ lines for woven preregs and 0° lines for UD preregs).

Obtained results are presented here after starting with woven plies in the top and bottom flange, then UD plies in the locations ($z=-135, -45, 45, 135$ mm) along the structure length. It was technically difficult to obtain clear measures for the web area using the orthogonal photo taking.

	Ply#1		Ply#3		Ply#7	Ply#9	Ply#14	Ply#18	Ply#20
Location (mm)	bottom flange	top flange	bottom flange	top flange	bottom flange	top flange	top flange	bottom flange	bottom flange
-135	-3.33	0.64	0.78	-0.96	0.56	-2.49	0.43	-0.71	2.63
-45	-0.62	-0.6	-0.73	0.32	-1.03	-1.5	1.62	0.45	1.58
45	-1.94	-1.43	-1.49	2.57	3.5	-3.7	-1.79	1.88	2.77
135	-2.39	-2.92	-3.35	2.66	4.1	-2.78	0.76	2.6	0.73

Table 3.4. Example of angle gap measurements for different woven plies in different locations (mean gap for 4 manufactured stringers) in z direction

Chapter 3

Starting from measurements from Table 3.4, we can obtain a range of variation of fibre orientation. The mean of different gaps is then calculated to be used in the mathematical representation. The chosen model is based on work from the literature [1] for UD and woven plies.

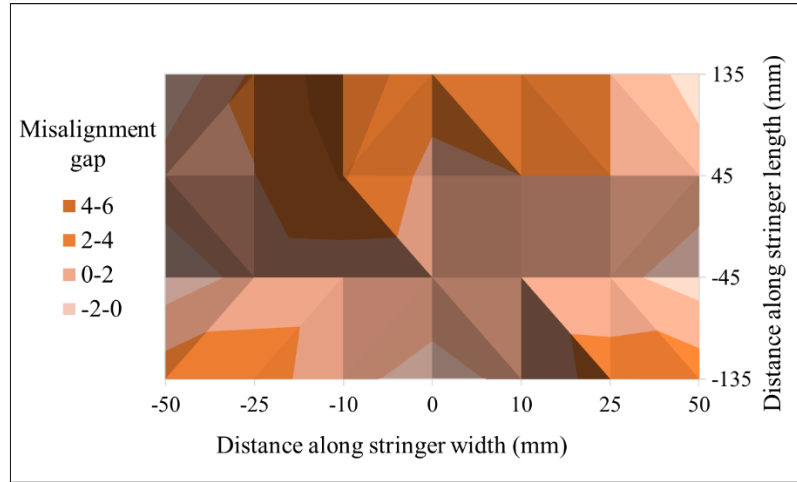


Fig. 3. 9. Example of map for misalignment gap for ply#1 along the structure (mean gap for 4 manufactured Ω -shape stringers)

Fig. 3. 9. presents a map for misalignment gaps for ply#1 chosen as examples in the stringer's length and width. As shown, gaps can be either positive or negative (the angle may be $>45^\circ$ or $<45^\circ$) depending on prepreg z location in the mould and on the operator himself. Ply#1 is a woven ply passing by all the corners and the bottom in the mould, which make it more sensible to fibre misalignment while lay-up. 3 peaks of misalignment gap are observed in the figure (between 4° and 6°).

Results for UD plies are presented in the Table 3.5. UD plies are localised in the structures top flange area.

Location (mm)	Ply#11	Ply#13	Ply#15	Ply#17	Ply#19
-135	2,27	0,56	0,27	0,33	1,12
-45	2,26	-0,29	0,68	0,32	1,09
45	1,14	-0,29	0,3	0	0,87
135	0,24	0,15	0,14	0,32	0,17

Table 3.5. Example of angle gap measurements in z direction for different UD plies in the top flange area (mean gap for 4 manufactured Ω -shape stringers)

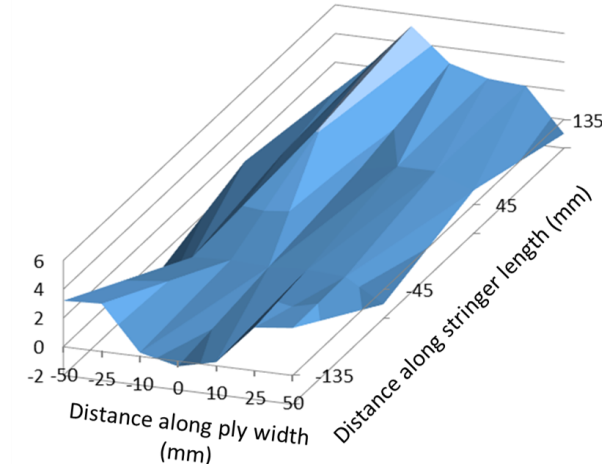


Fig. 3. 10. Example of map for misalignment gap for ply#11 in the top flange (mean gap for 4 manufactured Ω -shape stringers)

As noticed in the Fig. 3. 10., the fibre angle gap of ply#11 is less important than the gap of woven prepregs (between 0° and 2.5° mostly for this ply). This difference is due to the difficulty of woven plies installation in the mould and to the nature of woven plies, which are more deformable. Woven prepregs are delicate and easily triable especially after protectors' removal.

The aim here is to represent the fibre misalignment variability in the form of a geometric mathematical tendency in order to use it later for numerical modelling.

Based on the literature [7], to model the in-ply fibre orientations observing a continuity in the orientation values, the local variation of the misalignment can be described as a sum of i^{th} pseudo-Gaussian surfaces (Eq. 3.3) having a central point with coordinates X_i and Y_i , the length and the width of each perturbation is described by the parameters α_i and β_i , homologues to the standard deviation in a normal distribution. The amplitude of this perturbation is controlled by the parameter B_i (cf. Fig. 3. 11.). This set of equations allows the modelling of a continuous change in the fibre orientation properties [7].

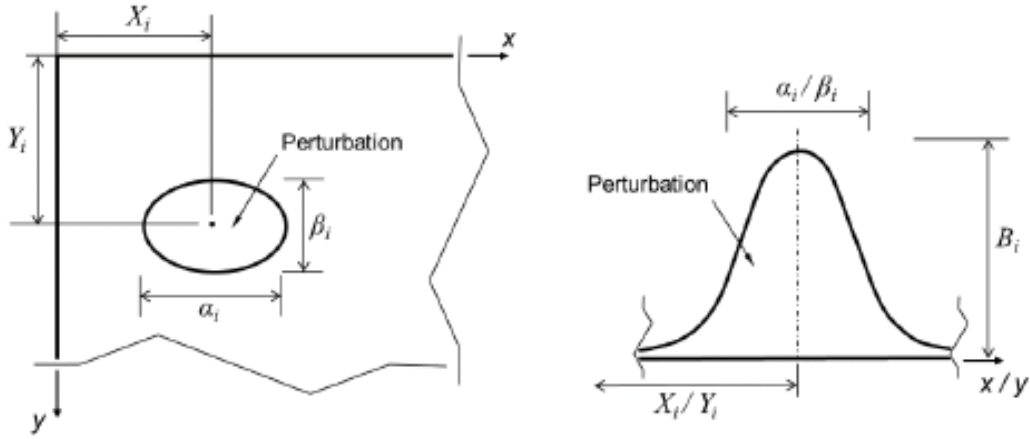


Fig. 3. 11. Schematic description by a pseudo-Gaussian surface (on right) of a localized zone of fibre orientation perturbation (on left) [8]

$$\theta_{pert}(x, y) = \sum_{i=1}^n B_i e^{-\left(\left(\frac{x-X_i}{\alpha_i}\right)^2 + \left(\frac{y-Y_i}{\beta_i}\right)^2\right)}$$

Eq. 3.3

In a surface of 380 mm x100 mm (stringer's surface), the number of present bosses is about 3 unlike the work in the literature that uses much larger composite structure. According to this, the parameter n in the equation (Eq. 3.3) is equal to 3.

The values of B_i are obtained by the identification algorithm, depending on the mean of variation related to each ply type (UD or Woven).

The continuous distribution of B_i has a mean of 0.071° and a standard deviation of 1.47° for woven preregs and for UD preregs 0.045 as mean and 0.35 as standard deviation. A normal distribution is used to draw the amplitude values.

The amplitudes α_i and β_i are generated independently, and are drawn from a uniform distribution with its limits ranging from 30 to 210 mm based on the literature work. The data are chosen from 20 different plies [8].

This distribution gives a continuous variation depending on geometric location of fibre orientation in the structure's surface.

4. Porosity distribution mathematical representation

Voids distribution in the structure is based essentially on data gathered from tomographic observations. Microscopic observation can show us voids in the cross-section, but measure

process is not able to obtain a global view. Tomography was able to provide a continuous section view over a representative length up to 100 mm. This observed length is able to give us valuable information about porosities shapes, location and size.

After reconstruction of images using the RX Solutions X-Act 2.0 software, they are transferred to Imagej to be cleared. The scale is settled also using Imagej.

Tomographic observation can provide a complete view of stringer's section plies. The obtained images give a clear view of plies, inter-ply, and porosities in each position along the stringer. The obtained images can't provide enough information related to the calculation of ply stop locations with the current adjustments (a different set of adjustments needs to be done on stringer area to obtain small voxel size to be able to identify ply stops) (cf. Fig. 3. 12.) but they provide the ability to measure porosities, their size, location, and fraction (cf. Fig. 3. 13.).

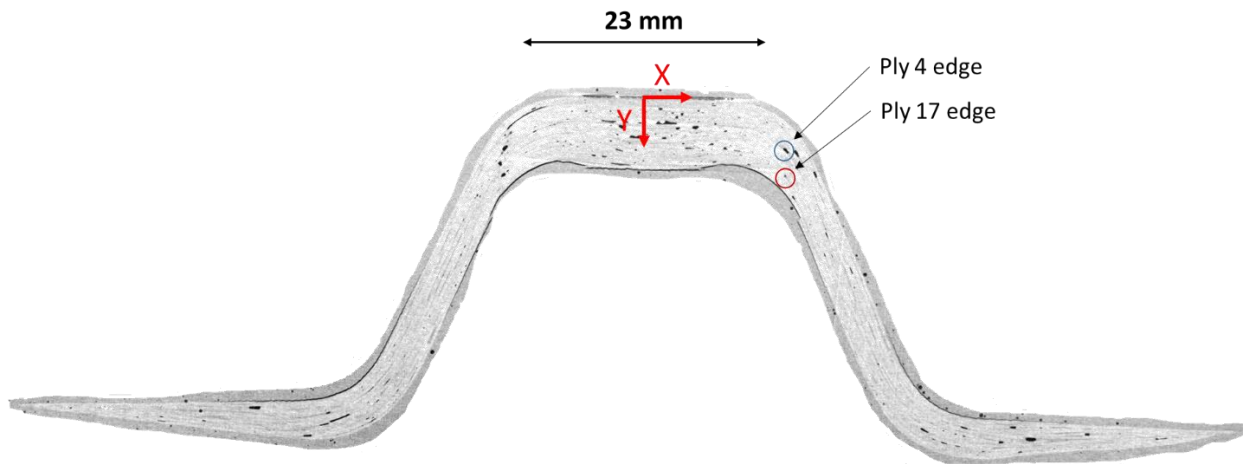


Fig. 3. 12. Tomographic observation of a stringer's section for the first manufactured Ω-shape stringer

Obtained images for the 1st manufactured stringer (cf. Fig. 3. 12.) for example show the presence of voids all over the section but especially in the top flange area. By focusing on the voids shape and length, we can determine stop locations for some plies (here for example ply#4 and ply#7). We chose to focus on porosity analysis using tomographic obtained images (cf. Fig. 3. 13. for example), where they are exploited for example to determine porosity locations.

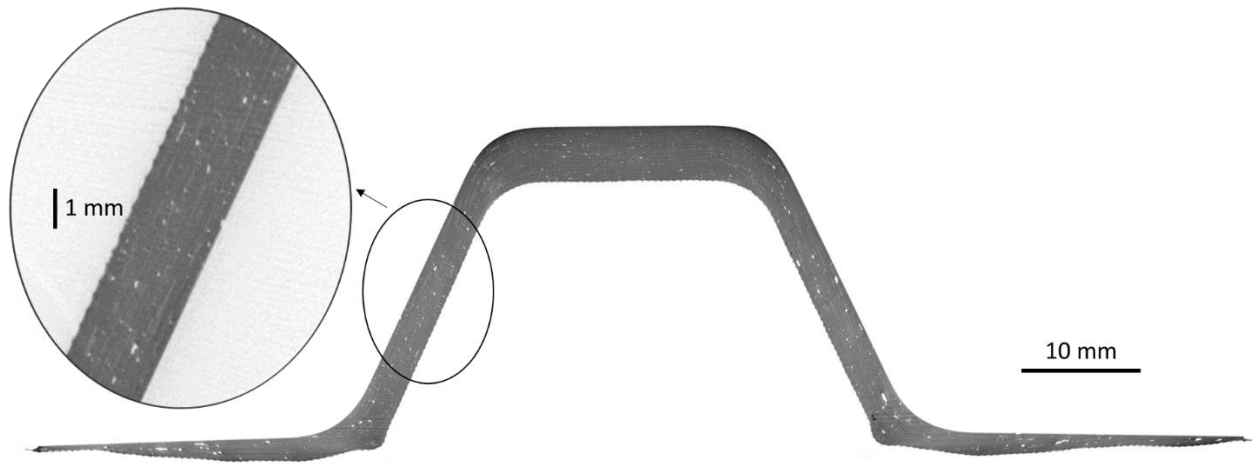


Fig. 3. 13. Tomographic observation of the complete stringer's section for the 2nd manufactured Ω -shape stringer

Fig. 3. 12. and Fig. 3. 13. correspond respectively to the first and the second manufactured stringers. Very different macroscopic defects are found between the two observed stringers. This difference is obvious especially in the top flange area, where for the first manufactured stringer, there is a larger porosity rate than the second stringer, which has an effect even on the area's thickness. These images confirm our prediction that even using the same manufacturing process under the same conditions, composite structures are different.

The assembly of different images obtained from tomography observation leads to obtain the shape and the length of voids inside the structure using Scan 3D software. Fig. 3. 14. shows porosities in a stringer's length equal to 10 cm. We can observe that in the webs, there are porosities that seem to be strongly linked to the crossing zones of the woven plies. In the top flange area, which has many UD plies at 0° , there are linear porosities, in the form of needles, in the direction of the length of the stringer.

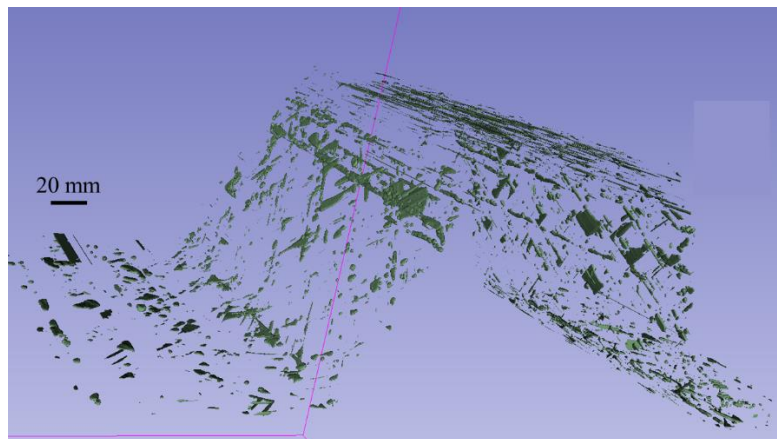


Fig. 3. 14. 3D representation of voids inside an Ω -shape stringer's length equal to 10 cm

From the images obtained from tomography, we can measure the porosity rate in specific areas, for example, the following position of a transverse cut of the top flange area of the first manufactured stringer (cf. Fig. 3. 15.). We can see the presence of voids in different shapes (large porosities in some areas, and small porosities in others).

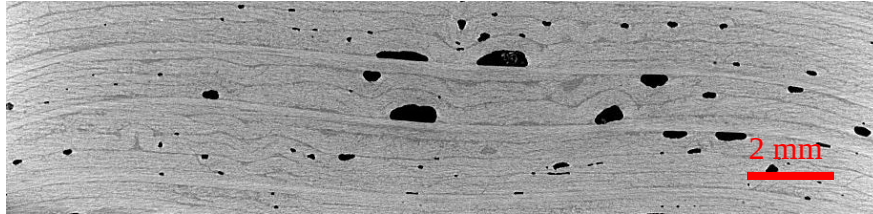


Fig. 3. 15. Zoom for tomographic image for the Ω -shape stringer's top flange

On a distance of 21 mm on the top flange, it is noticed, according to the curve below (Fig. 3. 16.), that the middle of the straight section of the beam contains the highest percentage of porosities.

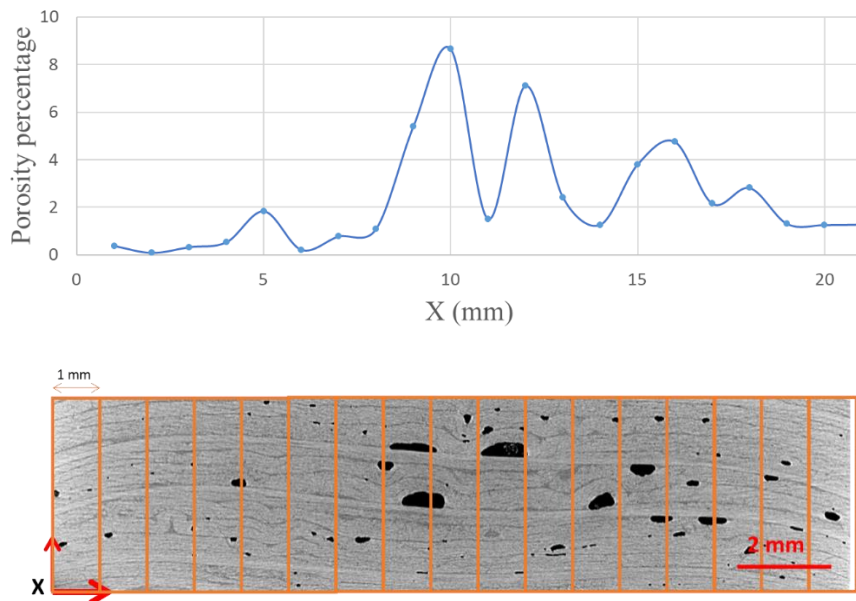


Fig. 3. 16. Curve of porosity rate calculation (top) and example of porosity in the top flange area (bottom)

This methodology of measurement is applied on the four manufactured stringers in the different areas (bottom flanges, webs and top flange). The mean calculations for all the stringers are used for results to be used for mathematical representation.

Measurement method consists in transferring obtained tomographic results to Imagej to obtain black and white images (black indicates porosity). The percentage of porosity is calculated using 2 different methods respectively Python code or Imagej.

The 1st method is Python code that consists in determining the percentage of dark areas in the total surface compared to an envelope, the envelope is determined from the predefined function "np.percentile" which allows determining the points (approximate) of the difference area between the part and the external surface from the percentage of detection of the external surface (parameter to be found according to the case of study). Using this envelope, the program can define the areas of porosity (the dark areas) and give us at the end the approximate percentage per image. The predefined "skimage.morphology" module of python is the most used for image analysis. The "convex_hull_image" function allows to determine the convex hull image of a binary image. This method is automatic so its advantage is to reduce analysis time, but it is not applicable for all image shapes since the edge's definition is limited. For our study, this method is used to determine porosity rates per different structure's area.

The 2nd method consists in importing the series of images in Imagej. This allows first to settle the contrast (dark areas are porosities) then, determine the outer edges of the part in each image using the find edges process. From the obtained results, we can determine the percentage of porosities in each sequence. This method is simple but it needs manual processing that takes time. In our work, voids rate in the total stringer's section is analysed using this process.

Taking a stringer length about 36 mm for example, obtained results for void percentage are represented in the Table 3.6, an image every 2 mm has been taken for the measure.

Using these data, composite material mechanical characteristics are recalculated with the probabilistic normal law applied to the void's volume fraction.

Z location from origin (mm)	Reference image	Void percentage
10	Slice 00033	0,035
12	Slice 00083	0,037
14	Slice 00133	0,041
16	Slice 00183	0,047
18	Slice 00233	0,048
20	Slice 00283	0,053
22	Slice 00333	0,054
24	Slice 00383	0,056
26	Slice 00433	0,058
28	Slice 00483	0,054
30	Slice 00533	0,052
32	Slice 00583	0,054
34	Slice 00633	0,049
36	Slice 00683	0,050
38	Slice 00733	0,054
40	Slice 00783	0,071
42	Slice 00833	0,065
44	Slice 00883	0,066
	Mean	0,05244444
	Standard deviation	0,00933193

Table 3.6. Voids rates in different sections of a stringer's length in the top flange area

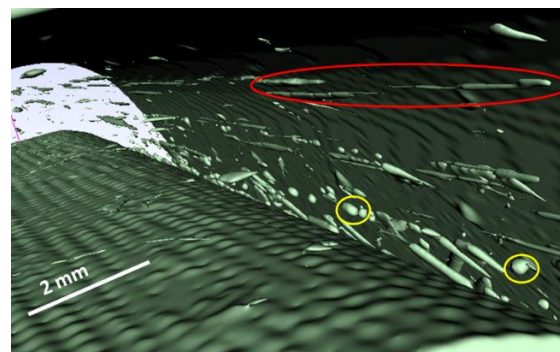


Fig. 3. 17. Porosity shapes representation by tomographic images reconstruction in the corner between web and top flange with (in red ellipse) long shaped porosity and (in yellow circles) circular shaped porosity

In addition to that, voids inside the structure have different shapes (cf. Fig. 3. 17. obtained by tomographic images reconstruction). We can see long porosities that can cover all the stringer's length. The location of this type is generally next to ply stops. We can observe also circular shaped voids all over the structure's section. These porosities appear to be arbitrarily distributed in the section.

The presented forms of voids distributed in the structure can be classified using several methods. For our case, the choice is to adopt two forms of mathematical representation, mathematical statistical representation for porosity rate in a chosen area, and void locations in each area. This choice is due to the variety of observed results where porosity rate is different from one area to another with different shapes that are localised all over the structure. this classification is not complicate to be introduced later in numerical modelling by affecting the material properties. 3D porosities and their effect on the structure's thickness is not taken into account. This limits the results to be obtained.

First of all, we need to introduce different analytical methods for porosity calculation to be used in our analysis. Voids in a structure impact essentially the material properties which are determined in a composite material using mixture rules.

4.1. Analytical approach for composite mechanical properties without porosity consideration

Several methods exist in the literature for void measurements. One of the simplest methods is to use composite mixture laws that give the mechanical properties of a composite material.

Starting with presenting mixture laws in the case of a composite material without porosity rates consideration, as follows,

The elastic constants of fibres E_{11} (cf. Eq. 3.4) and ϑ_{12} (cf. Eq. 3.5) are evaluated according to the rule of mixtures:

$$E_{11} = V_f E_{f1} + (1 - V_f) E_m \quad \text{Eq. 3.4}$$

$$\vartheta_{12} = \vartheta_{13} = V_f \vartheta_{f12} + V_m \vartheta_m \quad \text{Eq. 3.5}$$

E_{11} and ϑ_{12} are respectively the longitudinal elastic modulus and the shear Poisson ratio. E_{f1} and E_m are the elastic modulus respectively of fibre and matrix. V_f and V_m are the volume fractions of fibre and matrix respectively.

the transverse shear modulus (cf. Eq. 3.6) can be obtained by the following empirical formula (the most affected by porosity taking into account):

$$G_{23} = \frac{E_2}{2(1+\nu_{23})} \quad \text{Eq. 3.6}$$

4.2. Analytical approach for composite mechanical properties with porosity consideration

According to the theoretical study by MacKenzie in [9], the material stiffness of a solid which containing spherical holes can be approximately expressed as follows in Eq. 3.7:

$$E_v = E_s (1 - V_v)^2 \quad \text{Eq. 3.7}$$

where the subscripts s and v represent the fully dense material and the porous material, respectively

To generalise, the formula used in the literature studying the elastic modulus of fibre composite containing voids is as follows in Eq. 3.8:

$$E_{i-v} = E_i (1 - V_v)^2 \quad \text{Eq. 3.8}$$

where E_i represents the elastic constants under non-void conditions, and V_v represents the void contents. Notably, the elastic constants are the constants of the composite, instead of that of a single fibre or matrix material.

Nevertheless, the fibre component in polymer matrix composite mainly provides strength and stiffness, and the matrix component mainly supports and protects the fibre component. The existence of voids only weakens the properties of the matrix and exerts no effect on the performance of the fibre component. Hence, the properties of the matrix component should only be reduced (cf. Eq. 3.9):

$$E_{m-v} = E_m (1 - V_{m-v})^2 \quad \text{Eq. 3.9}$$

where E_{m-v} represents the elastic moduli of the matrix after reduction. V_{m-v} denotes the void content in the matrix component and can be expressed as follows in Eq. 3.10:

$$V_{m-v} = \frac{V_v}{V_m} \quad \text{Eq. 3.10}$$

where V_v and V_m are the void and matrix contents of the fibre material, respectively.

We assumed that the Poisson's ratio of the matrix is not affected by the void, and ν_{23} will remain stable according to the above equations.

This empirical calculation method is used in the literature and it's available for most composite structures with voids integration [9]. Nevertheless, there are several other methods, for example, the one used by authors in [8]. This method consists in calculating void volume fraction depending on the total thickness of the composite structure in a specific area. For the top flange area for example, Fig. 3. 18. represents the evolution of porosity rate in the top flange area depending on the thickness of the total area (16 plies in the top flange). The red dot in the curve represents the perfect case, where the total thickness is 4.8 mm and there is no porosity. The trend of the least square fit indicates that effectively the porosity varies with the plate thickness. The equation here after (cf. Eq. 3.11) represents the evolution of voids volume fraction in function of area thickness specified for the top flange:

$$V_p = \begin{cases} 0 & \text{if } t_{area} < 4.8 \text{ mm} \\ 4.75 t_{area} - 23.17 & \text{if } t_{area} > 4.8 \text{ mm} \end{cases} \quad \text{Eq. 3.11}$$

where V_p is the volume fraction of the porosity (%) and t_{plate} is the area thickness.

The coefficients 4.75 and -23.17 are obtained from a linear regression from the experimental data (cf. Fig. 3. 18.).

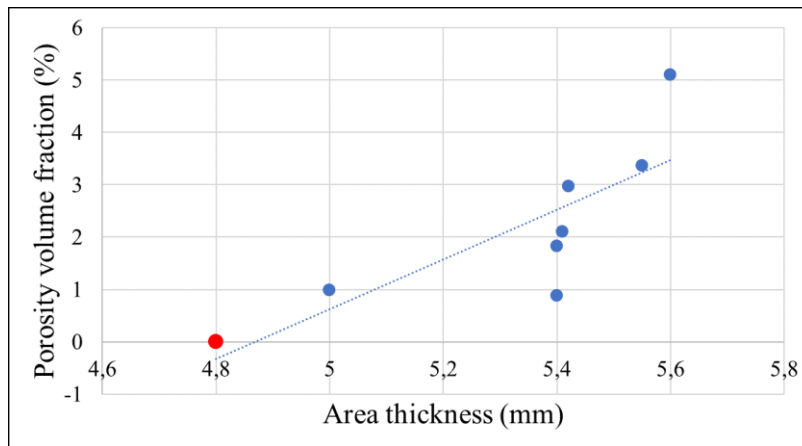


Fig. 3. 18. Curve of porosity volume fraction depending on the structure thickness in the top flange area

The same type of representation is applied on the web and the bottom flange areas.

We need now to determine porosity rate mathematical statistical representation in order to be able to calculate material properties. As mentioned earlier, porosities are not represented only by their rate but also by their location in a section in relation with its shape.

Voids rates are presented with a simple normal distribution based on the mean and standard deviation (Eq. 3.12) related to each area in different cross sections of the stringer. For this work, we chose to represent rates every 2 mm in a length of a stringer of 10 cm:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad \text{Eq. 3.12}$$

where μ is the mean and σ is the standard deviation and x represents the location in the cross-section.

For its locations related to shapes, long porosities for example, effect the material properties in the same z location along the stringer's length. That's why it is necessary to identify (even approbatively) the number of long shaped voids, their location in the section, length and volume. For example, in Fig. 3. 19, we took an area between the web and the top flange for the 2nd manufactured stringer. There is a long void with volume about 0.152 mm^3 . This form of void exists in several locations in the stringer. Their locations are represented also by a normal law related to each area where the mean is the mean location of long porosities in each area calculated for the four manufactured stringers. For other forms of porosity, the mean value is integrated per considered area in the stringer's section. In the bottom flange for example, the mean value is about 0.2 % while it is less important in the web area with only 0.12 %, where voids number is the smallest compared to other areas.

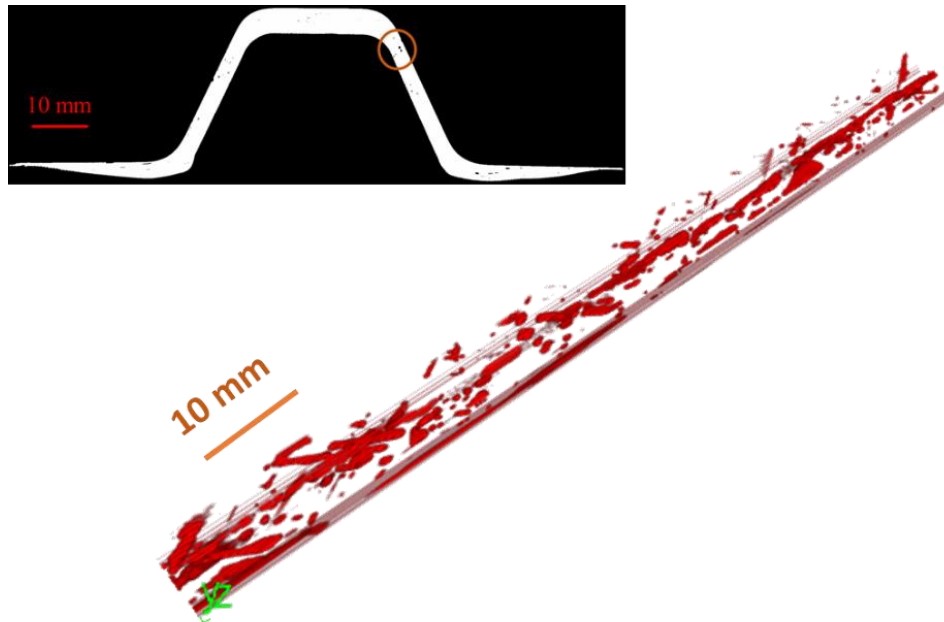


Fig. 3. 19. Porosities representation (bottom) in the corner between the web and the top flange (the red circle in the top)

Conclusion

The obtained data from observations are transformed into mathematical representations related to each type of variability. Ply stop locations are represented by 5th degree polynomials sufficient to describe the ply ends undulations observed over a stringer length of about 380 mm. The 6 parameters describing these polynomials were analysed from a statistical point of view, by separating the measurements into four families of plies, determined according to characteristics such as strand orientation or position with respect to a geometrical feature. The choice to manufacture 4 stringers was a compromise between statistical needs and manufacturing complexity. A total number of 64 random draws has been realised for polynomial parameters for 4 stringers. Fibre waviness was analysed using a continuous mathematical representation based on literature work represented by a finite sum of deformed zones, using a Gaussian surface centred on the centre of the perturbations. The amplitudes of the perturbations are statistically studied with an adaptation to the characteristics of the structure studied from the observations made. A total number of 10 random draws for 5 trend parameters per material. The porosity rates were evaluated by normal laws in terms of position and extent in the horizontal plane and represented by statistical laws conducted according to their location in the cross-section of the studied stringer based on tomographic observations.

All the gathered sets of data from the four manufactured stringers related to ply stop locations, fibre misalignment or porosity rates variabilities are to be assembled in global numerical modelling taking into account the studied forms of variability. The challenge is to be able to insert them in a simple shell model and to predict ranges of variability for further manufactured structures.

References

- [1] Davila Y, Crouzeix L, Douchin B, Collombet F, Grunevald Y-H. Identification and Modelling of the In-Plane Reinforcement Orientation Variations in a CFRP Laminate Produced by Manual Lay-Up. *Applied Composite Materials* 2018;25. <https://doi.org/10.1007/s10443-017-9642-4>.
- [2] Potter K, Khan B, Wisnom M, Bell T, Stevens J. Variability, fibre waviness and misalignment in the determination of the properties of composite materials and structures. *Composites Part A: Applied Science and Manufacturing* 2008;39:1343–54. <https://doi.org/10.1016/j.compositesa.2008.04.016>.
- [3] Alves MP, Cimini Junior CA, Ha SK. Fiber waviness and its effect on the mechanical performance of fiber reinforced polymer composites: An enhanced review. *Composites Part A: Applied Science and Manufacturing* 2021;149. <https://doi.org/10.1016/j.compositesa.2021.106526>.
- [4] Christos C. Chamis. Probabilistic Composite Design. American Society For Testing and Materials. 1997.
- [5] Trask RS, Hallett SR, Helenon FMM, Wisnom MR. Influence of process induced defects on the failure of composite T-joint specimens. *Composites Part A: Applied Science and Manufacturing* 2012;43:748–57. <https://doi.org/10.1016/j.compositesa.2011.12.021>.
- [6] Thor M, Sause MGR, Hinterhölzl RM. Mechanisms of origin and classification of out-of-plane fiber waviness in composite materials. A review. *Journal of Composites Science* 2020;4. <https://doi.org/10.3390/jcs4030130>.
- [7] Ardeshir Goshtasby, William D. O'Neill. Curve Fitting by a Sum of Gaussians. *Graphical Models and Image Processing* 1994;56:281–8.
- [8] Yves Angel Davila Montano. Etude multi-échelle du couplage matériau-procédé pour l'identification et la modélisation des variabilités au sein d'une structure composite. 2015.
- [9] J K Mackenzie. The Elastic Constants of a Solid containing Spherical Holes. *Proceedings of the Physical Society Section B*. 1950.

Chapter 4: Numerical analysis of variability and integration of different studied variability forms in the numerical modelling

1. Introduction to insertion of variabilities in numerical modelling

In this chapter, our goal is to create a family of numerical models representing virtual but realistic composite structures with similar complex geometry zones which include variabilities within them. The term realistic in our case means the ability to take into account the influence of manufacturing steps from raw material to the final part considering a sufficient amount of information. The model proposes a continuous evolution of certain properties from one finite element to the next. The created models are a numerical twin of real observed parts as a first step, and as a second step extrapolate the information obtained by observation of a finite number of real structures to provide information for an infinite number of future composite structures and over lengths greater than those observed. The objective as mentioned earlier (see introduction) is to propose variation ranges of the properties and behaviour of the structures based on the studied variability. The ambition is to refine the knockdown factors and to reduce a certain amplitude of manufacturing variability. A model for the stringer has been created using Python/Abaqus. Python code (see appendix 5) was totally generated by our team allowing to integrate the chosen mathematical representations in the numerical modelling. It was fed with observation results, to obtain a toolbox able to estimate variability effects on the final behaviour of the Ω -shape stringer and predict variability distribution.

2. Model presentation (objectives and methodology)

2.1. Mesh creation adapted to the chosen model and to the objectives

The objective of the model consists of translating the variabilities found experimentally into a numerical model of the Ω -shape stringer to see the effects of these variabilities on its behaviour. The chosen model is a composite shell model considering the hypothesis that the mould geometry is perfect.

Shell model choice is based on two main objectives, in terms of ability to include the variabilities in industrial type models, and to launch batches of several hundreds of calculations of all different objects, fed by realistic variabilities, to evaluate the sensitivity.

Shell chosen elements are S4R type with 4-node general-purpose shell, reduced integration with hourglass control, finite membrane strains. This choice is due to its simplicity and efficient representation of the studied structure. This type of modelling can represent studied variabilities and provide desired results.

Chapter 4

The Ω -shape structure was created by programming each node spatial coordinates, with the possibility to refine the mesh depending on the zone of interest. The objective here is to be able to control the numbering of each node, then later, of each element, to be able to control the mechanical properties associated with each finite element independently. 9 zones were created to describe the Ω -shape stringer (cf. Fig. 4. 1.) as explained in Table 4. 1. Experimental observations show that ply stops are not necessarily in the theoretical zones mentioned in Table 4. 1.. This strategy allows mastering mechanical properties elementary mesh by elementary mesh.

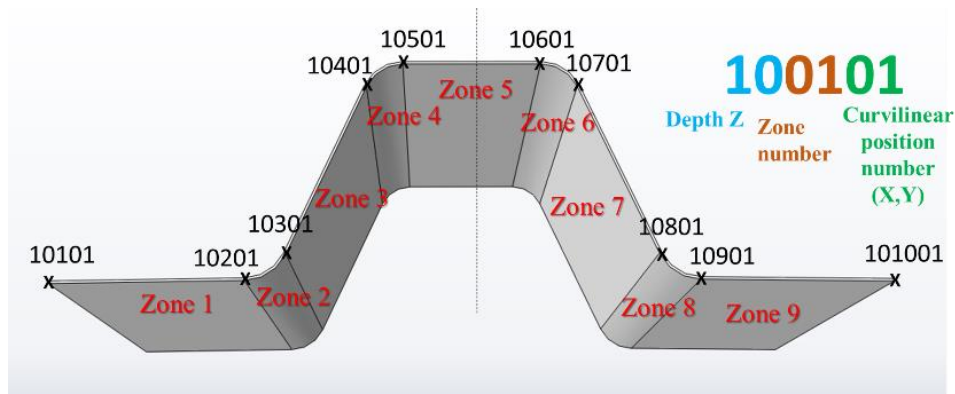


Fig. 4. 1. Numerical Ω -shape stringer cross-section with related zone numbers and symbol of each node

Zone number	Zone location	Zone characteristics	Ply number with theoretical stop location
1 and 9	Bottom flange	Standard zone with ply stops	1, 3, 9, 12, 18, 20, 5, 7, 14, 16
2 and 8	between bottom flange and web	convex radius with ply stops	$2_{b.flange}$, $19_{b.flange}$
3 and 7	Web	standard with ply stops in the upper part	5, 7, 14, 16
4 and 6	Between top flange and web	Concave radius with ply stops	$2_{t.flange}$, 4, 6, 8, 10, 13, 15, 17, $19_{t.flange}$
5	Top flange	Standard	-

Table 4. 1. Zone definitions with specifications and theoretical ply stops included in each cross-section area

For this work, the elementary element is chosen to be a millimetre in the representation of results, because the variability of the positions of the ply stops are on a millimetre scale (cf. Fig. 4. 2.). In total, 57000 elementary elements are chosen.

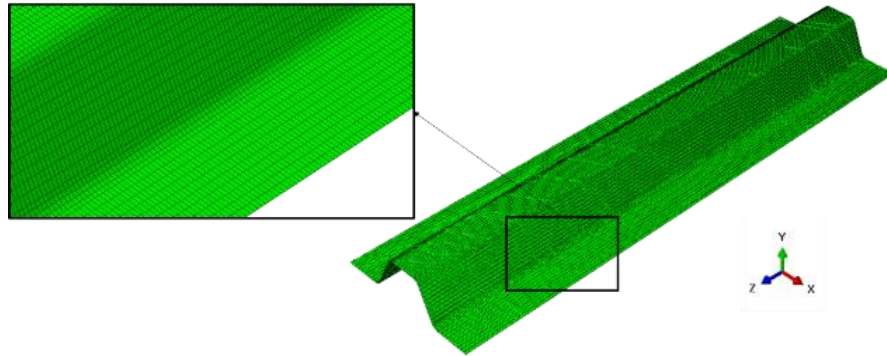


Fig. 4. 2. Mesh finesses in the Ω -shape stringer's numerical model

2.1.1. Material properties used by default in the numerical modelling without variability effects

Table 4. 2. presents material properties used by default in numerical modelling. s1t signifies Tensile stress limit in fibre direction X_t . s1c is the Compressive stress limit in fibre direction X_c . s2t represents the Tensile stress limit in the transverse direction Y_t . s2c is the Compressive stress limit in transverse direction Y_c and s12 defines the Shear strength in the X-Y plane. These data sets are taken from the constructor sheets. As defined in the previous chapter, these nominal properties will be modified for each element by taking into account porosity rates.

Failure parameters are respectively, Tsai coef. cross prod. which indicates cross-product term coefficient ($-1 < f < 1$) and used only for Tsai theory and ignored if sigma bia. x is given (default=0) and biaxial stress limit coef. Tsai that defines Biaxial stress limit sigma bia. x. This value is used for Tsai theory, if this entry is non zero, f is ignored. α_{ii} are the thermal expansion coefficients. These parameters are insert in Abaqus model as initial material properties. They will be changed when porosity rates are considered in the model.

