

Titre: Génération de chemins de couverture pour des opérations automatisées de contrôle non destructif appliquées dans l'industrie aérospatiale
Title:

Auteur: Pierre Olivier
Author:

Date: 2013

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Olivier, P. (2013). Génération de chemins de couverture pour des opérations automatisées de contrôle non destructif appliquées dans l'industrie aérospatiale
Citation: [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie.
<https://publications.polymtl.ca/1117/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie:
PolyPublie URL: <https://publications.polymtl.ca/1117/>

Directeurs de recherche: Lionel Birglen
Advisors:

Programme: Génie mécanique
Program:

UNIVERSITÉ DE MONTRÉAL

GÉNÉRATION DE CHEMINS DE COUVERTURE POUR DES OPÉRATIONS
AUTOMATISÉES DE CONTRÔLE NON DESTRUCTIF APPLIQUÉES DANS L'INDUSTRIE
AÉROSPATIALE

PIERRE OLIVIERI
DÉPARTEMENT DE GÉNIE MÉCANIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE MÉCANIQUE)
AVRIL 2013

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

GÉNÉRATION DE CHEMINS DE COUVERTURE POUR DES OPÉRATIONS
AUTOMATISÉES DE CONTRÔLE NON DESTRUCTIF APPLIQUÉES DANS L'INDUSTRIE
AÉROSPATIALE

présenté par : OLIVIERI Pierre

en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de :

M. BARON Luc, Ph.D., président

M. BIRGLEN Lionel, Ph.D., membre et directeur de recherche

M. BILODEAU Guillaume-Alexandre, Ph.D., membre

REMERCIEMENTS

Je veux d'abord remercier mon directeur de recherche, Lionel Birglen, pour m'avoir fait confiance depuis le début de ma maîtrise, pour avoir régulièrement suivi mes travaux et fait de nombreuses suggestions utiles pour m'orienter dans cette recherche. Je veux également remercier les organismes tels que le Consortium de Recherche et d'Innovation en Aérospatiale au Québec (CRIAQ) et le Conseil National de Recherche du Canada (CNRC), ainsi que les partenaires industriels tels que L-3 MAS et Pratt & Whitney Canada pour leurs soutiens financiers, matériels et humains à l'avancement du projet MANU-418 du CRIAQ dans lequel s'inscrit mon sujet de recherche (parmi d'autres).

J'adresse mes remerciements à tous les membres du projet (professeurs, étudiants et partenaires industriels) pour leurs suggestions, leur collaboration et leur gentillesse tout au long de la maîtrise. En particulier à Rami Massoud pour ses idées et son dynamisme qui ont permis de clarifier les objectifs et de prendre les bonnes décisions dans mon travail de recherche et plus globalement, dans le projet MANU-418. Je veux également souligner les efforts et le temps que Christian Corbeil m'a accordé pour réaliser des expériences au CNRC ainsi que le support de ses collègues (Yves et Martin). Je suis également très reconnaissant du support technique de François Menard et en particulier Alain Robidoux à l'école Polytechnique qui ont été indispensables dans ce projet.

Enfin, je veux remercier très sincèrement ma famille, mes collègues du lab et mes amis proches pour leur soutien moral continu qui m'a beaucoup aidé à persévérer dans mes travaux tout au long de la maîtrise.

RÉSUMÉ

L'industrie aérospatiale accorde un rôle majeur au contrôle non destructif (CND) dans le processus de fabrication et de maintenance des structures et il est utilisé, entre autres, pour détecter des défauts tels que des fissures à un stade précoce. Cependant, les techniques de CND sont encore principalement opérées manuellement, en particulier sur des structures aérospatiales complexes. Plusieurs inconvénients en résultent tels que la pénibilité et le temps d'opération des techniques. De plus, la fiabilité et la répétabilité des résultats d'inspection est susceptible de varier de manière significative puisqu'elles dépendent de l'expérience et de la dextérité de chaque opérateur. Dans le cadre du projet MANU-418 du Consortium de recherche et d'innovation en aérospatiale au Québec (CRIAQ) dans lequel s'inscrit le présent travail de recherche, il a été proposé de développer un système muni d'un bras manipulateur à 6 DDL permettant d'automatiser trois techniques particulières de CND utilisées dans l'industrie aérospatiale : l'inspection par courants de Foucault (ECT), l'inspection par ressuage fluorescent (FPI) et l'inspection par thermographie infrarouge (IRT). L'objectif global du projet MANU-418 est de démontrer que ce système automatisé permet d'obtenir des résultats de détection de défauts (des fissures généralement) en surface et légèrement sous la surface au moins aussi acceptables en termes de fiabilité et de répétabilité que ceux obtenus manuellement.

Un des objectifs spécifiques qui découle de ce projet est de développer une méthodologie et un outil logiciel de génération de chemins de couverture adaptés aux trois techniques de CND mentionnées précédemment pour l'inspection de surfaces complexes de structures aérospatiales. Le présent travail de recherche s'attache à la réalisation de cet objectif spécifique.

Dans un premier temps, les propriétés de géométrie et de topologie du type de surfaces considérées dans ce projet sont définies (surfaces planes, congés, bords droits, surfaces cylindriques, trou dans la surface). Il est également supposé que le modèle 3D de la surface à inspecter est connu à l'avance. Aussi, l'automatisation de l'inspection ECT étant prioritaire devant celles des techniques FPI et IRT dans le cadre du projet MANU-418, la méthodologie développée se base sur les propriétés de chemin qu'une sonde différentielle à courants de Foucault (appelée ici sonde EC) suit habituellement lors d'une inspection manuelle, tout en permettant de rester utilisable pour les autres techniques. Le chemin d'inspection d'une sonde EC est défini par un balayage en zigzag et cette sonde est généralement en contact léger avec la surface durant le balayage. Aussi, l'axe de la sonde doit en tout temps être normal à la surface et, l'alignement de ses deux bobines doit toujours être orienté le long de sa direction de déplacement.

Une première méthodologie est alors proposée pour générer des chemins de couverture sur toute la surface à inspecter en respectant les contraintes de la sonde EC. En premier lieu, la surface

complexe est maillée par des facettes triangulaires, puis subdivisée en plusieurs “zones” de géométrie et de topologie plus simples. Les chemins sont ensuite générés sur chacun des patchs par intersection de leurs facettes avec des plans de coupe décalés dans la direction de balayage. Une autre méthodologie est aussi développée pour générer des chemins autour d’une indication (une petite zone où la présence d’un défaut est suspectée) dont la position et l’orientation sont supposées connues *a priori*. Pour ces deux méthodologies, une fois les positions de l’outil définies le long d’un chemin, l’orientation de l’outil est calculée afin de respecter les contraintes d’une inspection ECT. Aussi, afin d’adapter les chemins ainsi générés pour les techniques FPI et IRT, un paramètre de distance entre la surface et l’extrémité de l’outil d’inspection est défini (le contact correspond donc à une valeur nulle de ce paramètre).

Un outil logiciel muni d’une interface utilisateur a alors été développé dans l’environnement MATLAB afin de générer des chemins basés sur ces méthodologies. Un ensemble de paramètres de chemin sont proposés et modifiables par l’utilisateur pour que celui-ci puisse générer les chemins désirés (distance entre deux passes adjacentes, direction de balayage, etc.). Une fois les chemins calculés, une liste de coordonnées des positions et orientations des chemins sont exportés dans une table EXCEL pour pouvoir être utilisée par un robot. Dans le cadre de ce travail de recherche, des simulations de trajectoires (chemins exprimés par rapport au temps) d’un modèle de robot MotoMan (SV3XL) sont effectuées avec le logiciel MotoSim à partir des données exportées des chemins. Après validation des trajectoires dans ce logiciel (absence de collisions, positions atteignables par le robot, etc.), des instructions sont enfin générées pour mener des tests expérimentaux sur le robot réel.

Les premières simulations et tests expérimentaux des chemins ainsi générés correspondent visuellement aux mouvements attendus pour les techniques d’inspection considérées, en particulier pour la technique ECT. Néanmoins, il conviendra de valider cette méthodologie avec davantage de tests expérimentaux, par exemple avec la présence d’un outil “test” pour vérifier son positionnement et son orientation vis-à-vis d’une surface complexe typique d’une structure aérospatiale.

ABSTRACT

Non destructive testing (NDT) plays an important role in the aerospace industry during the fabrication and maintenance of the structures built and is used, among other useful applications, to detect flaws such as cracks at an early stage. However, NDT techniques are still mainly done manually, especially on complex aeronautical structures, which then results in several drawbacks. In addition to be difficult and time-consuming, reliability and repeatability of inspection results are likely to be affected, since they rely on each operator's experience and dexterity. The present thesis is part of a larger project (MANU-418) of the Consortium for Research and Innovation in Aerospace in Québec (CRIAQ). In this project, it has been proposed to develop a system using a 6-DOF manipulator arm to automate three particular NDT techniques often needed in the aerospace industry: eddy current testing (ECT), fluorescent penetrant inspection (FPI), and infrared thermography (IRT). The main objective of the MANU-418 project is to demonstrate the efficiency of the developed system and provide inspection results of surface and near surface flaws (cracks usually) at least as reliably and repeatably as inspection results from a human operator.

One specific objective stemming from the main objective of the project is to develop a methodology and a software tool to generate covering paths adapted for the three aforementioned NDT techniques to inspect the complex surfaces of aerospace structures. The present thesis aims at reaching this specific objective.