Prepreg characteristics	UD	Woven
Thickness (mm)	0.297	0.297
Density (T/mm ³)	1466e-12	1471e-12
E1 (MPa)	131650	61225
E2 (MPa)	6550	61225
E3 (MPa)	6550	11249.5
ν_{12}	0.27	0.22
ν_{13}	0.27	0.2
ν_{23}	0.14	0.2
G12 (MPa)	3260	2400
G13 (MPa)	3260	2475
G23 (MPa)	2280	2475
s1t (MPa)	1814	1600
s1c (MPa)	-1325	-1168
s2t (MPa)	70	70
s2c (MPa)	-300	-300
s12 (MPa)	70	70
Tsai coef. cross prod.	1	1
Tsai biaxial stress limit coef.	1	1
α_{11}	-0.0000056	-0.0000115
α_{22}	0.00001434	-0.0000115
α_{33}	0.00001434	-0.0000115

Table 4. 2. Material theoretical properties and failure characteristics used in the numerical models

2.1.2. Definition of boundary conditions according to the studied cases

Two cases of boundary conditions are studied here:

- 1st case illustrates the structure behaviour during cooling at the end of the curing. A variation of temperature of - 100 °C is applied on all nodes while fixing the node located in the geometric centre of the structure and blocking its 6 degrees of freedom. This phase

represents the implementation of residual stresses within the structure during the cooling phase that follows the polymerization;

- 2nd case illustrates the structure behaviour during a cooling according to the first case, followed by a tensile loading by imposing a displacement of 1 mm on every node of a side of the structure and blocking the displacement of the nodes of another side (cf. Fig. 4.3).

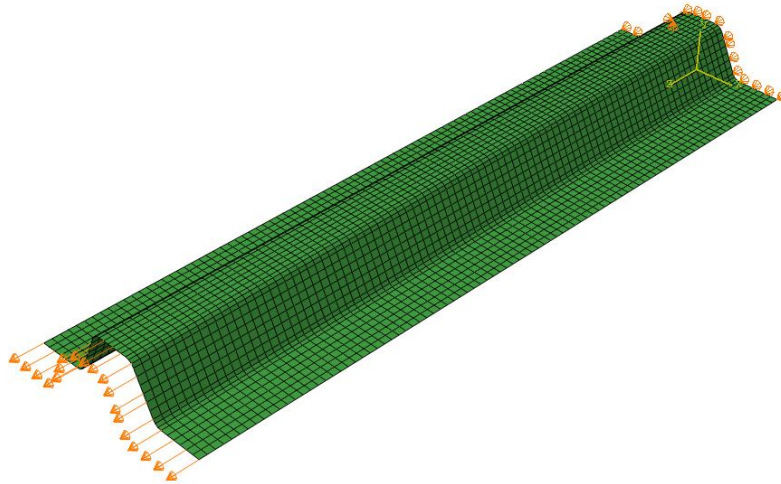


Fig. 4. 3. Schematic representation for applied mechanical loading on one side of the Ω -shape stringer with boundary conditions on the other side

3. Ply stops variability integration

3.1. Integration of variabilities due to cut-out and lay-up process

The originality of the proposed model consists in the fact that the plies do not stop systematically in the planned area, but can be located to the right or the left of the nominal position. It was therefore necessary to devise a special algorithm to determine the presence or not of each ply in each finite element of the model. The presence or absence of a ply depends on two types of variabilities resulting respectively, from the cutting process and the ply installation in the mould. As explained earlier, it has been chosen to define 2 categories related to cutting variability of woven plies and UD plies and 4 categories related to lay-up issues variabilities. To integrate variabilities, a random drawing, based on means and standard deviations identified in the previous chapter per category is applied for each location along the stringer's length. In order to guarantee the physical continuity of the ply edges, there is a random drawing of the presence or absence of a ply in each element. The evolution of the ply edge is

described by a polynomial of degree 5 (cf. Eq. 4.1), to be continuous and derivable, and to describe the physics. The random draw is done on the 6 parameters of the polynomial that describes this spatial evolution, based on the results of the previous chapter:

$$A(z) = \sum_{i=0}^5 a_i z^i \quad \text{Eq. 4.1}$$

The difference in ply widths in the theoretical and real cases has an impact on model results. We needed to identify it by defining the first theoretical edges of all 20 plies along with the stinger, then add this gap (identified experimentally in the second chapter) to both edges of each ply for each stringer. The integration of ply stop variability in a shell numerical model has been done on every structure element, to attribute properties by finite element and to be able to change these properties from one finite element to another depending on its geometric location. Fig. 4. 4. shows the numerical integration of variabilities in numerical modelling. The random draw is realised on each ply edge for variabilities due to the cut-out process and variabilities obtained after the polymerization process. The new ply stop locations are then calculated in the right and the left side of the stringer cross-section. The creation of each ply related location matrix is based on the presence or not of this ply in the finite element. The aim is to create an input file containing information related to each finite element of each ply along with the geometric creation of the structure. For Abaqus, each finite element contains the associated number of plies.

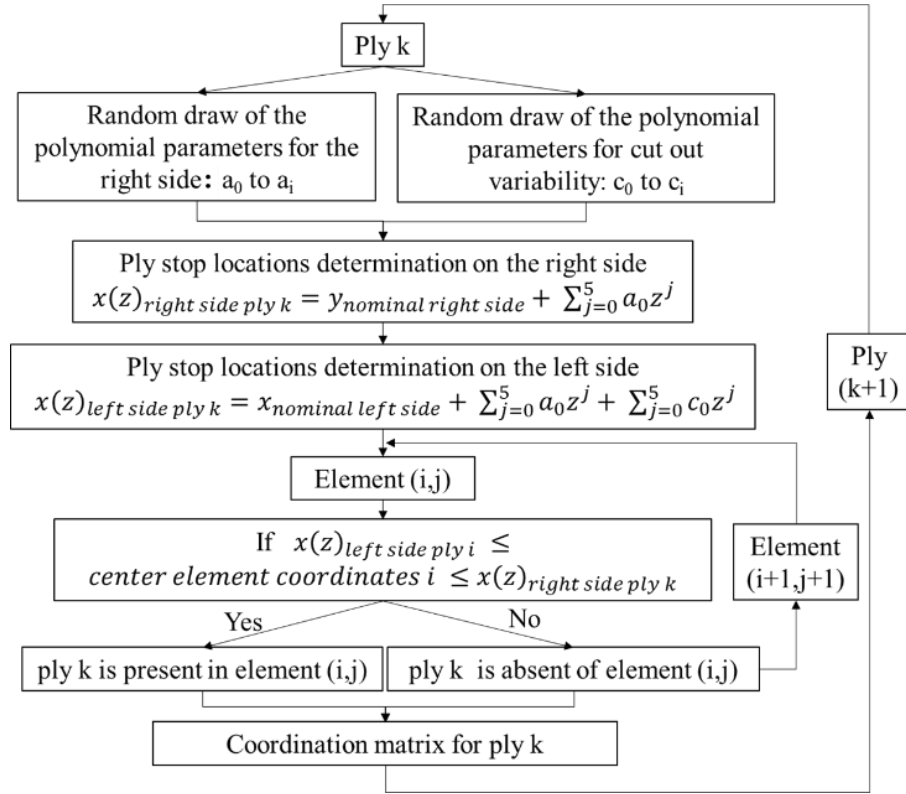


Fig. 4. 4. Flowchart for the integration of ply stop variability in numerical modelling

Fig. 4.5 represents an example of ply edges outline after ply stop location variability consideration in numerical modelling using input data for each finite element (total number of 57000 element). The yellow zone represents the ply area in the total surface (purple).

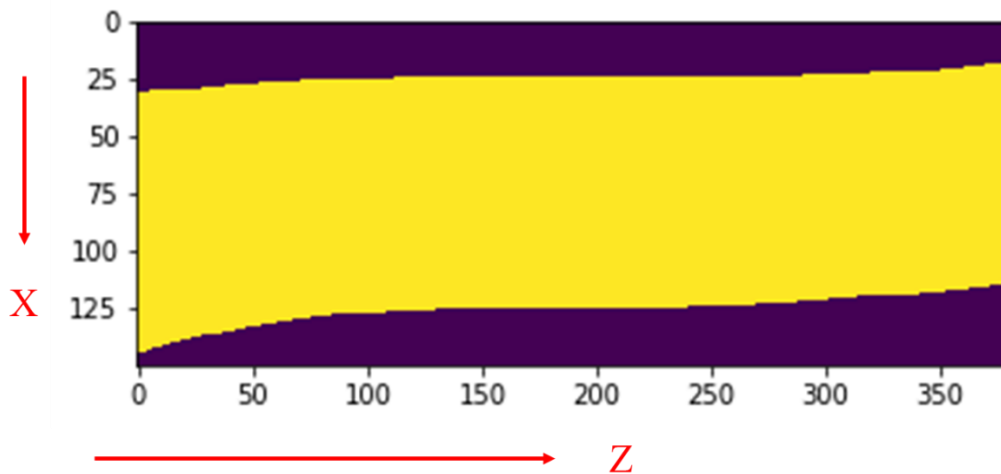


Fig. 4. 5. Example of ply #12 edges after ply stop location variability integration in the numerical modelling

4. Fibre misalignment integration

In order to evaluate the effects of the variability of fibre orientation in a composite part, the fibre misalignments are introduced into a finite element (FE) model as the sum of two different outcomes. The first outcome is the theoretical ply lay-up angle defined in the structure's description. The second outcome is the local variability due to the deformation after lay-up process of each ply depending on the structure's geometry.

The local variability of fibre orientation is calculated based on the model proposed in the previous chapter (cf. Eq. 4.2) for each finite element and for each ply. A random draw is conducted on the equation parameters based on the realised measurements. Two categories are chosen according to material nature, UD prepregs and woven prepregs. The coordinates of each finite element are defined by the geometric coordinated of its central point. The orientations are assumed to be unchanged during the curing phase of the composite material. However, this hypothesis cannot be proven, since the determination of actual fibre orientation after curing involves destructive techniques to measure the fibre in the composite cross-sections [1]:

$$\theta_{pert}(x, y) = \sum_{i=1}^n B_i e^{-\left(\left(\frac{x-X_i}{\alpha_i}\right)^2 + \left(\frac{y-Y_i}{\beta_i}\right)^2\right)} \quad \text{Eq. 4.2}$$

The originality of the process lies on its continuity from one finite element to another, in order to obtain more realistic model. The random draw of parameters allows to predict fibre misalignment variability effect on future structures as well as variability ranges to be taken into account in the designing phase.

5. Porosity integration

Porosity presence in a composite structure affects mainly its mechanical properties. So based on models defined in the previous chapter, material properties are redefined in numerical modelling.

First of all, two materials are defined per finite element respectively UD and Woven. Then, according to porosity rates measured for each element, we realise random draws on the normal distribution of voids rate per element. Fig. 4. 6. explains the adopted process and steps for porosity integration in the model.

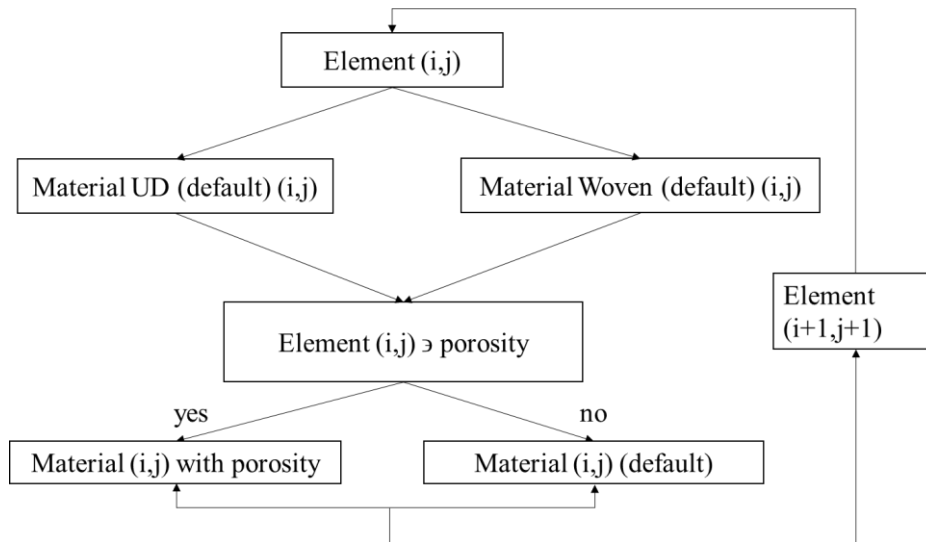


Fig. 4. 6. Flowchart for porosity integration in numerical modelling

The size and shape of voids is also taken into account by considering that the material properties of finite elements that contains a continuous porosity are identical both in x or z direction of the stringer. Random draws were conducted also on voids location in the stringer's cross-section.

This strategy is applied for 5 areas of the stringer's section shown in Fig. 4. 7. Area#1 is the left bottom flange with the left concave corner. This zone contains porosities especially at the plies stop locations. Area#2 is the left web with the left convex corner that contains the least porosities compared to other areas. Area#3 is the top flange. It presents the highest voids rate compared to other areas. Area#4 and Area#5 are symmetric to respectively Area#2 and Area#1 and they have equivalent characteristics.

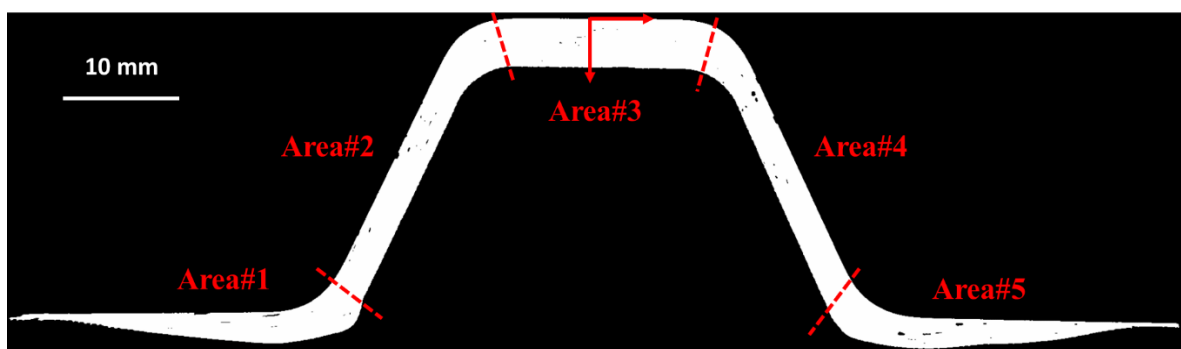


Fig. 4. 7. Schematic representation for different areas considered in porosity numerical study

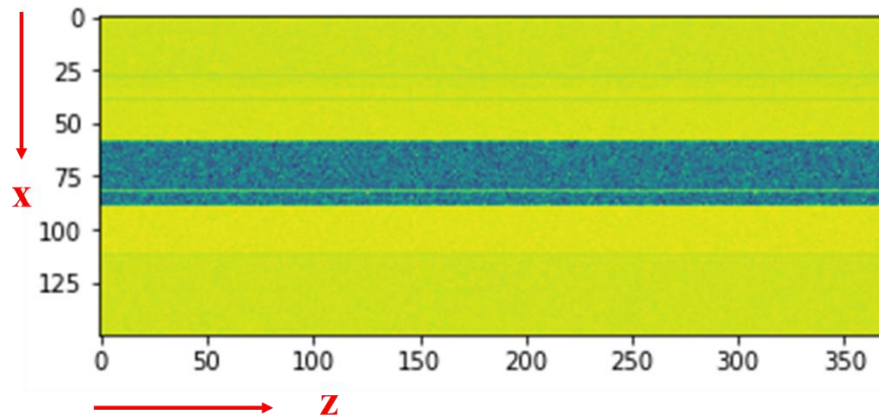


Fig. 4. 8. Schematic representation for porosity concentration in the Ω -shape stringer using finite element number

Fig. 4. 8. shows porosity concentration in the structure by finite element in a case of random draw (in this case we have 90 finite elements in z direction and 40 finite elements in x direction), where the darker colour presents the higher porosity rate. There are small voids in an element and long voids along z direction. The maximum void rates are in the top flange area. Fig. 4. 9. indicates the effect of voids consideration on the mechanical properties of a composite structure.

E1 Woven

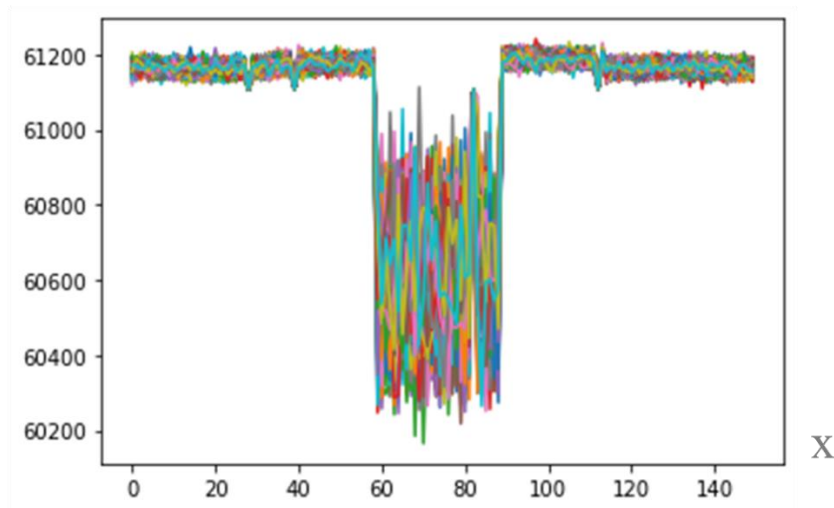


Fig. 4. 9. Elastic woven composite component along the transverse direction x with voids consideration

6. Results and discussion of taking into account each type of variability and the total variability types

6.1. Illustration of numerical results with ply stop variability

6.1.1. Twin models of the four manufactured stringers after cooling followed by tensile loading

This section presents the explanation of the numerical twin realised on the available manufactured Ω -shape stringers (4) using the real ply stop locations determined experimentally. The aim is to compare the behaviour of the different stringers to determine the variability effect in a family of parts manufactured using the same process.

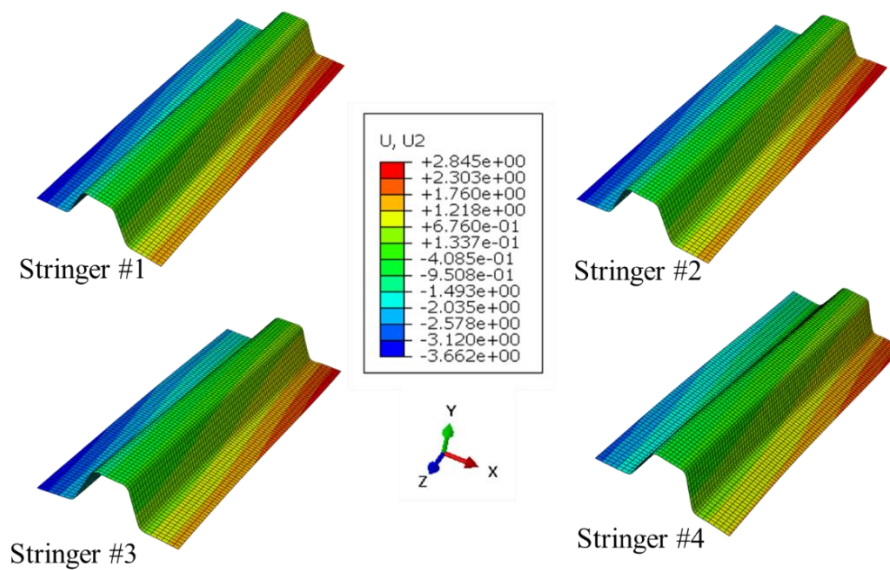


Fig. 4. 10. Twin model results in the case of vertical displacement fields (Y direction), after cooling followed by tensile loading for the 4 manufactured Ω -shape stringers

Fig. 4. 10. shows the results of the second case of boundary conditions applied to the twin models of the 4 manufactured stringers, indeed representing the real ply stop locations observed, without adding statistical random drawing for the parameters leading to the polynomial representation of ply stop positions. Differences between the four manufactured stringers are obvious in terms of displacement, with the unique representation of ply stop locations, enhancing the fact that the selected variability in this paper seems to have a sensitive effect on the behaviour of the structure. The following section is dedicated to gathering the input data got from the 4 twin models to carry out probabilistic modelling.

6.1.2. Numerical Results of stringers with random drawings after cooling

Secondly, the results obtained from observing the four of them are merged into a set of entry parameters used in the general model with a sufficient statistical set of parameters. Starting from results and observations obtained from these parts, the numerical model aims to predict the behaviour of an infinite number of parts, by applying a random draw on the studied variability (here ply stop locations). The random draw is realised on the polynomial parameters describing the evolution of ply stop locations along the stringer.

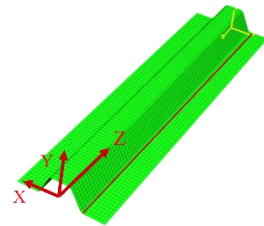
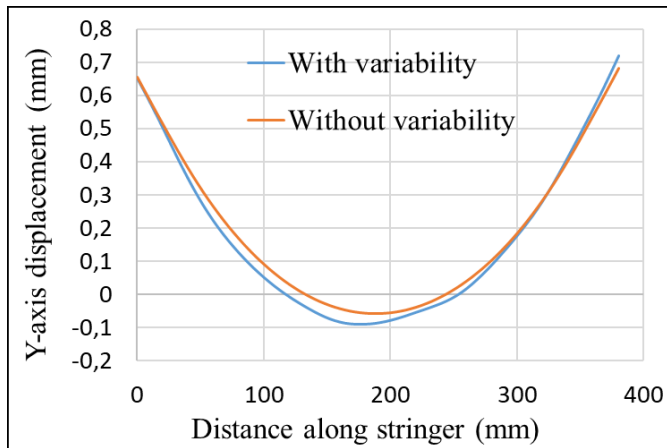


Fig. 4. 11. Curve of y -axis displacements of edge nodes along stringer length for the case of models with and without ply stop variability after cooling

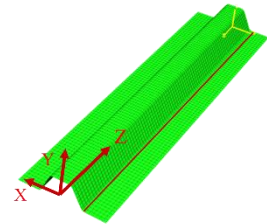
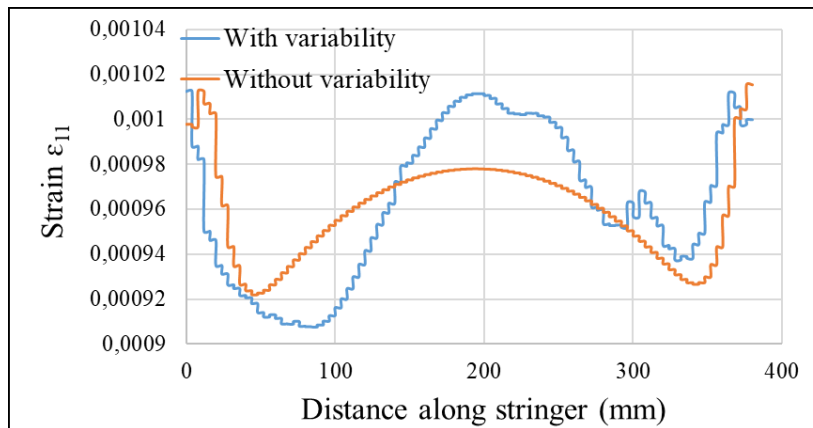


Fig. 4. 12. Curve of Strain ε_{11} of edge nodes along stringer length for the case of models with and without ply stop variability after cooling

Fig. 4. 11. represents the evolution of the vertical displacement along a line of nodes, the line located in the junction zone between the web of the stringer and the bottom flange after cooling. The orange curve represents the evolution of the displacement that would have been obtained for a stringer without any variability in the location of the ply stops, as foreseen at the time of

the design. The blue curve represents the same evolution for a virtual stringer containing ply stop variabilities, variabilities presenting realistic levels and evolutions concerning all the observations made. As expected, the presence of variability generates some differences in the magnitude illustrated compared to the design model, and the simplified model proposed seems able to quantify this difference. Here the major difference encountered in the behaviour between the designed part and the realistic virtual part leads to a deflection difference of less than a tenth of mm at the centre of the structure.

Additionally, the proposed model allows observing differences in stress levels, deformations, failure criteria, etc. For example, Fig. 4. 12. represents the evolution of the strain in the direction of the element oriented towards the global x-direction, always for a line of nodes located at the web/bottom flange junction, and along the stringer. This direction for result visualisation is chosen because it shows the variability effect the most remarkable. On the example chosen here for our illustration, the presence of variabilities generates spatial strain evolutions that are significantly different from the evolution observed on a design model that does not include variabilities, with in particular a loss of expected symmetry in the length. However, the order of magnitude of the observed deformations remains very close under this load case, seeming to demonstrate a low sensitivity of the observed magnitude to the selected variability extent (real position of the ply stops) to the selected manual draping method.

The proposed modelling and variability insertion strategy thus seem to yield results that note a variation in mechanical behaviour when inserting variabilities. The proposed tool, therefore, seems suitable for studying the influence of the selected variability.

According to Fig. 4. 11., gaps in displacement are not so huge between theoretical and numerical models. In Fig. 4. 12., the differences between theoretical and real models are obvious concerning symmetry loss, non-constant values ... especially when we consider strain analysis, where ϵ_{11} strain is chosen as an example. ϵ_{11} strain can be more or less large than the theoretical one depending on gap and position, but this difference is low (about 3 %).

6.1.3. Numerical Results after cooling and tensile loading

The created model is capable of determining variability effects in different cases of applied load. Fig. 4. 13. shows an example of mechanical behaviour under cooling followed by a tensile loading. This case of loading can give us an idea about the influence of ply stop variability on

a composite structure with complex geometry in mechanical cases, also, to conclude a range that defines the maximum and minimum variability value in the case of ply stop.

In Fig. 4. 13., the differences between the model without variabilities and a random model are presented qualitatively that including variabilities in the case of vertical displacement and shear strain fields.

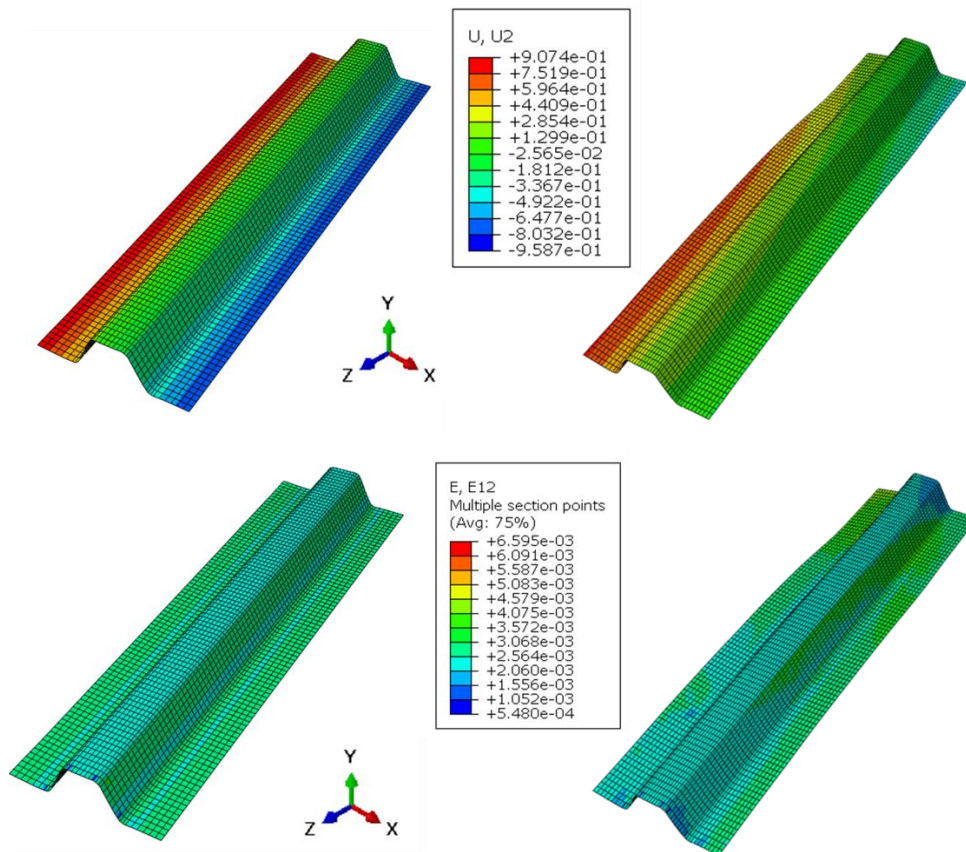


Fig. 4. 13. Comparison between models without and with variability (random draw from input data from the 4 stringers) in the case of (a) displacement in y-direction and (b) shear strain fields after tension preceded by cooling

This example seems to show again that the numerical strategy is sensitive to the variability studied, with a significant influence predicted on the ply stop location on the shape of the Ω -shape stringer under this case of loading. The displacement U2 gap reaches almost 0.9 mm between the designed structure and the virtual structure containing variable and realistic ply positions concerning the selected manufacturing process. Similarly, the shear strain field of the median ply shows significant variations, ranging for some elements in particular from + 6.6e-3 on the model without variabilities to 5.48e-4 for a random model (with variabilities) used in the illustration.

To have an order of magnitude of the gap variations, Monte-Carlo draw is conducted in the next paragraph. A repetitive analysis of the created model allows observing the influence of ply edge locations on the behavior for further structures.

6.1.4. Repetitive analysis for Monte-Carlo draws after cooling for ply stop variability

According to the previous results, the modelling strategy seems to fulfil the objective of allowing the insertion of ply position variabilities and having a mechanical response sensitive to this quantity. Moreover, this strategy leads to very short computation times, of the order of 360 seconds for a millimetre mesh and a 380 mm long structure, leading to about 57 000 finite elements. This strategy, therefore, allows considering launching a large number of calculations in which we can vary the position of the ply stops while being realistic about the extent and the evolution of these positions.

Repetitive analysis of the created model allows observing the influence of ply edge locations on the behavior for further structures. The gap between the theoretical designed model and the manufactured piece can be important due to many factors explained in previous sections. This influence is translated into output parameters such as strain analysis and Failure criterion (Tsai-Hill in our case).

As an example of the capacity of the proposed numerical strategy to perform numerous calculi, Fig. 4. 14. shows the maximal magnitude of the shear strain ε_{12} all along with the structure depending on the maximum ply position deviation in the model. The choice to show ε_{12} is due to the presence of woven plies at $\pm 45^\circ$ which a priori affects more the shear modulus in this case of loading. In Fig. 4. 14., the design Ω -shape stringer without variability is the red dot and twelve random stringers including ply stop variabilities are the blue dots. We can observe that shear strain slightly increases when the maximum gap (between theoretical and real ply stop locations) increases, which means when plies are more displaced from their theoretical position (red point). In this particular case, it can be noted that for ply stop position gaps less than 2 mm, the maximum value of shear strain seems to be less than the value obtained for the non-variability calculus. On the contrary, from a certain value of position error realized by the operator, the maximum value of shear deformation will exceed up to 60 % of the value predicted in the case without variabilities. For maximum ply location gap equal to 6 mm and 6.2 mm, the shear strain values are less than the value related to the case without variability consideration. Maybe in this case, the maximum gap is associated to plies that don't have much influence on the shear strain value.

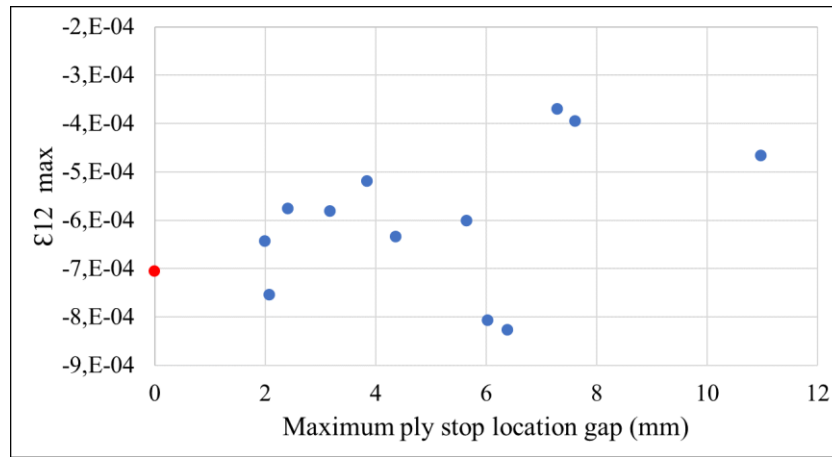


Fig. 4. 14. Curve of maximum strain in function of maximum gap between real and theoretical ply edge locations (red point indicates the value in theoretical case without variability) after cooling

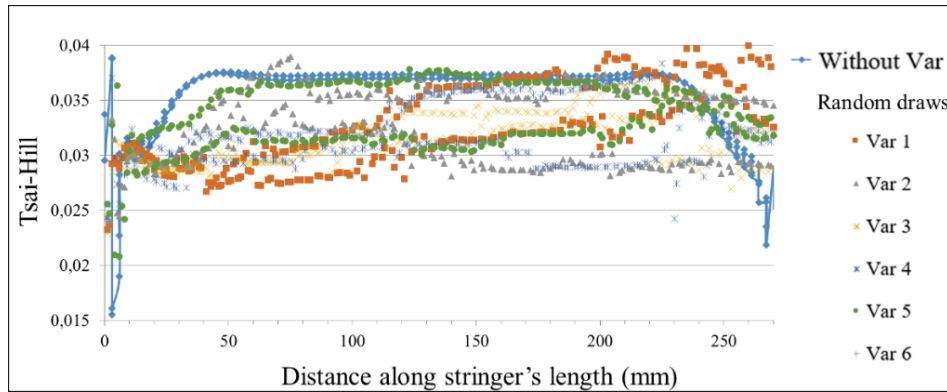


Fig. 4. 15. Tsai-Hill criterion along the Ω-shape stringer length (see complementary scheme) in different calculations of variability (Var 1 to 6) after cooling

The proposed numerical strategy also allows displaying breakage or failure criteria. As an example, Fig. 4. 15. represents the evolution of the Tsai-Hill failure criterion that takes into account the interaction between the different stress components on a length of stringer of 270 mm (not 380 mm total length) to reduce calculation time. Its limit is that it does not directly inform on the mode of failure modes including fibre failure, matrix failure, and fibre-matrix interface failure. Fig. 4. 15. shows that the constructed numerical model can launch large scale calculations, and to identify value intervals for the quantities of interest, here for example the criterion of Tsai after cooling for the web bottom. The blue curve represents the model without taking into account the variabilities. We can note on this example, that the variabilities of ply stop locations tend to decrease very slightly the values of the residual stresses that are set up after the cooling (8% increase and about 20% decrease).

6.2. Illustration of numerical results with fibre misalignment variability

6.2.1. Twin models taking into account fibre misalignment variability in the case of tensile loading after cooling

Twin numerical models are based on the actual variability levels found experimentally in order to evaluate their behaviour under different loading forms. In this section, we will consider only fibre misalignment variability into the numerical model without random draw. The goal here is to compare the behaviour of the different stringers to determine the fibre orientation variability effect in a family of parts manufactured using the same process. the case of loading chosen is the tensile loading after cooling because this case allows to visualise the variability effect more than the case after only cooling.

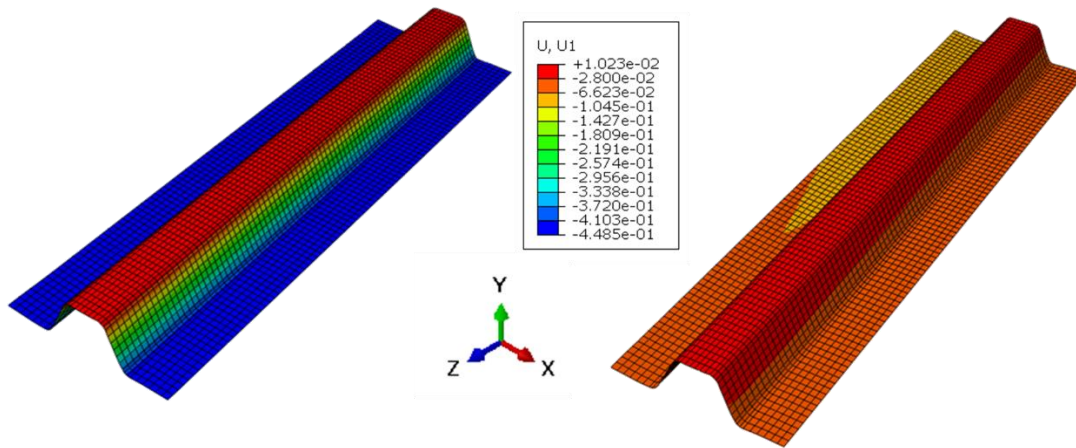


Fig. 4. 16. Comparison between displacements in the horizontal direction for numerical models without (left) and with (right) fibre misalignment consideration in the case of mechanical loading after cooling

Fig. 4. 16. shows the difference between models with and without fibre misalignment consideration in the case of tensile mechanical loading after cooling (imposed displacement of 1 mm explained earlier). Compared to the model without variability consideration, the model containing fibre orientation uncertainty present important horizontal displacement in the x direction. In the bottom flange areas, there is a difference in the displacement with about 0.38 mm. There is an obvious difference also in the web area with 0.19 mm in terms of displacement along x direction. The symmetry between the left and right side of the section is also lost. This aspect is clearer in the Fig. 4. 18. here after.

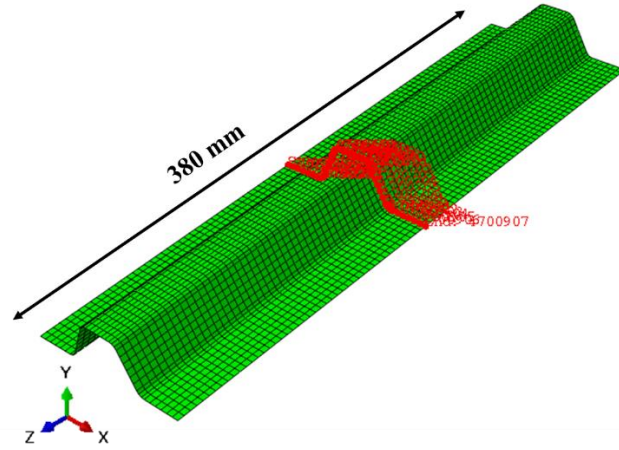


Fig. 4. 17. Schematic representation for the path used for results analysis

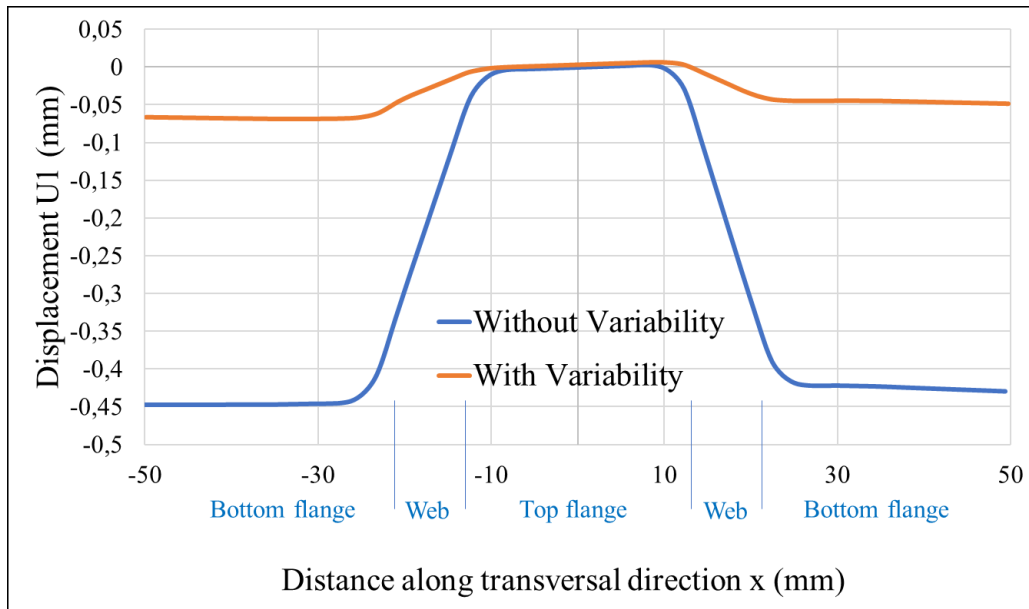


Fig. 4. 18. Curves for x -direction displacement U_1 along the transverse direction for cases with and without fibre misalignment variability consideration

In this case, z is fixed to the central position of the stringer's length (cf. Fig. 4. 17.). We can see that in the case without variability consideration, the transversal displacement is symmetric and between -0.01 and 0.01 mm. When adding fibre orientation variation, transversal displacement become more important especially in the bottom flange areas (about 0.08 mm). In the top flange area ($x \in [-12, 12]$ mm), the displacement increases by 0.01 mm. Symmetry loss is obvious especially in the top flange area ($x \in [-12, 12]$ mm) and bottom flange areas. The choice of this type of results visualisation is more or less arbitrary. It lies on the fact that the objective here is to show the effect of variabilities on the model no matter what we choose as output. The

displacements U1 seem to show here the difference between models with and without variability consideration.

Fibre misalignment has an effect on the behaviour of the structure in the case of imposed displacement after cooling. But this effect is not important compared to ply stop variability consideration. This is more noticeable in the case of Tsai-Hill criteria (cf. Fig. 4. 19.). There is almost no difference between model response with and without variability. The maximum value is about 0.7 situated in the stringer's top flange and corners between webs and bottom flanges.

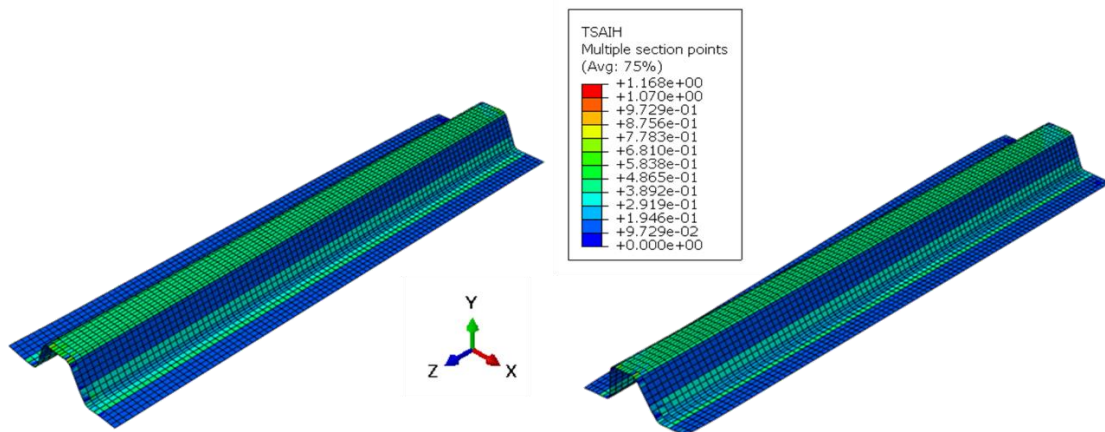


Fig. 4. 19. Comparison between Tsai criteria for numerical models without (left) and with (right) fibre misalignment consideration in the case of mechanical loading after cooling

6.2.2. Numerical repetitive results after thermal and/or mechanical loading

In order to be able to quantify a variation range related to porosity on the structure, a repetitive calculus is performed on the numerical model that takes into account fibre misalignment variability with different random draws of fibre angles in the structure.

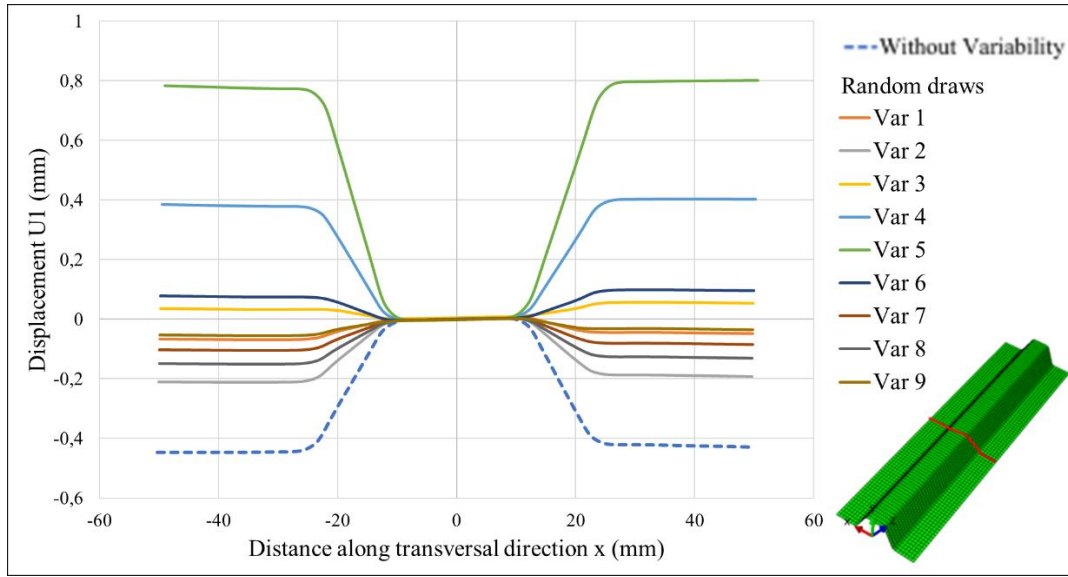


Fig. 4. 20. Repetitive calculation of horizontal displacement $U1(z=0)$ with different random draws (1 to 9) for fibre misalignment variability analysis in the case of mechanical loading after cooling

Fig. 4. 20. shows that the displacement $U1$ along the distance in the transversal direction in the location $z=0$ is between about -0.45 mm and 0.8 mm. The case without variability consideration presents the smaller displacement $U1$. The case without variability is symmetric also along $y=0$ unlike other cases where the displacement symmetry is lost. The curve named Var 5 presents a maximum gap between theoretical and variability fibre misalignment per ply about 6° . The maximum displacement for this variability value is 0.8 mm in the bottom flange area. For the curve named Var 9, the maximum angle gap is about 1.96° , which corresponds to the minimum displacement gap for models with random draws. We can notice that in this particular location and for this type of output, fibre misalignment can increase the displacement gap $U1$ when the angle gap is important, while it can also decrease it when the angle gap become small.

It appears that this type of variability has its effect on the mechanical response of the structure and on a family of stringers manufactured using the same process. When the angle gap between the model without variability and with variability increases, the displacement increases.

6.3. Illustration of numerical results with porosity rates variability

In this section, we are considering only porosity rates variability in the analysis of the structures response to the different chosen load types, thermal and mechanical. In the literature, porosity was been studied in different levels, experimentally, numerically etc. For numerical analysis, in [2] for example, multiscale 3D FE models are developed and periodical boundary conditions

are applied to the representative volume element models to obtain elastic modulus of the composite. The obtained results indicate that void defects exert significant influence on the elastic moduli of fibres. The elastic moduli of 3D woven composites are more sensitive to the voids in fibre tows than those in the matrix of woven composites [2]. Also, in [3], a numerical methodology is developed for simulating the mechanical behaviour of porous CFRP unidirectional (UD) laminates by exploiting data extracted from X-ray computed tomography scans. The numerical results show a small decrease of transverse stiffness (2.3%) and a significant decrease of transverse tensile strength (14%) with increasing the pore content (45%) in the case of a composite laminate.

Our methodology examines porosities in a composite structure with complex geometry. The originality lays in the numerical modelling using shell model to be able to lunch multiple simulations in a short time.

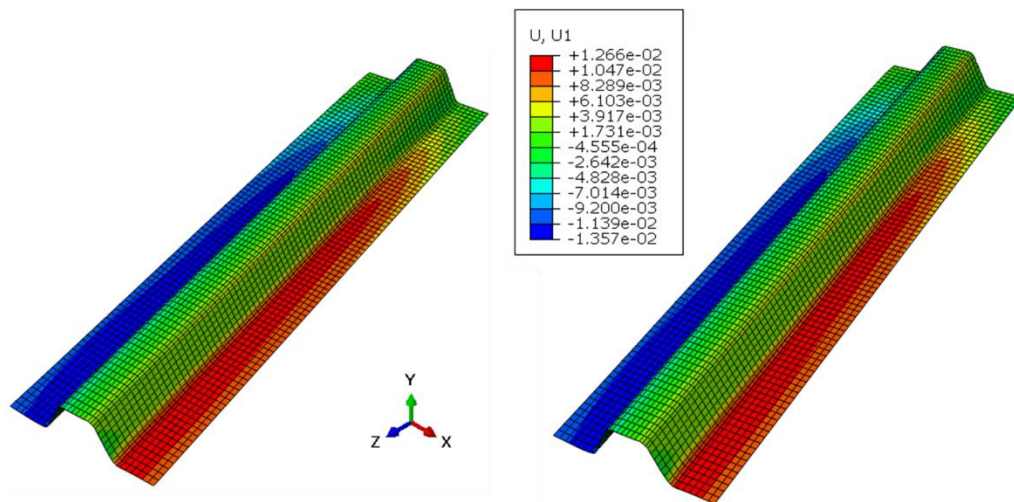


Fig. 4. 21. Comparison between numerical modelling without (left) and with (right) porosity variability consideration for mechanical response after cooling

Considering as an example the transverse displacement $U1$, the models with and without variability in the case of porosity variability taking into account do not present significant difference. This is more noticeable in Fig. 4. 22., where we trace the horizontal displacement $U1$ along transversal direction x for $z = 0$ for mechanical response after cooling.

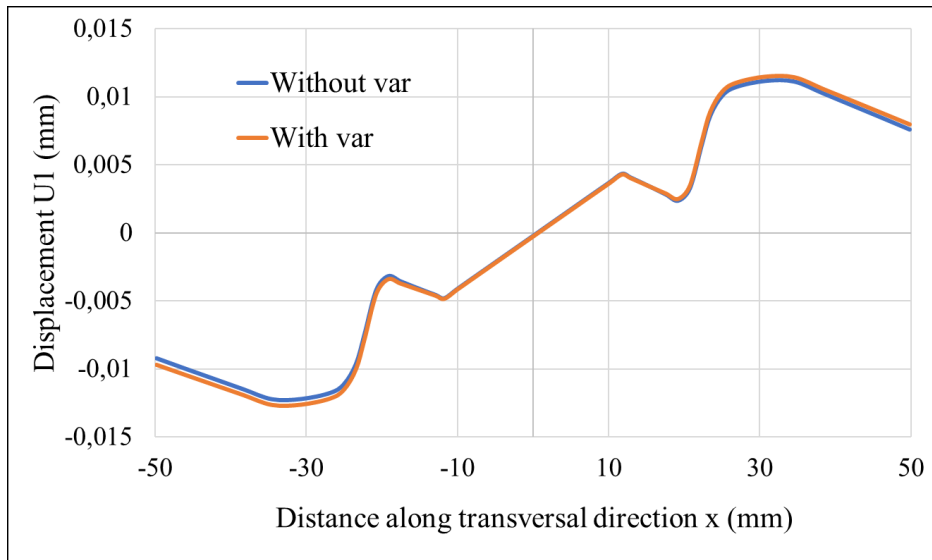


Fig. 4. 22. Curves for displacement $U1$ along the transverse direction for cases with (orange curve) and without (blue curve) porosity variability consideration for mechanical response after cooling with results location scheme in the stringer length

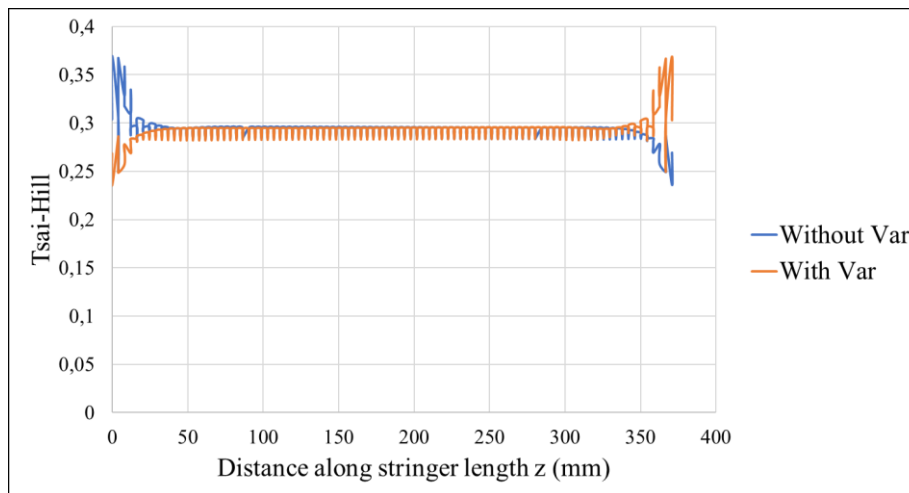


Fig. 4. 23. Curves for Tsai-Hill criterion along the transverse direction for cases with (orange curve) (one random draw) and without (blue curve) porosity variability consideration for mechanical response after cooling with results location scheme in the stringer length

Considering Tsai-Hill criterion in z direction along the stringer in the corner between the web and the bottom flange as result (cf. Fig. 4.23), the difference between curves between model without and with porosity variability consideration is not obvious in most of the structure. But in the stringer's edges there is a difference by about 0.1 on both sides. This direction is chosen because it shows the most difference between the two models.

Voids presence in a structure has an effect mainly on the mechanical properties of the considered material. This fact is proven also in many works of the literature.

7. Numerical model results with three types of variability taken into account

After studying the chosen forms of variability in this study, one by one, we are representing the effect of these variabilities on the global structure's behaviour in this section. The aim is to quantify ranges of variation of the structure's response to variability consideration and to show the difference between one structure and another manufactured using the same process.

We have to note that our overall objective is not to study all the forms of variability presented in a composite structure, but to introduce a methodology that is able to handle various forms of studied variability within a numerical model based on realistic data for structures presenting complex geometry zones.

7.1. Numerical modelling results taking into account variabilities after cooling

Starting with the initial state of the structure, this section contains the analysis of numerical modelling taking into account ply stop, fibre misalignment and porosity rate variabilities after cooling after polymerisation process.

Fig. 4. 24. illustrate this difference by showing results in the case of horizontal displacement along x axis for a case of one variability random draw. The symmetry loss is obvious especially at the stringer's edges. In the corner between the bottom flange and the web, there is a difference in displacement with about 0.01 mm.

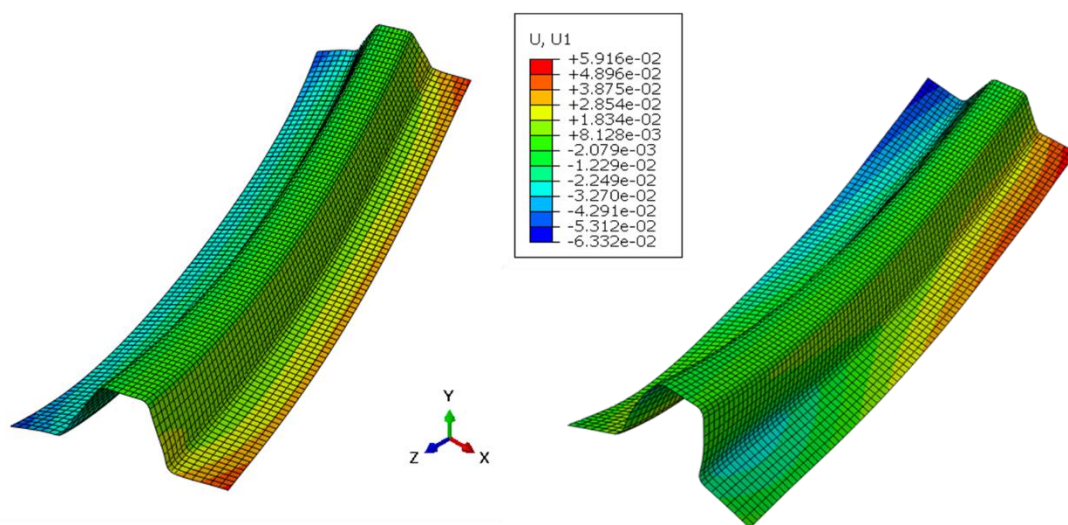


Fig. 4. 24. Comparison between numerical modelling of horizontal displacement U_1 without (left) and with (right) total studied variabilities consideration after cooling

This difference can be noticed also in Fig. 4. 25., where we represent the displacement $U1$ along the transversal direction when $z=0$. The difference occurs mainly in the area between $x= -50$ mm and $x= -20$ mm (bottom flange area) with about 0.003 mm.

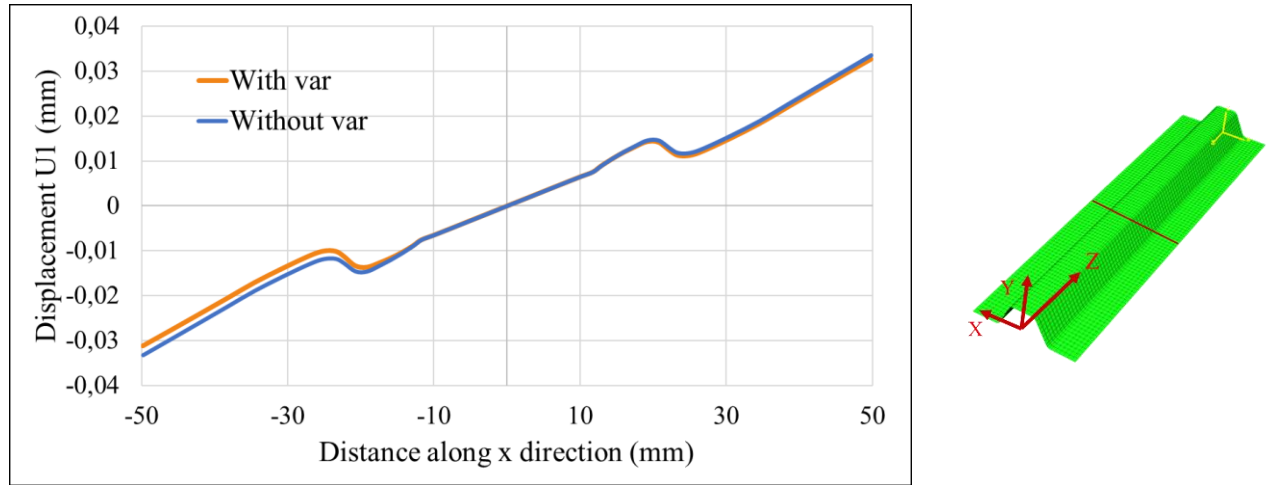


Fig. 4. 25. Curves for displacement $U1$ along the transverse direction for cases with (orange curve) and without (blue curve) variability consideration after cooling with scheme for result location in the stringer length

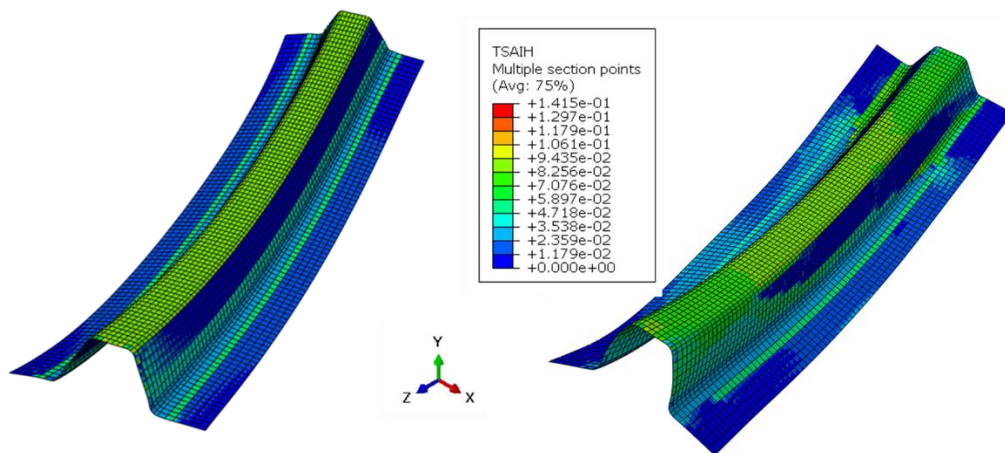


Fig. 4. 26. Comparison between numerical modelling of Tsai-Hill criterion without (left) and with (right) total studied variabilities consideration after cooling

Fig. 4. 26. shows that the difference between models with and without variability after cooling is not outstanding in terms of maximum values for Tsai-Hill criterion. But we can note that variabilities increase Tsai-Hill failure values in the web area by about 0.01.

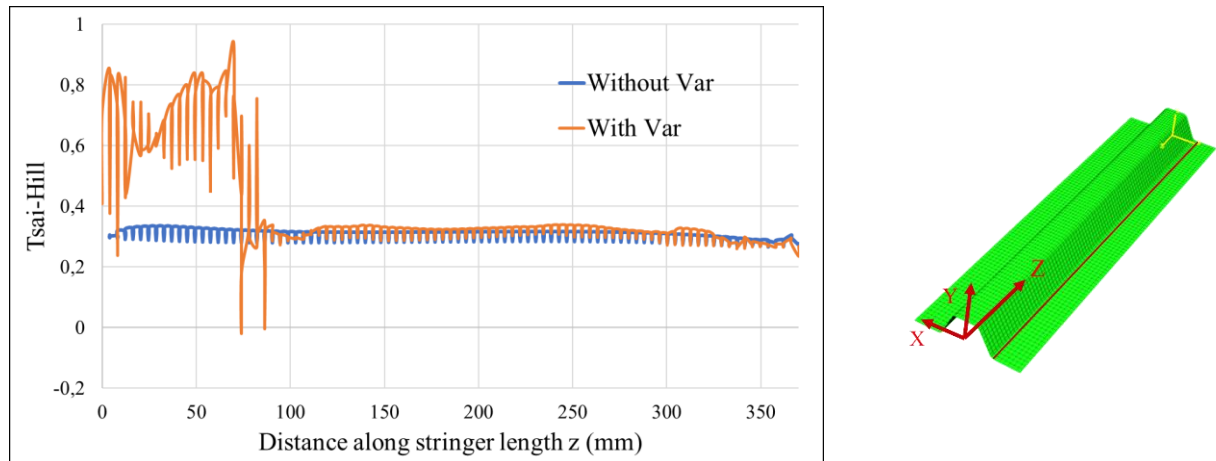


Fig. 4. 27. Curves for Tsai-Hill criterion along the stringer length for cases with (orange curve) and without (blue curve) variability consideration after cooling with scheme for results location

Fig.4.27 shows that in the corner between the bottom flange and the web, Tsai-Hill criterion is different in models with and without variability consideration after cooling especially in the zone where z is between 0 and 70 mm (difference up to 0.06).

Although the initial state of the structure after cooling is affected slightly (9% for Tsai-Hill criterion for example) by the presence of variabilities, this effect should not be ignored in the design phase.

7.2. Numerical modelling results of influence of variabilities after mechanical loading preceded by cooling

Fig. 4. 28. presents the difference between models with and without taking into account variabilities for a type of displacement which is horizontal displacement U_1 . In the top flange for example, this difference between models in term of displacement is about 0.04 mm and - 0.13 mm in the left bottom flange area for mechanical loading after cooling.

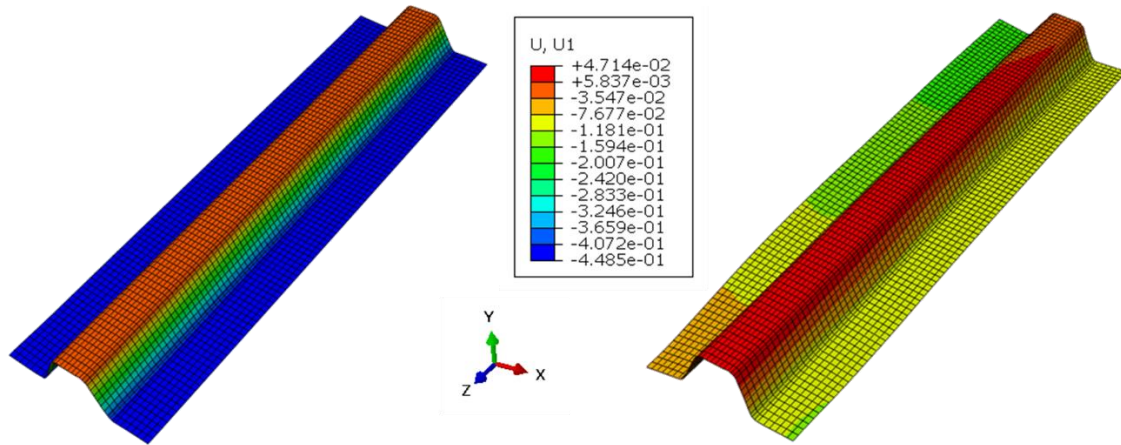


Fig. 4. 28. Comparison between numerical modelling of horizontal displacement $U1$ without (left) and with (right) total studied variabilities consideration for mechanical loading after cooling

Considering a specific location with $z = 0$ (cf. Fig. 4.29), the effect of considered variabilities is observable in the bottom flange areas where the symmetry is lost compared to the model without variability consideration. The difference in terms of maximum displacement in this location between the two models is about 0.2 mm in the bottom flange. It appears that variabilities in this case trend to reduce (in absolute value) the displacement in the bottom flange area. In the top flange area, the maximum value of displacement $U1$ is almost the same considering or not variabilities. This is due the fact that this area contains the minimum amount of variability (less ply stops, less fibre misalignment) although it contains the maximum values of porosity rates. This proves that porosity rates effect is not huge compared to the other types of variability taken into account.

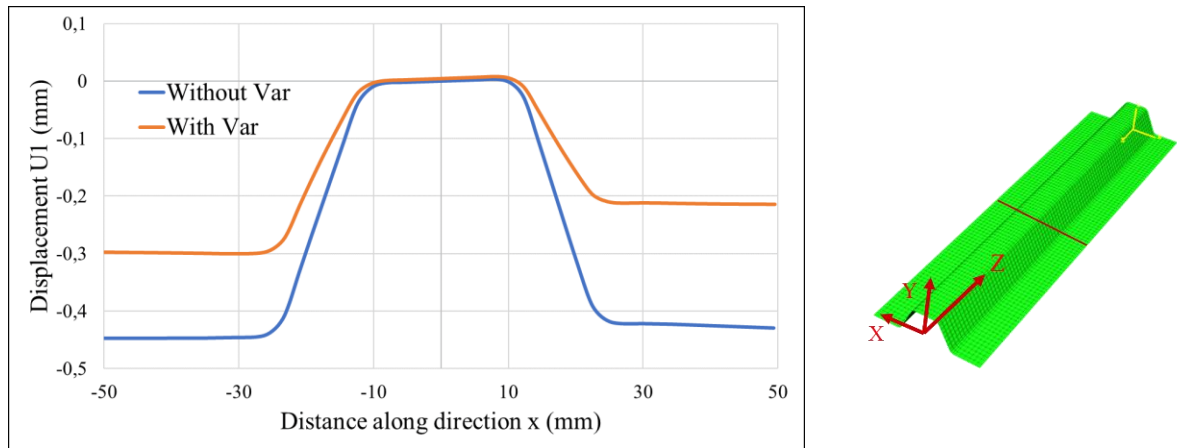


Fig. 4. 29. Curves for x-direction displacement U_1 along the transverse direction for cases with (orange curve) (one random draw) and without (blue curve) variabilities consideration after mechanical loading after cooling with scheme for result location

As for Tsai-Hill criterion (cf. Fig. 4.30), the symmetry loss in the stringer section is clear especially between the web areas for a random drawing. The criterion values increase mostly in the web area but it remains almost the same in the top flange areas between models with and without variability consideration.

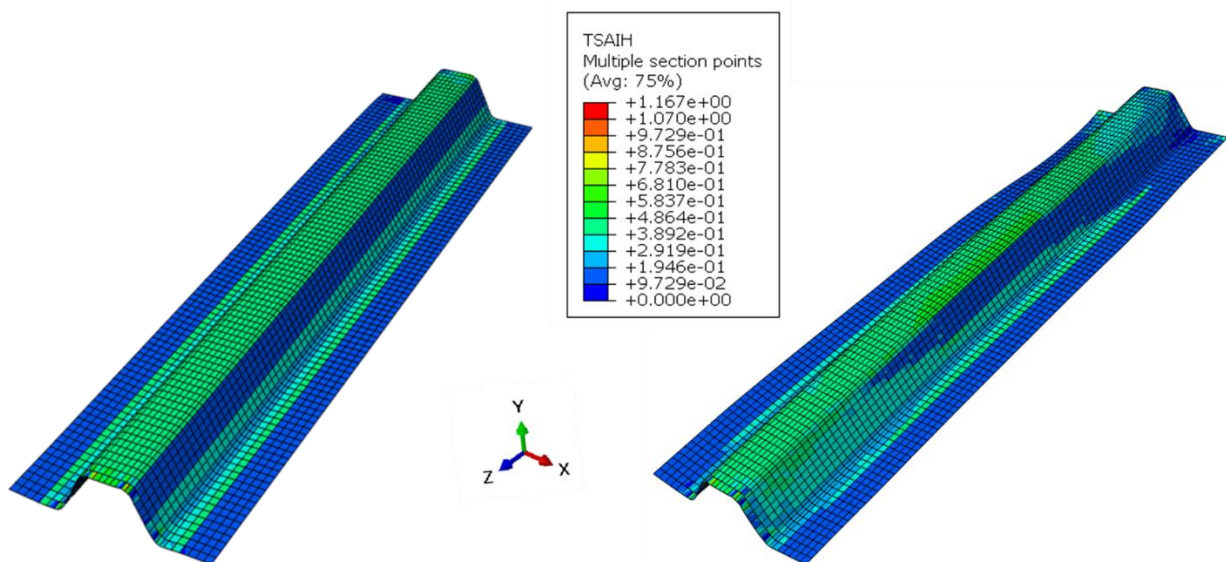


Fig. 4. 30. Comparison between numerical modelling of Tsai-Hill criterion without (left) and with (right) total studied variabilities consideration for mechanical response after cooling

Considering the location $z=0$ in the stringer's length, the Tsai-Hill criterion (cf. Fig. 4.31) with variabilities taken into account (ply stop location, fibre misalignment and porosity rate)

increases by 0.05 in the top flange area compared to the model without variability. There is a small difference also in the corner between web and bottom flange areas ($-35 \text{ mm} < x < -22 \text{ mm}$) by about 0.04. The symmetry is also lost in the Ω -shape stringer cross-section.

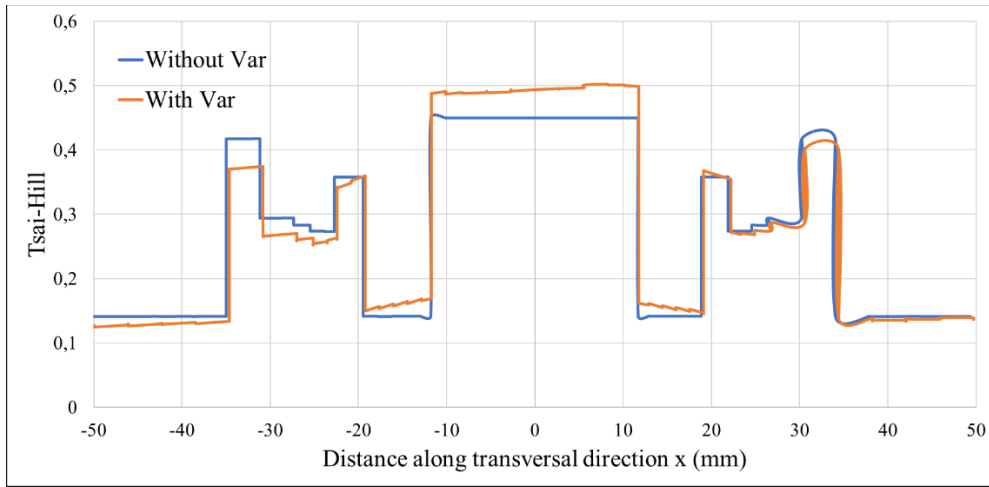


Fig. 4. 31. Curves for Tsai-Hill criterion along the transverse direction for cases with (orange curve) and without (blue curve) variabilities consideration for mechanical loading after cooling

7.3. Numerical repetitive results after thermal and/or mechanical loading taking into account all the studied variabilities

The goal of this section is to identify a variation range of selected outputs along random drawing of variabilities based on numerical modelling.

For example, considering the variation of the displacement U_1 (cf. Fig. 4.32) along x direction (horizontal) when $z=0$ (see attached scheme in Fig. 4.32), after 5 different random draws, variabilities seem to decrease the maximum value of displacement especially in the bottom flange area. The curve named Var 5 corresponds to the minimum value of variability (maximum angle gap = 1.25° and the maximum gap between theoretical and real ply stop location = 3.34 mm). The curve named Var 4 corresponds to variability values of 13.2 mm for ply stop location gap and 6.9° for fibre angle gap. The maximum value of displacement is obtained in the case of model without variability with -0.45 mm in the centre of the stringer ($z=0$).

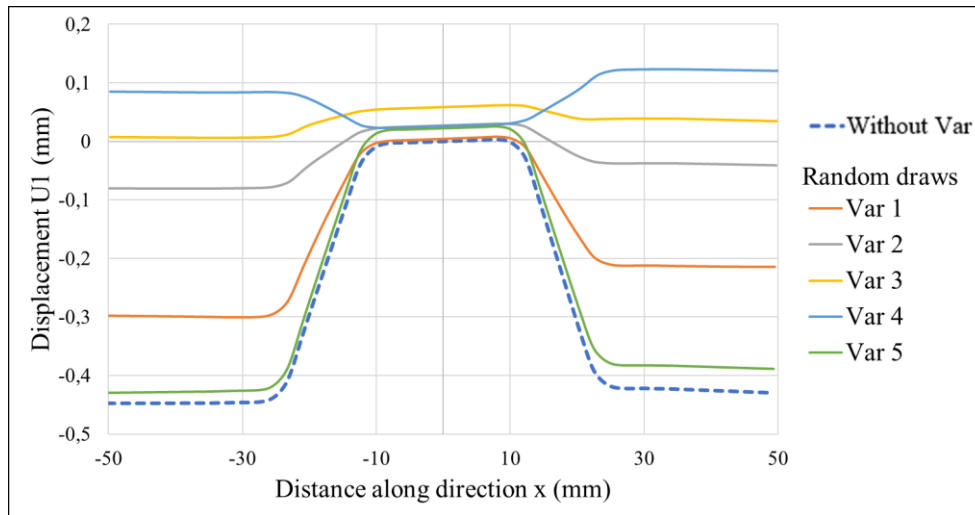


Fig. 4. 32. Repetitive calculation of horizontal displacement U_1 ($z=0$) with different random draws (Var from 1 to 5) for variabilities analysis in the case of mechanical loading after cooling

Moving to Tsai-Hill criterion analysis, the repetitive calculi in the corner between the web and the bottom flange give the curves here after (cf. Fig. 4.33). Variability influence tends to decrease very slightly the values of the residual stresses that are set up after the mechanical loading preceded by cooling. We can notice about 25% increase and about 60% decrease in the borders of the stringer ($z < 75$ mm and $z > 320$ mm) while in the stringer's centre area, the decrease is about 16%. Variabilities trend to smooth Tsai-Hill criterion in the chosen particular zone.

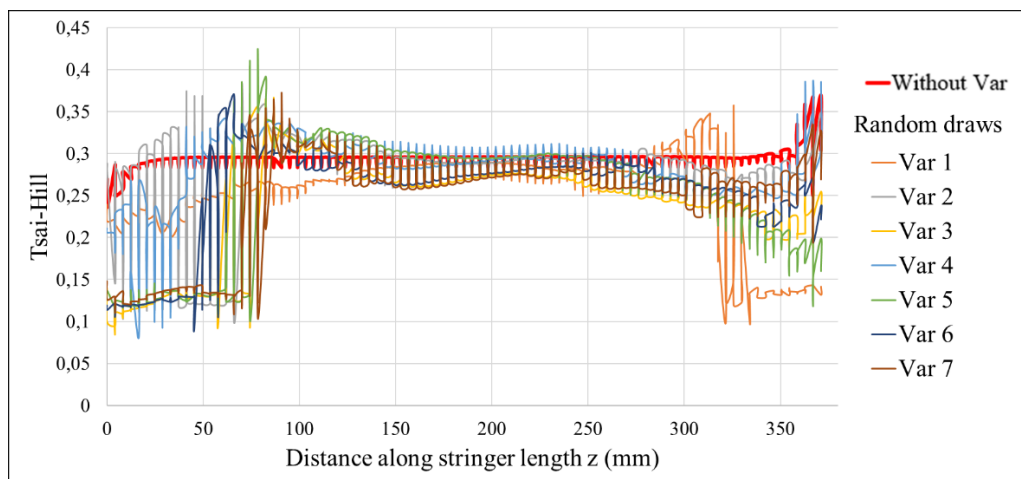


Fig. 4. 33. Repetitive calculation of Tsai-Hill criterion along Ω -shape stringer's length with different random draws (Var from 1 to 7) for variabilities analysis in the case of mechanical loading after cooling

Conclusion

This chapter studies the numerical modelling nourished using the statistical tendencies with the chosen type of variability. Random draws of statistical parameters were made to obtain a probabilistic model that can predict the shape and behaviour of stringers under thermal and/or mechanical loadings. The set of mathematical models from chapter 3, proposed for the representation of the spatial evolution within the part respectively of the ply stops, the fibre orientations and the porosity rate, were expended for the creation of a Finite Element model taking into account local variations of properties. The model was based on 2D shell elements, to be able to run a large number of simulations based on random draws of the parameters of the mathematical laws studied. In this proposed model, the values of the material and geometric parameters change not only from one element to another, but also for the same element, for each of the 20 plies constituting the shell element of the composite structure.

The modelling strategies have been designed to be light and allow for multiple calculations. It was observed also that although the “bad chosen” manufacturing conditions, huge differences were not found after numerical analysis. Results show that the chosen studied variabilities have an effect on the final shape and the behaviour of the structure. A difference between the numerical twins of the 4 manufactured stringers was observed after cooling.

Ply stops location variability is the most uncertainty that has an effect on the structure behaviour. The presence or not of a ply has the most obvious influence. In some cases, variabilities seem to decrease the strain or Tsai-Hill values. This might be due to the fact that variabilities can overlap each other. For example, for ply stop location variability, a ply may have an important gap compared to the theoretical case, but in the same location (x, z), another ply may have the same gap but in the other direction, which causes a decrease in the total variability.

All the found results prove that variabilities should be taken into account in composite structures design especially in the aeronautic field, in order to minimise their effect that leads to reduce the costs.

References

- [1] Davila Y, Crouzeix L, Douchin B, Collombet F, Grunevald Y-H. Identification and Modelling of the In-Plane Reinforcement Orientation Variations in a CFRP Laminate Produced by Manual Lay-Up. *Applied Composite Materials* 2018;25. <https://doi.org/10.1007/s10443-017-9642-4>.
- [2] Huang T, Gong Y. A multiscale analysis for predicting the elastic properties of 3D woven composites containing void defects. *Composite Structures* 2018;185:401–10. <https://doi.org/10.1016/j.compstruct.2017.11.046>.
- [3] Tserpes KI, Stamopoulos AG, Pantelakis SG. A numerical methodology for simulating the mechanical behavior of CFRP laminates containing pores using X-ray computed tomography data. *Composites Part B: Engineering* 2016;102:122–33. <https://doi.org/10.1016/j.compositesb.2016.07.019>.
- [4] Davila Y, Crouzeix L, Douchin B, Collombet F, Grunevald Y-H. Spatial Evolution of the Thickness Variations over a CFRP Laminated Structure *Applied Composite Materials* 2017;24. <https://doi.org/10.1007/s10443-016-9573-5>.

General conclusions and future work

The objective of this thesis was to set up an approach allowing the evaluation of the effects of variabilities in mechanical calculations applied to composite materials. As these materials are more subject to uncertainty problems than others, in particular metallic ones, we had to find a measurement methodology during and after manufacturing process to quantify the variabilities in order to have a realistic input data set to be able to evaluate their effects on the behaviour of the part. The choice was to work on a part inspired by an aeronautical composite fuselage, which represents geometric singularities such as the presence of ply stops, varying thickness, convex and concave shapes, and small radii.

The literature review, presented in chapter 1 entitled Bibliography, demonstrates that there is no common methodology for the study of variabilities in composite structures regardless of the evidence that composite materials present a dispersion in their properties. Usually during research, the variability in composite structures is considered either as local defects or as uncertainties in stochastic analysis. For the former consideration, dedicated models are created in order to study the effects of localised defect on the composite structure. For the second objective, a broader set of general approaches is employed such as probabilistic studies, reliability-based approaches and stochastic finite element methods. The Monte Carlo simulation is still the most common method due to the simplicity of its formulation. The model variables include the material properties which are obtained from probability functions. The end of this chapter is dedicated to the presentation of the working methodology and the associated object of study.

A part inspired from a composite fuselage stringer was studied. In addition to composite uncertainties, the chosen structure present geometric singularities (convex and concave shape, small radii, thickness variations, the association of several materials etc.) that enhance variability presence. The part considered had an omega-shaped structure consisting of 20 plies from preforms with unidirectional and woven reinforcements. The prepregs are Hexcel materials, respectively, M79/34%/UD300/CHS for the UD and M79/42%/285T2/CHS for the woven. The structure was manufactured using a dedicated mould, including a core for internal geometry control, and cured in a single stage in an oven.

The 2nd chapter entitled Experimental setups and protocols for variabilities observation within complex geometries: from raw material to final structure, details about the experimental

protocol used and the variability observations for the analysis of ply stops, porosity and fibre misalignment. The observations were carried out throughout the manufacturing process, using a wide range of measurement techniques such as optical, micrographic, tomographic observations, etc. Each observation was dedicated to the monitoring of one or more types of variability studied for different phases of the manufacturing process. Once the structure is consolidated, it was scanned in 3D to identify its real geometry, which, due to the presence of internal stresses, is different from the theoretical geometry. The fabricated part is then cut out for observation. Microscopic and tomographic analyses were performed to determine the variations of the studied parameters, mainly the positions of the ply edges, the fibre misalignment along the length and the size and the extent of the porosities that appeared during the consolidation phase. All the observations done during every manufacturing step serve to obtain a lot of data sets from raw material to complete structure analysis to feed a set of input data for numerical modelling. For example, in the lay-up phase, a minimum number of 384 measure was taken using microscopic observation. The observation and measurement time was up to 2 hours per cross-section. Tomographic observation takes about 6 hours per sample.