At first, geometrical and topological properties of the surfaces considered in this project are defined (flat surfaces, round and straight edges, cylindrical or near cylindrical surfaces, holes). It is also assumed that the 3D model of the surface to inspect is known in advance. Moreover, it has been decided within the framework of the MANU-418 project to give priority to the automation of ECT compared with the other techniques (FPI and IRT). As a result, the methodology developed to generate inspection paths is more closely focused on path constraints relative to the manual operations of ECT using a differential eddy current probe (named here EC probe), but it is developed to be flexible enough to be used with the other techniques as well. Common inspection paths for ECT are usually defined by a sweeping motion using a zigzag pattern with the EC probe in mild contact with the inspected surface. Moreover, the main axis of the probe must keep a normal orientation with the surface, and the alignment of its two coils must always be oriented along the direction of its motion.

A first methodology is then proposed to generate covering paths on the whole surface of interest while meeting all EC probe motion constraints. First, the surface is meshed with triangular facets, and then it is subdivided into several patches such that their geometry and topology are simpler than the whole surface. Paths are then generated on each patch by intersecting their facets with

offset section planes defined along a sweeping direction. Furthermore, another methodology is developed to generate paths around an indication (namely a small area where the presence of a flaw is suspected) whose position and orientation are assumed to be known *a priori*. For both these methodologies, once the positions along a path have been computed, orientations are deduced to meet ECT probe motion constraints. Also, in order to adapt generated paths based on these methodologies to FPI and IRT techniques, a distance parameter is defined between the surface and the end of the inspection tool (contact is obtained when this parameter is equal to zero).

Then, a software tool with a graphical user interface has been developed in the MATLAB environment to generate inspection paths based on these methodologies. A set of path parameters can be changed by the user to get desired paths (distance between passes, sweep direction, etc.). Once paths are computed, an ordered list of coordinates (positions and orientations) of the tool is exported in an EXCEL spreadsheet so that it could be used with a real robot. In this research, these data are then used to perform simulations of trajectories (path described as a function of the time) with a MotoMan robot (model SV3XL) using the MotoSim software. After validation of these trajectories in this software (absence of collisions, positions are all reachable, etc.), they are finally converted into instructions for the real MotoMan robot to proceed with experimental tests.

These first simulations and experimentations on a MotoMan robot of the generated paths have given results close to the expected inspection trajectories used manually in the NDT techniques considered, especially for the ECT technique. Nevertheless, it is strongly recommended to validate this path generation method with more experimental tests. For instance, a “test” tool could be manufactured to measure errors of position and orientation of this tool with respect to expected trajectories on a typical complex aeronautical structure.

TABLE DES MATIÈRES

REMERCIEMENTS	iii
RÉSUMÉ	iv
ABSTRACT	vi
TABLE DES MATIÈRES	viii
LISTE DES TABLEAUX	xi
LISTE DES FIGURES	xii
LISTE DES ANNEXES	xv
LISTE DES SIGLES ET ABRÉVIATIONS	xvi
CHAPITRE 1 INTRODUCTION	1
1.1 Cadre de la recherche	1
1.1.1 Inspection par ressuage fluorescent (FPI)	1
1.1.2 Inspection par courants de Foucault (ECT)	3
1.1.3 Inspection par thermographie infrarouge (IRT)	4
1.1.4 Problème soulevé dans le projet	5
1.2 Objectif de la recherche	6
1.3 Hypothèses de la recherche	7
CHAPITRE 2 REVUE CRITIQUE DE LA LITTÉRATURE	12
2.1 Systèmes automatisés existants de CND	12
2.1.1 Systèmes automatisés pour FPI	12
2.1.2 Systèmes automatisés pour ECT	13
2.1.3 Systèmes automatisés pour IRT	15
2.2 Techniques de génération de chemins de couverture	17
2.2.1 Méthodes de programmation des robots industriels	17
2.2.2 Les motifs de chemins de couverture	18
2.2.3 Approches avec ou sans connaissance de la géométrie de la surface	19

2.2.4	Techniques de génération de chemins en zigzag sur des surfaces tridimensionnelles	20
CHAPITRE 3 DÉMARCHE DE TRAVAIL		23
CHAPITRE 4 ARTICLE : COVERAGE PATH PLANNING FOR EDDY CURRENT INSPECTION ON COMPLEX AERONAUTICAL PARTS		25
4.1	Abstract	25
4.2	Introduction	25
4.3	Outline of the proposed method	27
4.4	Segmentation based on curvature: watershed segmentation	28
4.5	Segmentation based on surface normals	31
4.6	Segmentation based on topology: the Morse decomposition	33
4.6.1	Rationale	33
4.6.2	Morse Decomposition	34
4.6.3	Extension of the Boustrophedon Algorithm	36
4.7	Path generation	40
4.8	Results	41
4.9	Conclusion	43
4.10	Acknowledgments	44
CHAPITRE 5 ASPECTS MÉTHODOLOGIQUES ET RÉSULTATS COMPLÉMENTAIRES		48
5.1	Remaillage lors de la segmentation “watershed”	48
5.2	Calcul de l’orientation de l’outil	49
5.2.1	Calcul de l’orientation du référentiel de la sonde EC en chaque point du chemin	50
5.2.2	Orientation décrite avec la convention “X-Y-Z fixed angles”	51
5.3	Génération de chemins autour d’une indication de coordonnées connues	53
5.4	Programme de génération de chemins	55
5.5	Simulations d’un bras manipulateur sous le logiciel MotoSim	56
5.6	Tests expérimentaux	57
CHAPITRE 6 DISCUSSION GÉNÉRALE		63
6.1	Limitations du programme de génération de chemins	63
6.2	Améliorations futures	64
CHAPITRE 7 CONCLUSION		67

RÉFÉRENCES	70
ANNEXES	74

LISTE DES TABLEAUX

Table 4.1	Identification of critical points of the outer boundary $P(1)$	40
Table 4.2	Identification of critical points of the inner boundaries $P(j)$, $j \geq 2$	41

LISTE DES FIGURES

Figure 1.1	Étapes d'application de la technique FPI.	2
Figure 1.2	Schéma d'une sonde différentielle. Extrait de NDT Education Resource Center (2013c).	4
Figure 1.3	Signal représenté dans le plan complexe (phase et amplitude) de la variation d'impédance d'une sonde EC différentielle. Le signal correspond à la variation complexe de l'impédance (résistance et réactance) en fonction du temps.	5
Figure 1.4	Influence de l'orientation de la sonde différentielle par rapport à la direction de son déplacement.	6
Figure 1.5	Perturbations des courants de Foucault selon l'orientation d'une fissure et la direction de déplacement de la sonde EC. Adapté de Buckley (2013). . .	7
Figure 1.6	Exemple de système d'acquisition pour une inspection IRT avec une bobine comme dispositif de perturbation locale de la température de la surface. Adapté de Spicer et Osiander (2002).	8
Figure 1.7	Cartographie de la température d'une plaque de calibration contenant des défauts artificiels révélés en rouge. La légende colorée est représentative de la température (du bleu au rouge respectivement pour les températures basses et élevées). Extrait de Castenado (2005).	9
Figure 1.8	Pièce structurelle typique issue de l'industrie aéronautique.	10
Figure 1.9	Surface relativement plane de topologie complexe. La surface projetée dans son plan moyen n'est pas convexe : il existe au moins un couple de points (A, B) de la surface projetée tel que le segment $[AB]$ ne soit pas inclus dans cette dernière.	11
Figure 2.1	Système automatisé de ressuage par pénétrant fluorescent chez AIRBUS, extrait de Vetterlein <i>et al.</i> (2006).	13
Figure 2.2	Système automatisé de ressuage par pénétrant fluorescent pour l'inspection de composants de moteur de turbine, extrait de PROCECO (2013).	14
Figure 2.3	Système automatisé d'inspection par courants de Foucault sur un arbre à cames. MARPOSS (2013).	15
Figure 2.4	Système automatisé d'inspection par courants de Foucault avec un bras manipulateur (Eloscan). Extrait de Rohmann GmbH (2008).	16

Figure 2.5	Système IRT automatisé (modèle DIO 4000 de l'entreprise STARMANS) pour la détection de fissures sur des barres d'acier (billettes) de section carrée ou arrondie. Extrait de STARMANS electronics Ltd. (2013).	17
Figure 2.6	Exemples de motifs de chemin pour des applications de recouvrement. . .	18
Figure 2.7	Trois techniques de génération de chemin en zigzag sur une surface 3D, adapté de Atkar <i>et al.</i> (2003) et Misra <i>et al.</i> (2005).	22
Figure 4.1	Construction of paths with section planes (left) and with offset curves (right), adapted from Atkar <i>et al.</i> (2003).	27
Figure 4.2	Flowchart of the method.	29
Figure 4.3	Principle of the watershed segmentation, adapted from Eddins (2002). . .	30
Figure 4.4	Illustration of the watershed segmentation algorithm.	32
Figure 4.5	Three patches formed using the segmentation based on surface normals, $\theta_{thresh} = 120^\circ$	33
Figure 4.6	Path over a hole resulting in a long cycle-time.	33
Figure 4.7	Path over a hole resulting in collisions.	34
Figure 4.8	The region is decomposed in new cells by sweeping a "slice," adapted from Acar <i>et al.</i> (2002).	35
Figure 4.9	The Morse decomposition in a planar case leads here to six new cells (critical points are represented by black dots), adapted from Acar <i>et al.</i> (2002). . .	36
Figure 4.10	Boustrophedon decomposition in a planar case Choset et Pignon (1997). . .	37
Figure 4.11	Boustrophedon decomposition on non-convex obstacles.	38
Figure 4.12	Notations used for the search of critical points.	39
Figure 4.13	Modified boustrophedon decomposition applied on non-convex obstacles. .	42
Figure 4.14	Bounding box of the surface and resultant path with $sweep_dir = 0^\circ$, $d = 5mm$, $Os_d = -2mm$ and $offset_edge = 2mm$	43
Figure 4.15	Parameters of a path.	44
Figure 4.16	Limitation of the method with section planes.	45
Figure 4.17	Aeronautic structural part. The light yellow surface is selected here for an inspection with eddy currents.	45
Figure 4.18	First segmentation : watershed segmentation with $depth_{thresh} = 0.1$	46
Figure 4.19	Second segmentation : based on surface normals with $\theta_{thresh} = 100^\circ$. . .	46
Figure 4.20	Third segmentation : the Morse decomposition.	47
Figure 4.21	Path result with $sweep_dir = 0^\circ$, $d = 2mm$, $Os_d = -1mm$ and $offset_edge = 0mm$	47
Figure 5.1	Surface triangulée d'un parallélépipède rectangle. Le maillage n'a pas été raffiné dans ce cas-ci.	49