These observations are analysed in the 3rd chapter intituled Spatial evolution of variability within a complex geometry structure: Mathematical laws and statistical trends. First, the observed spatial evolutions are represented in the form of a continuous mathematical law representative of the observed evolutions. Here, polynomial evolutions have been selected to model ply stop variability in both stages of cutting and the ply deposit in the mould. The polynomials were chosen of degree 5, sufficient to describe the undulations observed over a stringer length of about 400 mm. The six parameters describing these polynomials were analysed from a statistical point of view, by separating the measurements into four families of plies, determined according to characteristics such as strand orientation or position with respect to a geometrical feature. The category 1 (6 plies and 1 measure/ply in a side) has $6 \times 4 = 24$ set of parameters (curve number $\times$ manufactured stringer's number). The category 2 (2 plies and 2 measures /ply in two sides) has $4 \times 4 = 16$ set of parameters. The category 3 (10 plies et 1 measure /ply in one side) has $10 \times 4 = 40$ set of parameters. The category 4 (4 plies et 2 measures /ply in two sides): $8 \times 4 = 32$ set of parameters. Fibre waviness was characterised using a continuous mathematical representation based on literature work and was represented by a finite sum of deformed zones, using a Gaussian surface centred on the centre of the perturbations. The amplitudes of the perturbations are statistically studied with an adaptation to the characteristics of the structure

studied from the observations made. From its side, the void content was evaluated by normal laws in terms of position and extent in the horizontal plane and represented by statistical laws conducted according to their location in the cross-section of the studied stringer based on tomographic observations.

These statistical tendencies are used to nourish numerical modelling with the chosen type of variability in the 4th chapter intitled Numerical analysis of variability and integration of different studied variability forms in the numerical modelling. Random draws of statistical parameters were made to obtain a probabilistic model that can predict the shape and behaviour of stringers under thermal and/or mechanical loadings. The set of mathematical models from chapter 3, proposed for the representation of the spatial evolution within the part respectively of the ply stops, the fibre orientations and the porosity rate, were extended for the creation of a Finite Element model taking into account local variations of properties. The model was based on 2D shell elements, to be able to run a large number of simulations based on random draws of the parameters of the mathematical laws studied. In this proposed model, the values of the material and geometric parameters change not only from one mesh to another, but also for the same mesh, for each of the 20 plies constituting the shell element of the composite structure.

The modelling strategies have been designed to be light and allow for multiple calculations. This approach opens the door to insert other types of variabilities in the same numerical model, such as ply thickness and mechanical properties for composite parts with complex geometry. It was also observed that although the “bad chosen” manufacturing conditions, relatively small differences were found after numerical analysis. Results show that the studied variabilities have an effect on the final shape and the behaviour of the structure. A difference between the numerical twins of the four manufactured stringers was observed after cooling

This proves that the variabilities should be taken into account in composite structures design especially in the aeronautic field, in order to minimise their effect that leads to reduce the costs.

Among the load cases selected to highlight the operability of the strategy, we applied our probabilistic approach to the case of a cooling at the end of the consolidation phase, when the residual deformations are set up. Cooling phase may also be studied taking into account physico-chemical effects. The long-term objective of this type of study is, after having accumulated a certain number of variabilities or by treating the variabilities independently of each other, to be able to redefine the amplitudes of admissible variabilities for a particular

structure. For example, if it is demonstrated that the positional deviation of certain plies can be increased without significantly degrading the performance of the composite structure, it is possible to consider reductions in the manufacturing costs of certain parts. In the same way, the numerical consideration of realistic variations between theoretical and manufactured quantities, either in terms of amplitudes or spatial evolutions, can help to redefine the safety factors or the allowances used during the validation of the structural behaviour of the composite part in the design phase.

It was demonstrated by the proposed finite element model; we are able to highlight the influence of variabilities on the mechanical responses of a composite structure. However, the FE model continues to be a work in progress, therefore, efforts leading to its constant development must be made.

The technical aspects of the FE model that should be addressed on the short-term deal with the automation of the input files generation, analysis and post-processing. This step is mandatory in order to be able to evaluate very large number of simulations needed for the probabilistic analysis of composite structures. An optimisation of the input files is necessary to reduce their size and computing time (loading, calculation and writing). The addition of damage analysis of the composite structure could be envisaged using current criteria. To properly assess this condition, failure criteria have to be updated in function of variations of geometry and fibre and matrix volume fractions, thus, leading to an iterative calculation per finite element.

Finally, although the proposed approach seems to return adequate results, it is indispensable to compare the proposed calculation methodology with a purely stochastic approach that a priori can reduce processing time and cost. This work also can be projected on other composite structures with complex geometry zones.

Appendix 1: Experimental protocol

The different steps carried out are described below.

1. Cut-out (estimated time: 1/2 day / omega-shape stringer)

The list of equipment used is as following:

- Prepreg: 10 plies at 45° and 10 plies at 0° of type M10.1/UD300/CHS are required. The total area of prepreg required to manufacture a 380 mm long stringer sample is equal to 481080 mm²;
- Rulers and squares: Rulers and squares can exhibit variability in alignment and measurement due to their frequent use by the operator;
- Pens: The cutting of the prepreg is done firstly by marking the edges using a permanent pen following a ruler, the marking on the roll depends on the user. The dimensions are therefore not exact, there are always variations in measurements due to the operator and even the section of the pen used;
- Cutters: hand cutting can cause variability and even small ripples (+/- 2mm);
- Cutting table: the table is covered by a silicone layer which allows easy cutting of prepreg.

The process is described as following:

- Measurement and cutting: the cutting measurements are recorded in a detailed table to allow the monitoring of the cutting operation of each ply and according to the desired dimensions;
- Scanning the plies: The plies were scanned in order to compare their dimensions before and after draping, which can create variability in the dimensions (width and length) after draping and compaction (a reference must be put on the folds to differentiate the left and right sides and this positioning must be respected during draping). The scan is carried out with the ply protectors in place so as not to damage either the scanner or the plies.

2. Lay-up and photo taking (estimated time: 2 days / omega-shape stringer of 380 mm length)

The list of equipment used is as following:

Appendix

- Release agent (demoulder);
- Omega shape stringer mould (allows for proper positioning of the creases and compaction of the stringer);
- Digital camera;
- Computer.

The process is described as following:

- Preparation of the mould (application of release agent);
- Preparation and implementation of the reference grid;
- Plies lay-up and compacting;
- Photo shooting.

Plies are draped according to their position in relation to the mould. The positions of the plies depend on the operator, hence the appearance of variability. The compaction is done after the first ply (to take the shape of the mould) and then after every 4 plies for about 20 minutes.

Photos were taken for each ply during draping in order to determine its position and the variability in position and orientation that are consecutive to the draping operation. Taking the grid as a reference, a minimum number of 4 photos must be taken after the draping of each ply (draped ply with protectors, ply without protectors, positioning of the top flange and the bottom flange). A device is installed in the clean room to take pictures of the stringer during the manufacturing process (draping). This device consists of a camera (Canon EOS 550D with a 100 mm lens) which is fixed above the work surface at a distance of 1.9 m. Taking into account the brightness and the conditions of the photo shooting, the camera is placed vertically on the draping table while making the necessary adjustments to obtain clearer photos. Verticality is ensured by observing the photo being taken. The mould must be kept in a well-defined position in relation to the camera by using wedges placed on the mould support table. The camera is associated with a computer to make the necessary adjustments remotely and to record the photos. The grid allows the draped pleats to be positioned after each drape with reference to the horizontal lines. The squares in the corners help to identify the 45° and -45° orientation of the plies with respect to the theoretical orientation. The measurement grid allows each ply to be positioned in relation to its theoretical position. The distance between one vertical axis and another is 2 cm (vertical position of the plies and orientation) which allows the positions of the plies to be clearly identified in relation to the grid when analysing the images. The horizontal

Appendix

axes indicate the extremities of the ply positions. The squares in the corners indicate the 45° and -45° orientations. The grid (29.7 cm x 42, cm) is printed on an A3 sheet (29.7 cm x 42, cm) and is glued to the mould with double-sided tape.

3. Polymerisation step (estimated time: 1 day)

The list of equipment used is as following:

- Vacuum bag;
- Thermocouples ;
- Oven.

The process is described as following:

- Cutting and preparing the environmental products and measuring their mass (to quantify the quantity of resin absorbed);
- Positioning the thermocouples on the stringer using adhesive tape that resists the high temperature during the cycle (to monitor the evolution of the temperature during the polymerisation cycle);
- Placement of environmental products;
- Preparation of the oven (polymerisation cycle, temperatures, etc.);
- Putting the part in the oven and connecting the thermocouples in the oven.

4. Analysis and observation of the part after polymerization

Three types of observation are performed respectively, Image analysis, Microscopy and Tomography.

4.1. Sample preparation

Cross-sectional samples aim to:

- See and locate porosities and defects in relation to the part;
- Positioning the ply stops in relation to a chosen reference frame.

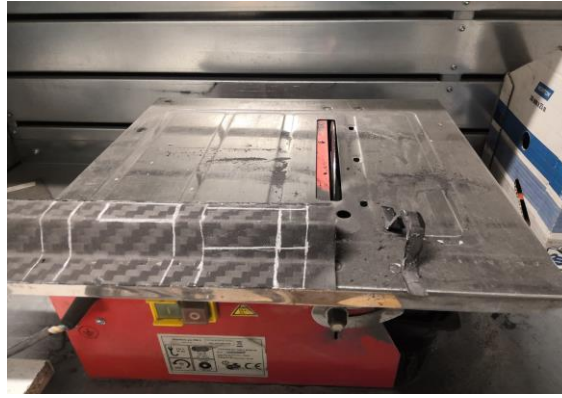


Fig. 1. Cutting circular saw

The disc circular saw (cf. Fig. 1) has a thickness of about 3 mm. This thickness can cause uncertainty in the dimensions of the samples. The cutting is done by hand, so it always depends on the operator, which causes errors in the measurement and dimensions of the samples compared to the theoretical ones. Sawing causes large burrs which can be toxic, so a mask, protective gloves and an apron must be worn. 16 specimens have been prepared.

5. Microscopy

The list of equipment used is as following:

- Chemical material for resin coating;
- Polishing machine;
- Polishing discs of the type indicated in the description of the polishing process of the samples;
- Microscope (Olympus);
- Protection equipment.

The process is described as following:

- Resin coating by mixing in a container 2 parts powder to 1 part liquid (by volume)

The coating hardens in 15 minutes (approximately) and the samples need about 200 ml of coating resin (cf. Fig. 2)



Fig. 2. Coating resin

The positioning of the samples is done by using special springs (cf. Fig. 3) that allow the samples to be hooked into the mould.



Fig. 3. Springs for fixing the samples in the resin

- Samples polishing as following respectively, using disc No. 320 for about 2 minutes, then clean with water, using disc No. 600 for about 2 minutes, then clean with water, using disc No. 800 for about 2 minutes, then clean with water, using disc N°1200 for about 3 min then clean with water + dry using a dryer (remove any type of residue that remains such as limescale), using 9-micron solution for about 2 min then cleaning with distilled water + drying and finally 3-micron solution for about 2 min then cleaning with distilled water + drying. The total operation takes about 30 min / specimen.

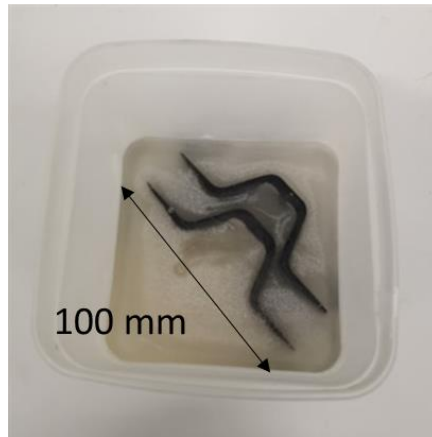


Fig. 4. Sample obtained after resin coating

The observation of the specimens (cf. Fig. 4) through a microscope (5x and possibly 10x magnification) is realised as follows; first of all, the specimen is fixed on the microscope bearing, then the horizontal and vertical position of the magnifying glass is checked in relation to the specimen (the position must be adjusted until a clear and sharp image is obtained). The brightness of the microscope is adjusted automatically. The microscope parameters must then be checked on the software: position, magnifying glass, sharpness The scanning of the sample is done by moving the bearing along the x and y directions. Although the microscope is based on precise software, there can still be errors in the image taking caused by the bearing not being perfectly fixed. The slightest destabilization of the microscope causes measurement errors, which leads to more or less important measurement variabilities. Acquisition and processing of results takes about 1 hour / section.

Appendix 2: Mould design and manufacturing

In the thesis beginning, the work ambition was to study the different composite fuselage components and the interaction between them (cf. ANR / Astrid Blast Project). So, a mould (cf. Fig. 5) was designed to manufacture a prototype for a composite fuselage part to be used in an innovative polymerisation process called “one shot”. This concept aims to polymerise all the fuselage components in one phase and using the same polymerisation process.

The mould is composed of 4 main components (cf. Fig. 6), (1) is the base that holds all the other components, (2) the frame mould, (3) the frame counter mould and (4) the stringers counter mould (cf. Fig.2). This form allows to manufacture a fuselage part containing a part from the skin with a frame and two stringers.

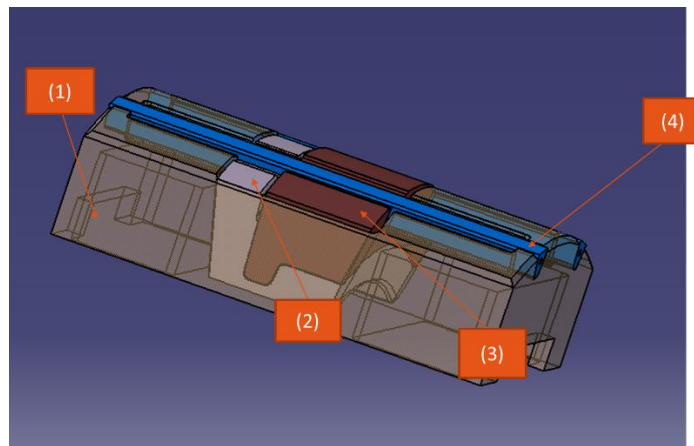


Fig. 5. One shot fuselage mould designed using Catia V5

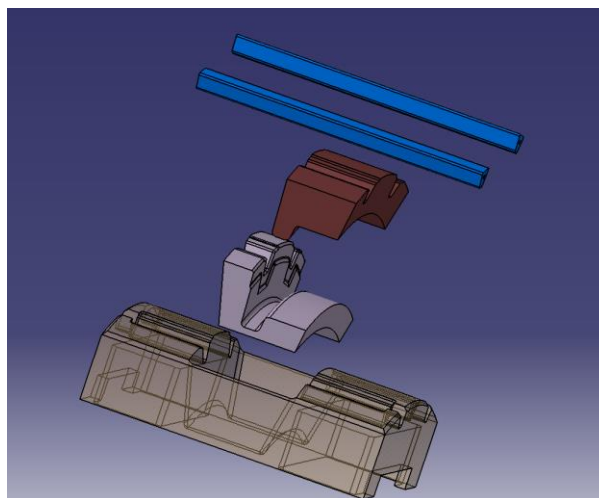


Fig. 6. Exploded view for the fuselage mould

Appendix

The different mould components were manufactured using the 5-axis machine and “AU4G (2017A)T451” material (cf. Fig. 7).

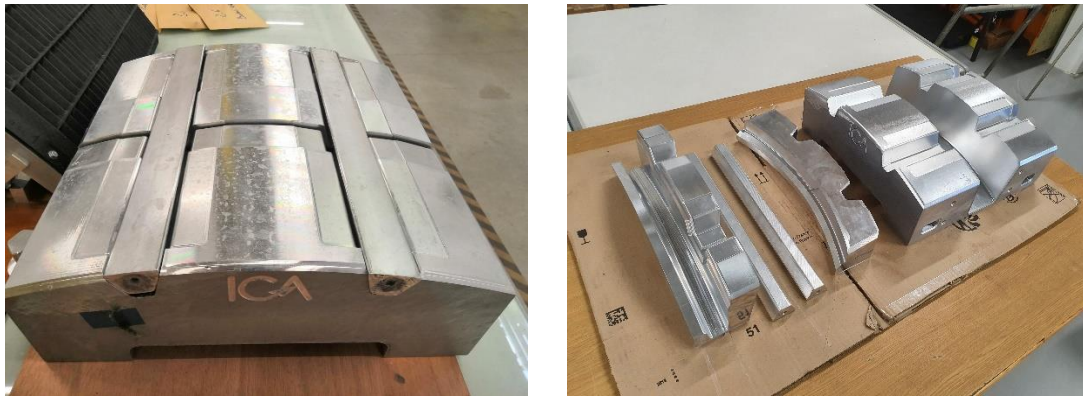


Fig. 7. Manufactured mould

The obtained fuselage part is presented in Fig. 8.



Fig. 8. Composite fuselage part obtained after curing

In this thesis, the stringer counter moulds were used as for the manufacturing of the Ω -shape 4 stringers used in variability study.

Appendix 3: Input sets after lay-up

The full data sets of gaps (mm) measured after polymerization process using microscopic observation is presented here for each ply category respectively, for 1st category (cf. Table 1), for 2nd category (cf. Table 2), for 3rd category (cf. Table 3) and for 4th category (cf. Table 4).

Category 1:

	Location along z direction (mm)			
Ply number	-135	-45	45	135
Ply1	-1,66	1,46	1,14	-0,31
Ply3	2,56	1,75	2,72	1,37
Ply9	1,95	2,43	2,41	1,1
Ply12	4,56	4,35	1,06	0,56
Ply18	1,82	3,34	3,5	2,15
Ply20	-1,42	2,16	1,75	0,41

Table 1. Gaps measured after polymerisation process for category 1 plies

Category 2:

	Location along z direction (mm)			
Ply number	-135	-45	45	135
Ply5	1,9	0,95	-1,32	0,58
Ply7	0,53	1,39	2,52	0,36
Ply14	0,49	1,71	2,32	-0,55
Ply16	-0,54	0,34	0,63	0,58

Table 2. Gaps measured after polymerisation process for category 2 plies

Appendix

Category 3:

	Location along z direction (mm)			
Ply number	-135	-45	45	135
Ply2 (Web)	-4,71	-5,01	3,7	2,69
Ply19 (Web)	-4,91	-5,72	-1,49	-4,66
Ply2 (b.flange)	-2,49	-1,50	-2,68	-3,02
Ply19 (b.flange)	-1,68	-0,2	3,73	2,4

Table 3. Gaps measured after polymerisation process for category 3 plies

Category 4:

	Location along z direction (mm)			
Ply number	-135	-45	45	135
Ply4	-0,97	-0,68	1,58	-0,69
Ply6	0,41	-0,43	-1,41	-1,06
Ply8	0,39	0,58	0,43	-0,84
Ply10	-0,61	-0,33	0,48	0,78
Ply11	0,69	0,36	0,24	0,52
Ply13	-1,85	0,26	1,04	-1,73
Ply15	-0,67	0,29	0,62	-0,4
Ply17	-0,99	0,64	1,29	-0,67

Table 4. Gaps measured after polymerisation process for category 4 plies

Appendix 4: Numerical study of a composite fuselage skin

Numerical modelling of the skin

In this section, we represent a numerical analysis related to another part of the composite fuselage. The skin of fuselage has a cylindrical form. It covers the stringers and the frames (cf. Fig. 9). Here we study $\frac{1}{4}$ of the total skin surface (without stringer or frame) used as prototype for ANR / Astrid Blast project presented in the first chapter.

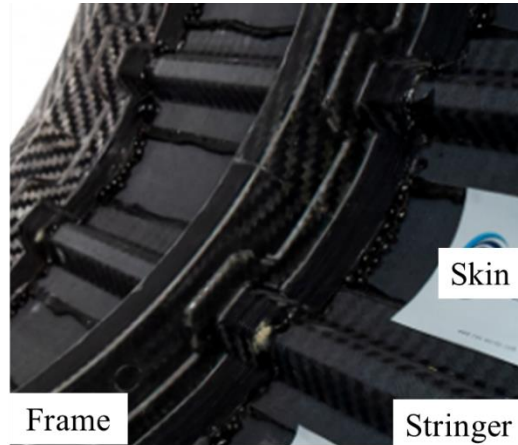


Fig. 9. Composite fuselage with different components, stringer, skin and frame (ANR/Astrid Blast Project)

The skin is composed from 8 UD plies with stacking sequence $[45, 90, -45, 0]_s$. The skin bearing geometrical cylinder has a radius of 615 mm, the skin angle is 36° with a depth of about 2.48 mm (cf. Fig. 10).

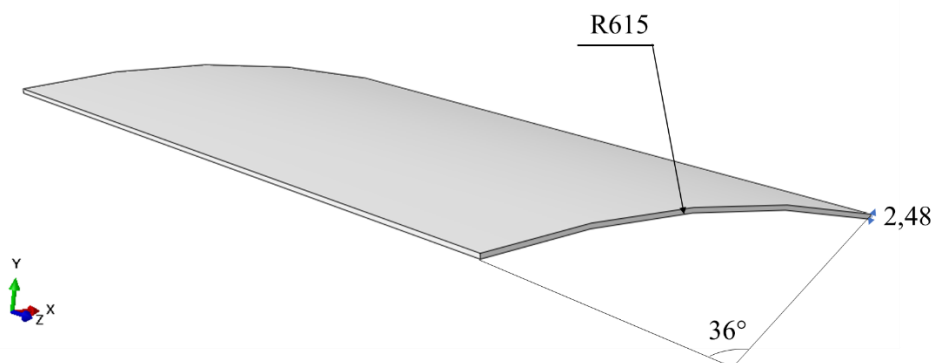


Fig. 10. Skin part dimensions

The considered variability in this section is fibre misalignment and ply thickness. Mathematical analysis is taken from work in [1].

Appendix

For fibre misalignment variability, the local variation of the misalignment was described as a sum of i^{th} pseudo-Gaussian surfaces, having a central point with coordinates X_i and Y_i , the length and the width of each perturbation is described by the parameters α_i and β_i , equivalent to the standard deviation in a normal distribution. The amplitude of the perturbation is controlled by B_i . This set of equations (cf. Eq.1) allows a continuous modelling of fibre orientation variability:

$$\theta_{pert}(x, y) = \sum_{i=1}^n B_i e^{-\left(\left(\frac{x-X_i}{\alpha_i}\right)^2 + \left(\frac{y-Y_i}{\beta_i}\right)^2\right)} \quad \text{Eq.1}$$

The Eq.1 parameters are obtained by an optimisation algorithm.

The structure thickness variability is represented by the following model (cf. Eq.2):

$$t_k(x) = \bar{t}_k + \sum_{i=1}^n A_i \sin\left(2\pi \frac{x}{\lambda_i} + \varphi_{k,i}\right) + \delta t_{plate}(x) \quad \text{Eq.2}$$

Where $t_k(x)$ is the evolution of the thickness of the k^{th} ply through x , $\bar{t}_k$ is the mean thickness of the k^{th} ply, A_i is the amplitude of the corresponding wavelength λ_i of the i^{th} peak, $\delta t_{plate}(x)$ is the contribution of the thickness variation of the plate on the ply mean thickness (means the desired plate profile divided by the number of plies). The parameters are taken from the literature work because the skin is constituted by UD plies (as in the literature).

Results after cooling

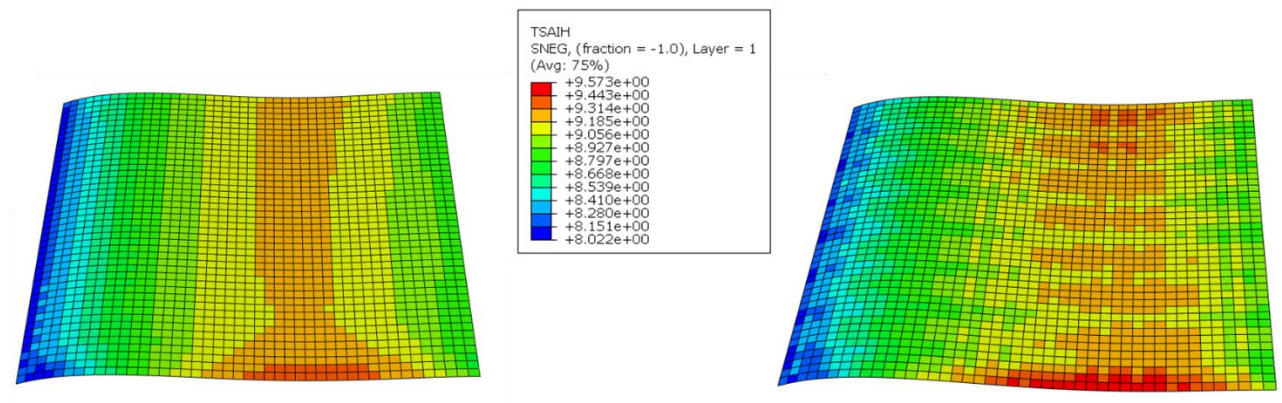


Fig. 11. Tsai-Hill criterion comparison between models without (left) and with (right) variability consideration after cooling

Tsai-Hill criterion (cf. Fig. 11) seems to increase the maximum values areas in the skin after cooling. A discontinuity also is observed especially in the centre. These results are different from results obtained in the composite structure with complex geometry structure.

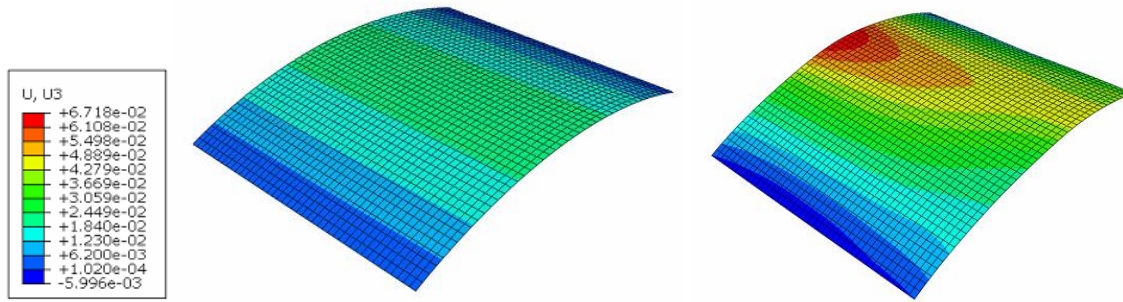


Fig. 12. Comparison between skin model without (left) and with (right) variability consideration in the case of displacement U3

The thickness and the fibre misalignment variabilities seem to influence the structure behaviour after cooling process. The displacement U3 increases with about 50% in the centre area (cf. Fig. 12). In a specific location ($y=0$), Fig. 13 shows clearly this difference in terms of displacement.

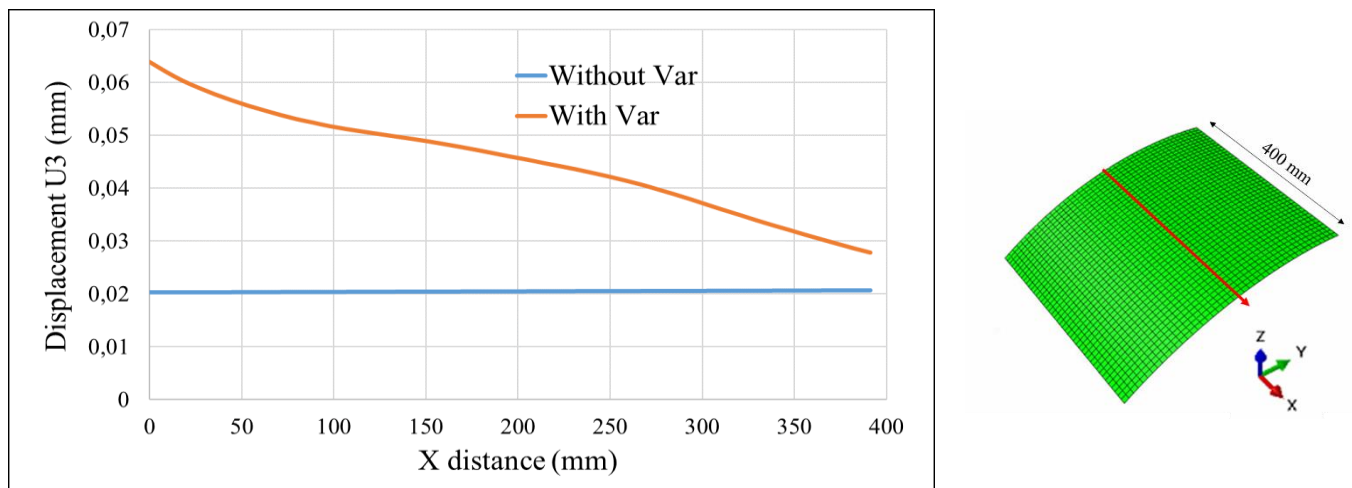


Fig. 13. Comparison between skin model without (blue curve) and with (orange curve) variability consideration in the case of displacement U3 in $y=0$

References

- [1] Davila Y, Crouzeix L, Douchin B, Collombet F, Grunevald Y-H, Rocher N. Variability in Monolithic Composite Parts: From Data Collection to FE Analysis, 2018, p. 75–97. https://doi.org/10.1007/978-981-13-2417-8_3.

Appendix 5: Numerical methodology using python code

```
# -*- coding: utf-8 -*-
Created on Fri Nov 21 10:56:55 2019
@author: khouloud
import numpy as np
import numpy as numpy
import matplotlib.pyplot as plt
import statistics
from random import random
from random import uniform
from random import gauss

# Programmation des coefficients des polynômes des écarts
#paramètres après découpe
#
X= [-135,-45,45,135]    # positions Z (largeur lisse)
#
#catégorie TISSE (plis 3, 5, 5, 7, 9, 12, 14 ,14, 16, 18 et 20)
#
#
#Ply 3
#
YPc3= [0.2,0.0,-0.8,-0.6,0,0]  #Ecarts
APc3= np.zeros ((6,6))    # matrice des Ecarts (entre position théorique et réelle)

for j in range (0,4):
    for i in range (0,6):
        APc3[j,i] = X[j]**(i)    # remplissage matrice partie polynome des écarts

for i in range (1,6):
    APc3[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche du polynome (choix pour continuité)
    APc3[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite du polynome (choix pour continuité)

PPc3=np.linalg.solve(APc3,YPc3) #vecteur solutions (coefficients du polynôme des écarts)

SPc3 =np.zeros((270)) # initialisation matrice des résultats pour l'identification des coef. du polynôme

for i in range (-135,135):
    SPc3 [i+135] =PPc3[0]+PPc3[1]*i+PPc3[2]*(i**2) + PPc3[3]*(i**3)+ PPc3[4]*(i**4)+PPc3[5]*(i**5) #remplissage matrice des
résultats pour l'identification des coef. du polynôme

IPc3 = np.dot(APc3,PPc3)-YPc3    #vérification de la cohérence physique des coefficients du polynôme sur la valeur des
écarts obtenus

plot5 = plt. figure(5)
plt.title("category Woven")
plt.plot (SPc3,label="Ply3",linestyle='--')
plt.legend()
```

Appendix

```
plt.xlabel('Distance along structure length (mm)')
plt.ylabel('Gap (mm)')

#
#Ply 5 (Gauche)
#
YP5Gc= [0.2,0.6,0.2,-0.4,0,0]  #Ecart

AP5Gc= np.zeros ((6,6))  # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP5Gc[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    AP5Gc[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP5Gc[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP5Gc=np.linalg.solve(AP5Gc,YP5Gc) #vecteur solutions (coefficients de polynome)

SP5Gc =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP5Gc [i+135] =PP5Gc[0]+PP5Gc[1]*i+PP5Gc[2]*(i**2) + PP5Gc[3]*(i**3)+ PP5Gc[4]*(i**4)+PP5Gc[5]*(i**5)  #remplissage
matrice des résultats

IP5Gc = np.dot(AP5Gc,PP5Gc)-YP5Gc  #vérification polynome

plt.plot (SP5Gc,label="Ply5")
plt.legend()

#
#Ply 7 (Gauche)
#
YP7Gc= [-1.6,-1.2,-1,-1,0,0]  #Ecart

AP7Gc= np.zeros ((6,6))  # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP7Gc[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    AP7Gc[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP7Gc[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP7Gc=np.linalg.solve(AP7Gc,YP7Gc) #vecteur solutions (coefficients de polynome)
```

Appendix

```
SP7Gc = np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP7Gc [i+135] =PP7Gc[0]+PP7Gc[1]*i+PP7Gc[2]*(i**2) + PP7Gc[3]*(i**3)+ PP7Gc[4]*(i**4)+PP7Gc[5]*(i**5) #remplissage
matrice des résultats

IP7Gc = np.dot(AP7Gc,PP7Gc)-YP7Gc #vérification polynome

plt.plot (SP7Gc,label="Ply7")
plt.legend()

# _____
#Ply 9
# _____

YPc9= [0.4,0,1,1,0,0] #Ecart

APc9= np.zeros ((6,6)) # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        APc9[j,i] = X[j]**(i) # remplissage matrice partie polynome

for i in range (1,6):
    APc9[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
    APc9[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite

PPc9=np.linalg.solve(APc9,YPc9) #vecteur solutions (coefficients de polynome)

SPc9 =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SPc9 [i+135] =PPc9[0]+PPc9[1]*i+PPc9[2]*(i**2) + PPc9[3]*(i**3)+ PPc9[4]*(i**4)+PPc9[5]*(i**5) #remplissage matrice des
résultats

IPc9 = np.dot(APc9,PPc9)-YPc9 #vérification polynome

plt.plot (SPc9,label="Ply9")
plt.legend()

# _____
#Ply 12
# _____

YPc12= [-0.6,-0.7,-0.8,-0.6,0,0] #Ecart

APc12= np.zeros ((6,6)) # matrice des Ecart
```

Appendix

```
for j in range (0,4):
    for i in range (0,6):
        APc12[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    APc12[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    APc12[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PPc12=np.linalg.solve(APc12,YPc12) #vecteur solutions (coefficients de polynome)

SPc12=np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SPc12 [i+135] =PPc12[0]+PPc12[1]*i+PPc12[2]*(i**2) + PPc12[3]*(i**3)+ PPc12[4]*(i**4)+PPc12[5]*(i**5) #remplissage matrice
des résultats

IPc12 = np.dot(APc12,PPc12)-YPc12    #vérification polynome

plt.plot (SPc12,label="Ply12")
plt.legend()

#
#Ply 14 (Gauche)
#

YP14Gc= [-1.0,-0.8,-0.6,0.0,0,0]    #Ecarts

AP14Gc= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP14Gc[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP14Gc[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP14Gc[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP14Gc=np.linalg.solve(AP14Gc,YP14Gc) #vecteur solutions (coefficients de polynome)

SP14Gc =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP14Gc [i+135] =PP14Gc[0]+PP14Gc[1]*i+PP14Gc[2]*(i**2) + PP14Gc[3]*(i**3)+ PP14Gc[4]*(i**4)+PP14Gc[5]*(i**5)
#remplissage matrice des résultats

IP14Gc = np.dot(AP14Gc,PP14Gc)-YP14Gc    #vérification polynome
```

Appendix

```
plt.plot (SP14Gc,label="Ply14")
plt.legend()

# _____
#Ply 16 (Gauche)
# _____

YP16Gc= [1,1.4,2,3,0,0]  #Ecart

AP16Gc= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP16Gc[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP16Gc[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP16Gc[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP16Gc=np.linalg.solve(AP16Gc,YP16Gc) #vecteur solutions (coefficients de polynome)

SP16Gc =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP16Gc [i+135] =PP16Gc[0]+PP16Gc[1]*i+PP16Gc[2]*(i**2) + PP16Gc[3]*(i**3)+ PP16Gc[4]*(i**4)+PP16Gc[5]*(i**5)
#remplissage matrice des résultats

IP16Gc = np.dot(AP16Gc,PP16Gc)-YP16Gc  #vérification polynome

plt.plot (SP16Gc,label="Ply16")
plt.legend()

# _____
#Ply 18
# _____

YPc18= [-2,-2,-1.5,-1.4,0,0]  #Ecart

APc18= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        APc18[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
```

Appendix

```
APc18[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
APc18[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PPc18=np.linalg.solve(APc18,YPc18) #vecteur solutions (coefficients de polynome)

SPc18 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SPc18 [i+135] =PPc18[0]+PPc18[1]*i+PPc18[2]*(i**2) + PPc18[3]*(i**3)+ PPc18[4]*(i**4)+PPc18[5]*(i**5)  #remplissage matrice
des résultats

IPc18 = np.dot(APc18,PPc18)-YPc18  #vérification polynome

plt.plot (SPc18,label="Ply18")
plt.legend()
# _____
#Ply 20
# _____

YPc20= [1.2,0.7,0.4,0.8,0,0]  #Ecart

APc20= np.zeros ((6,6))      # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        APc20[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    APc20[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    APc20[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PPc20=np.linalg.solve(APc20,YPc20) #vecteur solutions (coefficients de polynome)

SPc20 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SPc20 [i+135] =PPc20[0]+PPc20[1]*i+PPc20[2]*(i**2) + PPc20[3]*(i**3)+ PPc20[4]*(i**4)+PPc20[5]*(i**5)  #remplissage matrice
des résultats

IPc20 = np.dot(APc20,PPc20)-YPc20  #vérification polynome

plt.plot (SPc20,label="Ply20")
plt.legend()
plt.grid()

# _____
#détermination des moyennes et écart-types pour catégorie TISSE écart découpe
```

Appendix

```
# _____  
ecartype_coeff_catTISSE = np.zeros((6))  
moyenne_coeff_catTISSE = np.zeros((6))  
  
coeff_catTISSE0=[PPc3[0],PP5Gc[0],PP7Gc[0],PPc9[0],PPc12[0],PP14Gc[0],PP16Gc[0],PPc18[0],PPc20[0]]  
coeff_catTISSE1=[PPc3[1],PP5Gc[1],PP7Gc[1],PPc9[1],PPc12[1],PP14Gc[1],PP16Gc[1],PPc18[1],PPc20[1]]  
coeff_catTISSE2=[PPc3[2],PP5Gc[2],PP7Gc[2],PPc9[2],PPc12[2],PP14Gc[2],PP16Gc[2],PPc18[2],PPc20[2]]  
coeff_catTISSE3=[PPc3[3],PP5Gc[3],PP7Gc[3],PPc9[3],PPc12[3],PP14Gc[3],PP16Gc[3],PPc18[3],PPc20[3]]  
coeff_catTISSE4=[PPc3[4],PP5Gc[4],PP7Gc[4],PPc9[4],PPc12[4],PP14Gc[4],PP16Gc[4],PPc18[4],PPc20[4]]  
coeff_catTISSE5=[PPc3[5],PP5Gc[5],PP7Gc[5],PPc9[5],PPc12[5],PP14Gc[5],PP16Gc[5],PPc18[5],PPc20[5]]  
  
ecartype_coeff_catTISSE[0]= statistics.stdev(coeff_catTISSE0)  
ecartype_coeff_catTISSE[1]= statistics.stdev(coeff_catTISSE1)  
ecartype_coeff_catTISSE[2]= statistics.stdev(coeff_catTISSE2)  
ecartype_coeff_catTISSE[3]= statistics.stdev(coeff_catTISSE3)  
ecartype_coeff_catTISSE[4]= statistics.stdev(coeff_catTISSE4)  
ecartype_coeff_catTISSE[5]= statistics.stdev(coeff_catTISSE5)  
  
moyenne_coeff_catTISSE[0]= statistics.mean(coeff_catTISSE0)  
moyenne_coeff_catTISSE[1]= statistics.mean(coeff_catTISSE1)  
moyenne_coeff_catTISSE[2]= statistics.mean(coeff_catTISSE2)  
moyenne_coeff_catTISSE[3]= statistics.mean(coeff_catTISSE3)  
moyenne_coeff_catTISSE[4]= statistics.mean(coeff_catTISSE4)  
moyenne_coeff_catTISSE[5]= statistics.mean(coeff_catTISSE5)  
  
# _____  
#catégorie UD (2, 4, 6, 8, 10, 11, 13, 15, 17 et 19)  
# _____  
# _____  
#Ply 2 (Web) (âme de la lisse)  
# _____  
  
YP2Wc= [-0.4,0.2,0,-0.4,0,0]  #Ecarts  
  
AP2Wc= np.zeros ((6,6))  # matrice des Ecarts  
  
for j in range (0,4):  
    for i in range (0,6):  
        AP2Wc[j,i] = X[j]**(i)  # remplissage matrice partie polynome  
  
for i in range (1,6):  
    AP2Wc[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche  
    AP2Wc[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite  
  
PP2Wc=np.linalg.solve(AP2Wc,YP2Wc) #vecteur solutions (coefficients de polynome)  
  
SP2Wc =np.zeros((270))  # initialisation matrice des résultats
```

Appendix

```
for i in range (-135,135):
    SP2Wc [i+135] =PP2Wc[0]+PP2Wc[1]*i+PP2Wc[2]*(i**2) + PP2Wc[3]*(i**3)+ PP2Wc[4]*(i**4)+PP2Wc[5]*(i**5) #remplissage
matrice des résultats

IP2Wc = np.dot(AP2Wc,PP2Wc)-YP2Wc #vérification polynome

plot6 = plt. figure(6)
plt.title("category UD")
plt.plot (SP2Wc,label="Ply2 Web",linestyle='--')
plt.legend()
plt.xlabel('Distance along structure length (mm)')
plt.ylabel('Gap (mm)')

#
#Ply 2 (t.flange) (aile gauche de la lisse)
#
YP2Tc= [-0.4,0.2,0,-0.4,0,0] #Ecart

AP2Tc= np.zeros ((6,6)) # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP2Tc[j,i] = X[j]**(i) # remplissage matrice partie polynome

for i in range (1,6):
    AP2Tc[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
    AP2Tc[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite

PP2Tc=np.linalg.solve(AP2Tc,YP2Tc) #vecteur solutions (coefficients de polynome)

SP2Tc =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP2Tc [i+135] =PP2Tc[0]+PP2Tc[1]*i+PP2Tc[2]*(i**2) + PP2Tc[3]*(i**3)+ PP2Tc[4]*(i**4)+PP2Tc[5]*(i**5) #remplissage
matrice des résultats

IP2Tc = np.dot(AP2Tc,PP2Tc)-YP2Tc #vérification polynome

plt.plot (SP2Tc,label="Ply2 top flange")
plt.legend()

#
#Ply 2 (b.flange)
#
YP2Bc= [-0.4,0.2,0,-0.4,0,0] #Ecart
```

Appendix

```
AP2Bc= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP2Bc[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP2Bc[4,i] = i * X[0]**(i-1)    #remplissage matrice partie tangente gauche
    AP2Bc[5,i] = i * X[3]**(i-1)    #remplissage matrice partie tangente droite

PP2Bc=np.linalg.solve(AP2Bc,YP2Bc) #vecteur solutions (coefficients de polynome)

SP2Bc =np.zeros((270))    # initialisation matrice des résultats

for i in range (-135,135):
    SP2Bc [i+135] =PP2Bc[0]+PP2Bc[1]*i+PP2Bc[2]*(i**2) + PP2Bc[3]*(i**3)+ PP2Bc[4]*(i**4)+PP2Bc[5]*(i**5)    #remplissage
matrice des résultats

IP2Bc = np.dot(AP2Bc,PP2Bc)-YP2Bc    #vérification polynome

plt.plot (SP2Bc,label="Ply2 Bottom flange")
plt.legend()

# _____
#Ply 4 (continu sur la largeur de la lisse)
# _____

YPc4= [0,-0.3,-0.3,-0.4,0,0]    #Ecarts

APc4= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        APc4[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    APc4[4,i] = i * X[0]**(i-1)    #remplissage matrice partie tangente gauche
    APc4[5,i] = i * X[3]**(i-1)    #remplissage matrice partie tangente droite

PPc4=np.linalg.solve(APc4,YPc4) #vecteur solutions (coefficients de polynome)

SPc4 =np.zeros((270))    # initialisation matrice des résultats

for i in range (-135,135):
    SPc4 [i+135] =PPc4[0]+PPc4[1]*i+PPc4[2]*(i**2) + PPc4[3]*(i**3)+ PPc4[4]*(i**4)+PPc4[5]*(i**5)    #remplissage matrice des
résultats
```

Appendix

```
IPc4 = np.dot(APc4,PPc4)-YPc4  #vérification polynome

plt.plot (SPc4,label="Ply4")
plt.legend()
#
#Ply 6 (continu sur la largeur de la lisse)
#

YPc6= [-1.4,-1.6,-1.4,-2,0,0]  #Ecart

APc6= np.zeros ((6,6))      # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        APc6[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    APc6[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    APc6[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PPc6=np.linalg.solve(APc6,YPc6) #vecteur solutions (coefficients de polynome)

SPc6 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SPc6 [i+135] =PPc6[0]+PPc6[1]*i+PPc6[2]*(i**2) + PPc6[3]*(i**3)+ PPc6[4]*(i**4)+PPc6[5]*(i**5)  #remplissage matrice des
    résultats

IPc6 = np.dot(APc6,PPc6)-YPc6  #vérification polynome

plt.plot (SPc6,label="Ply6")
plt.legend()
#
#Ply 8 (continu sur la largeur de la lisse)
#

YPc8= [0,-0.3,-0.3,-0.4,0,0]  #Ecart

APc8= np.zeros ((6,6))      # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        APc8[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    APc8[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
```

Appendix

```
APc8[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PPc8=np.linalg.solve(APc8,YPc8)  #vecteur solutions (coefficients de polynome)

SPc8 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SPc8 [i+135] =PPc8[0]+PPc8[1]*i+PPc8[2]*(i**2) + PPc8[3]*(i**3)+ PPc8[4]*(i**4)+PPc8[5]*(i**5)  #remplissage matrice des
résultats

IPc8 = np.dot(APc8,PPc8)-YPc8  #vérification polynome

plt.plot (SPc8,label="Ply8")
plt.legend()

# _____
#Ply 10 (continu sur la largeur de la lisse)
# _____

YPc10= [-0.2,-0.4,-1.2,-2,0,0]  #Ecart

APc10= np.zeros ((6,6))  # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        APc10[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    APc10[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    APc10[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PPc10=np.linalg.solve(APc10,YPc10)  #vecteur solutions (coefficients de polynome)

SPc10 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SPc10 [i+135] =PPc10[0]+PPc10[1]*i+PPc10[2]*(i**2) + PPc10[3]*(i**3)+ PPc10[4]*(i**4)+PPc10[5]*(i**5)  #remplissage matrice
des résultats

IPc10 = np.dot(APc10,PPc10)-YPc10  #vérification polynome

plt.plot (SPc10,label="Ply10")
plt.legend()

# _____
#Ply11 (continu sur la largeur de la lisse)
# _____
```

Appendix

```
YPc11= [-0.6,-0.8,-0.4,-0.4,0,0] #Ecart
```

```
APc11= np.zeros ((6,6)) # matrice des Ecart
```

```
for j in range (0,4):
```

```
    for i in range (0,6):
```

```
        APc11[j,i] = X[j]**(i) # remplissage matrice partie polynome
```

```
for i in range (1,6):
```

```
    APc11[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
```

```
    APc11[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite
```

```
PPc11=np.linalg.solve(APc11,YPc11) #vecteur solutions (coefficients de polynome)
```

```
SPc11=np.zeros((270)) # initialisation matrice des résultats
```

```
for i in range (-135,135):
```

```
    SPc11 [i+135] =PPc11[0]+PPc11[1]*i+PPc11[2]*(i**2) + PPc11[3]*(i**3)+ PPc11[4]*(i**4)+PPc11[5]*(i**5) #remplissage matrice des résultats
```

```
IPc11 = np.dot(APc11,PPc11)-YPc11 #vérification polynome
```

```
plt.plot (SPc11,label="Ply11")
```

```
plt.legend()
```

```
#
```

```
#Ply 13 (continu sur la largeur de la lisse)
```

```
#
```

```
YPc13= [-2,-1.6,-1.4,-0.8,0,0] #Ecart
```

```
APc13= np.zeros ((6,6)) # matrice des Ecart
```

```
for j in range (0,4):
```

```
    for i in range (0,6):
```

```
        APc13[j,i] = X[j]**(i) # remplissage matrice partie polynome
```

```
for i in range (1,6):
```

```
    APc13[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
```

```
    APc13[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite
```

```
PPc13=np.linalg.solve(APc13,YPc13) #vecteur solutions (coefficients de polynome)
```

```
SPc13=np.zeros((270)) # initialisation matrice des résultats
```

```
for i in range (-135,135):
```

Appendix

```
SPc13 [i+135] =PPc13[0]+PPc13[1]*i+PPc13[2]*(i**2) + PPc13[3]*(i**3)+ PPc13[4]*(i**4)+PPc13[5]*(i**5)  #remplissage matrice  
des résultats
```

```
IPc13 = np.dot(APc13,PPc13)-YPc13  #vérification polynome
```

```
plt.plot (SPc13,label="Ply13")
```

```
plt.legend()
```

```
#
```

```
#Ply 15 (continu sur la largeur de la lisse)
```

```
#
```

```
YPc15= [-0.4,-0.8,-0.8,-0.6,0,0]  #Ecart
```

```
APc15= np.zeros ((6,6))  # matrice des Ecart
```

```
for j in range (0,4):
```

```
    for i in range (0,6):
```

```
        APc15[j,i] = X[j]**(i)  # remplissage matrice partie polynome
```

```
for i in range (1,6):
```

```
    APc15[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
```

```
    APc15[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite
```

```
PPc15=np.linalg.solve(APc15,YPc15)  #vecteur solutions (coefficients de polynome)
```

```
SPc15 =np.zeros((270))  # initialisation matrice des résultats
```

```
for i in range (-135,135):
```

```
    SPc15 [i+135] =PPc15[0]+PPc15[1]*i+PPc15[2]*(i**2) + PPc15[3]*(i**3)+ PPc15[4]*(i**4)+PPc15[5]*(i**5)  #remplissage matrice  
des résultats
```

```
IPc15 = np.dot(APc15,PPc15)-YPc15  #vérification polynome
```

```
plt.plot (SPc15,label="Ply15")
```

```
plt.legend()
```

```
#
```

```
#Ply 17 (continu sur la largeur de la lisse)
```

```
#
```

```
YPc17= [-1.4,-0.6,0,0,0,0]  #Ecart
```

```
APc17= np.zeros ((6,6))  # matrice des Ecart
```

```
for j in range (0,4):
```

```
    for i in range (0,6):
```

Appendix

```
APc17[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    APc17[4,i] = i * X[0]**(i-1)    #remplissage matrice partie tangente gauche
    APc17[5,i] = i * X[3]**(i-1)    #remplissage matrice partie tangente droite

PPc17=np.linalg.solve(APc17,YPc17) #vecteur solutions (coefficients de polynome)

SPc17 =np.zeros((270))    # initialisation matrice des résultats

for i in range (-135,135):
    SPc17 [i+135] =PPc17[0]+PPc17[1]*i+PPc17[2]*(i**2) + PPc17[3]*(i**3)+ PPc17[4]*(i**4)+PPc17[5]*(i**5)    #remplissage matrice
des résultats

IPc17 = np.dot(APc17,PPc17)-YPc17    #vérification polynome

plt.plot (SPc17,label="Ply17")
plt.legend()

# _____
#Ply 19 (t.flange) (aile gauche)
# _____

YP19Tc= [-1.4,-1,-1,-1,0,0]    #Ecarts

AP19Tc= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP19Tc[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP19Tc[4,i] = i * X[0]**(i-1)    #remplissage matrice partie tangente gauche
    AP19Tc[5,i] = i * X[3]**(i-1)    #remplissage matrice partie tangente droite

PP19Tc=np.linalg.solve(AP19Tc,YP19Tc) #vecteur solutions (coefficients de polynome)

SP19Tc =np.zeros((270))    # initialisation matrice des résultats

for i in range (-135,135):
    SP19Tc [i+135] =PP19Tc[0]+PP19Tc[1]*i+PP19Tc[2]*(i**2) + PP19Tc[3]*(i**3)+ PP19Tc[4]*(i**4)+PP19Tc[5]*(i**5)
#remplissage matrice des résultats

IP19Tc = np.dot(AP19Tc,PP19Tc)-YP19Tc    #vérification polynome

plt.plot (SP19Tc,label="Ply19 Top flange")
plt.legend()
```

Appendix

```
#
#Ply 19 (Web) (âme de la lisse)
#

YP19Wc= [-1.4,-1,-1,-1,0,0] #Ecart
AP19Wc= np.zeros ((6,6)) # matrice des Ecart
for j in range (0,4):
    for i in range (0,6):
        AP19Wc[j,i] = X[j]**(i) # remplissage matrice partie polynome

for i in range (1,6):
    AP19Wc[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
    AP19Wc[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite

PP19Wc=np.linalg.solve(AP19Wc,YP19Wc) #vecteur solutions (coefficients de polynome)

SP19Wc =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP19Wc [i+135] =PP19Wc[0]+PP19Wc[1]*i+PP19Wc[2]*(i**2) + PP19Wc[3]*(i**3)+ PP19Wc[4]*(i**4)+PP19Wc[5]*(i**5)
#remplissage matrice des résultats

IP19Wc = np.dot(AP19Wc,PP19Wc)-YP19Wc #vérification polynome

plt.plot (SP19Wc,label="Ply19 Web")
plt.legend()

#
#Ply 19 (b.flange) (aile gauche de la lisse)
#

YP19Bc= [-1.4,-1,-1,-1,0,0] #Ecart

AP19Bc= np.zeros ((6,6)) # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP19Bc[j,i] = X[j]**(i) # remplissage matrice partie polynome

for i in range (1,6):
    AP19Bc[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
    AP19Bc[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite

PP19Bc=np.linalg.solve(AP19Bc,YP19Bc) #vecteur solutions (coefficients de polynome)

SP19Bc =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
```

Appendix

```
SP19Bc [i+135] = PP19Bc[0]+PP19Bc[1]*i+PP19Bc[2]*(i**2) + PP19Bc[3]*(i**3)+ PP19Bc[4]*(i**4)+PP19Bc[5]*(i**5)
#remplissage matrice des résultats

IP19Bc = np.dot(AP19Bc,PP19Bc)-YP19Bc    #vérification polynome

plt.plot (SP19Bc,label="Ply19 Bottom flange")
plt.legend()
plt.grid()

## _____
#détermination des moyennes et écart-types pour catégorie UD  écart découpe
# _____

ecartype_coeff_catUD =np.zeros((6))
moyenne_coeff_catUD =np.zeros((6))

coeff_catUD0=[PP2Tc[0],PP2Wc[0],PP2Bc[0],PPc4[0],PPc6[0],PPc8[0],PPc10[0],PPc11[0],PPc13[0],PPc15[0],PPc17[0],PP19Tc[0],P
P19Wc[0],PP19Bc[0]]
coeff_catUD1=[PP2Tc[1],PP2Wc[1],PP2Bc[1],PPc4[1],PPc6[1],PPc8[1],PPc10[1],PPc11[1],PPc13[1],PPc15[1],PPc17[1],PP19Tc[1],P
P19Wc[1],PP19Bc[1]]
coeff_catUD2=[PP2Tc[2],PP2Wc[2],PP2Bc[2],PPc4[2],PPc6[2],PPc8[2],PPc10[2],PPc11[2],PPc13[2],PPc15[2],PPc17[2],PP19Tc[2],P
P19Wc[2],PP19Bc[2]]
coeff_catUD3=[PP2Tc[3],PP2Wc[3],PP2Bc[3],PPc4[3],PPc6[3],PPc8[3],PPc10[3],PPc11[3],PPc13[3],PPc15[3],PPc17[3],PP19Tc[3],P
P19Wc[3],PP19Bc[3]]
coeff_catUD4=[PP2Tc[4],PP2Wc[4],PP2Bc[4],PPc4[4],PPc6[4],PPc8[4],PPc10[4],PPc11[4],PPc13[4],PPc15[4],PPc17[4],PP19Tc[4],P
P19Wc[4],PP19Bc[4]]
coeff_catUD5=[PP2Tc[5],PP2Wc[5],PP2Bc[5],PPc4[5],PPc6[5],PPc8[5],PPc10[5],PPc11[5],PPc13[5],PPc15[5],PPc17[5],PP19Tc[5],P
P19Wc[5],PP19Bc[5]]

ecartype_coeff_catUD[0]= statistics.stdev(coeff_catUD0)
ecartype_coeff_catUD[1]= statistics.stdev(coeff_catUD1)
ecartype_coeff_catUD[2]= statistics.stdev(coeff_catUD2)
ecartype_coeff_catUD[3]= statistics.stdev(coeff_catUD3)
ecartype_coeff_catUD[4]= statistics.stdev(coeff_catUD4)
ecartype_coeff_catUD[5]= statistics.stdev(coeff_catUD5)

moyenne_coeff_catUD[0]= statistics.mean(coeff_catUD0)
moyenne_coeff_catUD[1]= statistics.mean(coeff_catUD1)
moyenne_coeff_catUD[2]= statistics.mean(coeff_catUD2)
moyenne_coeff_catUD[3]= statistics.mean(coeff_catUD3)
moyenne_coeff_catUD[4]= statistics.mean(coeff_catUD4)
moyenne_coeff_catUD[5]= statistics.mean(coeff_catUD5)

#paramètres après polymérisation (deuxième source d'écarts) (cat 1, 2, 3 et 4 : quatre catégories de pli liées à la
polymérisation)
# _____
#category 1 (plis 3, 9, 12 , 18 et 20)
# _____
```

Appendix

```
# _____
#Ply 3
# _____
YP3= [2.37,1.68,2.71,1.35,0,0]  #Ecart
AP3= np.zeros ((6,6))  # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP3[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    AP3[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP3[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP3=np.linalg.solve(AP3,YP3)  #vecteur solutions (coefficients de polynome)

SP3 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP3 [i+135] =PP3[0]+PP3[1]*i+PP3[2]*(i**2) + PP3[3]*(i**3)+ PP3[4]*(i**4)+PP3[5]*(i**5)  #remplissage matrice des résultats

IP3 = np.dot(AP3,PP3)-YP3  #vérification polynome

plot1 = plt. figure(1)
plt.title("category 1")
plt.plot (SP3,label="Ply3",linestyle='--')
plt.legend()
plt.xlabel('Distance along structure length (mm)')
plt.ylabel('Gap (mm)')

# _____
#Ply 9
# _____

YP9= [1.8,2.42,2.41,1.07,0,0]  #Ecart
AP9= np.zeros ((6,6))  # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP9[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    AP9[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP9[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP9=np.linalg.solve(AP9,YP9)  #vecteur solutions (coefficients de polynome)
```

Appendix

```
SP9=np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP9 [i+135] =PP9[0]+PP9[1]*i+PP9[2]*(i**2) + PP9[3]*(i**3)+ PP9[4]*(i**4)+PP9[5]*(i**5) #remplissage matrice des résultats

IP9 = np.dot(AP9,PP9)-YP9 #vérification polynome

plt.plot (SP9,label="Ply9")
plt.legend()
# _____
#Ply 12
# _____

X= [-135,-45,45,135] # positions Z

YP12= [4.46,4.32,1.01,-0.32,0,0] #Ecart

AP12= np.zeros ((6,6)) # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP12[j,i] = X[j]**(i) # remplissage matrice partie polynome

for i in range (1,6):
    AP12[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
    AP12[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite

PP12=np.linalg.solve(AP12,YP12) #vecteur solutions (coefficients de polynome)

SP12 =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP12 [i+135] =PP12[0]+PP12[1]*i+PP12[2]*(i**2) + PP12[3]*(i**3)+ PP12[4]*(i**4)+PP12[5]*(i**5) #remplissage matrice des
résultats

IP12 = np.dot(AP12,PP12)-YP12 #vérification polynome

plt.plot (SP12,label="Ply12",linestyle='-'.)
plt.legend()

# _____
#Ply 18
# _____

X= [-135,-45,45,135] # positions Z
```

Appendix

```
YP18= [1.45,3.25,3.5,2.15,0,0]  #Ecarts
AP18= np.zeros ((6,6))      # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP18[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP18[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP18[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP18=np.linalg.solve(AP18,YP18) #vecteur solutions (coefficients de polynome)

SP18 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP18 [i+135] =PP18[0]+PP18[1]*i+PP18[2]*(i**2) + PP18[3]*(i**3)+ PP18[4]*(i**4)+PP18[5]*(i**5)  #remplissage matrice des
résultats

IP18 = np.dot(AP18,PP18)-YP18  #vérification polynome

plt.plot (SP18,label="Ply18",linestyle=':')
plt.legend()
# _____
#Ply 20
# _____

X= [-135,-45,45,135]    # positions Z

YP20= [0.93,2.06,1.75,0.39,0,0]  #Ecarts

AP20= np.zeros ((6,6))      # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP20[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP20[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP20[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP20=np.linalg.solve(AP20,YP20) #vecteur solutions (coefficients de polynome)

SP20 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
```

Appendix

```
SP20 [i+135] =PP20[0]+PP20[1]*i+PP20[2]*(i**2) + PP20[3]*(i**3)+ PP20[4]*(i**4)+PP20[5]*(i**5)  #remplissage matrice des
résultats
```

```
IP20 = np.dot(AP20,PP20)-YP20  #vérification polynome
```

```
plt.plot (SP20,label="Ply20")
```

```
plt.legend()
```

```
plt.grid()
```

```
# _____
```

```
#détermination des moyennes et écart-types pour catégorie 1
```

```
# _____
```

```
ecartype_coeff_cat1 =np.zeros((6))
```

```
moyenne_coeff_cat1 =np.zeros((6))
```

```
coeff_cat10=[PP3[0],PP9[0],PP12[0],PP18[0],PP20[0]]
```

```
coeff_cat11=[PP3[1],PP9[1],PP12[1],PP18[1],PP20[1]]
```

```
coeff_cat12=[PP3[2],PP9[2],PP12[2],PP18[2],PP20[2]]
```

```
coeff_cat13=[PP3[3],PP9[3],PP12[3],PP18[3],PP20[3]]
```

```
coeff_cat14=[PP3[4],PP9[4],PP12[4],PP18[4],PP20[4]]
```

```
coeff_cat15=[PP3[5],PP9[5],PP12[5],PP18[5],PP20[5]]
```

```
ecartype_coeff_cat1[0]= statistics.stdev(coeff_cat10)
```

```
ecartype_coeff_cat1[1]= statistics.stdev(coeff_cat11)
```

```
ecartype_coeff_cat1[2]= statistics.stdev(coeff_cat12)
```

```
ecartype_coeff_cat1[3]= statistics.stdev(coeff_cat13)
```

```
ecartype_coeff_cat1[4]= statistics.stdev(coeff_cat14)
```

```
ecartype_coeff_cat1[5]= statistics.stdev(coeff_cat15)
```

```
moyenne_coeff_cat1[0]= statistics.mean(coeff_cat10)
```

```
moyenne_coeff_cat1[1]= statistics.mean(coeff_cat11)
```

```
moyenne_coeff_cat1[2]= statistics.mean(coeff_cat12)
```

```
moyenne_coeff_cat1[3]= statistics.mean(coeff_cat13)
```

```
moyenne_coeff_cat1[4]= statistics.mean(coeff_cat14)
```

```
moyenne_coeff_cat1[5]= statistics.mean(coeff_cat15)
```

```
# _____
```

```
# category 2 (plis 2 et 19)
```

```
# _____
```

```
# _____
```

```
#Ply 2 (Web)
```

```
# _____
```

```
X= [-135,-45,45,135]  # positions Z
```

```
YP2W= [0.68,0.26,3.39,2.08,0,0]  #Ecart
```

Appendix

```
AP2W= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP2W[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP2W[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP2W[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP2W=np.linalg.solve(AP2W,YP2W) #vecteur solutions (coefficients de polynome)

SP2W =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP2W [i+135] =PP2W[0]+PP2W[1]*i+PP2W[2]*(i**2) + PP2W[3]*(i**3)+ PP2W[4]*(i**4)+PP2W[5]*(i**5)  #remplissage matrice
des résultats

IP2W = np.dot(AP2W,PP2W)-YP2W    #vérification polynome
plot2 = plt. figure(2)
plt.title("category 2")
plt.plot (SP2W,label="Ply2 Web",color='red',linestyle='--')
plt.legend()
plt.xlabel('Distance along structure length (mm)')
plt.ylabel('Gap (mm)')
# _____
#Ply 19 (Web)
# _____

X= [-135,-45,45,135]    # positions Z

YP19W= [-5.2,-5.8,-1.35,-4.6,0,0]  #Ecarts

AP19W= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP19W[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP19W[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP19W[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP19W=np.linalg.solve(AP19W,YP19W) #vecteur solutions (coefficients de polynome)

SP19W =np.zeros((270))  # initialisation matrice des résultats
```

Appendix

```
for i in range (-135,135):
    SP19W [i+135] =PP19W[0]+PP19W[1]*i+PP19W[2]*(i**2) + PP19W[3]*(i**3)+ PP19W[4]*(i**4)+PP19W[5]*(i**5)  #remplissage
matrice des résultats

IP19W = np.dot(AP19W,PP19W)-YP19W    #vérification polynome

plt.plot (SP19W,label="Ply19 Web",color='grey',linestyle='--')
plt.legend()

# _____
#Ply 2 (b.flange)
# _____

X= [-135,-45,45,135]    # positions Z

YP2B= [-2.49,-1.5,-4.68,-6.02,0,0]  #Ecart

AP2B= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP2B[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP2B[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP2B[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP2B=np.linalg.solve(AP2B,YP2B) #vecteur solutions (coefficients de polynome)

SP2B =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP2B [i+135] =PP2B[0]+PP2B[1]*i+PP2B[2]*(i**2) + PP2B[3]*(i**3)+ PP2B[4]*(i**4)+PP2B[5]*(i**5)  #remplissage matrice des
résultats

IP2B = np.dot(AP2B,PP2B)-YP2B    #vérification polynome

plt.plot (SP2B,label="Ply2 Bottom flange",color='red')
plt.legend()

# _____
#Ply 19 (b.flange)
# _____

X= [-135,-45,45,135]    # positions Z
```

Appendix

```
YP19B= [-1.68,-0.2,3.73,2.4,0,0]  #Ecart

AP19B= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP19B[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP19B[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP19B[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP19B=np.linalg.solve(AP19B,YP19B) #vecteur solutions (coefficients de polynome)

SP19B =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP19B [i+135] =PP19B[0]+PP19B[1]*i+PP19B[2]*(i**2) + PP19B[3]*(i**3)+ PP19B[4]*(i**4)+PP19B[5]*(i**5)  #remplissage
    matrice des résultats

IP19B = np.dot(AP19B,PP19B)-YP19B  #vérification polynome

plt.plot (SP19B,label="Ply19 Bottom flange",color='grey')
plt.legend()
plt.grid()

##
#détermination des moyennes et écart-types pour catégorie 2 (plis 2 et 19)
#

ecartype_coef_cat2 =np.zeros((6))
moyenne_coef_cat2 =np.zeros((6))

coeff_cat20=[PP2W[0],PP19W[0],PP2B[0],PP19B[0]]
coeff_cat21=[PP2W[1],PP19W[1],PP2B[1],PP19B[1]]
coeff_cat22=[PP2W[2],PP19W[2],PP2B[2],PP19B[2]]
coeff_cat23=[PP2W[3],PP19W[3],PP2B[3],PP19B[3]]
coeff_cat24=[PP2W[4],PP19W[4],PP2B[4],PP19B[4]]
coeff_cat25=[PP2W[5],PP19W[5],PP2B[5],PP19B[5]]

ecartype_coef_cat2[0]= statistics.stdev(coeff_cat20)
ecartype_coef_cat2[1]= statistics.stdev(coeff_cat21)
ecartype_coef_cat2[2]= statistics.stdev(coeff_cat22)
ecartype_coef_cat2[3]= statistics.stdev(coeff_cat23)
ecartype_coef_cat2[4]= statistics.stdev(coeff_cat24)
ecartype_coef_cat2[5]= statistics.stdev(coeff_cat25)
```

Appendix

```
moyenne_coeff_cat2[0]= statistics.mean(coeff_cat20)
moyenne_coeff_cat2[1]= statistics.mean(coeff_cat21)
moyenne_coeff_cat2[2]= statistics.mean(coeff_cat22)
moyenne_coeff_cat2[3]= statistics.mean(coeff_cat23)
moyenne_coeff_cat2[4]= statistics.mean(coeff_cat24)
moyenne_coeff_cat2[5]= statistics.mean(coeff_cat25)

#
#category 3 (plis UD 2, 4, 6, 8, 10, 11, 13, 15, 17 et 19)
#
#
#Ply 2 (t.flange)
#

X= [-135,-45,45,135]    # positions Z

YP2T= [1.63,1.5,2.38,2.02,0,0]  #Ecart

AP2T= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP2T[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP2T[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP2T[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP2T=np.linalg.solve(AP2T,YP2T) #vecteur solutions (coefficients de polynome)

SP2T =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP2T [i+135] =PP2T[0]+PP2T[1]*i+PP2T[2]*(i**2) + PP2T[3]*(i**3)+ PP2T[4]*(i**4)+PP2T[5]*(i**5)  #remplissage matrice des
résultats

IP2T = np.dot(AP2T,PP2T)-YP2T    #vérification polynome

plot3 = plt.figure(3)
plt.title("category 3")
plt.plot (SP2T,label="Ply2 Top flange")
plt.legend()
plt.xlabel('Distance along structure length (mm)')
plt.ylabel('Gap (mm)')
#
#Ply 4
#
```

Appendix

```
YP4= [-0.89,-0.65,0.52,-0.69,0,0] #Ecart
```

```
AP4= np.zeros ((6,6)) # matrice des Ecart
```

```
for j in range (0,4):
```

```
    for i in range (0,6):
```

```
        AP4[j,i] = X[j]**(i) # remplissage matrice partie polynome
```

```
for i in range (1,6):
```

```
    AP4[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
```

```
    AP4[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite
```

```
PP4=np.linalg.solve(AP4,YP4) #vecteur solutions (coefficients de polynome)
```

```
SP4 =np.zeros((270)) # initialisation matrice des résultats
```

```
for i in range (-135,135):
```

```
    SP4 [i+135] =PP4[0]+PP4[1]*i+PP4[2]*(i**2) + PP4[3]*(i**3)+ PP4[4]*(i**4)+PP4[5]*(i**5) #remplissage matrice des résultats
```

```
IP4 = np.dot(AP4,PP4)-YP4 #vérification polynome
```

```
plt.plot (SP4,label="Ply4")
```

```
plt.legend()
```

```
# _____
```

```
#Ply 6
```

```
# _____
```

```
YP6= [0.006,-0.35,-1.32,-0.94,0,0] #Ecart
```

```
AP6= np.zeros ((6,6)) # matrice des Ecart
```

```
for j in range (0,4):
```

```
    for i in range (0,6):
```

```
        AP6[j,i] = X[j]**(i) # remplissage matrice partie polynome
```

```
for i in range (1,6):
```

```
    AP6[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
```

```
    AP6[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite
```

```
PP6=np.linalg.solve(AP6,YP6) #vecteur solutions (coefficients de polynome)
```

```
SP6 =np.zeros((270)) # initialisation matrice des résultats
```

```
for i in range (-135,135):
```

```
    SP6 [i+135] =PP6[0]+PP6[1]*i+PP6[2]*(i**2) + PP6[3]*(i**3)+ PP6[4]*(i**4)+PP6[5]*(i**5) #remplissage matrice des résultats
```

Appendix

```
IP6 = np.dot(AP6,PP6)-YP6    #vérification polynome

plt.plot (SP6,label="Ply6")
plt.legend()

#
#Ply 8
#

YP8= [0.11,0.064,-0.21,-0.68,0,0]    #Ecarts

AP8= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP8[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP8[4,i] = i * X[0]**(i-1)    #remplissage matrice partie tangente gauche
    AP8[5,i] = i * X[3]**(i-1)    #remplissage matrice partie tangente droite

PP8=np.linalg.solve(AP8,YP8) #vecteur solutions (coefficients de polynome)

SP8 =np.zeros((270))    # initialisation matrice des résultats

for i in range (-135,135):
    SP8 [i+135] =PP8[0]+PP8[1]*i+PP8[2]*(i**2) + PP8[3]*(i**3)+ PP8[4]*(i**4)+PP8[5]*(i**5)    #remplissage matrice des résultats

IP8 = np.dot(AP8,PP8)-YP8    #vérification polynome

plt.plot (SP8,label="Ply8")
plt.legend()

#
#Ply 10
#

YP10= [-0.29,-0.3,0.089,0.35,0,0]    #Ecarts

AP10= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP10[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP10[4,i] = i * X[0]**(i-1)    #remplissage matrice partie tangente gauche
    AP10[5,i] = i * X[3]**(i-1)    #remplissage matrice partie tangente droite
```

Appendix

```
PP10=np.linalg.solve(AP10,YP10) #vecteur solutions (coefficients de polynome)

SP10 =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP10 [i+135] =PP10[0]+PP10[1]*i+PP10[2]*(i**2) + PP10[3]*(i**3)+ PP10[4]*(i**4)+PP10[5]*(i**5) #remplissage matrice des
résultats

IP10 = np.dot(AP10,PP10)-YP10 #vérification polynome

plt.plot (SP10,label="Ply10")
plt.legend()
# _____
#Ply11
# _____

YP11= [0.56,0.16,-0.14,0.31,0,0] #Ecart

AP11= np.zeros ((6,6)) # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP11[j,i] = X[j]**(i) # remplissage matrice partie polynome

for i in range (1,6):
    AP11[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
    AP11[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite

PP11=np.linalg.solve(AP11,YP11) #vecteur solutions (coefficients de polynome)

SP11 =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP11 [i+135] =PP11[0]+PP11[1]*i+PP11[2]*(i**2) + PP11[3]*(i**3)+ PP11[4]*(i**4)+PP11[5]*(i**5) #remplissage matrice des
résultats

IP11 = np.dot(AP11,PP11)-YP11 #vérification polynome

plt.plot (SP11,label="Ply11")
plt.legend()

# _____
#Ply 13
# _____

YP13= [-1.17,0.026,0.64,-1.37,0,0] #Ecart
```

Appendix

```
AP13= np.zeros ((6,6))      # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP13[j,i] = X[j]**(i)      # remplissage matrice partie polynome

for i in range (1,6):
    AP13[4,i] = i * X[0]**(i-1)      #remplissage matrice partie tangente gauche
    AP13[5,i] = i * X[3]**(i-1)      #remplissage matrice partie tangente droite

PP13=np.linalg.solve(AP13,YP13) #vecteur solutions (coefficients de polynome)

SP13 =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP13 [i+135] =PP13[0]+PP13[1]*i+PP13[2]*(i**2) + PP13[3]*(i**3)+ PP13[4]*(i**4)+PP13[5]*(i**5) #remplissage matrice des
résultats

IP13 = np.dot(AP13,PP13)-YP13      #vérification polynome

plt.plot (SP13,label="Ply13")
plt.legend()
#
#Ply 15
#
YP15= [-0.29,0.06,0.25,-0.34,0,0] #Ecarts

AP15= np.zeros ((6,6))      # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP15[j,i] = X[j]**(i)      # remplissage matrice partie polynome

for i in range (1,6):
    AP15[4,i] = i * X[0]**(i-1)      #remplissage matrice partie tangente gauche
    AP15[5,i] = i * X[3]**(i-1)      #remplissage matrice partie tangente droite

PP15=np.linalg.solve(AP15,YP15) #vecteur solutions (coefficients de polynome)

SP15 =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP15 [i+135] =PP15[0]+PP15[1]*i+PP15[2]*(i**2) + PP15[3]*(i**3)+ PP15[4]*(i**4)+PP15[5]*(i**5) #remplissage matrice des
résultats

IP15 = np.dot(AP15,PP15)-YP15      #vérification polynome
```

Appendix

```
plt.plot (SP15,label="Ply15")
plt.legend()

# _____
#Ply 17
# _____

YP17= [-0.37,0.31,0.82,-0.43,0,0]  #Ecart

AP17= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP17[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP17[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP17[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP17=np.linalg.solve(AP17,YP17) #vecteur solutions (coefficients de polynome)

SP17 =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP17 [i+135] =PP17[0]+PP17[1]*i+PP17[2]*(i**2) + PP17[3]*(i**3)+ PP17[4]*(i**4)+PP17[5]*(i**5)  #remplissage matrice des
résultats

IP17 = np.dot(AP17,PP17)-YP17  #vérification polynome

plt.plot (SP17,label="Ply17")
plt.legend()

# _____
#Ply 19 (t.flange)
# _____

YP19T= [-0.55,0.72,1.021,0.71,0,0]  #Ecart

AP19T= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP19T[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP19T[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP19T[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite
```

Appendix

```
PP19T=np.linalg.solve(AP19T,YP19T) #vecteur solutions (coefficients de polynome)

SP19T=np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP19T [i+135]=PP19T[0]+PP19T[1]*i+PP19T[2]*(i**2) + PP19T[3]*(i**3)+ PP19T[4]*(i**4)+PP19T[5]*(i**5) #remplissage
matrice des résultats

IP19T = np.dot(AP19T,PP19T)-YP19T #vérification polynome

plt.plot (SP19T,label="Ply19 Top flange")
plt.legend()
plt.grid()

## _____
#détermination des moyennes et écart-types pour categorie 3
# _____

ecartype_coeff_cat3 =np.zeros((6))
moyenne_coeff_cat3 =np.zeros((6))

coeff_cat30=[PP2T[0],PP4[0],PP6[0],PP8[0],PP10[0],PP11[0],PP13[0],PP15[0],PP17[0],PP19T[0]]
coeff_cat31=[PP2T[1],PP4[1],PP6[1],PP8[1],PP10[1],PP11[1],PP13[1],PP15[1],PP17[1],PP19T[1]]
coeff_cat32=[PP2T[2],PP4[2],PP6[2],PP8[2],PP10[2],PP11[2],PP13[2],PP15[2],PP17[2],PP19T[2]]
coeff_cat33=[PP2T[3],PP4[3],PP6[3],PP8[3],PP10[3],PP11[3],PP13[3],PP15[3],PP17[3],PP19T[3]]
coeff_cat34=[PP2T[4],PP4[4],PP6[4],PP8[4],PP10[4],PP11[4],PP13[4],PP15[4],PP17[4],PP19T[4]]
coeff_cat35=[PP2T[5],PP4[5],PP6[5],PP8[5],PP10[5],PP11[5],PP13[5],PP15[5],PP17[5],PP19T[5]]

ecartype_coeff_cat3[0]= statistics.stdev(coeff_cat30)
ecartype_coeff_cat3[1]= statistics.stdev(coeff_cat31)
ecartype_coeff_cat3[2]= statistics.stdev(coeff_cat32)
ecartype_coeff_cat3[3]= statistics.stdev(coeff_cat33)
ecartype_coeff_cat3[4]= statistics.stdev(coeff_cat34)
ecartype_coeff_cat3[5]= statistics.stdev(coeff_cat35)

moyenne_coeff_cat3[0]= statistics.mean(coeff_cat30)
moyenne_coeff_cat3[1]= statistics.mean(coeff_cat31)
moyenne_coeff_cat3[2]= statistics.mean(coeff_cat32)
moyenne_coeff_cat3[3]= statistics.mean(coeff_cat33)
moyenne_coeff_cat3[4]= statistics.mean(coeff_cat34)
moyenne_coeff_cat3[5]= statistics.mean(coeff_cat35)

# _____
#categorie 4 (plis 5, 7, 14 et 16 drapés sur l'âme, partie gauche et droite)
# _____
```

Appendix

```
#  
#Ply 5 (Gauche)  
#  
  
X= [-135,-45,45,135]    # positions Z  
  
YP5G= [0.72,0.86,-1.3,-0.23,0,0]  #Ecart  
  
AP5G= np.zeros ((6,6))    # matrice des Ecart  
  
for j in range (0,4):  
    for i in range (0,6):  
        AP5G[j,i] = X[j]**(i)    # remplissage matrice partie polynome  
  
for i in range (1,6):  
    AP5G[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche  
    AP5G[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite  
  
PP5G=np.linalg.solve(AP5G,YP5G) #vecteur solutions (coefficients de polynome)  
  
SP5G =np.zeros((270))  # initialisation matrice des résultats  
  
for i in range (-135,135):  
    SP5G [i+135] =PP5G[0]+PP5G[1]*i+PP5G[2]*(i**2) + PP5G[3]*(i**3)+ PP5G[4]*(i**4)+PP5G[5]*(i**5)  #remplissage matrice des  
    résultats  
  
IP5G = np.dot(AP5G,PP5G)-YP5G    #vérification polynome  
  
plot4 = plt.figure(4)  
plt.title("category 4")  
plt.plot (SP5G,label="Ply5")  
plt.legend()  
plt.xlabel('Distance along structure length (mm)')  
plt.ylabel('Gap (mm)')  
#  
#Ply 7 (Gauche)  
#  
  
X= [-135,-45,45,135]    # positions Z  
  
YP7G= [0.11,0.42,1.03,-0.11,0,0]  #Ecart  
  
AP7G= np.zeros ((6,6))    # matrice des Ecart  
  
for j in range (0,4):  
    for i in range (0,6):
```

Appendix

```
AP7G[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP7G[4,i] = i * X[0]**(i-1)    #remplissage matrice partie tangente gauche
    AP7G[5,i] = i * X[3]**(i-1)    #remplissage matrice partie tangente droite

PP7G=np.linalg.solve(AP7G,YP7G) #vecteur solutions (coefficients de polynome)

SP7G =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP7G [i+135] =PP7G[0]+PP7G[1]*i+PP7G[2]*(i**2) + PP7G[3]*(i**3)+ PP7G[4]*(i**4)+PP7G[5]*(i**5) #remplissage matrice des
résultats

IP7G = np.dot(AP7G,PP7G)-YP7G    #vérification polynome

plt.plot (SP7G,label="Ply7")
plt.legend()

# _____
#Ply 14 (Gauche)
# _____

X= [-135,-45,45,135]    # positions Z

YP14G= [0.48,1.01,1.5,-0.25,0,0] #Ecarts

AP14G= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP14G[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP14G[4,i] = i * X[0]**(i-1)    #remplissage matrice partie tangente gauche
    AP14G[5,i] = i * X[3]**(i-1)    #remplissage matrice partie tangente droite

PP14G=np.linalg.solve(AP14G,YP14G) #vecteur solutions (coefficients de polynome)

SP14G =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP14G [i+135] =PP14G[0]+PP14G[1]*i+PP14G[2]*(i**2) + PP14G[3]*(i**3)+ PP14G[4]*(i**4)+PP14G[5]*(i**5) #remplissage
matrice des résultats

IP14G = np.dot(AP14G,PP14G)-YP14G    #vérification polynome
```