Figure 5.2	Surface remaillée. Un sommet est à présent défini au milieu de chacune des faces.	49
Figure 5.3	Représentations des référentiels de la sonde en chaque point du chemin. . .	52
Figure 5.4	Méthode de génération de chemin autour d'une indication par plans de coupe régulièrement espacés et normaux le long de la ligne brisée représentant l'indication.	54
Figure 5.5	Paramètres du chemin de couverture autour d'une indication.	55
Figure 5.6	Chemin d'inspection autour d'une indication avec les valeurs de paramètres suivants : $d_Ind = 5mm$, $W_Ind = 4mm$ et $offset_Ind = 5mm$	56
Figure 5.7	Architecture globale du programme de génération de chemins de couverture.	57
Figure 5.8	Interface graphique pour la génération de chemins sur une surface globale.	58
Figure 5.9	Interface graphique pour la génération de chemins autour d'une indication.	59
Figure 5.10	Simulation sous le logiciel MotoSim d'une trajectoire sur une face d'une plaque.	60
Figure 5.11	Fichier "Plaque.jbi" produit par MotoSim et contenant les informations de la trajectoire. Dans ce fichier sont indiqués, entre autres, l'outil utilisé, les configurations du robot (16 configurations pour cette trajectoire) exprimées en pulse le long de la trajectoire ainsi que la vitesse associée à chacune des configurations.	61
Figure 5.12	Test expérimental avec le robot MotoMan SV3XL.	62
Figure 6.1	Illustration d'une surface paramétrique constituant la surface d'une pièce. Les frontières de la surface paramétrique sélectionnée sont marquées en pointillés orange.	64
Figure 6.2	Limitation de la méthode des plans de coupe. La distance entre les passes adjacentes peut augmenter significativement, en particulier sur des surfaces de courbure élevées.	65
Figure 6.3	Exemple d'outil "test" mesurant l'erreur de positionnement et d'orientation dans le cas d'une inspection ECT.	66

LISTE DES ANNEXES

Annexe A	MANUEL UTILISATEUR	74
----------	------------------------------	----

LISTE DES SIGLES ET ABRÉVIATIONS

CND	Contrôle non destructif (ou NDT : non destructive testing)
ECT	Eddy current testing : inspection par courant de Foucault
EC	Eddy current : courant de Foucault
FPI	Fluorescent penetrant inspection : inspection par ressuage de pénétrant fluorescent
IRT	Infrared thermography testing : inspection par thermographie infrarouge
CFRP	Carbon Fiber Reinforced Plastic : plastique à renfort fibre de carbone
CAO	Conception assistée par ordinateur (ou CAD : computer assisted design)
B-Rep	Boundary representation : représentation par les bords
CSG	Constructive Solid Geometry : géométrie de construction de solides
STL	Stereolithography : stéréolithographie
DDL	Degré de liberté (ou DOF : degree(s) of freedom)
UV	Ultraviolet
GUI	Graphical User Interface : interface graphique utilisateur

CHAPITRE 1

INTRODUCTION

1.1 Cadre de la recherche

L'inspection des structures dans l'industrie aéronautique est une étape cruciale permettant de prévenir leur rupture et ainsi garantir leur qualité, autant au cours de la phase de fabrication que durant la maintenance. Pour réaliser ces inspections, des techniques de contrôle non destructif (CND) ont été développées dans le but de vérifier la présence de défauts en surface ou sous la surface (fissures, porosités, cavités, inclusions de matériaux étrangers, etc.) et de les caractériser (position, dimensions, etc.) sans endommager ces structures. Dans le cadre du projet MANU-418 du Consortium de recherche et d'innovation en aéronautique au Québec (CRIAQ) dans lequel s'inscrit ce travail de recherche, trois techniques de CND fréquemment utilisées dans l'industrie aéronautique sont considérées : l'inspection par ressuage fluorescent (FPI), l'inspection par courants de Foucault (ECT) et l'inspection par thermographie infrarouge (IRT).

1.1.1 Inspection par ressuage fluorescent (FPI)

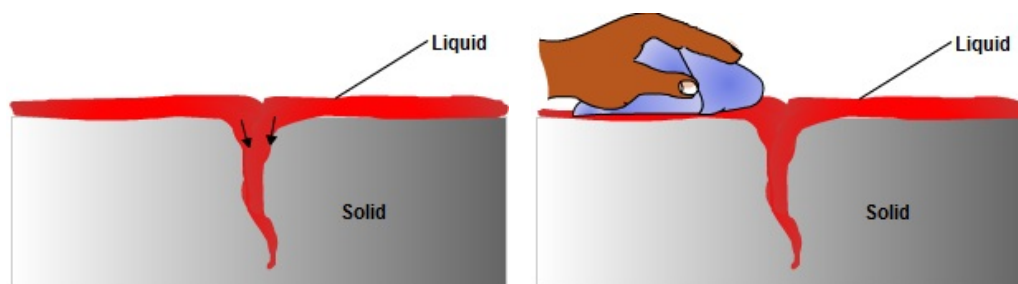
La première technique de CND mentionnée précédemment, FPI, est utilisée uniquement pour la détection de “vides” formés en surface (défauts débouchants). L'application de cette technique s'effectue en sept étapes successives :

1. Nettoyer la surface à inspecter pour retirer les agents contaminants (saletés, traces d'huile, etc.) pouvant nuire aux processus d'inspection (pré-nettoyage).
2. Appliquer un produit liquide fluorescent appelé “pénétrant” qui permettra d'infiltrer facilement les défauts débouchants (Figure 1.1(a)).
3. Laisser le pénétrant s'infiltrer dans les défauts pendant une durée appelée “ temps d'imprégnation ” (en moyenne 30 min).
4. Retirer l'excès de pénétrant sur la surface pour ne conserver que le pénétrant qui a imprégné les défauts. Cette opération peut être effectuée à l'aide d'un chiffon non pelucheux légèrement imprégné de solvant (Figure 1.1(b)). Si la pièce est lavable à l'eau, une autre manière de retirer l'excès de pénétrant consiste à appliquer un produit émulsifiant à la surface, puis de la rincer avec un jet d'eau.
5. Appliquer un produit à base de talc appelé “révélateur” qui permettra de faire ressortir le pénétrant contenu dans les défauts et faire apparaître des tâches fluorescentes sur la surface

(Figure 1.1(c)).

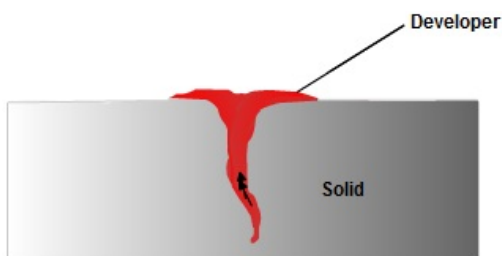
6. Inspecter visuellement sous un faible éclairage en lumière visible et utiliser une lumière ultraviolet (UV) pour faire apparaître clairement les indications fluorescentes formées sur la surface. Ainsi, il est possible de déterminer la position et la taille de chacune de ces indications. Selon des critères définis à l'avance, les dimensions de l'indication permettront de juger si le défaut est acceptable ou non sur la pièce considérée (Figure 1.1(d)).
7. Nettoyer la surface de toutes traces de pénétrant et de révélateur appliqués durant le processus (post-nettoyage).

Cette technique a l'avantage d'être utilisée sur de nombreux types de matériaux, à l'exception des matériaux poreux ou présentant des surfaces rugueuses.

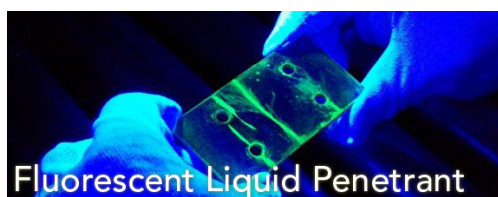


(a) Application du pénétrant sur la surface à inspecter, extrait de NDT Education Resource Center (2013a).

(b) Retrait de l'excès de pénétrant à la surface pour ne laisser que le pénétrant dans les défauts débouchants, extrait de NDT Education Resource Center (2013a).



(c) Application du révélateur (developer en anglais) pour faire ressortir en surface le pénétrant contenu dans les défauts débouchants, extrait de NDT Education Resource Center (2013a).



(d) Inspection visuelle sous lumière UV de la surface. Les indications vertes fluorescentes correspondent au pénétrant contenu dans les défauts en surface. Extrait de Atmospheres (2013).

Figure 1.1 Étapes d'application de la technique FPI.

1.1.2 Inspection par courants de Foucault (ECT)

La deuxième technique considérée ici est la technique ECT. Celle-ci se base sur les principes de l'électromagnétisme pour détecter des défauts en surface ou légèrement sous la surface (fissures, cavités ou inclusions de matériaux étrangers). En pratique, une sonde à courant de Foucault (notée plus simplement sonde EC) est déplacée selon un mouvement de **balayage en zigzag** sur la surface à inspecter (souvent en léger contact avec celle-ci) en conservant une **orientation normale** entre elles. Différentes formes de sondes existent pour s'adapter à des géométries particulières de surface (surfaces plates, congés, trou, etc.). Ayant souvent l'apparence d'un stylo, une sonde EC est constituée d'au moins une bobine électrique placée à son extrémité. Cette bobine est parcourue par un courant alternatif qui génère alors un champ magnétique alternatif, appelé **champ magnétique primaire**. Lorsque la bobine est approchée d'un matériau conducteur, des courants (alternatifs également) sont alors induits à sa surface et légèrement sous sa surface (courants de Foucault), lesquels génèrent à leur tour leur propre champ magnétique alternatif, appelé **champ magnétique secondaire**. Selon la loi de Lenz, ce champ magnétique secondaire va s'opposer au champ magnétique primaire. Lorsque les courants de Foucault sont interceptés par un défaut, le champ magnétique secondaire est alors perturbé, provoquant une variation d'impédance électrique dans le circuit de la sonde. C'est cette variation d'impédance qui est mesurée et utilisée comme signal de sortie pour la détection et la caractérisation des défauts rencontrés.

Différentes catégories de sondes basées sur ce principe existent. Elles sont classées selon leur mode d'opération : sonde absolue, sonde différentielle, sonde à réflexion et sonde hybride (Community College Education Resource). Dans le cadre de ce projet de recherche, des sondes différentielles sont considérées. En comparaison à une sonde absolue ne comportant qu'une seule bobine, une sonde différentielle en possède deux qui sont connectées en série avec des enroulements dans des sens opposés (cf. Figure 1.2). Une telle configuration des bobines permet une plus grande sensibilité pour la détection de fissures comparativement à une sonde absolue. En effet, le signal mesuré correspond à la différence des signaux d'impédance des deux bobines. Dans le cas où la sonde n'est pas vis-à-vis d'une fissure, les impédances des deux bobines seront alors égales et le signal différentiel observé dans le plan d'impédance restera centré (cf. Figure 1.3(a)). À l'inverse, lorsqu'une seule des bobines est vis-à-vis d'une fissure, une variation du signal différentiel est observée. De même, lorsque cette bobine quitte la zone de la fissure et que l'autre bobine apparaît devant la fissure, une nouvelle variation du signal différentiel est mesurée, résultant en une double boucle de signal au passage des deux bobines devant la fissure (cf. Figure 1.3(b)). De plus, un avantage de la sonde différentielle est qu'elle permet de limiter les bruits de mesure souvent causés par de légères variations d'orientation autour de la normale locale à la surface (suite à des tremblements de l'opérateur par exemple). Cependant, il est important pour ce type de sonde que l'alignement de ses deux bobines soit parallèle à la direction de son déplacement afin de maximiser la différence

entre les signaux des bobines au passage devant une fissure (cf. Figure 1.4).

Contrairement à l'inspection FPI, la technique ECT est restreinte uniquement aux surfaces qui conduisent le courant. De plus, dans le cas où une fissure est tangente aux lignes de courants de Foucault et que la direction de déplacement de la sonde est parallèle à la direction de la fissure, ces courants ne seront pas perturbés de manière significative au passage de la sonde et aucune variation d'impédance ne sera observée (cf. Figure 1.5). Par conséquent, si aucune information sur la direction principale des fissures n'est connue, il convient de balayer la surface dans deux directions perpendiculaires afin de maximiser les chances de détecter toutes les fissures. En revanche, la technique ECT présente une meilleure sensibilité pour la détection de défaut que FPI.

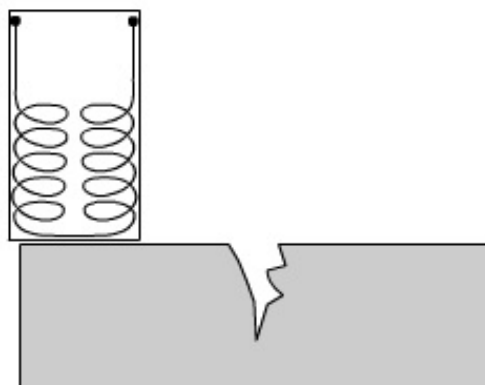
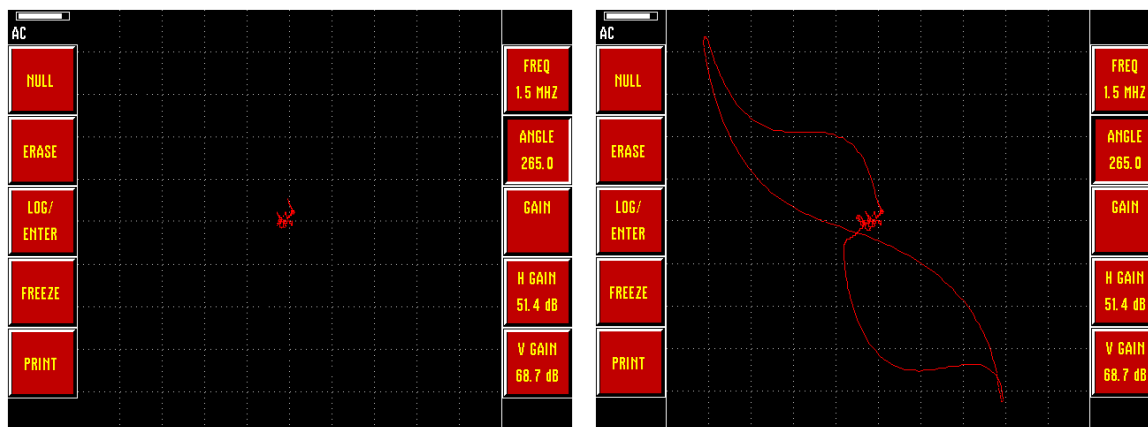


Figure 1.2 Schéma d'une sonde différentielle. Extrait de NDT Education Resource Center (2013c).

1.1.3 Inspection par thermographie infrarouge (IRT)

La troisième technique de CND considérée est la technique par thermographie infrarouge. D'une manière générale, la thermographie infrarouge consiste à mesurer la température en tout point d'une surface, ce qui permet également d'évaluer le flux de chaleur à travers celle-ci au cours du temps. Elle est utilisée dans des domaines variés tels que la surveillance, le secourisme (recherche de personnes), la médecine et l'ingénierie NDT Education Resource Center (2013b). Dans la branche de l'ingénierie, la thermographie trouve également plusieurs utilisations dont des opérations de CND, ce qui fait l'objet de sa présentation dans cette section. L'utilité de cette application pour le CND vient du fait que la présence d'une discontinuité dans la matière (fissures, vides, inclusions, etc.) perturbent le flux de chaleur au sein de celle-ci. Cette perturbation peut alors être mesurée et être analysée pour obtenir les caractéristiques du défaut (positions, dimensions, etc.).

Dans le cas de la détection de défauts, une technique d'application de la thermographie infrarouge consiste à perturber localement la température de la surface à inspecter, puis à analyser l'évolution de sa température au cours du temps à l'aide d'une caméra infrarouge. Il existe plu-



(a) Le signal reste centré sur l'écran lorsque la sonde inspecte une surface sans présence de défauts.

(b) Variation d'impédance observée au passage de la sonde devant une fissure.

Figure 1.3 Signal représenté dans le plan complexe (phase et amplitude) de la variation d'impédance d'une sonde EC différentielle. Le signal correspond à la variation complexe de l'impédance (résistance et réactance) en fonction du temps.

sieurs manières de perturber localement la température d'un matériau avec, entre autres, l'usage d'appareils optiques (e.g. une lampe produisant une lumière intense pulsée, appelée aussi lampe "flash"), d'appareils mécaniques (e.g. un traducteur à ultrason en contact avec la surface) ou encore électromagnétique (e.g. dissipation de courants de Foucault produits par une bobine placée devant la surface). Un ensemble d'équipements pour l'instrumentation est illustré Figure 1.6. Dans le cas d'une perturbation par une augmentation locale de la température, il est observé que la zone occupée par un défaut se refroidit plus lentement qu'une zone sans défauts. En effet, la présence d'un vide tel qu'une fissure aura tendance à ralentir le flux de chaleur au moment du refroidissement. Au fur et à mesure que la surface se refroidit, la cartographie révélera alors des défauts qui sont situés dans des zones (souvent isolées) de température relative élevée (cf. Figure 1.7). Dans le cas d'une inspection continue de la surface, un déplacement suffisamment lent de l'équipement de mesure devant cette surface est nécessaire pour laisser le temps à la chaleur de se propager dans le matériau.

1.1.4 Problème soulevé dans le projet

Dans l'industrie aéronautique, les techniques FPI et ECT sont encore principalement appliquées manuellement, ce qui présente plusieurs inconvénients :

- coûts induits et temps d'opération généralement élevés,
- pénibilité liées à ces tâches avec des positions parfois contraignantes dans des zones difficiles

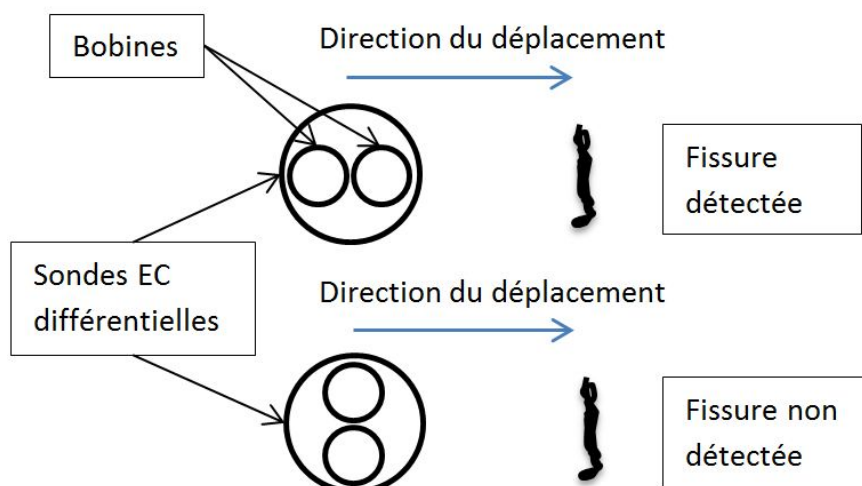


Figure 1.4 Influence de l'orientation de la sonde différentielle par rapport à la direction de son déplacement.

d'accès,

- exposition à des produits chimiques (tels que les produits utilisés pour la technique FPI),
- dépendamment de l'expérience et de la dextérité de chaque opérateur humain, la fiabilité, la précision et la répétabilité des résultats de l'inspection peuvent être affectés.