Appendix

```
plt.plot (SP14G,label="Ply14")
plt.legend()
# _____

#Ply 16 (Gauche)
# _____

X= [-135,-45,45,135]    # positions Z

YP16G= [-0.22,0.09,0.32,0.04,0,0]  #Ecart

AP16G= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP16G[j,i] = X[j]**(i)    # remplissage matrice partie polynome

for i in range (1,6):
    AP16G[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP16G[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP16G=np.linalg.solve(AP16G,YP16G) #vecteur solutions (coefficients de polynome)

SP16G =np.zeros((270))  # initialisation matrice des résultats

for i in range (-135,135):
    SP16G [i+135] =PP16G[0]+PP16G[1]*i+PP16G[2]*(i**2) + PP16G[3]*(i**3)+ PP16G[4]*(i**4)+PP16G[5]*(i**5)  #remplissage matrice des résultats

IP16G = np.dot(AP16G,PP16G)-YP16G    #vérification polynome

plt.plot (SP16G,label="Ply16")
plt.legend()
plt.grid()
# _____

#Ply 5 (Droite)
# _____

X= [-135,-45,45,135]    # positions Z

YP5D= [2.12,0.67,1.92,0.57,0,0]  #Ecart

AP5D= np.zeros ((6,6))    # matrice des Ecart

for j in range (0,4):
    for i in range (0,6):
        AP5D[j,i] = X[j]**(i)    # remplissage matrice partie polynome
```

Appendix

```
for i in range (1,6):
    AP5D[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP5D[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP5D=np.linalg.solve(AP5D,YP5D) #vecteur solutions (coefficients de polynome)

SP5D =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP5D [i+135] =PP5D[0]+PP5D[1]*i+PP5D[2]*(i**2) + PP5D[3]*(i**3)+ PP5D[4]*(i**4)+PP5D[5]*(i**5)  #remplissage matrice
des résultats

IP5D = np.dot(AP5D,PP5D)-YP5D  #vérification polynome

# _____
#Ply 7 (Droite)
# _____

X= [-135,-45,45,135]    # positions Z

YP7D= [4.2,3.31,4.79,4.44,0,0]  #Ecarts

AP7D= np.zeros ((6,6))    # matrice des Ecarts

for j in range (0,4):
    for i in range (0,6):
        AP7D[j,i] = X[j]**(i)  # remplissage matrice partie polynome

for i in range (1,6):
    AP7D[4,i] = i * X[0]**(i-1)  #remplissage matrice partie tangente gauche
    AP7D[5,i] = i * X[3]**(i-1)  #remplissage matrice partie tangente droite

PP7D=np.linalg.solve(AP7D,YP7D) #vecteur solutions (coefficients de polynome)

SP7D =np.zeros((270)) # initialisation matrice des résultats

for i in range (-135,135):
    SP7D [i+135] =PP7D[0]+PP7D[1]*i+PP7D[2]*(i**2) + PP7D[3]*(i**3)+ PP7D[4]*(i**4)+PP7D[5]*(i**5)  #remplissage matrice
des résultats

IP7D = np.dot(AP7D,PP7D)-YP7D  #vérification polynome

# _____
#Ply 14 (Droite)
# _____

X= [-135,-45,45,135]    # positions Z
```

Appendix

```
YP14D= [2.95,3.81,3.82,2.48,0,0] #Ecart
```

```
AP14D= np.zeros ((6,6)) # matrice des Ecart
```

```
for j in range (0,4):
```

```
    for i in range (0,6):
```

```
        AP14D[j,i] = X[j]**(i) # remplissage matrice partie polynome
```

```
for i in range (1,6):
```

```
    AP14D[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
```

```
    AP14D[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite
```

```
PP14D=np.linalg.solve(AP14D,YP14D) #vecteur solutions (coefficients de polynome)
```

```
SP14D =np.zeros((270)) # initialisation matrice des résultats
```

```
for i in range (-135,135):
```

```
    SP14D [i+135] =PP14D[0]+PP14D[1]*i+PP14D[2]*(i**2) + PP14D[3]*(i**3)+ PP14D[4]*(i**4)+PP14D[5]*(i**5) #remplissage  
matrice des résultats
```

```
IP14D = np.dot(AP14D,PP14D)-YP14D #vérification polynome
```

```
#
```

```
#Ply 16 (Droite)
```

```
#
```

```
X= [-135,-45,45,135] # positions Z
```

```
YP16D= [-0.63,0.36,2.61,1.27,0,0] #Ecart
```

```
AP16D= np.zeros ((6,6)) # matrice des Ecart
```

```
for j in range (0,4):
```

```
    for i in range (0,6):
```

```
        AP16D[j,i] = X[j]**(i) # remplissage matrice partie polynome
```

```
for i in range (1,6):
```

```
    AP16D[4,i] = i * X[0]**(i-1) #remplissage matrice partie tangente gauche
```

```
    AP16D[5,i] = i * X[3]**(i-1) #remplissage matrice partie tangente droite
```

```
PP16D=np.linalg.solve(AP16D,YP16D) #vecteur solutions (coefficients de polynome)
```

```
SP16D =np.zeros((270)) # initialisation matrice des résultats
```

```
for i in range (-135,135):
```

```
    SP16D[i+135] =PP16D[0]+PP16D[1]*i+PP16D[2]*(i**2) + PP16D[3]*(i**3)+ PP16D[4]*(i**4)+PP16D[5]*(i**5) #remplissage  
matrice des résultats
```

Appendix

```
IP16D = np.dot(AP16D,PP16D)-YP16D    #vérification polynome
```

```
##
```

```
#détermination des moyennes et écart-types pour catégorie 4
```

```
#
```

```
ecartype_coeff_cat4=np.zeros((6))
```

```
moyenne_coeff_cat4=np.zeros((6))
```

```
coeff_cat40=[PP5G[0],PP7G[0],PP14G[0],PP16G[0],PP5D[0],PP7D[0],PP14D[0],PP16D[0]]
```

```
coeff_cat41=[PP5G[1],PP7G[1],PP14G[1],PP16G[1],PP5D[1],PP7D[1],PP14D[1],PP16D[1]]
```

```
coeff_cat42=[PP5G[2],PP7G[2],PP14G[2],PP16G[2],PP5D[2],PP7D[2],PP14D[2],PP16D[2]]
```

```
coeff_cat43=[PP5G[3],PP7G[3],PP14G[3],PP16G[3],PP5D[3],PP7D[3],PP14D[3],PP16D[3]]
```

```
coeff_cat44=[PP5G[4],PP7G[4],PP14G[4],PP16G[4],PP5D[4],PP7D[4],PP14D[4],PP16D[4]]
```

```
coeff_cat45=[PP5G[5],PP7G[5],PP14G[5],PP16G[5],PP5D[5],PP7D[5],PP14D[5],PP16D[5]]
```

```
ecartype_coeff_cat4[0]= statistics.stdev(coeff_cat40)
```

```
ecartype_coeff_cat4[1]= statistics.stdev(coeff_cat41)
```

```
ecartype_coeff_cat4[2]= statistics.stdev(coeff_cat42)
```

```
ecartype_coeff_cat4[3]= statistics.stdev(coeff_cat43)
```

```
ecartype_coeff_cat4[4]= statistics.stdev(coeff_cat44)
```

```
ecartype_coeff_cat4[5]= statistics.stdev(coeff_cat45)
```

```
moyenne_coeff_cat4[0]= statistics.mean(coeff_cat40)
```

```
moyenne_coeff_cat4[1]= statistics.mean(coeff_cat41)
```

```
moyenne_coeff_cat4[2]= statistics.mean(coeff_cat42)
```

```
moyenne_coeff_cat4[3]= statistics.mean(coeff_cat43)
```

```
moyenne_coeff_cat4[4]= statistics.mean(coeff_cat44)
```

```
moyenne_coeff_cat4[5]= statistics.mean(coeff_cat45)
```

```
#####  
#####
```

```
from time import gmtime, strftime
```

```
date=strftime("%d %b %Y %H_%M ", gmtime())
```

```
f = open('test generation lisse%s .INP'%(date),'w')
```

```
# Analyse stochastique
```

```
# Modèles stochastique : spectrale et chaos polynomial
```

```
# Forme spectrale pour l'orientation :
```

```
# somme des fonctions trigonométriques avec amplitudes et phases aléatoires
```

```
import numpy as np
```

```
#definition des paramètres géométriques de la lisse (rayon de courbure et longueur en mm)
```

```
R=5.1
```

Appendix

L=380

#programmation des différentes zones maillées (de 1 à 9) : nombres fixés a priori

N1=30 **#nombre de maille sur zone 1**

N2=5 **#nombre de maille sur zone 2**

N3=20 **#nombre de maille sur zone 3**

N4=5 **#nombre de maille sur zone 4**

N5=20 **#nombre de maille sur zone 5**

N6=5 **#nombre de maille sur zone 6**

N7=20 **#nombre de maille sur zone 7**

N8=5 **#nombre de maille sur zone 8**

N9=30 **#nombre de maille sur zone 9**

Nz=380 **#nombre de maille sur z**

coordonnées des points de début des zones (à gauche)

coord100101=(-49.789472,3.59)

coord100201=(-26.860851,3.888326)

coord100301=(-22.303152,6.832566)

coord100401=(-12.873111,27.055353)

coord100501=(-8.250942,30)

#symétrique (à droite)

coord100601=(8.250942,30)

coord100701=(12.873111,27.055353)

coord100801=(22.303152,6.832566)

coord100901=(26.860851,3.888326)

coord101001=(49.789472,3.59)

#programmation des différents arcs (de 1 à 4)

#C1 centre arc 1

coordC1=(-26.925322,8.987919) **#à vérifier**

#C2 centre arc 2

coordC2=(-8.25,24.9)

#C3 centre arc 3

coordC4=(26.925322,8.987919)

#C4 centre arc 4

coordC3=(8.25,24.9)

épaisseur des plis

ep = np.zeros((20))

ep = ([0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3,0.3])

from math import cos

from math import sin

from math import acos

#noeudX=np.eye(2*Nx,2*Ny) (alternative de programmation non utilisée)

noeudX1=numpy.zeros((N1+1,Nz+1)) **#initialisation matrice des coordonnées en X**

noeudY1=numpy.zeros((N1+1,Nz+1)) **#initialisation matrice des coordonnées en Y**

noeudZ1=numpy.zeros((N1+1,Nz+1)) **#initialisation matrice des coordonnées en Z**

noeudX2=numpy.zeros((N2+1,Nz+1)) **#initialisation matrice des coordonnées en X**

noeudY2=numpy.zeros((N2+1,Nz+1)) **#initialisation matrice des coordonnées en Y**

Appendix

```
noeudZ2=numpy.zeros((N2+1,Nz+1)) #initialisation matrice des coordonnées en Z
noeudX3=numpy.zeros((N3+1,Nz+1)) #initialisation matrice des coordonnées en X
noeudY3=numpy.zeros((N3+1,Nz+1)) #initialisation matrice des coordonnées en Y
noeudZ3=numpy.zeros((N3+1,Nz+1)) #initialisation matrice des coordonnées en Z
noeudX4=numpy.zeros((N4+1,Nz+1)) #initialisation matrice des coordonnées en X
noeudY4=numpy.zeros((N4+1,Nz+1)) #initialisation matrice des coordonnées en Y
noeudZ4=numpy.zeros((N4+1,Nz+1)) #initialisation matrice des coordonnées en Z
noeudX5=numpy.zeros((N5+1,Nz+1)) #initialisation matrice des coordonnées en X
noeudY5=numpy.zeros((N5+1,Nz+1)) #initialisation matrice des coordonnées en Y
noeudZ5=numpy.zeros((N5+1,Nz+1)) #initialisation matrice des coordonnées en Z
noeudX6=numpy.zeros((N6+1,Nz+1)) #initialisation matrice des coordonnées en X
noeudY6=numpy.zeros((N6+1,Nz+1)) #initialisation matrice des coordonnées en Y
noeudZ6=numpy.zeros((N6+1,Nz+1)) #initialisation matrice des coordonnées en Z
noeudX7=numpy.zeros((N7+1,Nz+1)) #initialisation matrice des coordonnées en X
noeudY7=numpy.zeros((N7+1,Nz+1)) #initialisation matrice des coordonnées en Y
noeudZ7=numpy.zeros((N7+1,Nz+1)) #initialisation matrice des coordonnées en Z
noeudX8=numpy.zeros((N8+1,Nz+1)) #initialisation matrice des coordonnées en X
noeudY8=numpy.zeros((N8+1,Nz+1)) #initialisation matrice des coordonnées en Y
noeudZ8=numpy.zeros((N8+1,Nz+1)) #initialisation matrice des coordonnées en Z
noeudX9=numpy.zeros((N9+1,Nz+1)) #initialisation matrice des coordonnées en X
noeudY9=numpy.zeros((N9+1,Nz+1)) #initialisation matrice des coordonnées en Y
noeudZ9=numpy.zeros((N9+1,Nz+1)) #initialisation matrice des coordonnées en Z
noeudN1=np.eye(N1+1,Nz+1) #initialisation matrice des numéros des noeuds
noeudN2=np.eye(N2+1,Nz+1) #initialisation matrice des numéros des noeuds
noeudN3=np.eye(N3+1,Nz+1) #initialisation matrice des numéros des noeuds
noeudN4=np.eye(N4+1,Nz+1) #initialisation matrice des numéros des noeuds
noeudN5=np.eye(N5+1,Nz+1) #initialisation matrice des numéros des noeuds
noeudN6=np.eye(N6+1,Nz+1) #initialisation matrice des numéros des noeuds
noeudN7=np.eye(N7+1,Nz+1) #initialisation matrice des numéros des noeuds
noeudN8=np.eye(N8+1,Nz+1) #initialisation matrice des numéros des noeuds
noeudN9=np.eye(N9+1,Nz+1) #initialisation matrice des numéros des noeuds
```

$N_x = N_1 + N_2 + N_3 + N_4 + N_5 + N_6 + N_7 + N_8 + N_9$ #nombre total des mailles en x et y

```
noeudXt= np.zeros ((Nx,Nz))
noeudYt= np.zeros ((Nx,Nz))
noeudZt= np.zeros ((Nx,Nz))
noeudNt= np.zeros ((Nx,Nz))
noeudNtot= np.zeros ((Nx+1,Nz+1))
noeudXtot= np.zeros ((Nx+1,Nz+1))
```

```
C1N=numpy.zeros([2,2]) #initialisation de l'arc de cercle C1
C1O=numpy.zeros([2,2])
C2P=numpy.zeros([2,2]) #initialisation de l'arc de cercle C2
C2Q=numpy.zeros([2,2])
C3S=numpy.zeros([2,2]) #initialisation de l'arc de cercle C3
C3T=numpy.zeros([2,2])
```

Appendix

```
C4U=numpy.zeros([2,2]) #initialisation de l'arc de cercle C4
```

```
C4V=numpy.zeros([2,2])
```

```
f.write('*Heading\n')
```

```
f.write('** Job name: try Model name: Model-1\n')
```

```
f.write('** Generated by: Abaqus/CAE 2018\n')
```

```
f.write('Preprint, echo=NO, model=NO, history=NO, contact=NO\n')
```

```
f.write('*\n')
```

```
f.write('** PARTS\n')
```

```
f.write('*\n')
```

```
f.write('*Part, name=PART-1\n')
```

```
f.write('*Node\n')
```

```
#
```

```
# GENERATION DE LA GEOMETRIE De la lisse
```

```
#
```

```
#définition des noeuds et leurs numéros
```

```
# programmation zone 1 (ligne)
```

```
for i in range (1,N1+2):
```

```
    for j in range (1,Nz+2):
```

```
        #Numero de noeud
```

```
        noeudN1[i-1,j-1]= 100000*j+100+i
```

```
        sn=str(noeudN1[i-1,j-1].astype(int))
```

```
        f.write(sn)
```

```
        f.write(', ')
```

```
        #noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L #équation des positions des noeuds en X (perpendiculaire à l'axe de la lisse)
```

```
        noeudX1[i-1,j-1]=coord100101[0]+((coord100201[0]-coord100101[0])/N1*(i-1))
```

```
        sx=str(round(noeudX1[i-1,j-1],5))
```

```
        #noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny)) #équation des positions des noeuds en Y (hauteur de la lisse)
```

```
        noeudY1[i-1,j-1]=coord100101[1]+((coord100201[1]-coord100101[1])/N1*(i-1))
```

```
        sy=str(round(noeudY1[i-1,j-1],5))
```

```
        noeudZ1[i-1,j-1]= L/Nz*(j-1) #équation des positions des noeuds en Z (axe de la lisse)
```

```
        sz=str(round(noeudZ1[i-1,j-1],5))
```

```
        f.write(sx)
```

```
        f.write(', ')
```

```
        f.write(sy)
```

```
        f.write(', ')
```

```
        f.write(sz)
```

```
        f.write('\n')
```

```
# programmation zone 2 (arc de cercle)
```

```
for i in range (1,N2+2):
```

```
    for j in range (1,Nz+2):
```

```
        #Numero de noeud
```

Appendix

```
noeudN2[i-1,j-1]= 100000*j+200+i
sn=str(noeudN2[i-1,j-1].astype(int))
f.write(sn)
f.write(' ')
C1N[0,0]=(coord100201[0]-coordC1[0])/R  #norme 5.1 (rayon)
C1N[1,0]=(coord100201[1]-coordC1[1])/R
C1O[0,0]=(coord100301[0]-coordC1[0])/R  #norme
C1O[1,0]=(coord100301[1]-coordC1[1])/R
costheta1=C1N[0,0]*C1O[0,0]+C1N[1,0]*C1O[1,0]
theta1=-acos(costheta1)
thetaN1=+acos(C1N[0,0])
#noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L  #équation des positions des noeuds en X
noeudX2[i-1,j-1]= coordC1[0]+R*cos(thetaN1+(theta1/N2)*(i-1))
sx=str(round(noeudX2[i-1,j-1],5))
#noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny))  #équation des positions des noeuds en Y
noeudY2[i-1,j-1]=coordC1[1]-R*sin(thetaN1+(theta1/N2)*(i-1))
sy=str(round(noeudY2[i-1,j-1],5))
noeudZ2[i-1,j-1]= L/Nz*(j-1)  #équation des positions des noeuds en Z
sz=str(round(noeudZ2[i-1,j-1],5))
f.write(sx)
f.write(' ')
f.write(sy)
f.write(' ')
f.write(sz)
f.write('\n')
```

programmation zone 3 (ligne)

```
for i in range (1,N3+2):
    for j in range (1,Nz+2):
        #Numero de noeud
        noeudN3[i-1,j-1]= 100000*j+300+i
        sn=str(noeudN3[i-1,j-1].astype(int))
        f.write(sn)
        f.write(' ')
        #noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L  #équation des positions des noeuds en X
        noeudX3[i-1,j-1]=coord100301[0]+((coord100401[0]-coord100301[0])/N3*(i-1))
        sx=str(round(noeudX3[i-1,j-1],5))
        #noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny))  #équation des positions des noeuds en Y
        noeudY3[i-1,j-1]=coord100301[1]+((coord100401[1]-coord100301[1])/N3*(i-1))
        sy=str(round(noeudY3[i-1,j-1],5))
        #noeudZ[i-1,j-1]= L/Nz*(j-1)  #équation des positions des noeuds en Z
        sz=str(round(noeudZ3[i-1,j-1],5))
        f.write(sx)
        f.write(' ')
        f.write(sy)
        f.write(' ')
```

Appendix

```
f.write(sz)
f.write('\n')
```

programmation zone 4 (arc de cercle)

```
for i in range (1,N4+2):
    for j in range (1,Nz+2):
        #Numero de noeud
        noeudN4[i-1,j-1]= 100000*j+400+i    #par exemple noeud1 : 1004001
        sn=str(noeudN4[i-1,j-1].astype(int))
        f.write(sn)
        f.write(' ')
        C2P[0,0]=(coord100401[0]-coordC2[0])/R  #/norme 5.1 (rayon)
        C2P[1,0]=(coord100401[1]-coordC2[1])/R
        C2Q[0,0]=(coord100501[0]-coordC2[0])/R  #/norme
        C2Q[1,0]=(coord100501[1]-coordC2[1])/R
        costheta2=C2P[0,0]*C2Q[0,0]+C2P[1,0]*C2Q[1,0]
        theta2=-acos(costheta2)
        thetaN2=acos(C2P[0,0])

        #noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L  #équation des positions des noeuds en X
        noeudX4[i-1,j-1]= coordC2[0]+R*cos(thetaN2+(theta2/N4)*(i-1))
        sx=str(round(noeudX4[i-1,j-1],5))
        #noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny))  #équation des positions des noeuds en Y
        noeudY4[i-1,j-1]=coordC2[1]+R*sin(thetaN2+(theta2/N4)*(i-1))
        sy=str(round(noeudY4[i-1,j-1],5))
        noeudZ4[i-1,j-1]= L/Nz*(j-1)  #équation des positions des noeuds en Z
        sz=str(round(noeudZ4[i-1,j-1],5))
        f.write(sx)
        f.write(' ')
        f.write(sy)
        f.write(' ')
        f.write(sz)
        f.write('\n')
```

programmation zone 5 (ligne)

```
for i in range (1,N5+2):
    for j in range (1,Nz+2):
        #Numero de noeud
        noeudN5[i-1,j-1]= 100000*j+500+i
        sn=str(noeudN5[i-1,j-1].astype(int))
        f.write(sn)
        f.write(' ')
        #noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L  #équation des positions des noeuds en X
        noeudX5[i-1,j-1]=coord100501[0]+((coord100601[0]-coord100501[0])/N5*(i-1))
        sx=str(round(noeudX5[i-1,j-1],5))
        #noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny))  #équation des positions des noeuds en Y
```

Appendix

```
noeudY5[i-1,j-1]=coord100501[1]+((coord100601[1]-coord100501[1])/N5*(i-1))
sy=str(round(noeudY5[i-1,j-1],5))
noeudZ5[i-1,j-1]= L/Nz*(j-1)  #équation des positions des noeuds en Z
sz=str(round(noeudZ5[i-1,j-1],5))
f.write(sx)
f.write(' ')
f.write(sy)
f.write(' ')
f.write(sz)
f.write('\n')
```

programmation zone 6 (arc de cercle)

```
for i in range (1,N6+2):
    for j in range (1,Nz+2):
        #Numero de noeud
        noeudN6[i-1,j-1]= 100000*j+600+i
        sn=str(noeudN6[i-1,j-1].astype(int))
        f.write(sn)
        f.write(' ')
        C3S[0,0]=(coord100601[0]-coordC3[0])/R  #/norme 5.1 (rayon)
        C3S[1,0]=(coord100601[1]-coordC3[1])/R
        C3T[0,0]=(coord100701[0]-coordC3[0])/R  #/norme
        C3T[1,0]=(coord100701[1]-coordC3[1])/R
        costheta3=(C3S[0,0]*C3T[0,0])+(C3S[1,0]*C3T[1,0])
        theta3=-acos(costheta3)
        thetaN3=acos(C3S[0,0])
        #noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L  #équation des positions des noeuds en X
        noeudX6[i-1,j-1]= coordC3[0]+R*cos(thetaN3+(theta3/N6)*(i-1))
        sx=str(round(noeudX6[i-1,j-1],5))
        #noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny))  #équation des positions des noeuds en Y
        noeudY6[i-1,j-1]=coordC3[1]+R*sin(thetaN3+(theta3/N6)*(i-1))
        sy=str(round(noeudY6[i-1,j-1],5))
        noeudZ6[i-1,j-1]= L/Nz*(j-1)  #équation des positions des noeuds en Z
        sz=str(round(noeudZ6[i-1,j-1],5))
        f.write(sx)
        f.write(' ')
        f.write(sy)
        f.write(' ')
        f.write(sz)
        f.write('\n')
```

programmation zone 7 (ligne)

```
for i in range (1,N7+2):
    for j in range (1,Nz+2):
        #Numero de noeud
        noeudN7[i-1,j-1]= 100000*j+700+i
```

Appendix

```
sn=str(noeudN7[i-1,j-1].astype(int))
f.write(sn)
f.write(' ')
#noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L  #équation des positions des noeuds en X
noeudX7[i-1,j-1]=coord100701[0]+((coord100801[0]-coord100701[0])/N7*(i-1))
sx=str(round(noeudX7[i-1,j-1],5))
#noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny))  #équation des positions des noeuds en Y
noeudY7[i-1,j-1]=coord100701[1]+((coord100801[1]-coord100701[1])/N7*(i-1))
sy=str(round(noeudY7[i-1,j-1],5))
noeudZ7[i-1,j-1]= L/Nz*(j-1)  #équation des positions des noeuds en Z
sz=str(round(noeudZ7[i-1,j-1],5))
f.write(sx)
f.write(' ')
f.write(sy)
f.write(' ')
f.write(sz)
f.write('\n')
```

programmation zone 8 (arc de cercle)

```
for i in range (1,N8+2):
```

```
    for j in range (1,Nz+2):
```

```
        #Numero de noeud
```

```
        noeudN8[i-1,j-1]= 100000*j+800+i      #
```

```
        sn=str(noeudN8[i-1,j-1].astype(int))
```

```
        f.write(sn)
```

```
        f.write(' ')
```

```
        C4U[0,0]=(coord100801[0]-coordC4[0])/R  #/norme 5.1 (rayon)
```

```
        C4U[1,0]=(coord100801[1]-coordC4[1])/R
```

```
        C4V[0,0]=(coord100901[0]-coordC4[0])/R  #/norme
```

```
        C4V[1,0]=(coord100901[1]-coordC4[1])/R
```

```
        costheta4=C4U[0,0]*C4V[0,0]+C4U[1,0]*C4V[1,0]
```

```
        theta4=-acos(costheta4)
```

```
        thetaN4=acos(C4U[0,0])
```

```
        #noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L  #équation des positions des noeuds en X
```

```
        noeudX8[i-1,j-1]= coordC4[0]+R*cos(thetaN4+(theta4/N8)*(i-1))
```

```
        sx=str(round(noeudX8[i-1,j-1],5))
```

```
        #noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny))  #équation des positions des noeuds en Y
```

```
        noeudY8[i-1,j-1]=coordC4[1]-R*sin(thetaN4+(theta4/N8)*(i-1))
```

```
        sy=str(round(noeudY8[i-1,j-1],5))
```

```
        noeudZ8[i-1,j-1]= L/Nz*(j-1)  #équation des positions des noeuds en Z
```

```
        sz=str(round(noeudZ8[i-1,j-1],5))
```

```
        f.write(sx)
```

```
        f.write(' ')
```

```
        f.write(sy)
```

```
        f.write(' ')
```

```
        f.write(sz)
```

```
        f.write('\n')
```

Appendix

programmation zone 9 (ligne)

```
for i in range (1,N9+2):
    for j in range (1,Nz+2):
        #Numero de noeud
        noeudN9[i-1,j-1]= 100000*j+900+i
        sn=str(noeudN9[i-1,j-1].astype(int))
        f.write(sn)
        f.write(', ')
        #noeudX[i-1,j-1]= -L/2 + (i-1)/Nx*L  #équation des positions des noeuds en X
        noeudX9[i-1,j-1]=coord100901[0]+((coord101001[0]-coord100901[0])/N9*(i-1))
        sx=str(round(noeudX9[i-1,j-1],5))
        #noeudY[i-1,j-1]= R*sin(Theta/2*(-1+(j-1)*Theta/Ny))  #équation des positions des noeuds en Y
        noeudY9[i-1,j-1]=coord100901[1]+((coord101001[1]-coord100901[1])/N9*(i-1))
        sy=str(round(noeudY9[i-1,j-1],5))
        noeudZ9[i-1,j-1]= L/Nz*(j-1)  #équation des positions des noeuds en Z
        sz=str(round(noeudZ9[i-1,j-1],5))
        f.write(sx)
        f.write(', ')
        f.write(sy)
        f.write(', ')
        f.write(sz)
        f.write('\n')
```

#Définition et numérotation des mailles zone 1

chaque maille est définie par 4 noeuds

```
O1=np.eye(N1,Nz)
O2=np.eye(N2,Nz)
O3=np.eye(N3,Nz)
O4=np.eye(N4,Nz)
O5=np.eye(N5,Nz)
O6=np.eye(N6,Nz)
O7=np.eye(N7,Nz)
O8=np.eye(N8,Nz)
O9=np.eye(N9,Nz)
```

```
f.write('*Element, type=S4\n')
for i in range (0,N1-1):
    for j in range(0,Nz):
        O1[i,j]=noeudN1[i,j]
        OO1=str(O1[i,j].astype(int))
        f.write(OO1)
        f.write(', ')
        a=str(noeudN1[i,j].astype(int))
        f.write(a)
        f.write(', ')
        b=str(noeudN1[i,j+1].astype(int))
```

Appendix

```
f.write(b)
f.write(', ')
d=str(noeudN1[i+1,j+1].astype(int))
f.write(d)
f.write(', ')
c=str(noeudN1[i+1,j].astype(int))
f.write(c)
f.write('\n')
noeudXt[i,j]=noeudX1[i,j]
noeudYt[i,j]=noeudY1[i,j]
noeudZt[i,j]=noeudZ1[i,j]
noeudNt[i,j]=noeudN1[i,j]
noeudNtot[i,j]=noeudN1[i,j]
noeudXtot[i,j]=noeudX1[i,j]
```

jonction 1_2

```
for j in range(0,Nz):
    O1[N1-1,j]=noeudN1[N1-1,j]
    OO1=str(O1[N1-1,j].astype(int))
    f.write(OO1)
    f.write(', ')
    a=str(noeudN1[N1-1,j].astype(int))
    f.write(a)
    f.write(', ')
    b=str(noeudN1[N1-1,j+1].astype(int))
    f.write(b)
    f.write(', ')
    d=str(noeudN2[0,j+1].astype(int))
    f.write(d)
    f.write(', ')
    c=str(noeudN2[0,j].astype(int))
    f.write(c)
    f.write('\n')

    noeudXt[N1-1,j]=noeudX1[N1-1,j]
    noeudYt[N1-1,j]=noeudY1[N1-1,j]
    noeudZt[N1-1,j]=noeudZ1[N1-1,j]
    noeudNt[N1-1,j]=noeudN1[N1-1,j]
    noeudNtot[N1-1,j]=noeudN1[N1-1,j]
    noeudXtot[N1-1,j]=noeudX1[N1-1,j]

for i in range (-1,N1):
    for j in range (-1,Nz):
        noeudNtot[i+1,j+1]=noeudN1[i+1,j+1]
        noeudXtot[i+1,j+1]=noeudX1[i+1,j+1]
```

Appendix

#Définition et numérotation des mailles zone 2

```
for i in range(0,N2-1):
    for j in range(0,Nz):
        O2[i,j]=noeudN2[i,j]
        OO2=str(O2[i,j].astype(int))
        f.write(OO2)
        f.write(', ')
        a=str(noeudN2[i,j].astype(int))
        f.write(a)
        f.write(', ')
        b=str(noeudN2[i,j+1].astype(int))
        f.write(b)
        f.write(', ')
        d=str(noeudN2[i+1,j+1].astype(int))
        f.write(d)
        f.write(', ')
        c=str(noeudN2[i+1,j].astype(int))
        f.write(c)
        f.write("\n")

    noeudXt[N1+i,j]=noeudX2[i,j]
    noeudYt[N1+i,j]=noeudY2[i,j]
    noeudZt[N1+i,j]=noeudZ2[i,j]
    noeudNt[N1+i,j]=noeudN2[i,j]
    noeudNtot[N1+i,j]=noeudN2[i,j]
    noeudXtot[N1+i,j]=noeudX2[i,j]
```

jonction 2_3

```
for j in range(0,Nz):
    O2[N2-1,j]=noeudN2[N2-1,j]
    OO2=str(O2[N2-1,j].astype(int))
    f.write(OO2)
    f.write(', ')
    a=str(noeudN2[N2-1,j].astype(int))
    f.write(a)
    f.write(', ')
    b=str(noeudN2[N2-1,j+1].astype(int))
    f.write(b)
    f.write(', ')
    d=str(noeudN3[0,j+1].astype(int))
    f.write(d)
    f.write(', ')
    c=str(noeudN3[0,j].astype(int))
    f.write(c)
    f.write("\n")
```

Appendix

```
noeudXt[N1+N2-1,j]=noeudX2[N2-1,j]
noeudYt[N1+N2-1,j]=noeudY2[N2-1,j]
noeudZt[N1+N2-1,j]=noeudZ2[N2-1,j]
noeudNt[N1+N2-1,j]=noeudN2[N2-1,j]
noeudNtot[N1+N2-1,j]=noeudN2[N2-1,j]
noeudXtot[N1+N2-1,j]=noeudX2[N2-1,j]
```

```
for i in range (0,N2):
```

```
    for j in range (0,Nz):
```

```
        noeudNtot[N1+i+1,j+1]=noeudN2[i+1,j+1]
```

```
        noeudXtot[N1+i+1,j+1]=noeudX2[i+1,j+1]
```

#Définition et numérotation des mailles zone 3

chaque maille est définie par 4 noeuds

```
for i in range (0,N3-1):
```

```
    for j in range(0,Nz):
```

```
        O3[i,j]=noeudN3[i,j]
```

```
        OO3=str(O3[i,j].astype(int))
```

```
        f.write(OO3)
```

```
        f.write(',  ')
```

```
        #p=str(i)
```

```
        #f.write(p)
```

```
        a=str(noeudN3[i,j].astype(int))
```

```
        f.write(a)
```

```
        f.write(',  ')
```

```
        b=str(noeudN3[i,j+1].astype(int))
```

```
        f.write(b)
```

```
        f.write(',  ')
```

```
        d=str(noeudN3[i+1,j+1].astype(int))
```

```
        f.write(d)
```

```
        f.write(',  ')
```

```
        c=str(noeudN3[i+1,j].astype(int))
```

```
        f.write(c)
```

```
        f.write("\n")
```

```
        noeudXt[N1+N2+i,j]=noeudX3[i,j]
```

```
        noeudYt[N1+N2+i,j]=noeudY3[i,j]
```

```
        noeudZt[N1+N2+i,j]=noeudZ3[i,j]
```

```
        noeudNt[N1+N2+i,j]=noeudN3[i,j]
```

```
        noeudNtot[N1+N2+i,j]=noeudN3[i,j]
```

```
        noeudXtot[N1+N2+i,j]=noeudX3[i,j]
```

jonction 3_4

```
for j in range(0,Nz):
```

```
    O3[N3-1,j]=noeudN3[N3-1,j]
```

```
    OO3=str(O3[N3-1,j].astype(int))
```

```
    f.write(OO3)
```

Appendix

```
f.write(' ')
a=str(noeudN3[N3-1,j].astype(int))
f.write(a)
f.write(' ')
b=str(noeudN3[N3-1,j+1].astype(int))
f.write(b)
f.write(' ')
d=str(noeudN4[0,j+1].astype(int))
f.write(d)
f.write(' ')
c=str(noeudN4[0,j].astype(int))
f.write(c)
f.write("\n")
noeudXt[N1+N2+N3-1,j]=noeudX3[N3-1,j]
noeudYt[N1+N2+N3-1,j]=noeudY3[N3-1,j]
noeudZt[N1+N2+N3-1,j]=noeudZ3[N3-1,j]
noeudNt[N1+N2+N3-1,j]=noeudN3[N3-1,j]
noeudNtot[N1+N2+N3-1,j]=noeudN3[N3-1,j]
noeudXtot[N1+N2+N3-1,j]=noeudX3[N3-1,j]
```

```
for i in range (0,N3):
    for j in range (0,Nz):
        noeudNtot[N1+N2+i+1,j+1]=noeudN3[i+1,j+1]
        noeudXtot[N1+N2+i+1,j+1]=noeudX3[i+1,j+1]
```

#Définition et numérotation des mailles zone 4

```
for i in range (0,N4-1):
    for j in range(0,Nz):
        O4[i,j]=noeudN4[i,j]
        OO4=str(O4[i,j].astype(int))
        f.write(OO4)
        f.write(' ')
        a=str(noeudN4[i,j].astype(int))
        f.write(a)
        f.write(' ')
        b=str(noeudN4[i,j+1].astype(int))
        f.write(b)
        f.write(' ')
        d=str(noeudN4[i+1,j+1].astype(int))
        f.write(d)
        f.write(' ')
        c=str(noeudN4[i+1,j].astype(int))
        f.write(c)
        f.write("\n")
        noeudXt[N1+N2+N3+i,j]=noeudX4[i,j]
        noeudYt[N1+N2+N3+i,j]=noeudY4[i,j]
        noeudZt[N1+N2+N3+i,j]=noeudZ4[i,j]
```

Appendix

```
noeudNt[N1+N2+N3+i,j]=noeudN4[i,j]
noeudNtot[N1+N2+N3+i,j]=noeudN4[i,j]
noeudXtot[N1+N2+N3+i,j]=noeudX4[i,j]
```

jonction 4_5

```
for j in range(0,Nz):
    O4[N4-1,j]=noeudN4[N4-1,j]
    OO4=str(O4[N4-1,j].astype(int))
    f.write(OO4)
    f.write(', ')
    a=str(noeudN4[N4-1,j].astype(int))
    f.write(a)
    f.write(', ')
    b=str(noeudN4[N4-1,j+1].astype(int))
    f.write(b)
    f.write(', ')
    d=str(noeudN5[0,j+1].astype(int))
    f.write(d)
    f.write(', ')
    c=str(noeudN5[0,j].astype(int))
    f.write(c)
    f.write('\n')
    noeudXt[N1+N2+N3+N4-1,j]=noeudX4[N4-1,j]
    noeudYt[N1+N2+N3+N4-1,j]=noeudY4[N4-1,j]
    noeudZt[N1+N2+N3+N4-1,j]=noeudZ4[N4-1,j]
    noeudNt[N1+N2+N3+N4-1,j]=noeudN4[N4-1,j]
    noeudNtot[N1+N2+N3+N4-1,j]=noeudN4[N4-1,j]
    noeudXtot[N1+N2+N3+N4-1,j]=noeudX4[N4-1,j]

for i in range (0,N4):
    for j in range (0,Nz):
        noeudNtot[N1+N2+N3+i+1,j+1]=noeudN4[i+1,j+1]
        noeudXtot[N1+N2+N3+i+1,j+1]=noeudX4[i+1,j+1]
```

#Définition et numérotation des mailles zone 5

chaque maille est définie par 4 noeuds

```
for i in range (0,N5-1):
    for j in range(0,Nz):
        O5[i,j]=noeudN5[i,j]
        OO5=str(O5[i,j].astype(int))
        f.write(OO5)
        f.write(', ')
        #p=str(i)
        #f.write(p)
        a=str(noeudN5[i,j].astype(int))
        f.write(a)
```

Appendix

```
f.write(' ')
b=str(noeudN5[i,j+1].astype(int))
f.write(b)
f.write(' ')
d=str(noeudN5[i+1,j+1].astype(int))
f.write(d)
f.write(' ')
c=str(noeudN5[i+1,j].astype(int))
f.write(c)
f.write("\n")
noeudXt[N1+N2+N3+N4+i,j]=noeudX5[i,j]
noeudYt[N1+N2+N3+N4+i,j]=noeudY5[i,j]
noeudZt[N1+N2+N3+N4+i,j]=noeudZ5[i,j]
noeudNt[N1+N2+N3+N4+i,j]=noeudN5[i,j]
noeudNtot[N1+N2+N3+N4+i,j]=noeudN5[i,j]
noeudXtot[N1+N2+N3+N4+i,j]=noeudX5[i,j]
```

jonction 5_6

```
for j in range(0,Nz):
    O5[N5-1,j]=noeudN5[N5-1,j]
    OO5=str(O5[N5-1,j].astype(int))
    f.write(OO5)
    f.write(' ')
    a=str(noeudN5[N5-1,j].astype(int))
    f.write(a)
    f.write(' ')
    b=str(noeudN5[N5-1,j+1].astype(int))
    f.write(b)
    f.write(' ')
    d=str(noeudN6[0,j+1].astype(int))
    f.write(d)
    f.write(' ')
    c=str(noeudN6[0,j].astype(int))
    f.write(c)
    f.write("\n")
    noeudXt[N1+N2+N3+N4+N5-1,j]=noeudX5[N5-1,j]
    noeudYt[N1+N2+N3+N4+N5-1,j]=noeudY5[N5-1,j]
    noeudZt[N1+N2+N3+N4+N5-1,j]=noeudZ5[N5-1,j]
    noeudNt[N1+N2+N3+N4+N5-1,j]=noeudN5[N5-1,j]
    noeudNtot[N1+N2+N3+N4+N5-1,j]=noeudN5[N5-1,j]
    noeudXtot[N1+N2+N3+N4+N5-1,j]=noeudX5[N5-1,j]
for i in range (0,N5):
    for j in range (0,Nz):
        noeudNtot[N1+N2+N3+N4+i+1,j+1]=noeudN5[i+1,j+1]
        noeudXtot[N1+N2+N3+N4+i+1,j+1]=noeudX5[i+1,j+1]
```