L'automatisation à l'aide de bras manipulateurs robotisés a permis de s'affranchir de tels inconvénients dans des domaines tels que l'application de peinture en spray sur des carrosseries de voitures ou le rivetage de structures d'avions. On peut donc aussi s'attendre à ce qu'une solution d'automatisation utilisant un bras robotisé à 6-DDL pour la manipulation des instruments des techniques ECT, FPI et IRT permettent de limiter ces mêmes inconvénients.

1.2 Objectif de la recherche

Dans le but d'améliorer les caractéristiques des résultats d'inspection et de s'affranchir des contraintes des tâches manuelles, l'objectif général du projet MANU-418, dans lequel s'inscrit le présent travail de recherche, est de démontrer que l'automatisation des techniques FPI, ECT et IRT à l'aide d'un bras manipulateur robotisé permet d'obtenir des résultats pour la détection de défauts en surface et légèrement sous la surface au moins aussi acceptables en termes de fiabilité et de répétabilité que ceux obtenus manuellement.

Pour remplir cet objectif général, plusieurs objectifs spécifiques sont identifiés :

1. Concevoir un système permettant à l'effecteur du robot de manipuler les instruments associés aux techniques ECT, FPI et IRT.
2. Planifier des trajectoires de couverture adaptées aux techniques de CND utilisées pour ins-

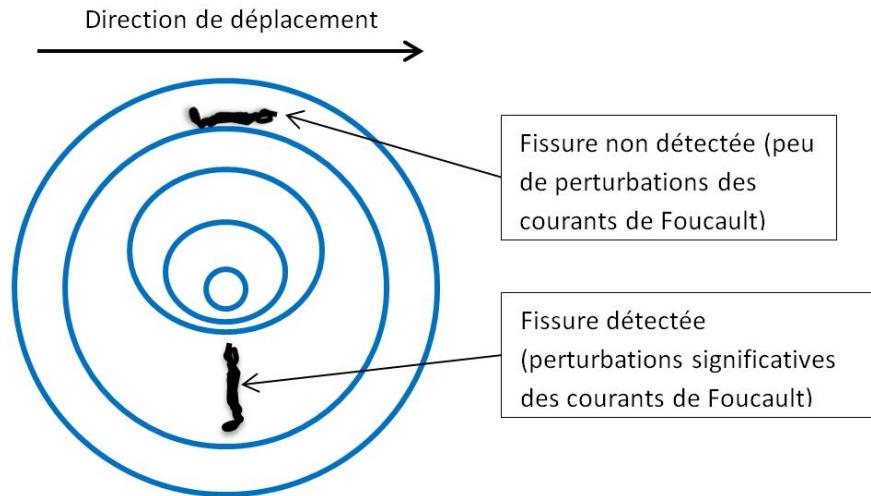


Figure 1.5 Perturbations des courants de Foucault selon l'orientation d'une fissure et la direction de déplacement de la sonde EC. Adapté de Buckley (2013).

pecter la surface des pièces à l'aide d'un bras manipulateur.

3. Interpréter automatiquement les informations provenant des instruments de CND utilisés afin de caractériser les défauts en surface ou légèrement sous la surface et décider si ces derniers sont acceptables ou non.
4. Réaliser un prototype de cellule robotisée intégrant les éléments cités précédemment et effectuer des tests expérimentaux.
5. Comparer les résultats d'inspection provenant des tests d'automatisation et ceux provenant d'opérations manuelles en termes de fiabilité, répétabilité, et de durée totale des opérations.

L'objectif du présent travail de recherche est de développer une méthodologie et un outil logiciel qui permettront de résoudre l'objectif 2, c'est-à-dire planifier des trajectoires de couverture adaptées à l'automatisation des techniques FPI, ECT et IRT à l'aide d'un bras manipulateur pour la détection de défauts en surface (et légèrement sous la surface) sur des pièces tridimensionnelles complexes issues de structures typiques de l'industrie aérospatiale telles que celle représentée Figure 1.8. La complexité des surfaces considérées est définie dans le paragraphe suivant.

1.3 Hypothèses de la recherche

Une hypothèse importante formulée dans cette recherche pour générer des chemins d'inspection à la surface d'une pièce tridimensionnelle est de supposer qu'un modèle CAO de la surface à inspecter est connu *a priori*. En d'autres termes, la géométrie de la surface est connue à l'avance. De plus, la complexité de la **géométrie** des surfaces considérées dans ce projet est caractérisée par un ensemble de sous-surfaces telles que :

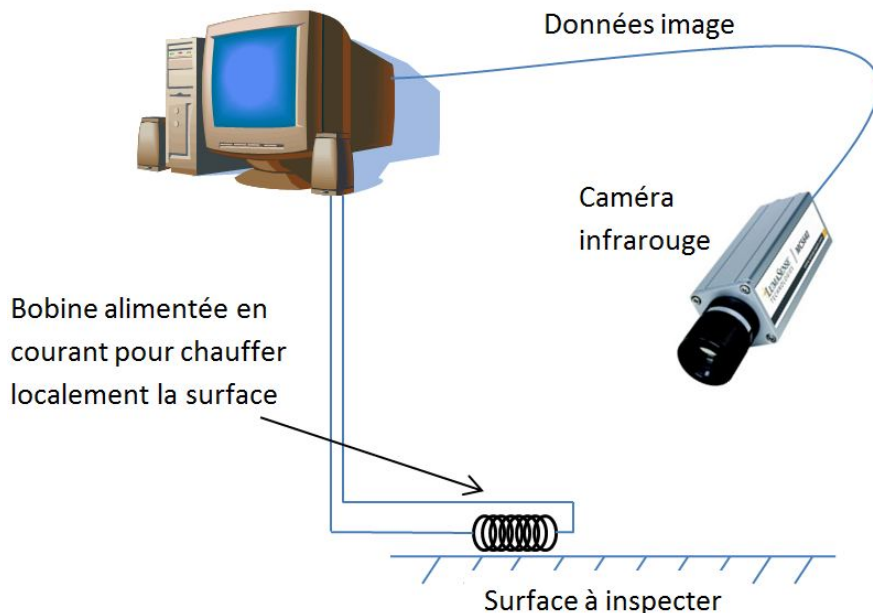


Figure 1.6 Exemple de système d’acquisition pour une inspection IRT avec une bobine comme dispositif de perturbation locale de la température de la surface. Adapté de Spicer et Osiander (2002).

- des surfaces planes ou relativement planes,
- des bords arrondis (conçus),
- des bords droits,
- des surfaces arrondies avec une géométrie régulière (proche d’une surface cylindrique).

Ces sous-surfaces sont également caractérisées par une **topologie** complexe. Afin de décrire plus précisément leurs propriétés de topologie, supposons que pour chacune de ces sous-surfaces il existe un plan dans lequel elles peuvent être projetées sans “recouvrements” (i.e. chaque point de la sous-surface projetée est l’image par projection d’un et un seul point de la sous-surface originale). Il est alors supposé que ces sous-surfaces projetées peuvent être éventuellement non-convexes¹ (i.e. concaves). Par exemple, celles-ci peuvent contenir des trous (cf. Figure 1.9).

Une autre hypothèse formulée dans ce projet est de supposer qu’une sonde de type différentielle (cf. Figure 1.2) est utilisée pour l’automatisation de la technique ECT. En effet, cette catégorie de sonde EC utilisée est particulièrement adaptée pour la détection de fissures. Il est important de mentionner cette hypothèse car les contraintes de déplacement de la sonde différentielle afin d’optimiser la mesure de ses signaux (cf. Figure 1.4) auront une influence directe sur le calcul de l’orientation de l’instrument le long du chemin (cf. Section 5.2).

1. Une surface S est dite convexe si pour tout couple de points $(A, B) \in S^2$, le segment $[AB]$ est inclus dans S .

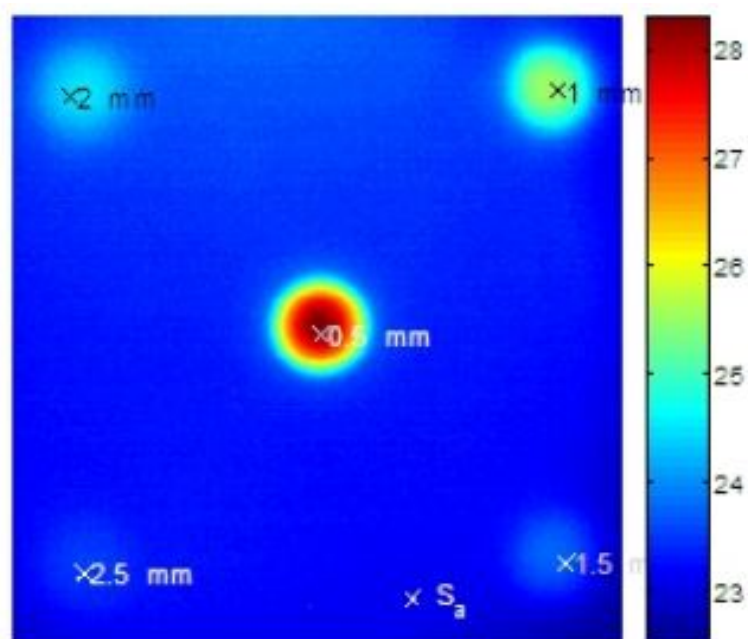


Figure 1.7 Cartographie de la température d'une plaque de calibration contenant des défauts artificiels révélés en rouge. La légende colorée est représentative de la température (du bleu au rouge respectivement pour les températures basses et élevées). Extrait de Castenado (2005).

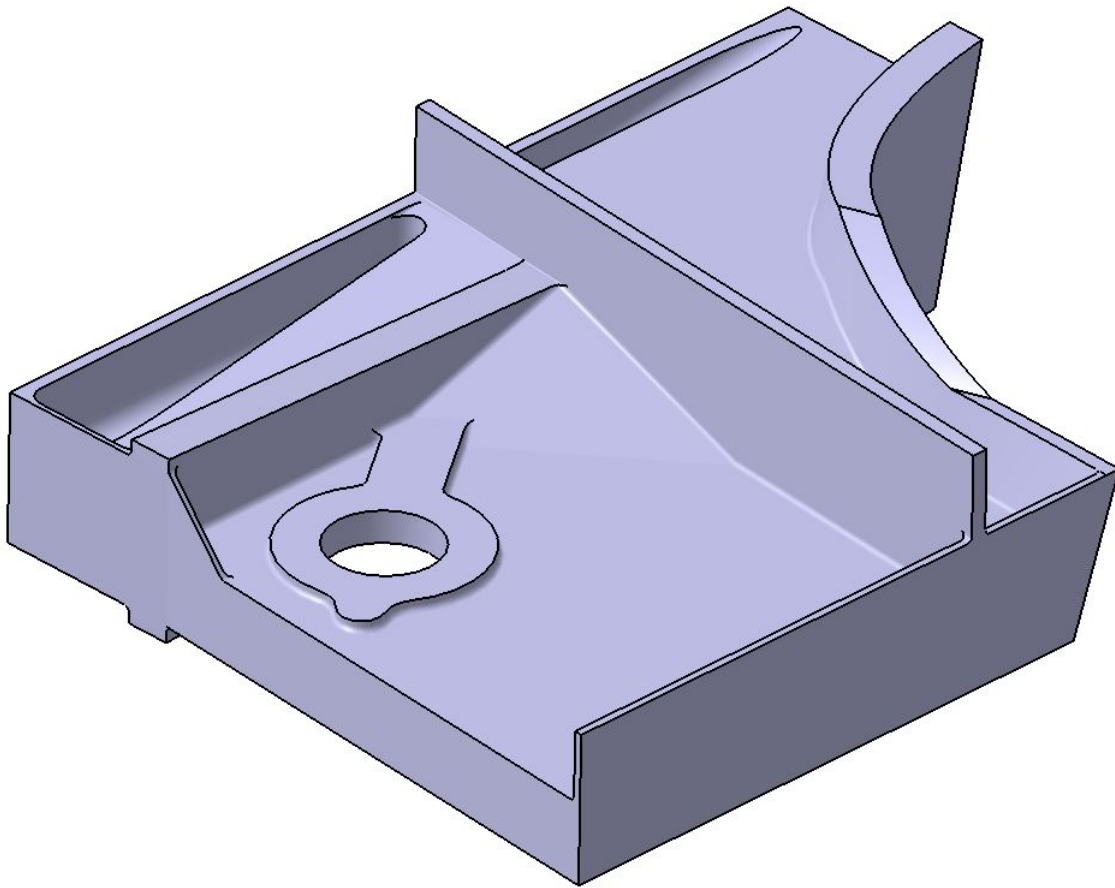


Figure 1.8 Pièce structurelle typique issue de l'industrie aérospatiale.

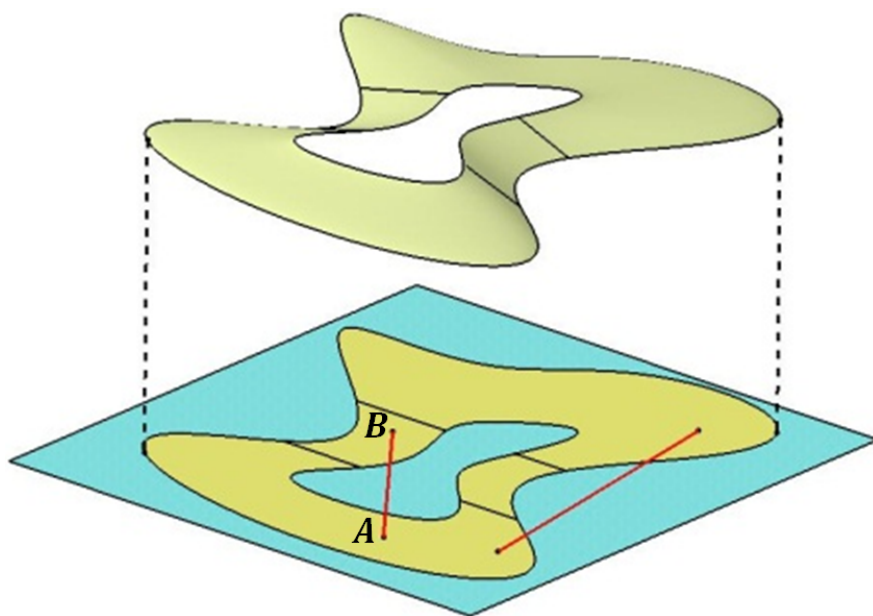


Figure 1.9 Surface relativement plane de topologie complexe. La surface projetée dans son plan moyen n'est pas convexe : il existe au moins un couple de points (A, B) de la surface projetée tel que le segment $[AB]$ ne soit pas inclus dans cette dernière.

CHAPITRE 2

REVUE CRITIQUE DE LA LITTÉRATURE

Comme décrit dans la Section 1.1.4, l'application manuelle des tâches de CND présentent plusieurs inconvénients. Un intérêt est donc de plus en plus croissant pour l'automatisation du CND afin de s'affranchir de ces inconvénients. Cependant, pour inspecter automatiquement la surface de pièces, il importe d'abord de s'intéresser au problème de génération de chemins de couverture afin de guider les instruments de CND. Dans le cas de la méthode FPI, il pourrait s'agir de générer des chemins pour l'application des produits (pénétrant, émulsifiant, eau et révélateur) à la surface de la pièce, ou pour l'inspection visuelle à l'aide d'une caméra sous éclairage UV. De même dans la méthode ECT, une sonde à courants de Foucault doit suivre des chemins de couverture en zig-zag tout en conservant un contact et en étant orienté normalement à la surface. Les paragraphes suivants présentent plusieurs systèmes existants pour l'automatisation des techniques ECT et FPI, et exposent différentes approches qui ont été proposées pour la génération de chemin dans des domaines d'application variés.

2.1 Systèmes automatisés existants de CND

2.1.1 Systèmes automatisés pour FPI

Des systèmes automatisés de ressuage par pénétrant fluorescent sont conçus selon un modèle de "chaîne de montage" Borucki et Jordan (1989). Plus précisément, un convoyeur achemine les pièces à inspecter à différents postes pour y subir les étapes du processus de contrôle par ressuage : préparation de la surface, application du pénétrant par immersion, prélavage à l'eau, application d'émulsifiant, rinçage à l'eau, séchage, et application du révélateur. Dans le cas de pièces de petites dimensions (rotors de turbine, éléments de roulements, éléments de train d'atterrissage, etc.), les produits tels que le pénétrant et l'émulsifiant (pour des pièces lavable à l'eau) sont appliqués sur celles-ci par immersion dans des réservoirs. Des systèmes automatisés utilisant des bains d'immersion pour l'application de FPI sont illustrés sur les Figures 2.1 et 2.2. Cependant, la taille des pièces à inspecter est contrainte par la capacité des bains d'immersion de chacun des postes de la chaîne. Par exemple, pour des pièces plus larges telles que des ailes d'avion, un système automatisé utilisant des bains d'immersion n'est plus envisageable. De plus, l'étape d'inspection visuelle sous la lumière UV est encore effectuée manuellement, cf. Adair *et al.* (1998) et PROCECO (2013).

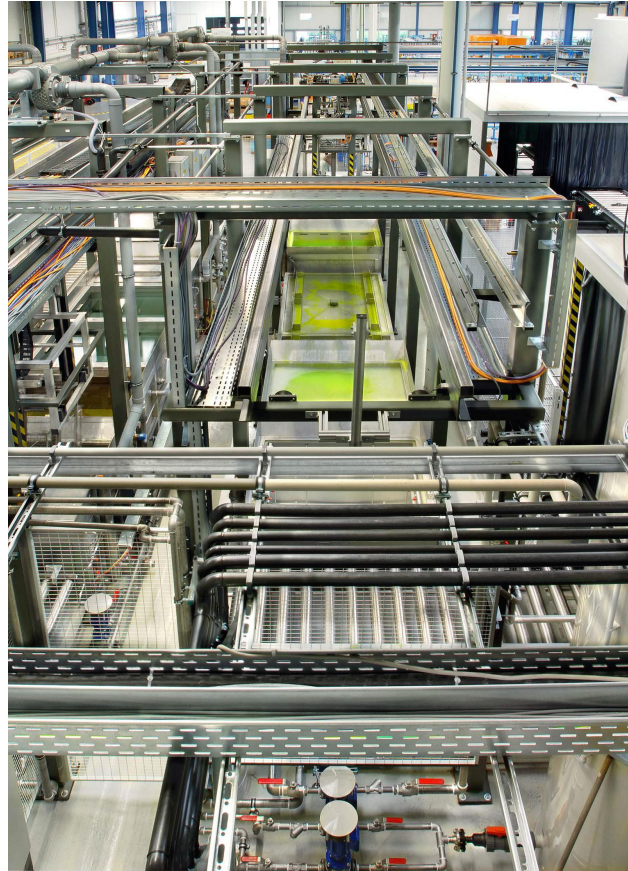


Figure 2.1 Système automatisé de ressuage par pénétrant fluorescent chez AIRBUS, extrait de Vetterlein *et al.* (2006).