Appendix

#Définition et numérotation des mailles zone 6

```
for i in range (0,N6-1):
    for j in range(0,Nz):
        O6[i,j]=noeudN6[i,j]
        OO6=str(O6[i,j].astype(int))
        f.write(OO6)
        f.write(' ', ' ')
        a=str(noeudN6[i,j].astype(int))
        f.write(a)
        f.write(' ', ' ')
        b=str(noeudN6[i,j+1].astype(int))
        f.write(b)
        f.write(' ', ' ')
        d=str(noeudN6[i+1,j+1].astype(int))
        f.write(d)
        f.write(' ', ' ')
        c=str(noeudN6[i+1,j].astype(int))
        f.write(c)
        f.write("\n")
        noeudXt[N1+N2+N3+N4+N5+i,j]=noeudX6[i,j]
        noeudYt[N1+N2+N3+N4+N5+i,j]=noeudY6[i,j]
        noeudZt[N1+N2+N3+N4+N5+i,j]=noeudZ6[i,j]
        noeudNt[N1+N2+N3+N4+N5+i,j]=noeudN6[i,j]
        noeudNtot[N1+N2+N3+N4+N5+i,j]=noeudN6[i,j]
        noeudXtot[N1+N2+N3+N4+N5+i,j]=noeudX6[i,j]
```

jonction 6_7

```
for j in range(0,Nz):
    O6[N6-1,j]=noeudN6[N6-1,j]
    OO6=str(O6[N6-1,j].astype(int))
    f.write(OO6)
    f.write(' ', ' ')
    a=str(noeudN6[N6-1,j].astype(int))
    f.write(a)
    f.write(' ', ' ')
    b=str(noeudN6[N6-1,j+1].astype(int))
    f.write(b)
    f.write(' ', ' ')
    d=str(noeudN7[0,j+1].astype(int))
    f.write(d)
    f.write(' ', ' ')
    c=str(noeudN7[0,j].astype(int))
    f.write(c)
    f.write("\n")
    noeudXt[N1+N2+N3+N4+N5+N6-1,j]=noeudX6[N6-1,j]
    noeudYt[N1+N2+N3+N4+N5+N6-1,j]=noeudY6[N6-1,j]
    noeudZt[N1+N2+N3+N4+N5+N6-1,j]=noeudZ6[N6-1,j]
```

Appendix

```
noeudNt[N1+N2+N3+N4+N5+N6-1,j]=noeudN6[N6-1,j]
noeudNtot[N1+N2+N3+N4+N5+N6-1,j]=noeudN6[N6-1,j]
noeudXtot[N1+N2+N3+N4+N5+N6-1,j]=noeudX6[N6-1,j]
```

```
for i in range (0,N6):
    for j in range (0,Nz):
        noeudNtot[N1+N2+N3+N4+N5+i+1,j+1]=noeudN6[i+1,j+1]
        noeudXtot[N1+N2+N3+N4+N5+i+1,j+1]=noeudX6[i+1,j+1]
```

#Définition et numérotation des mailles zone 7

```
for i in range (0,N7-1):
    for j in range(0,Nz):
        O7[i,j]=noeudN7[i,j]
        OO7=str(O7[i,j].astype(int))
        f.write(OO7)
        f.write(', ')
        a=str(noeudN7[i,j].astype(int))
        f.write(a)
        f.write(', ')
        b=str(noeudN7[i,j+1].astype(int))
        f.write(b)
        f.write(', ')
        d=str(noeudN7[i+1,j+1].astype(int))
        f.write(d)
        f.write(', ')
        c=str(noeudN7[i+1,j].astype(int))
        f.write(c)
        f.write('\n')
        noeudXt[N1+N2+N3+N4+N5+N6+i,j]=noeudX7[i,j]
        noeudYt[N1+N2+N3+N4+N5+N6+i,j]=noeudY7[i,j]
        noeudZt[N1+N2+N3+N4+N5+N6+i,j]=noeudZ7[i,j]
        noeudNt[N1+N2+N3+N4+N5+N6+i,j]=noeudN7[i,j]
        noeudNtot[N1+N2+N3+N4+N5+N6+i,j]=noeudN7[i,j]
        noeudXtot[N1+N2+N3+N4+N5+N6+i,j]=noeudX7[i,j]
```

jonction 7_8

```
for j in range(0,Nz):
    O7[N7-1,j]=noeudN7[N7-1,j]
    OO7=str(O7[N7-1,j].astype(int))
    f.write(OO7)
    f.write(', ')
    a=str(noeudN7[N7-1,j].astype(int))
    f.write(a)
    f.write(', ')
    b=str(noeudN7[N7-1,j+1].astype(int))
    f.write(b)
```

Appendix

```
f.write(' ')
d=str(noeudN8[0,j+1].astype(int))
f.write(d)
f.write(' ')
c=str(noeudN8[0,j].astype(int))
f.write(c)
f.write('\n')
noeudXt[N1+N2+N3+N4+N5+N6+N7-1,j]=noeudX7[N7-1,j]
noeudYt[N1+N2+N3+N4+N5+N6+N7-1,j]=noeudY7[N7-1,j]
noeudZt[N1+N2+N3+N4+N5+N6+N7-1,j]=noeudZ7[N7-1,j]
noeudNt[N1+N2+N3+N4+N5+N6+N7-1,j]=noeudN7[N7-1,j]
noeudNtot[N1+N2+N3+N4+N5+N6+N7-1,j]=noeudN7[N7-1,j]
noeudXtot[N1+N2+N3+N4+N5+N6+N7-1,j]=noeudX7[N7-1,j]
```

```
for i in range (0,N7):
    for j in range (0,Nz):
        noeudNtot[N1+N2+N3+N4+N5+N6+i+1,j+1]=noeudN7[i+1,j+1]
        noeudXtot[N1+N2+N3+N4+N5+N6+i+1,j+1]=noeudX7[i+1,j+1]
```

#Définition et numérotation des mailles zone 8

```
for i in range (0,N8-1):
    for j in range(0,Nz):
        O8[i,j]=noeudN8[i,j]
        OO8=str(O8[i,j].astype(int))
        f.write(OO8)
        f.write(' ')
        a=str(noeudN8[i,j].astype(int))
        f.write(a)
        f.write(' ')
        b=str(noeudN8[i+1,j].astype(int))
        f.write(b)
        f.write(' ')
        d=str(noeudN8[i+1,j+1].astype(int))
        f.write(d)
        f.write(' ')
        c=str(noeudN8[i,j+1].astype(int))
        f.write(c)
        f.write('\n')
        noeudXt[N1+N2+N3+N4+N5+N6+N7+i,j]=noeudX8[i,j]
        noeudYt[N1+N2+N3+N4+N5+N6+N7+i,j]=noeudY8[i,j]
        noeudZt[N1+N2+N3+N4+N5+N6+N7+i,j]=noeudZ8[i,j]
        noeudNt[N1+N2+N3+N4+N5+N6+N7+i,j]=noeudN8[i,j]
        noeudNtot[N1+N2+N3+N4+N5+N6+N7+i,j]=noeudN8[i,j]
        noeudXtot[N1+N2+N3+N4+N5+N6+N7+i,j]=noeudX8[i,j]
```

Appendix

jonction 8_9

```
for j in range(0,Nz):
    O8[N8-1,j]=noeudN8[N8-1,j]
    OO8=str(O8[N8-1,j].astype(int))
    f.write(OO8)
    f.write(', ')
    a=str(noeudN8[N8-1,j].astype(int))
    f.write(a)
    f.write(', ')
    b=str(noeudN8[N8-1,j+1].astype(int))
    f.write(b)
    f.write(', ')
    d=str(noeudN9[0,j+1].astype(int))
    f.write(d)
    f.write(', ')
    c=str(noeudN9[0,j].astype(int))
    f.write(c)
    f.write('\n')
    noeudXt[N1+N2+N3+N4+N5+N6+N7+N8-1,j]=noeudX8[N8-1,j]
    noeudYt[N1+N2+N3+N4+N5+N6+N7+N8-1,j]=noeudY8[N8-1,j]
    noeudZt[N1+N2+N3+N4+N5+N6+N7+N8-1,j]=noeudZ8[N8-1,j]
    noeudNt[N1+N2+N3+N4+N5+N6+N7+N8-1,j]=noeudN8[N8-1,j]
    noeudNtot[N1+N2+N3+N4+N5+N6+N7+N8-1,j]=noeudN8[N8-1,j]
    noeudXtot[N1+N2+N3+N4+N5+N6+N7+N8-1,j]=noeudX8[N8-1,j]

for i in range (0,N8):
    for j in range (0,Nz):
        noeudNtot[N1+N2+N3+N4+N5+N6+N7+i+1,j+1]=noeudN8[i+1,j+1]
        noeudXtot[N1+N2+N3+N4+N5+N6+N7+i+1,j+1]=noeudX8[i+1,j+1]
```

#Définition et numérotation des mailles zone 9

```
for i in range (0,N9):
    for j in range(0,Nz):
        O9[i,j]=noeudN9[i,j]
        OO9=str(O9[i,j].astype(int))
        f.write(OO9)
        f.write(', ')
        a=str(noeudN9[i,j].astype(int))
        f.write(a)
        f.write(', ')
        b=str(noeudN9[i+1,j].astype(int))
        f.write(b)
        f.write(', ')
        d=str(noeudN9[i+1,j+1].astype(int))
        f.write(d)
        f.write(', ')
        c=str(noeudN9[i,j+1].astype(int))
```

Appendix

```
f.write(c)
f.write('\n')
noeudXt[N1+N2+N3+N4+N5+N6+N7+N8+i,j]=noeudX9[i,j]
noeudYt[N1+N2+N3+N4+N5+N6+N7+N8+i,j]=noeudY9[i,j]
noeudZt[N1+N2+N3+N4+N5+N6+N7+N8+i,j]=noeudZ9[i,j]
noeudNt[N1+N2+N3+N4+N5+N6+N7+N8+i,j]=noeudN9[i,j]
noeudNtot[N1+N2+N3+N4+N5+N6+N7+N8+i,j]=noeudN9[i,j]
noeudXtot[N1+N2+N3+N4+N5+N6+N7+N8+i,j]=noeudX9[i,j]

for j in range(0,Nz+1):
    noeudXt[Nx-1,j-1]=noeudX9[N9-1,j-1]
    noeudYt[Nx-1,j-1]=noeudY9[N9-1,j-1]
    noeudZt[Nx-1,j-1]=noeudZ9[N9-1,j-1]
for j in range(0,Nz+2):
    noeudNtot[Nx-1,j-1]=noeudN9[N9-1,j-1]
    noeudXtot[Nx-1,j-1]=noeudX9[N9-1,j-1]
for i in range (-1,N9):
    for j in range (-1,Nz):
        noeudNtot[N1+N2+N3+N4+N5+N6+N7+N8+i+1,j+1]=noeudN9[i+1,j+1]
        noeudXtot[N1+N2+N3+N4+N5+N6+N7+N8+i+1,j+1]=noeudX9[i+1,j+1]
f.write('\n')

#
#matrice des coordonnées des noeuds au centre des mailles
#
noeudXt_cent= np.zeros((Nx,Nz))

for i in range (Nx):
    for j in range (Nz):
        noeudXt_cent [i,j]= (noeudXtot[i+1,j+1]+noeudXt[i,j])/2
#
# définition de matériau composite UD
#
#Vv1=uniform(0.002,0.001)
E1UD=np.zeros((Nx,Nz))
E2UD=np.zeros((Nx,Nz))
E3UD=np.zeros((Nx,Nz))
G12UD=np.zeros((Nx,Nz))
G13UD=np.zeros((Nx,Nz))
G23UD=np.zeros((Nx,Nz))
E1UDs=131650
E2UDs=6550
E3UDs=6550
G12UDs=3260
G13UDs=3260
G23UDs=2280
```

Appendix

#zone 1

```
for i in range (0,N1+N2+1):
    for j in range (0,Nz+1):
        Vv1=gauss(0.0005,0.0001)    #with void variability
        #Vv1=0                      #without void variability
        E1UD[i-1,j-1]=round(E1UDs*(1-Vv1)**2,3)
        E2UD[i-1,j-1]=round(E2UDs*(1-Vv1)**2,3)
        E3UD[i-1,j-1]=round(E3UDs*(1-Vv1)**2,3)
        G12UD[i-1,j-1]=round(G12UDs*(1-Vv1)**2,3)
        G13UD[i-1,j-1]=round(G13UDs*(1-Vv1)**2,3)
        G23UD[i-1,j-1]=round(G23UDs*(1-Vv1)**2,3)
```

#zone 2

```
for i in range (N1+N2,N1+N2+N3+N4+1):
    for j in range (0,Nz+1):
        Vv2=gauss(0.0004,0.0001)    #with void variability
        #Vv1=0                      #without void variability
        E1UD[i-1,j-1]=round(E1UDs*(1-Vv2)**2,3)
        E2UD[i-1,j-1]=round(E2UDs*(1-Vv2)**2,3)
        E3UD[i-1,j-1]=round(E3UDs*(1-Vv2)**2,3)
        G12UD[i-1,j-1]=round(G12UDs*(1-Vv2)**2,3)
        G13UD[i-1,j-1]=round(G13UDs*(1-Vv2)**2,3)
        G23UD[i-1,j-1]=round(G23UDs*(1-Vv2)**2,3)
```

#zone 3

```
for i in range (N1+N2+N3+N4,N1+N2+N3+N4+N5+1):
    for j in range (0,Nz+1):
        Vv3=gauss(0.005,0.001)    #with void variability
        #Vv1=0                      #without void variability
        E1UD[i-1,j-1]=round(E1UDs*(1-Vv3)**2,3)
        E2UD[i-1,j-1]=round(E2UDs*(1-Vv3)**2,3)
        E3UD[i-1,j-1]=round(E3UDs*(1-Vv3)**2,3)
        G12UD[i-1,j-1]=round(G12UDs*(1-Vv3)**2,3)
        G13UD[i-1,j-1]=round(G13UDs*(1-Vv3)**2,3)
        G23UD[i-1,j-1]=round(G23UDs*(1-Vv3)**2,3)
```

#zone 4

```
for i in range (N1+N2+N3+N4+N5,N1+N2+N3+N4+N5+N6+N7+1):
    for j in range (0,Nz+1):
        Vv4=gauss(0.0003,0.0001)    #with void variability
        #Vv1=0                      #without void variability
        E1UD[i-1,j-1]=round(E1UDs*(1-Vv4)**2,3)
        E2UD[i-1,j-1]=round(E2UDs*(1-Vv4)**2,3)
        E3UD[i-1,j-1]=round(E3UDs*(1-Vv4)**2,3)
        G12UD[i-1,j-1]=round(G12UDs*(1-Vv4)**2,3)
        G13UD[i-1,j-1]=round(G13UDs*(1-Vv4)**2,3)
        G23UD[i-1,j-1]=round(G23UDs*(1-Vv4)**2,3)
```

#zone 5

```
for i in range (N1+N2+N3+N4+N5+N6+N7,N1+N2+N3+N4+N5+N6+N7+N8+N9+1):
    for j in range (0,Nz+1):
```

Appendix

```
Vv5=gauss(0.0005,0.0001)    #with void variability
#Vv1=0                      #without void variability
E1UD[i-1,j-1]=round(E1UDs*(1-Vv5)**2,3)
E2UD[i-1,j-1]=round(E2UDs*(1-Vv5)**2,3)
E3UD[i-1,j-1]=round(E3UDs*(1-Vv5)**2,3)
G12UD[i-1,j-1]=round(G12UDs*(1-Vv5)**2,3)
G13UD[i-1,j-1]=round(G13UDs*(1-Vv5)**2,3)
G23UD[i-1,j-1]=round(G23UDs*(1-Vv5)**2,3)

from random import randint

pos_void=[[randint(round(Nx/7),round(Nx/5)),randint(round(Nx/5)+1,round(Nx/3)),randint(round(Nx/3),round(Nx/2))+ro
und(Nx/6),randint(round(Nx/2)+round(Nx/4),round(Nx))-1]]
Vv1l=gauss(0.001,0.0001)
for j in range (0,Nz+1):
    E1UD[pos_void,j-1]=round(E1UDs*(1-Vv1l)**2,3) #Module élastique avec porosités
    E2UD[pos_void,j-1]=round(E2UDs*(1-Vv1l)**2,3) #Module élastique avec porosités
    E3UD[pos_void,j-1]=round(E3UDs*(1-Vv1l)**2,3) #Module élastique avec porosités
    G12UD[pos_void,j-1]=round(G12UDs*(1-Vv1l)**2,3) #Module de cisaillement avec porosités
    G13UD[pos_void,j-1]=round(G13UDs*(1-Vv1l)**2,3) #Module de cisaillement avec porosités
    G23UD[pos_void,j-1]=round(G23UDs*(1-Vv1l)**2,3) #Module de cisaillement avec porosités

Nu12UD=str(0.27)
Nu13UD=str(0.27)
Nu23UD=str(0.14)
#coefficients d'expansion composite UD
alpha_11UD=str(-0.0000056)
alpha_22UD=str(0.00001434)
alpha_33UD=str(0.00001434)

Ten1UD=str(1814.) #Tensile stress limit in fiber direction Xt
Com1UD=str(-1325.) #Compressive stress limit in fiber direction Xc
Ten2UD=str(70.) #Tensile stress limit in transverse direction Yt
Com2UD=str(-300.) #Compressive stress limit in transverse direction Yc
SUD=str(70.) #Shear strength in the X-Y plane
CrossUD=str(1.) #cross product term coefficient (-1<f<1): used only for Tsai-Wu theory and ignored if sigma bia x is given
(default=0)
Stress_L=str(1200.) #Biaxial stress limit sigma bia x , this value is used for Tsai-Wu theory, if this entry is non zero, f is
ignored

# _____
# définition de matériau composite TISSE
# _____

#paramètres densité
density_nom_TISSE=str(1482E-12) #densité nominale composite TISSE (T/mm3)
```

Appendix

#

#paramètres porosités avec variabilités

#Vv1=uniform(0.002,0.002) #fraction volumique des porosités

Programmmations des porosités longues/grandes

E1WOVEN=np.zeros((Nx,Nz))

E2WOVEN=np.zeros((Nx,Nz))

E3WOVEN=np.zeros((Nx,Nz))

G12WOVEN=np.zeros((Nx,Nz))

G13WOVEN=np.zeros((Nx,Nz))

G23WOVEN=np.zeros((Nx,Nz))

E1WOVENs=61225 #Module élastique sans porosités

E2WOVENs=61225 #Module élastique sans porosités

E3WOVENs=11249.5 #Module élastique sans porosités

G12WOVENs=2400 #Module de cisaillement sans porosités

G13WOVENs=2475 #Module de cisaillement sans porosités

G23WOVENs=2475 #Module de cisaillement sans porosités

#zone1

for i in range (0,N1+N2+1):

for j in range (0,Nz+1):

Vv1=gauss(0.0005,0.0001) #with void variability

#Vv1=0 #without void variability

E1WOVEN[i-1,j-1]=round(E1WOVENs*(1-Vv1)**2,3) #Module élastique avec porosités

E2WOVEN[i-1,j-1]=round(E2WOVENs*(1-Vv1)**2,3) #Module élastique avec porosités

E3WOVEN[i-1,j-1]=round(E3WOVENs*(1-Vv1)**2,3) #Module élastique avec porosités

G12WOVEN[i-1,j-1]=round(G12WOVENs*(1-Vv1)**2,3) #Module de cisaillement avec porosités

G13WOVEN[i-1,j-1]=round(G13WOVENs*(1-Vv1)**2,3) #Module de cisaillement avec porosités

G23WOVEN[i-1,j-1]=round(G23WOVENs*(1-Vv1)**2,3) #Module de cisaillement avec porosités

#zone2

for i in range (N1+N2,N1+N2+N3+N4+1):

for j in range (0,Nz+1):

Vv2=gauss(0.0004,0.0001) #with void variability

#Vv1=0 #without void variability

E1WOVEN[i-1,j-1]=round(E1WOVENs*(1-Vv2)**2,3) #Module élastique avec porosités

E2WOVEN[i-1,j-1]=round(E2WOVENs*(1-Vv2)**2,3) #Module élastique avec porosités

E3WOVEN[i-1,j-1]=round(E3WOVENs*(1-Vv2)**2,3) #Module élastique avec porosités

G12WOVEN[i-1,j-1]=round(G12WOVENs*(1-Vv2)**2,3) #Module de cisaillement avec porosités

G13WOVEN[i-1,j-1]=round(G13WOVENs*(1-Vv2)**2,3) #Module de cisaillement avec porosités

G23WOVEN[i-1,j-1]=round(G23WOVENs*(1-Vv2)**2,3) #Module de cisaillement avec porosités

#zone3

Appendix

```
for i in range (N1+N2+N3+N4,N1+N2+N3+N4+N5+1):
    for j in range (0,Nz+1):
        Vv3=gauss(0.005,0.001)    #with void variability
        #Vv1=0                    #without void variability
        E1WOVEN[i-1,j-1]=round(E1WOVENs*(1-Vv3)**2,3) #Module élastique avec porosités
        E2WOVEN[i-1,j-1]=round(E2WOVENs*(1-Vv3)**2,3) #Module élastique avec porosités
        E3WOVEN[i-1,j-1]=round(E3WOVENs*(1-Vv3)**2,3) #Module élastique avec porosités
        G12WOVEN[i-1,j-1]=round(G12WOVENs*(1-Vv3)**2,3) #Module de cisaillement avec porosités
        G13WOVEN[i-1,j-1]=round(G13WOVENs*(1-Vv3)**2,3) #Module de cisaillement avec porosités
        G23WOVEN[i-1,j-1]=round(G23WOVENs*(1-Vv3)**2,3) #Module de cisaillement avec porosités

#zone4
for i in range (N1+N2+N3+N4+N5,N1+N2+N3+N4+N5+N6+N7+1):
    for j in range (0,Nz+1):
        Vv4=gauss(0.0003,0.0001)    #with void variability
        #Vv1=0                    #without void variability
        E1WOVEN[i-1,j-1]=round(E1WOVENs*(1-Vv4)**2,3) #Module élastique avec porosités
        E2WOVEN[i-1,j-1]=round(E2WOVENs*(1-Vv4)**2,3) #Module élastique avec porosités
        E3WOVEN[i-1,j-1]=round(E3WOVENs*(1-Vv4)**2,3) #Module élastique avec porosités
        G12WOVEN[i-1,j-1]=round(G12WOVENs*(1-Vv4)**2,3) #Module de cisaillement avec porosités
        G13WOVEN[i-1,j-1]=round(G13WOVENs*(1-Vv4)**2,3) #Module de cisaillement avec porosités
        G23WOVEN[i-1,j-1]=round(G23WOVENs*(1-Vv4)**2,3) #Module de cisaillement avec porosités

#zone5
for i in range (N1+N2+N3+N4+N5+N6+N7,N1+N2+N3+N4+N5+N6+N7+N8+N9+1):
    for j in range (0,Nz+1):
        Vv5=gauss(0.0005,0.0001)    #with void variability
        #Vv1=0                    #without void variability
        E1WOVEN[i-1,j-1]=round(E1WOVENs*(1-Vv5)**2,3) #Module élastique avec porosités
        E2WOVEN[i-1,j-1]=round(E2WOVENs*(1-Vv5)**2,3) #Module élastique avec porosités
        E3WOVEN[i-1,j-1]=round(E3WOVENs*(1-Vv5)**2,3) #Module élastique avec porosités
        G12WOVEN[i-1,j-1]=round(G12WOVENs*(1-Vv5)**2,3) #Module de cisaillement avec porosités
        G13WOVEN[i-1,j-1]=round(G13WOVENs*(1-Vv5)**2,3) #Module de cisaillement avec porosités
        G23WOVEN[i-1,j-1]=round(G23WOVENs*(1-Vv5)**2,3) #Module de cisaillement avec porosités

    for j in range (0,Nz+1):
        E1WOVEN[pos_void,j-1]=round(E1WOVENs*(1-Vv11)**2,3) #Module élastique avec porosités
        E2WOVEN[pos_void,j-1]=round(E2WOVENs*(1-Vv11)**2,3) #Module élastique avec porosités
        E3WOVEN[pos_void,j-1]=round(E3WOVENs*(1-Vv11)**2,3) #Module élastique avec porosités
        G12WOVEN[pos_void,j-1]=round(G12WOVENs*(1-Vv11)**2,3) #Module de cisaillement avec porosités
        G13WOVEN[pos_void,j-1]=round(G13WOVENs*(1-Vv11)**2,3) #Module de cisaillement avec porosités
        G23WOVEN[pos_void,j-1]=round(G23WOVENs*(1-Vv11)**2,3) #Module de cisaillement avec porosités
    Nu12WOVEN=str(0.22)
    Nu13WOVEN=str(0.2)
    Nu23WOVEN=str(0.2)

#coefficients d'expansion composite tisse
```

Appendix

```
alpha_11WOVEN=str(-0.0000115)
alpha_22WOVEN=str(-0.0000115)
alpha_33WOVEN=str(0.00001434)
```

```
Ten1WOVEN=str(1600.) #Tensile stress limit in fiber direction Xt
Com1WOVEN=str(-1168.) #Compressive stress limit in fiber direction Xc
Ten2WOVEN=str(70.) #Tensile stress limit in transverse direction Yt
Com2WOVEN=str(-300.) #Compressive stress limit in transverse direction Yc
SWOVEN=str(70.) #Shear strength in the X-Y plane
CrossWOVEN=str(1.)#cross product term coefficient (-1<f<1): used only for Tsai-Wu theory
```

#materiau par pli

Material

```
=(['COMPOSITEWOVEN','COMPOSITEUD','COMPOSITEWOVEN','COMPOSITEUD','COMPOSITEWOVEN','COMPOSITEUD',
'COMPOSITEWOVEN','COMPOSITEUD','COMPOSITEWOVEN','COMPOSITEUD','COMPOSITEUD','COMPOSITEWOVEN',
'COMPOSITEUD','COMPOSITEWOVEN','COMPOSITEUD','COMPOSITEWOVEN','COMPOSITEUD','COMPOSITEWOVEN',
'COMPOSITEUD','COMPOSITEWOVEN'])
```

#

TIRAGE ALEATOIRE DES EXTREMITES DE PLIS

#

#initialisation des matrices utiles pour le tirage

```
Coeff_moyenne_categorie1=numpy.zeros((1,6))
Coeff_moyenne_categorie2=numpy.zeros((1,6))
Coeff_moyenne_categorie3=numpy.zeros((1,6))
Coeff_moyenne_categorie4=numpy.zeros((1,6))
```

```
Coeff_moyenne_categorieUD=numpy.zeros((1,6))
Coeff_moyenne_categorieTISSE=numpy.zeros((1,6))
```

```
Coeff_ecarttype_categorie1=numpy.zeros((1,6))
Coeff_ecarttype_categorie2=numpy.zeros((1,6))
Coeff_ecarttype_categorie3=numpy.zeros((1,6))
Coeff_ecarttype_categorie4=numpy.zeros((1,6))
```

```
Coeff_ecarttype_categorieUD=numpy.zeros((1,6))
Coeff_ecarttype_categorieTISSE=numpy.zeros((1,6))
coefficient_mat_plis_categorie1=numpy.zeros((6,6)) # plis 1 3 9 12 18 20
coefficient_mat_plis_categorie2=numpy.zeros((2,6)) # plis 2 19 semelle
coefficient_mat_plis_categorie3=numpy.zeros((10,6)) # plis 2 4 6 8 10 11 13 15 17 19 aile
coefficient_mat_plis_categorie4=numpy.zeros((4,6)) # plis 5 7 14 16
coefficient_mat_plis_categorieUD=numpy.zeros((10,6)) # plis UD
coefficient_mat_plis_categorieTISSE=numpy.zeros((10,6)) # plis tissés
```

paramètres issus de l'analyse ecart après polym

```
Coeff_moyenne_categorie1=([moyenne_coeff_cat1[5],moyenne_coeff_cat1[4],moyenne_coeff_cat1[3],moyenne_coeff_cat1[2],moyenne_coeff_cat1[1],moyenne_coeff_cat1[0]]) #paramètres a5 a4 a3 a2 a1 a0
```

Appendix

Coeff_moyenne_categorie2=(moyenne_coeff_cat2[5],moyenne_coeff_cat2[4],moyenne_coeff_cat2[3],moyenne_coeff_cat2[2],moyenne_coeff_cat2[1],moyenne_coeff_cat2[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_moyenne_categorie3=(moyenne_coeff_cat3[5],moyenne_coeff_cat3[4],moyenne_coeff_cat3[3],moyenne_coeff_cat3[2],moyenne_coeff_cat3[1],moyenne_coeff_cat3[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_moyenne_categorie4=(moyenne_coeff_cat4[5],moyenne_coeff_cat4[4],moyenne_coeff_cat4[3],moyenne_coeff_cat4[2],moyenne_coeff_cat4[1],moyenne_coeff_cat4[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_ecarttype_categorie1=(ecartype_coeff_cat1[5],ecartype_coeff_cat1[4],ecartype_coeff_cat1[3],ecartype_coeff_cat1[2],ecartype_coeff_cat1[1],ecartype_coeff_cat1[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_ecarttype_categorie2=(ecartype_coeff_cat2[5],ecartype_coeff_cat2[4],ecartype_coeff_cat2[3],ecartype_coeff_cat2[2],ecartype_coeff_cat2[1],ecartype_coeff_cat2[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_ecarttype_categorie3=(ecartype_coeff_cat3[5],ecartype_coeff_cat3[4],ecartype_coeff_cat3[3],ecartype_coeff_cat3[2],ecartype_coeff_cat3[1],ecartype_coeff_cat3[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_ecarttype_categorie4=(ecartype_coeff_cat4[5],ecartype_coeff_cat4[4],ecartype_coeff_cat4[3],ecartype_coeff_cat4[2],ecartype_coeff_cat4[1],ecartype_coeff_cat4[0]) **#paramètres a5 a4 a3 a2 a1 a0**

paramètres issus de l'analyse ecart de découpe

Coeff_moyenne_categorieUD=(moyenne_coeff_catUD[5],moyenne_coeff_catUD[4],moyenne_coeff_catUD[3],moyenne_coeff_catUD[2],moyenne_coeff_catUD[1],moyenne_coeff_catUD[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_moyenne_categorieTISSE=(moyenne_coeff_catTISSE[5],moyenne_coeff_catTISSE[4],moyenne_coeff_catTISSE[3],moyenne_coeff_catTISSE[2],moyenne_coeff_catTISSE[1],moyenne_coeff_catTISSE[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_ecarttype_categorieUD=(ecartype_coeff_catUD[5],ecartype_coeff_catUD[4],ecartype_coeff_catUD[3],ecartype_coeff_catUD[2],ecartype_coeff_catUD[1],ecartype_coeff_catUD[0]) **#paramètres a5 a4 a3 a2 a1 a0**

Coeff_ecarttype_categorieTISSE=(ecartype_coeff_catTISSE[5],ecartype_coeff_catTISSE[4],ecartype_coeff_catTISSE[3],ecartype_coeff_catTISSE[2],ecartype_coeff_catTISSE[1],ecartype_coeff_catTISSE[0]) **#paramètres a5 a4 a3 a2 a1 a0**

#

paramètres issus de l'analyse des écarts des arrêts des plis

#

for i in range (0,5):

 for j in range (0,3):

 coefficient_mat_plis_categorie1[i,j]=gauss(Coeff_moyenne_categorie1[j],Coeff_ecarttype_categorie1[j])

for i in range (0,1):

 for j in range (0,3):

 coefficient_mat_plis_categorie2[i,j]=gauss(Coeff_moyenne_categorie2[j],Coeff_ecarttype_categorie2[j])

for i in range (0,9):

 for j in range (0,3):

 coefficient_mat_plis_categorie3[i,j]=gauss(Coeff_moyenne_categorie3[j],Coeff_ecarttype_categorie3[j])

for i in range (0,3):

 for j in range (0,3):

 coefficient_mat_plis_categorie4[i,j]=gauss(Coeff_moyenne_categorie4[j],Coeff_ecarttype_categorie4[j])

#

Appendix

paramètres issus de l'analyse ecart de découpe

#

for i in range (0,9):

 for j in range (0,3):

 coefficient_mat_plis_categorieUD[i,j]=gauss(Coeff_moyenne_categorieUD[j],Coeff_ecarttype_categorieUD[j])

for i in range (0,9):

 for j in range (0,3):

 coefficient_mat_plis_categorieTISSE[i,j]=gauss(Coeff_moyenne_categorieTISSE[j],Coeff_ecarttype_categorieTISSE[j])

#

#cas sans variabilités (coefficients =0)

#

#for i in range (0,6):

 #for j in range (0,4):

 #coefficient_mat_plis_categorie1[i,j]=0

#for i in range (0,2):

 #for j in range (0,4):

 #coefficient_mat_plis_categorie2[i,j]=0

#for i in range (0,10):

 #for j in range (0,4):

 #coefficient_mat_plis_categorie3[i,j]=0

#for i in range (0,4):

 #for j in range (0,4):

 #coefficient_mat_plis_categorie4[i,j]=0

#for i in range (0,9):

 #for j in range (0,3):

 #coefficient_mat_plis_categorieUD[i,j]=0

#for i in range (0,9):

 #for j in range (0,3):

 #coefficient_mat_plis_categorieTISSE[i,j]=0

#

Definition des positions des plis theoriques

#

initialisation des matrices position

pos_theo_mat_p1d = numpy.zeros([1,2])

pos_theo_mat_p2d = numpy.zeros([3,2])

pos_theo_mat_p3d = numpy.zeros([1,2])

pos_theo_mat_p4d = numpy.zeros([1,2])

pos_theo_mat_p5d = numpy.zeros([2,2])

pos_theo_mat_p6d = numpy.zeros([1,2])

pos_theo_mat_p7d = numpy.zeros([2,2])

pos_theo_mat_p8d = numpy.zeros([1,2])

pos_theo_mat_p9d = numpy.zeros([1,2])

pos_theo_mat_p10d = numpy.zeros([1,2])

pos_theo_mat_p11d = numpy.zeros([1,2])

pos_theo_mat_p12d = numpy.zeros([1,2])

Appendix

```
pos_theo_mat_p13d = numpy.zeros([1,2])
pos_theo_mat_p14d = numpy.zeros([2,2])
pos_theo_mat_p15d = numpy.zeros([1,2])
pos_theo_mat_p16d = numpy.zeros([2,2])
pos_theo_mat_p17d = numpy.zeros([1,2])
pos_theo_mat_p18d = numpy.zeros([1,2])
pos_theo_mat_p19d = numpy.zeros([3,2])
pos_theo_mat_p20d = numpy.zeros([1,2])
```

```
pos_theo_mat_p1g = numpy.zeros([1,Nz])
pos_theo_mat_p2g = numpy.zeros([3,Nz])
pos_theo_mat_p3g = numpy.zeros([1,Nz])
pos_theo_mat_p4g = numpy.zeros([1,Nz])
pos_theo_mat_p5g = numpy.zeros([2,Nz])
pos_theo_mat_p6g = numpy.zeros([1,Nz])
pos_theo_mat_p7g = numpy.zeros([2,Nz])
pos_theo_mat_p8g = numpy.zeros([1,Nz])
pos_theo_mat_p9g = numpy.zeros([1,Nz])
pos_theo_mat_p10g = numpy.zeros([1,Nz])
pos_theo_mat_p11g = numpy.zeros([1,Nz])
pos_theo_mat_p12g = numpy.zeros([1,Nz])
pos_theo_mat_p13g = numpy.zeros([1,Nz])
pos_theo_mat_p14g = numpy.zeros([2,Nz])
pos_theo_mat_p15g = numpy.zeros([1,Nz])
pos_theo_mat_p16g = numpy.zeros([2,Nz])
pos_theo_mat_p17g = numpy.zeros([1,Nz])
pos_theo_mat_p18g = numpy.zeros([1,Nz])
pos_theo_mat_p19g = numpy.zeros([3,Nz])
pos_theo_mat_p20g = numpy.zeros([1,Nz])
```

```
#
```

```
#matrices positions theoriques (droite)
```

```
#
```

```
pos_theo_mat_p1d = np.array ([49.79,26.55])
pos_theo_mat_p2d = np.array([[12.22 ,2.66] , [19.24, 23.12] , [33.64 ,26.64]])
pos_theo_mat_p3d = np.array([44.45,26.77])
pos_theo_mat_p4d = np.array([12.22,3.31])
pos_theo_mat_p5d = np.array([[12.22,3.31],[24.76,26.74]])
pos_theo_mat_p6d = np.array([12.13,3.76])
pos_theo_mat_p7d = np.array([[12.13,3.76],[39.98,27]])
pos_theo_mat_p8d = np.array([12.02,4.21])
pos_theo_mat_p9d = np.array([30.1,27.47])
pos_theo_mat_p10d = np.array([11.91,4.98])
pos_theo_mat_p11d = np.array([11.9,5.31])
pos_theo_mat_p12d = np.array([30.56,27.57])
pos_theo_mat_p13d = np.array([11.9,5.9])
pos_theo_mat_p14d = np.array([[12.01,6.24],[40.56,27.03]])
pos_theo_mat_p15d = np.array([12.01,6.24])
```

Appendix

```
pos_theo_mat_p16d = np.array([[12.01,6.24],[27.68,28.31]])
pos_theo_mat_p17d = np.array([11.88,6.66])
pos_theo_mat_p18d = np.array([43.71,27.05])
pos_theo_mat_p19d = np.array([[11.87,7.33],[21.9,23.35],[29.37,28.62]])
pos_theo_mat_p20d = np.array([47.6,26.81])
posZt=np.zeros([Nx,Nz])
for i in range (0,Nx):
    for j in range (0,Nz):
        posZt[i,j]=noeudZt[i,j]-(L/2)
#
#matrices positions apres ecart découpe (gauche)
#
pos_theo_mat_p1g = ([49.79,26.55])
#position gauche pli 2 (aile,s emelle)
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p2g [1,j-1] = pos_theo_mat_p2d [2,0] -
((coefficient_mat_plis_categorieUD[0,5])+(coefficient_mat_plis_categorieUD[0,4]*posZt[i-1,j-
1]))+(coefficient_mat_plis_categorieUD[0,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[0,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieUD[0,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[0,0]*(posZt[i-1,j-1]**5)) )
        pos_theo_mat_p2g [0,j-1] = - pos_theo_mat_p2d [0,0] +
(coefficient_mat_plis_categorieUD[0,5])+(coefficient_mat_plis_categorieUD[0,4]*posZt[i-1,j-
1]))+(coefficient_mat_plis_categorieUD[0,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[0,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieUD[0,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[0,0]*(posZt[i-1,j-1]**5))
        pos_theo_mat_p2g [2,j-1] = -pos_theo_mat_p2d [2,0] +
(coefficient_mat_plis_categorieUD[0,5])+(coefficient_mat_plis_categorieUD[0,4]*posZt[i-1,j-
1]))+(coefficient_mat_plis_categorieUD[0,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[0,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieUD[0,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[0,0]*(posZt[i-1,j-1]**5))

#position gauche pli 3

for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p3g [0,j-1] = - pos_theo_mat_p3d [0] +
(coefficient_mat_plis_categorieTISSE[0,5])+(coefficient_mat_plis_categorieTISSE[0,4]*posZt[i-1,j-
1]))+(coefficient_mat_plis_categorieTISSE[0,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[0,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieTISSE[0,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[0,0]*(posZt[i-1,j-1]**5))

#position gauche pli 4

for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p4g [0,j-1] = - pos_theo_mat_p4d [0] +
(coefficient_mat_plis_categorieUD[1,5])+(coefficient_mat_plis_categorieUD[1,4]*posZt[i-1,j-
1]))+(coefficient_mat_plis_categorieUD[1,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[1,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieUD[1,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[1,0]*(posZt[i-1,j-1]**5))
```

Appendix

#position gauche pli 5

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p5g [0,j-1] = pos_theo_mat_p5d [1][0] -
        ((coefficient_mat_plis_categorieTISSE[1,5])+(coefficient_mat_plis_categorieTISSE[1,4]*posZt[i-1,j-
        1])+(coefficient_mat_plis_categorieTISSE[1,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[1,2]*posZt[i-1,j-1]**3) +
        (coefficient_mat_plis_categorieTISSE[1,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[1,0]*(posZt[i-1,j-1]**5)))
        pos_theo_mat_p5g [1,j-1] = -pos_theo_mat_p5d [1][0] +
        (coefficient_mat_plis_categorieTISSE[1,5])+(coefficient_mat_plis_categorieTISSE[1,4]*posZt[i-1,j-
        1])+(coefficient_mat_plis_categorieTISSE[1,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[1,2]*posZt[i-1,j-1]**3) +
        (coefficient_mat_plis_categorieTISSE[1,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[1,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 6

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p6g [0,j-1] = - pos_theo_mat_p6d [0] +
        (coefficient_mat_plis_categorieUD[2,5])+(coefficient_mat_plis_categorieUD[2,4]*posZt[i-1,j-
        1])+(coefficient_mat_plis_categorieUD[2,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[2,2]*posZt[i-1,j-1]**3) +
        (coefficient_mat_plis_categorieUD[2,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[2,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 7

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p7g [0,j-1] = pos_theo_mat_p7d [1][0] -
        ((coefficient_mat_plis_categorieTISSE[2,5])+(coefficient_mat_plis_categorieTISSE[2,4]*posZt[i-1,j-
        1])+(coefficient_mat_plis_categorieTISSE[2,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[2,2]*posZt[i-1,j-1]**3) +
        (coefficient_mat_plis_categorieTISSE[2,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[2,0]*(posZt[i-1,j-1]**5)))
        pos_theo_mat_p7g [1,j-1] = -pos_theo_mat_p7d [1][0] +
        (coefficient_mat_plis_categorieTISSE[2,5])+(coefficient_mat_plis_categorieTISSE[2,4]*posZt[i-1,j-
        1])+(coefficient_mat_plis_categorieTISSE[2,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[2,2]*posZt[i-1,j-1]**3) +
        (coefficient_mat_plis_categorieTISSE[2,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[2,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 8

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p8g [0,j-1] = - pos_theo_mat_p8d [0] +
        (coefficient_mat_plis_categorieUD[3,5])+(coefficient_mat_plis_categorieUD[3,4]*posZt[i-1,j-
        1])+(coefficient_mat_plis_categorieUD[3,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[3,2]*posZt[i-1,j-1]**3) +
        (coefficient_mat_plis_categorieUD[3,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[3,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 9

```
for i in range (1,Nx+1):
```

Appendix

```
for j in range (1,Nz+1):
    pos_theo_mat_p9g [0,j-1] = - pos_theo_mat_p9d [0] +
(coefficient_mat_plis_categorieTISSE[3,5])+(coefficient_mat_plis_categorieTISSE[3,4]*posZt[i-1,j-
1])+(coefficient_mat_plis_categorieTISSE[3,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[3,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieTISSE[3,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[3,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 10

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p10g [0,j-1] = - pos_theo_mat_p10d [0] +
(coefficient_mat_plis_categorieUD[4,5])+(coefficient_mat_plis_categorieUD[4,4]*posZt[i-1,j-
1])+(coefficient_mat_plis_categorieUD[4,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[4,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieUD[4,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[4,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 11

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p11g [0,j-1] = - pos_theo_mat_p11d [0] +
(coefficient_mat_plis_categorieUD[5,5])+(coefficient_mat_plis_categorieUD[5,4]*posZt[i-1,j-
1])+(coefficient_mat_plis_categorieUD[5,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[5,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieUD[5,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[5,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 12

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p12g [0,j-1] = - pos_theo_mat_p12d [0] +
(coefficient_mat_plis_categorieTISSE[4,5])+(coefficient_mat_plis_categorieTISSE[4,4]*posZt[i-1,j-
1])+(coefficient_mat_plis_categorieTISSE[4,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[4,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieTISSE[4,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[4,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 13

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
        pos_theo_mat_p13g [0,j-1] = - pos_theo_mat_p13d [0] +
(coefficient_mat_plis_categorieUD[6,5])+(coefficient_mat_plis_categorieUD[6,4]*posZt[i-1,j-
1])+(coefficient_mat_plis_categorieUD[6,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[6,2]*posZt[i-1,j-1]**3) +
(coefficient_mat_plis_categorieUD[6,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[6,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 14

```
for i in range (1,Nx+1):
    for j in range (1,Nz+1):
```

Appendix

```
pos_theo_mat_p14g [0,j-1] = pos_theo_mat_p14d [1][0] -  
((coefficient_mat_plis_categorieTISSE[5,5])+(coefficient_mat_plis_categorieTISSE[5,4]*posZt[i-1,j-  
1])+(coefficient_mat_plis_categorieTISSE[5,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[5,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieTISSE[5,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[5,0]*(posZt[i-1,j-1]**5)))  
pos_theo_mat_p14g [1,j-1] = -pos_theo_mat_p14d [1][0] +  
(coefficient_mat_plis_categorieTISSE[5,5])+(coefficient_mat_plis_categorieTISSE[5,4]*posZt[i-1,j-  
1])+(coefficient_mat_plis_categorieTISSE[5,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[5,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieTISSE[5,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[5,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 15

```
for i in range (1,Nx+1):  
    for j in range (1,Nz+1):  
        pos_theo_mat_p15g [0,j-1] = - pos_theo_mat_p15d [0] +  
(coefficient_mat_plis_categorieUD[7,5])+(coefficient_mat_plis_categorieUD[7,4]*posZt[i-1,j-  
1])+(coefficient_mat_plis_categorieUD[7,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[7,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieUD[7,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[7,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 16

```
for i in range (1,Nx+1):  
    for j in range (1,Nz+1):  
        pos_theo_mat_p16g [0,j-1] = pos_theo_mat_p16d [1][0] -  
((coefficient_mat_plis_categorieTISSE[6,5])+(coefficient_mat_plis_categorieTISSE[6,4]*posZt[i-1,j-  
1])+(coefficient_mat_plis_categorieTISSE[6,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[6,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieTISSE[6,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[6,0]*(posZt[i-1,j-1]**5)))  
pos_theo_mat_p16g [1,j-1] = -pos_theo_mat_p16d [1][0] +  
(coefficient_mat_plis_categorieTISSE[6,5])+(coefficient_mat_plis_categorieTISSE[6,4]*posZt[i-1,j-  
1])+(coefficient_mat_plis_categorieTISSE[6,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[6,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieTISSE[6,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[6,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 17

```
for i in range (1,Nx+1):  
    for j in range (1,Nz+1):  
        pos_theo_mat_p17g [0,j-1] = - pos_theo_mat_p17d [0] +  
(coefficient_mat_plis_categorieUD[8,5])+(coefficient_mat_plis_categorieUD[8,4]*posZt[i-1,j-  
1])+(coefficient_mat_plis_categorieUD[8,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[8,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieUD[8,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[8,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 18

```
for i in range (1,Nx+1):  
    for j in range (1,Nz+1):  
        pos_theo_mat_p18g [0,j-1] = - pos_theo_mat_p18d [0] +  
(coefficient_mat_plis_categorieTISSE[7,5])+(coefficient_mat_plis_categorieTISSE[7,4]*posZt[i-1,j-
```

Appendix

```
1))+(coefficient_mat_plis_categorieTISSE[7,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[7,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieTISSE[7,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[7,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 19

```
for i in range (1,Nx+1):  
    for j in range (1,Nz+1):  
        pos_theo_mat_p19g [1,j-1] = pos_theo_mat_p19d [2,0] -(  
(coefficient_mat_plis_categorieUD[9,5])+(coefficient_mat_plis_categorieUD[9,4]*posZt[i-1,j-  
1))+(coefficient_mat_plis_categorieUD[9,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[9,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieUD[9,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[9,0]*(posZt[i-1,j-1]**5)))  
        pos_theo_mat_p19g [0,j-1] = - pos_theo_mat_p19d [0,0] +  
(coefficient_mat_plis_categorieUD[9,5])+(coefficient_mat_plis_categorieUD[9,4]*posZt[i-1,j-  
1))+(coefficient_mat_plis_categorieUD[9,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[9,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieUD[9,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[9,0]*(posZt[i-1,j-1]**5))  
        pos_theo_mat_p19g [2,j-1] = -pos_theo_mat_p19d [2,0] +  
(coefficient_mat_plis_categorieUD[9,5])+(coefficient_mat_plis_categorieUD[9,4]*posZt[i-1,j-  
1))+(coefficient_mat_plis_categorieUD[9,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieUD[9,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieUD[9,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieUD[9,0]*(posZt[i-1,j-1]**5))
```

#position gauche pli 20

```
for i in range (1,Nx+1):  
    for j in range (1,Nz+1):  
        pos_theo_mat_p20g [0,j-1] = - pos_theo_mat_p20d [0] +  
(coefficient_mat_plis_categorieTISSE[8,5])+(coefficient_mat_plis_categorieTISSE[8,4]*posZt[i-1,j-  
1))+(coefficient_mat_plis_categorieTISSE[8,3]*posZt[i-1,j-1]**2)+(coefficient_mat_plis_categorieTISSE[8,2]*posZt[i-1,j-1]**3) +  
(coefficient_mat_plis_categorieTISSE[8,1]*(posZt[i-1,j-1]**4)) + (coefficient_mat_plis_categorieTISSE[8,0]*(posZt[i-1,j-1]**5))
```

```
#
```

#programmation des écarts

```
#
```

```
Mat_position_p1 =np.ones([Nx,Nz])  
Mat_position_p2 =np.zeros([Nx,Nz])  
Mat_position_p3 =np.zeros([Nx,Nz])  
Mat_position_p4 =np.zeros([Nx,Nz])  
Mat_position_p5 =np.zeros([Nx,Nz])  
Mat_position_p6 =np.zeros([Nx,Nz])  
Mat_position_p7 =np.zeros([Nx,Nz])  
Mat_position_p8 =np.zeros([Nx,Nz])  
Mat_position_p9 =np.zeros([Nx,Nz])  
Mat_position_p10 =np.zeros([Nx,Nz])  
Mat_position_p11 =np.zeros([Nx,Nz])  
Mat_position_p12 =np.zeros([Nx,Nz])  
Mat_position_p13 =np.zeros([Nx,Nz])  
Mat_position_p14 =np.zeros([Nx,Nz])  
Mat_position_p15 =np.zeros([Nx,Nz])  
Mat_position_p16 =np.zeros([Nx,Nz])
```

Appendix

```
Mat_position_p17 = np.zeros([Nx,Nz])
Mat_position_p18 = np.zeros([Nx,Nz])
Mat_position_p19 = np.zeros([Nx,Nz])
Mat_position_p20 = np.zeros([Nx,Nz])
#delta_p1 = np.zeros([Nz,1])
delta_p21 = np.zeros([Nz,1])
delta_p22 = np.zeros([Nz,1])
delta_p23= np.zeros([Nz,1])
delta_p3 = np.zeros([Nz,1])
delta_p4 = np.zeros([Nz,1])
delta_p51 = np.zeros([Nz,1])
delta_p52 = np.zeros([Nz,1])
delta_p6 = np.zeros([Nz,1])
delta_p71 = np.zeros([Nz,1])
delta_p72 = np.zeros([Nz,1])
delta_p8 = np.zeros([Nz,1])
delta_p9 = np.zeros([Nz,1])
delta_p10 = np.zeros([Nz,1])
delta_p11 = np.zeros([Nz,1])
delta_p12 = np.zeros([Nz,1])
delta_p13 = np.zeros([Nz,1])
delta_p141 = np.zeros([Nz,1])
delta_p142 = np.zeros([Nz,1])
delta_p15 = np.zeros([Nz,1])
delta_p161 = np.zeros([Nz,1])
delta_p162 = np.zeros([Nz,1])
delta_p17 = np.zeros([Nz,1])
delta_p18 = np.zeros([Nz,1])
delta_p191 = np.zeros([Nz,1])
delta_p192 = np.zeros([Nz,1])
delta_p193 = np.zeros([Nz,1])
delta_p20 = np.zeros([Nz,1])
#
# extrémité pli 1
#
#for i in range (1,Nx+1):
#    #for j in range (1,Nz+1):
#        #delta_p1[j-1,0]=coefficient_mat_plis_categorie1[0,3]+(coefficient_mat_plis_categorie1[0,2]*noeudZt[i-1,j-1]) +
#        (coefficient_mat_plis_categorie1[0,1]*noeudZt[i-1,j-1]**2) + (coefficient_mat_plis_categorie1[0,0]*noeudZt[i-1,j-1]**3)
#        # if noeudXt [i-1,j-1] > (-pos_theo_mat_p1[0]-delta_p1[j-1,0]):
#        # if noeudXt [i-1,j-1] < (pos_theo_mat_p1[0]+delta_p1[j-1,0]):
#
#        # Mat_position_p1 [i-1,j-1]=1
#
#
# extrémité pli 2
#
```

Appendix

```
delta_max=0
for i in range (Nx):
    for j in range (Nz):
        delta_p21[j,0]=(coefficient_mat_plis_categorie2[0,5])+(coefficient_mat_plis_categorie2[0,4]*posZt[i,j])+(coefficient_mat_plis_categorie2[0,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie2[0,2]*posZt[i,j]**3) +
        (coefficient_mat_plis_categorie2[0,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie2[0,0]*(posZt[i,j]**5))
        if (delta_max<=delta_p21[j,0]):
            delta_max=delta_p21[j,0]
        delta_p22[j,0]=(coefficient_mat_plis_categorie3[0,5])+(coefficient_mat_plis_categorie3[0,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[0,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[0,2]*posZt[i,j]**3) +
        (coefficient_mat_plis_categorie3[0,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie3[0,0]*(posZt[i,j]**5))
        if (delta_max<=delta_p22[j,0]):
            delta_max=delta_p22[j,0]
        delta_p23[j,0]=(coefficient_mat_plis_categorie2[0,5])+(coefficient_mat_plis_categorie2[0,4]*posZt[i,j])+(coefficient_mat_plis_categorie2[0,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie2[0,2]*posZt[i,j]**3) +
        (coefficient_mat_plis_categorie2[0,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie2[0,0]*(posZt[i,j]**5))
        if (delta_max<=delta_p23[j,0]):
            delta_max=delta_p23[j,0]
        if noeudXt_cent [i,j] < (-pos_theo_mat_p2d[1,0]-delta_p21[j,0]):
            if noeudXt_cent [i,j] > (pos_theo_mat_p2g[2,j]):
                Mat_position_p2 [i,j]=1
        if noeudXt_cent [i,j] > (pos_theo_mat_p2g[0,j]):
            if noeudXt_cent [i,j] < (pos_theo_mat_p2d[0,0]+delta_p22[j,0]):

                Mat_position_p2 [i,j]=1

        if noeudXt_cent [i,j] > (pos_theo_mat_p2d[1,0]+delta_p23[j,0]):
            if noeudXt_cent [i,j] < (pos_theo_mat_p2g[1,j]):

                Mat_position_p2 [i,j]=1
#
# extrémité pli 3
#
```

```
for i in range (Nx):
    for j in range (Nz):

        delta_p3[j,0]=(coefficient_mat_plis_categorie1[1,5])+(coefficient_mat_plis_categorie1[1,4]*posZt[i,j])+(coefficient_mat_plis_categorie1[1,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie1[1,2]*posZt[i,j]**3) +
        (coefficient_mat_plis_categorie1[1,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie1[1,0]*(posZt[i,j]**5))
        if (delta_max<=delta_p3[j,0]):
            delta_max=delta_p3[j,0]
        if noeudXt_cent [i,j] > (pos_theo_mat_p3g [0,j]):
            if noeudXt_cent [i,j] < (pos_theo_mat_p3d[0]+delta_p3[j,0]):

                Mat_position_p3 [i,j]=1
```

Appendix

```
#
# extrémité pli 4
#
for i in range (Nx):
    for j in range (Nz):
delta_p4[j,0]=(coefficient_mat_plis_categorie3[1,5])+(coefficient_mat_plis_categorie3[1,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[1,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[1,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[1,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie3[1,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p4[j,0]):
        delta_max=delta_p4[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p4g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p4d[0]+delta_p4[j,0]):
            Mat_position_p4 [i,j]=1
#
# extrémité pli 5
#
for i in range (Nx):
    for j in range (Nz):

delta_p51[j,0]=(coefficient_mat_plis_categorie4[0,5])+(coefficient_mat_plis_categorie4[0,4]*posZt[i,j])+(coefficient_mat_plis_categorie4[0,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie4[0,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie4[0,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie4[0,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p51[j,0]):
        delta_max=delta_p51[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p5d[0][0]+delta_p51[j,0]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p5g[0,j]):
            Mat_position_p5 [i,j]=1
delta_p52[j,0]=(coefficient_mat_plis_categorie4[0,5])+(coefficient_mat_plis_categorie4[0,4]*posZt[i,j])+(coefficient_mat_plis_categorie4[0,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie4[0,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie4[0,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie4[0,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p52[j,0]):
        delta_max=delta_p52[j,0]
    if noeudXt_cent [i,j] < (-pos_theo_mat_p5d[0][0]-delta_p52[j,0]):
        if noeudXt_cent [i,j] > (pos_theo_mat_p5g[1,j]):
            Mat_position_p5 [i,j]=1
#
# extrémité pli 6
#
for i in range (Nx):
    for j in range (Nz):

delta_p6[j,0]=(coefficient_mat_plis_categorie3[2,5])+(coefficient_mat_plis_categorie3[2,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[2,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[2,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[2,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie3[2,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p6[j,0]):
        delta_max=delta_p6[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p6g[0,j]):
```

Appendix

```
    if noeudXt_cent [i,j] < (pos_theo_mat_p6d[0]+delta_p6[j,0]):
        Mat_position_p6 [i,j]=1
#
# extrémité pli 7
#
for i in range (Nx):
    for j in range (Nz):

delta_p71[j,0]=(coefficient_mat_plis_categorie4[1,5])+(coefficient_mat_plis_categorie4[1,4]*posZt[i,j])+(coefficient_mat_plis_categorie4[1,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie4[1,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie4[1,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie4[1,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p71[j,0]):
        delta_max=delta_p71[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p7d[0][0]+delta_p71[j,0]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p7g[0,j]):
            Mat_position_p7 [i,j]=1

delta_p72[j,0]=(coefficient_mat_plis_categorie4[1,5])+(coefficient_mat_plis_categorie4[1,4]*posZt[i,j])+(coefficient_mat_plis_categorie4[1,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie4[1,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie4[1,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie4[1,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p72[j,0]):
        delta_max=delta_p72[j,0]
    if noeudXt_cent [i,j] < (-pos_theo_mat_p7d[0][0]-delta_p72[j,0]):
        if noeudXt_cent [i,j] > (pos_theo_mat_p7g[1,j]):
            Mat_position_p7 [i,j]=1
#
# extrémité pli 8
#
for i in range (Nx):
    for j in range (Nz):

delta_p8[j,0]=(coefficient_mat_plis_categorie3[3,5])+(coefficient_mat_plis_categorie3[3,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[3,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[3,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[3,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie3[3,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p8[j,0]):
        delta_max=delta_p8[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p8g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p8d[0]+delta_p8[j,0]):
            Mat_position_p8 [i,j]=1
#
# extrémité pli 9
#
for i in range (Nx):
    for j in range (Nz):

delta_p9[j,0]=(coefficient_mat_plis_categorie1[2,5])+(coefficient_mat_plis_categorie1[2,4]*posZt[i,j])+(coefficient_mat_plis_categorie1[2,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie1[2,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie1[2,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie1[2,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p9[j,0]):
        delta_max=delta_p9[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p9g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p9d[0]+delta_p9[j,0]):
            Mat_position_p9 [i,j]=1
#
```

Appendix

```
orie1[2,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie1[2,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie1[2,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie1[2,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p9[j,0]):
        delta_max=delta_p9[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p9g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p9d[0]+delta_p9[j,0]):
            Mat_position_p9 [i,j]=1
#
# extrémité pli 10
#
for i in range (Nx):
    for j in range (Nz):

delta_p10[j,0]=(coefficient_mat_plis_categorie3[4,5])+(coefficient_mat_plis_categorie3[4,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[4,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[4,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[4,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie3[4,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p10[j,0]):
        delta_max=delta_p10[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p10g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p10d[0]+delta_p10[j,0]):
            Mat_position_p10 [i,j]=1
#
# extrémité pli 11
#
for i in range (Nx):
    for j in range (Nz):
delta_p11[j,0]=(coefficient_mat_plis_categorie3[5,5])+(coefficient_mat_plis_categorie3[5,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[5,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[5,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[5,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie3[5,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p11[j,0]):
        delta_max=delta_p11[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p11g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p11d[0]+delta_p11[j,0]):
            Mat_position_p11 [i,j]=1
#
# extrémité pli 12
#
for i in range (Nx):
    for j in range (Nz):

delta_p12[j,0]=(coefficient_mat_plis_categorie1[3,5])+(coefficient_mat_plis_categorie1[3,4]*posZt[i,j])+(coefficient_mat_plis_categorie1[3,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie1[3,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie1[3,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie1[3,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p12[j,0]):
        delta_max=delta_p12[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p12g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p12d[0]+delta_p12[j,0]):
```

Appendix

```
Mat_position_p12 [i,j]=1
#
# extrémité pli 13
#
for i in range (Nx):
    for j in range (Nz):
delta_p13[j,0]=(coefficient_mat_plis_categorie3[6,5])+(coefficient_mat_plis_categorie3[6,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[6,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[6,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[6,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie3[6,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p13[j,0]):
        delta_max=delta_p13[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p13g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p13d[0]+delta_p13[j,0]):
            Mat_position_p13 [i,j]=1
#
# extrémité pli 14
#
for i in range (Nx):
    for j in range (Nz):

delta_p141[j,0]=(coefficient_mat_plis_categorie4[2,5])+(coefficient_mat_plis_categorie4[2,4]*posZt[i,j])+(coefficient_mat_plis_categorie4[2,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie4[2,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie4[2,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie4[2,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p141[j,0]):
        delta_max=delta_p141[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p14d[0][0]+delta_p141[j,0]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p14g[0,j]):
            Mat_position_p14 [i,j]=1

delta_p142[j,0]=(coefficient_mat_plis_categorie4[2,5])+(coefficient_mat_plis_categorie4[2,4]*posZt[i,j])+(coefficient_mat_plis_categorie4[2,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie4[2,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie4[2,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie4[2,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p142[j,0]):
        delta_max=delta_p142[j,0]
    if noeudXt_cent [i,j] < (-pos_theo_mat_p14d[0][0]-delta_p142[j,0]):
        if noeudXt_cent [i,j] > (pos_theo_mat_p14g[1,j]):
            Mat_position_p14 [i,j]=1
#
# extrémité pli 15
#
for i in range (Nx):
    for j in range (Nz):

delta_p15[j,0]=(coefficient_mat_plis_categorie3[7,5])+(coefficient_mat_plis_categorie3[7,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[7,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[7,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[7,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie3[7,0]*(posZt[i,j]**5))
```

Appendix

```
    if (delta_max<=delta_p15[j,0]):
        delta_max=delta_p15[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p15g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p15d[0]+delta_p15[j,0]):
            Mat_position_p15 [i,j]=1
#
# extrémité pli 16
#
for i in range (Nx):
    for j in range (Nz):

delta_p161[j,0]=(coefficient_mat_plis_categorie4[3,5])+(coefficient_mat_plis_categorie4[3,4]*posZt[i,j])+(coefficient_mat_plis_cat
egorie4[3,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie4[3,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie4[3,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie4[3,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p161[j,0]):
        delta_max=delta_p161[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p16d[0][0]+delta_p161[j,0]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p16g[0,j]):
            Mat_position_p16 [i,j]=1

delta_p162[j,0]=(coefficient_mat_plis_categorie4[3,5])+(coefficient_mat_plis_categorie4[3,4]*posZt[i,j])+(coefficient_mat_plis_cat
egorie4[3,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie4[3,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie4[3,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie4[3,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p162[j,0]):
        delta_max=delta_p162[j,0]
    if noeudXt_cent [i,j] < (-pos_theo_mat_p16d[0][0]-delta_p162[j,0]):
        if noeudXt_cent [i,j] > (pos_theo_mat_p16g[1,j]):
            Mat_position_p16 [i,j]=1

#
# extrémité pli 17
#
for i in range (Nx):
    for j in range (Nz):

delta_p17[j,0]=(coefficient_mat_plis_categorie3[8,5])+(coefficient_mat_plis_categorie3[8,4]*posZt[i,j])+(coefficient_mat_plis_cat
egorie3[8,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[8,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[8,1]*(posZt[i,j]**4) + (coefficient_mat_plis_categorie3[8,0]*(posZt[i,j]**5))
    if (delta_max<=delta_p17[j,0]):
        delta_max=delta_p17[j,0]
    if noeudXt_cent [i,j] > (pos_theo_mat_p17g[0,j]):
        if noeudXt_cent [i,j] < (pos_theo_mat_p17d[0]+delta_p17[j,0]):
            Mat_position_p17 [i,j]=1
#
# extrémité pli 18
#
```

Appendix

for i in range (Nx):

 for j in range (Nz):

 delta_p18[j,0]=(coefficient_mat_plis_categorie1[4,5])+(coefficient_mat_plis_categorie1[4,4]*posZt[i,j])+(coefficient_mat_plis_categorie1[4,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie1[4,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie1[4,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie1[4,0]*(posZt[i,j]**5))

 if (delta_max<=delta_p18[j,0]):

 delta_max=delta_p18[j,0]

 if noeudXt_cent [i,j] > (pos_theo_mat_p18g[0,j]):

 if noeudXt_cent [i,j] < (pos_theo_mat_p18d[0]+delta_p18[j,0]):

 Mat_position_p18 [i,j]=1

#

extrémité pli 19

#

for i in range (Nx):

 for j in range (Nz):

 delta_p191[j,0]=(coefficient_mat_plis_categorie2[1,5])+(coefficient_mat_plis_categorie2[1,4]*posZt[i,j])+(coefficient_mat_plis_categorie2[1,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie2[1,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie2[1,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie2[1,0]*(posZt[i,j]**5))

 if (delta_max<=delta_p191[j,0]):

 delta_max=delta_p191[j,0]

 delta_p192[j,0]=(coefficient_mat_plis_categorie3[9,5])+(coefficient_mat_plis_categorie3[9,4]*posZt[i,j])+(coefficient_mat_plis_categorie3[9,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie3[9,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie3[9,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie3[9,0]*(posZt[i,j]**5))

 if (delta_max<=delta_p192[j,0]):

 delta_max=delta_p192[j,0]

 delta_p193[j,0]=(coefficient_mat_plis_categorie2[1,5])+(coefficient_mat_plis_categorie2[1,4]*posZt[i,j])+(coefficient_mat_plis_categorie2[1,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie2[1,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie2[1,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie2[1,0]*(posZt[i,j]**5))

 if (delta_max<=delta_p193[j,0]):

 delta_max=delta_p193[j,0]

 if noeudXt_cent [i,j] < (-pos_theo_mat_p19d[1,0]-delta_p191[j,0]):

 if noeudXt_cent [i,j] > (pos_theo_mat_p19g[2,j]):

 Mat_position_p19[i,j]=1

 if noeudXt_cent [i,j] > (pos_theo_mat_p19g[0,j]):

 if noeudXt_cent [i,j] < (pos_theo_mat_p19d[0,0]+delta_p192[j,0]):

 Mat_position_p19[i,j]=1

 if noeudXt_cent [i,j] > (pos_theo_mat_p19d[1,0]+delta_p193[j,0]):

 if noeudXt_cent [i,j] < (pos_theo_mat_p19g[1,j]):

 Mat_position_p19 [i,j]=1

#

Appendix

extrémité pli 20

#

for i in range (Nx):

 for j in range (Nz):

 delta_p20[j,0]=(coefficient_mat_plis_categorie1[5,5])+(coefficient_mat_plis_categorie1[5,4]*posZt[i,j])+(coefficient_mat_plis_categorie1[5,3]*posZt[i,j]**2)+(coefficient_mat_plis_categorie1[5,2]*posZt[i,j]**3) +
(coefficient_mat_plis_categorie1[5,1]*(posZt[i,j]**4)) + (coefficient_mat_plis_categorie1[5,0]*(posZt[i,j]**5))

 if (delta_max<=delta_p20[j,0]):

 delta_max=delta_p20[j,0]

 if noeudXt_cent [i,j] > (pos_theo_mat_p20g[0,j]):

 if noeudXt_cent [i,j] < (pos_theo_mat_p20d[0]+delta_p20[j,0]):

 Mat_position_p20 [i,j]=1

Mat_position_total = np.zeros([Nx+1,Nz+1,20])

for i in range (0,Nx+1):

 for j in range (0,Nz+1):

 Mat_position_total[i-1,j-1,0]=Mat_position_p1 [i-1,j-1]

 Mat_position_total[i-1,j-1,1]=Mat_position_p2 [i-1,j-1]

 Mat_position_total[i-1,j-1,2]=Mat_position_p3 [i-1,j-1]

 Mat_position_total[i-1,j-1,3]=Mat_position_p4 [i-1,j-1]

 Mat_position_total[i-1,j-1,4]=Mat_position_p5 [i-1,j-1]

 Mat_position_total[i-1,j-1,5]=Mat_position_p6 [i-1,j-1]

 Mat_position_total[i-1,j-1,6]=Mat_position_p7 [i-1,j-1]

 Mat_position_total[i-1,j-1,7]=Mat_position_p8 [i-1,j-1]

 Mat_position_total[i-1,j-1,8]=Mat_position_p9 [i-1,j-1]

 Mat_position_total[i-1,j-1,9]=Mat_position_p10 [i-1,j-1]

 Mat_position_total[i-1,j-1,10]=Mat_position_p11 [i-1,j-1]

 Mat_position_total[i-1,j-1,11]=Mat_position_p12 [i-1,j-1]

 Mat_position_total[i-1,j-1,12]=Mat_position_p13 [i-1,j-1]

 Mat_position_total[i-1,j-1,13]=Mat_position_p14 [i-1,j-1]

 Mat_position_total[i-1,j-1,14]=Mat_position_p15 [i-1,j-1]

 Mat_position_total[i-1,j-1,15]=Mat_position_p16 [i-1,j-1]

 Mat_position_total[i-1,j-1,16]=Mat_position_p17 [i-1,j-1]

 Mat_position_total[i-1,j-1,17]=Mat_position_p18 [i-1,j-1]

 Mat_position_total[i-1,j-1,18]=Mat_position_p19 [i-1,j-1]

 Mat_position_total[i-1,j-1,19]=Mat_position_p20 [i-1,j-1]

#

exportation des matrices sur abaqus

#

for i in range (0,Nx):

 #écriture des lignes de définition des plis (épaisseur et orientation)

 for j in range (0,Nz):

 Npart=str(noeudNt[i,j].astype(int))

 f.write('*Elset, elset=COMPOSITELAYUP-')

 f.write(Npart)

 f.write('-1\n')

 f.write(Npart)

 f.write(', \n')

Appendix

```
#
#orientation
#
orientation = np.zeros((20))
orientation = ([45,0,45,0,45,0,45,0,45,0,45,0,45,0,45,0,45,0,45])

teta=npumpy.zeros((Nx+1,Nz+1,20))
Alpha=npumpy.zeros((13,20))
Beta=npumpy.zeros((13,20))
B=npumpy.zeros((13,20))
X=npumpy.zeros((13,20))    #Les coord des points de Gauss
Y=npumpy.zeros((13,20))    #...
Alphai_lim=([0,200])
Betai_lim=([0,200])

#orientation without variability
tetan = np.zeros((20))
tetan = ([45,0,45,0,45,0,45,0,45,0,45,0,45,0,45,0,45,0,45])

for p in range (0,20):
# orientation with variability
    pert=npumpy.zeros((Nx,Nz,20))
    #k=k+1
    for b in range (0,12):

        Alpha[b,p]=(Alphai_lim[0]+(Alphai_lim[1]-Alphai_lim[0])*random()) #paramètres de l'équation de l'orientation
        Beta[b,p]=(Betai_lim[0]+(Betai_lim[1]-Betai_lim[0])*random())    #...
        B[b,p]=gauss(0.08,2.47)
        X[b,p]=(-200+(200+200)*random())
        Y[b,p]=(-200+(200+200)*random())

        for i in range (0,Nx+1):
            for j in range(0,Nz+1):
                pert[i-1,j-1,p]=pert[i-1,j-1,p]+B[b,p]*np.exp(-(((noeudXt_cent[i-1,j-1]-X[b,p])/Alpha[b,p])**2+((noeudZt[i-1,j-1]-Y[b,p])/Beta[b,p])**2))
                teta[i-1,j-1,p]=orientation[p]+pert[i-1,j-1,p]    #équation des orientations pour chaque maille et chaque pli
teta10=npumpy.zeros((Nx,Nz+1))
for i in range (0,Nx):
    for j in range(0,Nz+1):
        teta10[i-1,j-1]=teta[i-1,j-1,0]

cptn=1
for i in range (1,Nx+1):
    #écriture des lignes de définition des plis (épaisseur et orientation)
    for j in range (1,Nz+1):
        Npart=str(noeudNt[i-1,j-1].astype(int))
        f.write(*Orientation, name=Ori-)
        f.write(str(cptn))
```

Appendix

```
f.write(' \n')
f.write('1., 0., 0., 0., 1., 0.\n')
f.write('2, 0.\n')
f.write('** Region: (COMPOSITELAYUP-)')
f.write(Npart)
# f.write(', ')
f.write('-1: Generated From Layup')
f.write(' \n')
f.write('*Elset, elset=COMPOSITELAYUP-')
f.write(Npart)
f.write('-1-1')
f.write(' \n')
f.write(' ')
f.write(Npart)
f.write(', \n')
f.write('** Section: COMPOSITELAYUP-')
f.write(Npart)
f.write('-1 \n')
f.write('*Shell Section, elset=COMPOSITELAYUP-')
f.write(Npart)
f.write('-1, COMPOSITE, layup=COMPOSITELAYUP-')
f.write(Npart)
f.write(' \n')
cptn=cptn+1
for p in range (0,20):
    if Mat_position_total[i-1,j-1,p] == 1:
        epai=str(ep[p])
        f.write(epai)
        f.write(', ')
        f.write('3')
        f.write(', ')
        Mat=str(Material[p])
        f.write(Mat)          #matériau pour chaque pli
        f.write(str(noedNt[i-1,j-1].astype(int)))
        f.write(', ')
        angle=str(round(teta[i-1,j-1,p],5))      #angle d'orientation with variability
        #angle=str(round(tetan[p],5))
        f.write(angle)
        f.write(', ')
        f.write('PLY-')
        kk=str(p+1)
        f.write(kk)
        f.write('-')
        f.write(str(noedNt[i-1,j-1].astype(int)))
        f.write(' \n')
f.write(' \n')
f.write('*End Part \n')
```

Appendix

```
f.write('** \n')
f.write('**\n')
f.write('** ASSEMBLY\n')
f.write('**\n')
f.write('*Assembly, name=Assembly\n')
f.write('** \n')
f.write('*Instance, name=PART-1-1, part=PART-1\n')
f.write('*End Instance\n')
f.write('** \n')
#définition des set (points ou zones d'application des conditions aux limites et des chargements)
f.write('*Nset, nset=Set-1, instance=PART-1-1\n')
moyx=round(N5/2)
moyz=round(Nz/2)
f.write(str(noeudN5[moyx,moyz].astype(int)))
f.write(',\n')
f.write('*Nset, nset=Set-5, instance=PART-1-1\n') #point d'application de blockage x et y
pointx=round(N5/2)
f.write(str(noeudN5[0,pointx].astype(int)))
f.write(',\n')
f.write('*Nset, nset=temp, instance=PART-1-1\n')
cptnn =0
for i in range (0,Nx+1):
    for j in range (0,Nz+1):
        ipt=noeudNtot[i,j].astype(int)
        f.write(str(ipt))
        f.write(',')
        cptnn= cptnn+1
        if cptnn>14:
            f.write ('\n')
            cptnn=0
f.write ('\n')
cptnnn =0
f.write ('*Nset, nset=encastre, instance=PART-1-1\n')
for i in range (0,Nx+1):
    ipt=noeudNtot[i,0].astype(int)
    f.write(str(ipt))
    f.write(',')
    cptnnn= cptnnn+1
    if cptnnn>14:
        f.write ('\n')
        cptnnn=0
f.write ('\n')
cptnnnn =0
f.write ('*Nset, nset=traction, instance=PART-1-1\n')
for i in range (0,Nx+1):
    ipt=noeudNtot[i,Nz].astype(int)
    f.write(str(ipt))
```

Appendix

```
f.write(',')
cptnnnn= cptnnnn+1
if cptnnnn>14:
    f.write ('\n')
    cptnnnn=0
f.write ('\n')
f.write('*End Assembly\n')
f.write('**\n')
f.write('**MATERIALS\n')
f.write('**\n')
#définition composite UD
for i in range (0,Nx+1):
    for j in range (0,Nz+1):
        f.write('*Material, name=COMPOSITEUD')
        f.write(str(noeadNt[i-1,j-1].astype(int)))
        f.write ('\n')
        f.write('*Density\n')
        f.write(density_nom_TISSE)
        f.write(',\n')
        f.write('*Elastic, type=ENGINEERING CONSTANTS\n')
#caractéristiques matériau UD
        f.write(str(E1UD[i-1,j-1]))
        f.write(', ')
        f.write(str(E2UD[i-1,j-1]))
        f.write(', ')
        f.write(str(E3UD[i-1,j-1]))
        f.write(', ')
        f.write(Nu12UD)
        f.write(', ')
        f.write(Nu13UD)
        f.write(', ')
        f.write(Nu23UD)
        f.write(', ')
        f.write(str(G12UD[i-1,j-1]))
        f.write(', ')
        f.write(str(G13UD[i-1,j-1]))
        f.write(', \n')
        f.write(str(G23UD[i-1,j-1]))
        f.write(', \n')
        f.write('*Fail Stress\n')
        f.write(Ten1UD)
        f.write(',')
        f.write(Com1UD)
        f.write(', ')
        f.write(Ten2UD)
        f.write(', ')
        f.write(Com2UD)
```

```
f.write(', ')
f.write(SUD)
f.write(', ')
f.write(CrossUD)
f.write(', ')
f.write(Stress_L)
f.write('\n')
f.write(*Expansion, type=ORTHO\n')
f.write(alpha_11UD)
f.write(', ')
f.write(alpha_22UD)
f.write(', ')
f.write(alpha_33UD)
f.write('\n')
```

#définition composite tissé par maille

```
for i in range (0,Nx+1):
    for j in range (0,Nz+1):
        f.write(*Material, name=COMPOSITEWOVEN')
        f.write(str(noeudNt[i-1,j-1].astype(int)))
        f.write('\n')
        f.write(*Density\n')
        f.write(density_nom_TISSE)
        f.write(', \n')
        f.write(*Elastic, type=ENGINEERING CONSTANTS\n')
```

#caractéristiques matériau tissé

```
f.write(str(E1WOVEN[i-1,j-1]))
f.write(', ')
f.write(str(E2WOVEN[i-1,j-1]))
f.write(', ')
f.write(str(E3WOVEN[i-1,j-1]))
f.write(', ')
f.write(Nu12WOVEN)
f.write(', ')
f.write(Nu13WOVEN)
f.write(', ')
f.write(Nu23WOVEN)
f.write(', ')
f.write(str(G12WOVEN[i-1,j-1]))
f.write(', ')
f.write(str(G13WOVEN[i-1,j-1]))
f.write(', \n')
f.write(str(G23WOVEN[i-1,j-1]))
f.write(', \n')
f.write(*Fail Stress\n')
f.write(Ten1WOVEN)
f.write(',')
f.write(Com1WOVEN)
```

Appendix

```
f.write(' ')
f.write(Ten2WOVEN)
f.write(' ')
f.write(Com2WOVEN)
f.write(' ')
f.write(SWOVEN)
f.write(' ')
f.write(CrossWOVEN)
f.write(' ')
f.write(Stress_L)
f.write('\n')
f.write('*Expansion, type=ORTHO\n')
f.write(alpha_11WOVEN)
f.write(' ')
f.write(alpha_22WOVEN)
f.write(' ')
f.write(alpha_33WOVEN)
f.write('\n')
f.write('**\n')
#définition des conditions aux limites
f.write('*-----\n')
f.write('* \n')
f.write('* STEP: temp\n')
f.write('* \n')
f.write('*Step, name=temp, nlgeom=NO\n')
f.write('*Static\n')
f.write('1., 1., 1e-05, 1.\n')
f.write('* \n')
f.write('* PREDEFINED FIELDS\n')
f.write('* \n')
f.write('* Name: Predefined Field-1 Type: Temperature\n')
f.write('*Temperature\n')
f.write('temp, -100.\n')
f.write('* \n')
f.write('* OUTPUT REQUESTS\n')
f.write('* \n')
f.write('*Restart, write, frequency=0\n')
f.write('* \n')
f.write('* FIELD OUTPUT: F-Output-3\n')
f.write('* \n')
f.write('*Output, field\n')
f.write('*Contact Output\n')
f.write('CDISP, CStress\n')
f.write('* \n')
f.write('* FIELD OUTPUT: F-Output-1\n')
f.write('* \n')
f.write('*Node Output\n')
```

Appendix

```
f.write('CF, RF, U\n')
f.write('** \n')
f.write('** FIELD OUTPUT: F-Output-2\n')
f.write('** \n')
f.write('*Element Output, directions=YES\n')
f.write('CFAILURE, LE, PE, PEEQ, PEMAG, S\n')
f.write('** \n')
f.write('** HISTORY OUTPUT: H-Output-1\n')
f.write('** \n')
f.write('*Output, history, variable=PRESELECT\n')
f.write('*End Step\n')
f.write('** -----\n')
f.write('** \n')
f.write('** STEP: traction \n')
f.write('** \n')
f.write('*Step, name=traction, nlgeom=NO \n')
f.write('*Static \n')
f.write('1., 1., 1e-05, 1. \n')
f.write('** \n')
f.write('** BOUNDARY CONDITIONS\n')
f.write('**\n')
f.write('** Name: BC-4 Type: Displacement/Rotation\n')
f.write('*Boundary\n')
f.write('encastre, 3, 3 \n')
f.write('** Name: BC-3 Type: Displacement/Rotation\n')
f.write('*Boundary\n')
f.write('Set-5, 1, 1\n')
f.write('Set-5, 2, 2\n')
f.write('** Name: BC-5 Type: Displacement/Rotation\n')
f.write('*Boundary\n')
f.write('traction, 3, 3, 1. \n')
f.write('** \n')
f.write('**OUTPUT REQUESTS\n')
f.write('**\n')
f.write('*Restart, write, frequency=0\n')
f.write('**\n')
f.write('**FIELD OUTPUT: F-Output-1\n')
f.write('**\n')
f.write('*Output, field\n')
f.write('*Node Output\n')
f.write('CF, RF, U\n')
f.write('*Element Output, directions=YES\n')
f.write('CFAILURE, LE, PE, PEEQ, PEMAG, S\n')
f.write('*Contact Output\n')
f.write('CDISP, CSTRESS\n')
f.write('**\n')
f.write('**HISTORY OUTPUT: H-Output-1\n')
```

Appendix

```
f.write('**\n')
f.write('*Output, history, variable=PRESELECT\n')
f.write('*End Step\n')
f.close()
print ("l'ecart max est : ", delta_max)
```