2.1.2 Systèmes automatisés pour ECT

De même que pour la méthode FPI, des systèmes automatisés pour l'inspection par courants de Foucault ont été développés et commercialisés. Cependant, leur utilisation est souvent restreinte à des géométries particulières de pièces à inspecter, notamment des surfaces ayant un axe de symétrie (arbres moteur, bagues de roulements, tubes, etc.). Des pièces légèrement asymétriques telles que des arbres à cames (cf. Figure 2.3) peuvent aussi être inspectées avec ces systèmes Bangert (2009).

D'autres systèmes automatisés sont en cours de développement pour des surfaces complexes et de grandes dimensions. Plusieurs chercheurs ont étudiés des solutions utilisant des robots mobiles, équipés de sondes EC, qui se déplacent directement sur les surfaces à inspecter. Par exemple, un robot mobile a été conçu pour inspecter des aubes de turbine en se déplaçant le long de la périphérie du disque de la turbine Wu et Hsiao (2002). Ce système est toutefois dédié uniquement à l'inspection d'aubes montées sur un rotor de dimension spécifique. De même, l'équipe du centre de recherche en mécatronique, robotique et essais non destructifs de l'université London South Bank a développé de nombreux robots mobiles pour inspecter de grandes structures tout en s'adaptant

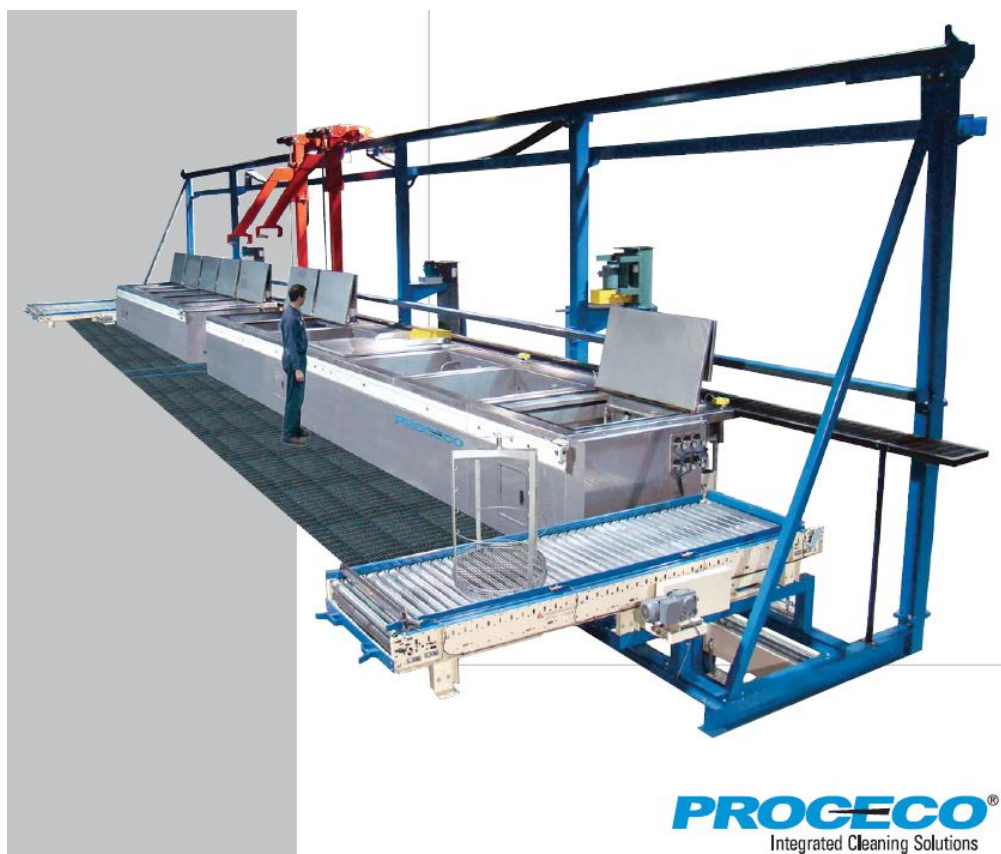


Figure 2.2 Système automatisé de ressuage par pénétrant fluorescent pour l'inspection de composants de moteur de turbine, extrait de PROCECO (2013).

à des zones dangereuses. Cependant, ces robots sont davantage adaptés pour se déplacer le long de surfaces de faible courbure (fuselages et ailes d'avions, coques de navires, réservoirs nucléaires sous pression, ...), à l'exception des oléoducs ou des rails de chemin de fer Sattar (2013). Enfin, des méthodes ont été proposées pour automatiser l'inspection par courants de Foucault en guidant les sondes EC à l'aide d'un bras manipulateur. Un exemple de système utilisant un bras manipulateur est le système *Eloscan*, développé par l'entreprise Rohmann GmbH (cf. Figure 2.4). Malgré la flexibilité d'utilisation que ce système peut proposer, la manipulation des sondes EC a été étudiée principalement pour l'inspection de pièces aéronautiques ayant un axe de symétrie (tels que des rotors de turbine). Dans Sattar et Brenner (2009), une solution pour l'inspection ECT sur des surfaces inconnues basée sur le contrôle en force du bras manipulateur est étudiée. Bien que cette solution puisse s'adapter à de nombreuses géométries de pièces, la vitesse d'inspection est limitée à $0,5 \text{ mm s}^{-1}$. Au-delà de cette vitesse, des pertes de contact entre la sonde et la surface peuvent survenir et entraîner alors des pertes de données.

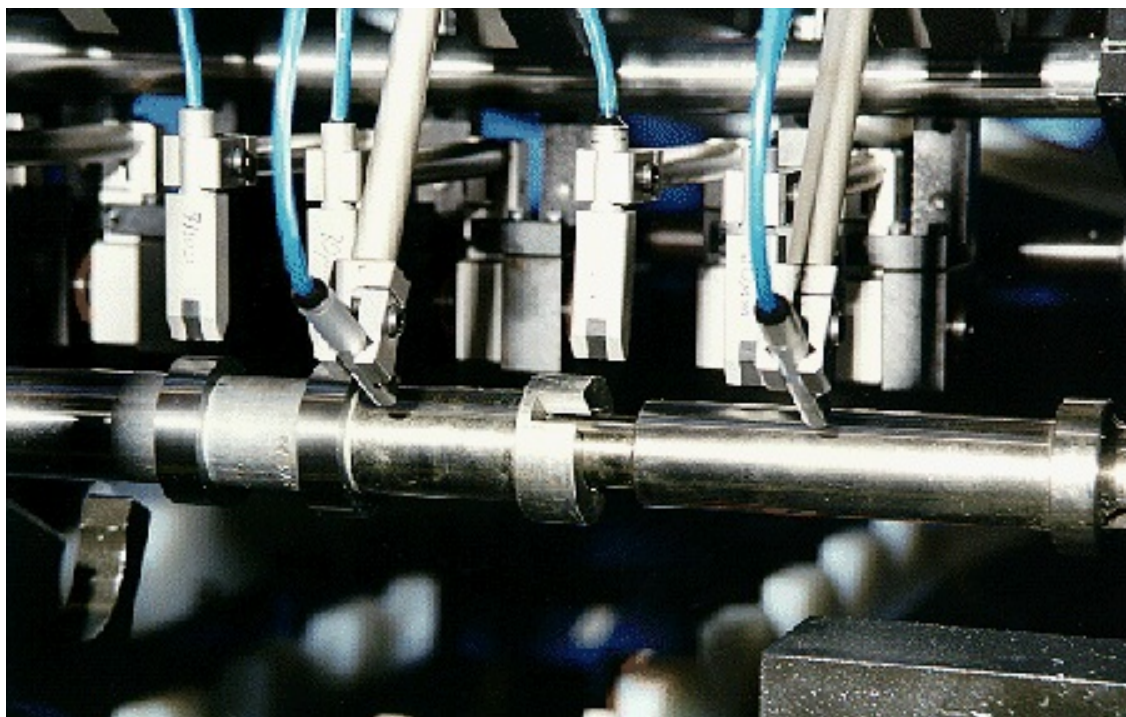


Figure 2.3 Système automatisé d’inspection par courants de Foucault sur un arbre à cames. MAR-POSS (2013).

2.1.3 Systèmes automatisés pour IRT

Plusieurs systèmes automatisés utilisant la thermographie infrarouge sont récemment à l’étude. D’ailleurs, des solutions utilisant des bras manipulateurs industriels pour déplacer des caméras infrarouges ainsi que des appareils optiques (lampe “flash”) qui permettent d’exciter thermiquement la surface d’inspection ont été proposées. Dans Luo *et al.* (2011), un logiciel nommé “Sentence” a été développé pour automatiser le processus complet d’inspection par la technique IRT (définition géométrique de la surface à inspecter, génération d’un chemin d’inspection, conversion de ce chemin en commandes robot, acquisition des images et identification des défauts et l’évaluation de leurs dimensions). Cependant, ce système a été conçu pour la détection de défauts résultant d’impacts dont la forme est généralement circulaire, et se restreint à des surfaces formées par balayage le long d’une courbe (“panel surfaces”). De plus, bien que ce système a prouvé son efficacité pour détecter des défauts circulaires d’au moins 5,5 mm de diamètre, la détection de défauts plus petits semblables à une fissure n’a pas encore testée. Un autre exemple de système est discuté dans Schmidt *et al.* (2012) pour automatiser l’inspection IRT et assurer la qualité de fabrication dans une ligne de production de structures d’avions de dernière génération faits de plastique à renfort fibre de carbone (CFRP). Toutefois, ce système est encore à l’étude au “DLR - Center for Lightweight-Production-Technology” à Augsburg.



Figure 2.4 Système automatisé d'inspection par courants de Foucault avec un bras manipulateur (Eloscan). Extrait de Rohmann GmbH (2008).

Un autre système déjà commercialisé (cf. Figure 2.5) permet une détection efficace de fissures de largeur de moins de 0,3 mm et s'intègre bien dans des lignes de production avec sa cadence rapide d'inspection Starman et Matz (2012). Cependant, ce système est principalement conçu pour l'inspection de billettes (barres d'acier de section carrée ou arrondie) qui seront utilisées par la suite pour la fabrication de pièces automobiles Starman et Matz (2012).

Des systèmes automatisés pour l'inspection IRT existent et font encore l'objet d'études en vue de les implanter dans des lignes de production. Toutefois, un système d'inspection IRT adapté pour la détection de fissures sur des surfaces tridimensionnelles complexes telles que celle présentée en Figure 1.8 n'a pas encore été développé.

Dans le paragraphe qui suit, plusieurs techniques de génération de chemins de couverture utilisées dans différents champs d'application sont présentées.



Figure 2.5 Système IRT automatisé (modèle DIO 4000 de l'entreprise STARMANS) pour la détection de fissures sur des barres d'acier (billettes) de section carrée ou arrondie. Extrait de STARMANS electronics Ltd. (2013).

2.2 Techniques de génération de chemins de couverture

2.2.1 Méthodes de programmation des robots industriels

Jusqu'au début des années 1990, la plupart des robots industriels étaient programmés "en ligne" (online programming) à l'aide d'outils d'apprentissage comme le pupitre de commande (teach- pendant features), permettant à l'opérateur d'apprendre au robot la tâche à réaliser. Cependant, si l'opérateur décide d'améliorer l'automatisation d'une tâche en modifiant le cycle d'un robot, cette méthode de programmation nécessite souvent l'arrêt de la production ou peut amener cet opérateur à intervenir dans des zones dangereuses où le robot travaille. Par la suite, la programmation hors-ligne (off-line programming) est devenue de plus en plus répandue grâce au développement de logiciels qui permettent de simuler et de visualiser des cellules robotisées virtuelles. On peut citer comme exemples les logiciels *MotoSim*, *ABB RobotStudio* ou *Delmia*. Les avantages de cette méthode par rapport à la programmation en ligne sont nombreux Pires (2007) :

- ne nécessite pas l'arrêt de la production,
- permet de simuler et d'optimiser le cycle du robot (en termes de rentabilité, de productivité et de sécurité) avant sa mise en application,
- permet d'envisager plus facilement d'autres stratégies de cycle du robot,
- permet de tester des logiciels développés pour la commande du robot.

Néanmoins, la programmation hors-ligne nécessite de calibrer les cellules robotisées virtuelles et réelles. Plus précisément, il s'agit de s'assurer que les référentiels associés à la cellule virtuelle correspondent à ceux associés à la cellule réelle afin que les trajectoires réelles du robot corres-

pondent aux trajectoires simulées.

Ainsi, avec cette dernière méthode de programmation, plusieurs techniques de génération de chemins de couverture ont été mises au point, chacune étant adaptée aux besoins de la tâche à réaliser. Les paragraphes suivants présentent différentes approches de génération de chemin de couverture, utilisées aussi bien pour des tâches de robots industriels (usinage numérique, dépôt de matière en spray) que pour des tâches dans d'autres champs d'applications (agriculture, nettoyage de sols, déminage, etc.).

2.2.2 Les motifs de chemins de couverture

Connaissant la zone couverte par un outil embarqué sur un robot, un chemin de couverture sur une surface donnée est défini comme un chemin que doit suivre l'outil pour que ce dernier recouvre la totalité de cette surface. À partir d'une telle définition, un grand nombre de chemins de couverture peuvent couvrir une surface donnée. Un chemin peut donc être optimisé en fonction de l'application que l'on souhaite en faire. Les critères d'optimalité sont par exemple la minimisation du temps du parcours Current et Schilling (1989) ou encore l'uniformité du recouvrement Antonio *et al.* (1997). Cependant, la recherche d'un tel chemin peut conduire à des problèmes d'optimisation difficiles à résoudre Antonio *et al.* (1997). Un moyen courant de simplifier ce problème est de formuler des hypothèses sur le chemin, notamment en décidant d'un choix du motif de recouvrement Antonio *et al.* (1997); Tam (1999); Chen *et al.* (2009).

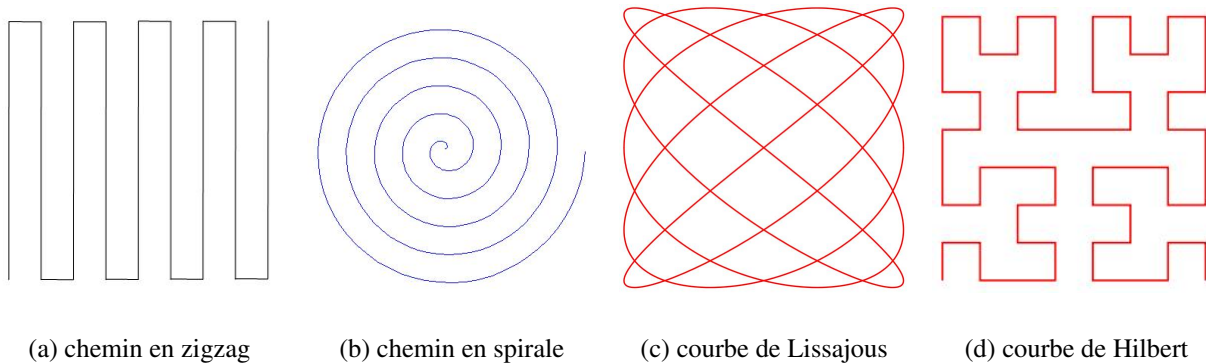


Figure 2.6 Exemples de motifs de chemin pour des applications de recouvrement.

Plusieurs motifs (Figure 2.6) ont été proposés comme par exemple des chemins en zigzag, en spirale, à base de courbes de Lissajous ou des courbes de remplissage (e.g. courbes de Hilbert, courbes de Peano, etc.). Chacun des motifs ont leurs avantages et inconvénients en fonction de la tâche que doit accomplir l'outil. Par exemple, en vue d'automatiser des procédés de moulage par projection, Sheng *et al.* (2005) ont comparé les performances des chemins en zigzag et en spirale

concernant l'uniformité de dépôt de matière à la frontière commune entre deux surfaces adjacentes d'un moule. Le chemin en spirale s'est révélé plus performant que le chemin en zigzag, bien que ce premier puisse conduire plus facilement à des interruptions temporaires du spray afin de respecter le critère d'uniformité du dépôt. De même, la courbe de Lissajous a présenté de bons résultats pour le polissage de surfaces relativement planes, mais peut difficilement respecter un retrait uniforme de matière pour des surfaces de géométrie plus complexe Tam (1999). Bien que les motifs de la Figure 2.6 soient représentés dans un plan, des techniques de génération de chemins basés sur ces motifs existent pour recouvrir des surfaces tridimensionnelles non nécessairement planes. En particulier, des techniques seront présentées à la Section 2.2.4 pour construire des chemins de couverture en zigzag sur des surfaces tridimensionnelles.

2.2.3 Approches avec ou sans connaissance de la géométrie de la surface

Les approches de génération de chemin varient également selon les connaissances *a priori* de la géométrie de la surface. En effet, de nombreuses méthodes ont été développées à partir du modèle CAO de la surface. Deux représentations du modèle CAO d'une surface sont principalement utilisées pour la génération de chemins de couverture : la **représentation paramétrique** (ou B-rep), et la **représentation à l'aide d'un maillage**, cf. Dragomatz et Mann (1997) et Chen *et al.* (2009). Tous deux présentent à nouveau bien sûr des avantages et des inconvénients pour la génération de chemins, dépendamment de l'application qu'on souhaite en faire. Par exemple, l'article Chen *et al.* (2009) recommande l'utilisation de la représentation à l'aide d'un maillage triangulaire pour des applications de peinture en spray car cette représentation est plus adaptée par rapport à la représentation paramétrique pour simuler le dépôt de peinture. De plus, la représentation paramétrique des surfaces de géométries complexes devient plus lourde car ces surfaces sont alors décomposées en de nombreuses petites sous-surfaces, chacune avec des paramètres différents, et autant de chemins indépendants seront générés. Néanmoins, la représentation paramétrique a l'avantage d'être précise et contient des données utiles telles que l'emplacement des arêtes vives ou des congés. À l'inverse, une surface représentée à l'aide d'un maillage a l'avantage d'être considérée comme une seule entité (composées de facettes dont la position, l'aire et la normale sont connues) ce qui ne contraint plus de générer des chemins sur de nombreuses sous-surfaces comme l'impose la représentation paramétrique dans le cas d'une géométrie complexe. En revanche, les informations sur les arêtes vives ou les congés ne sont plus disponibles. Par ailleurs, la surface est discrétisée et rend donc le modèle moins précis. Une solution pour limiter cette imprécision serait d'affiner le maillage, mais cela augmente alors la quantité de données à traiter. Une autre représentation géométrique d'une pièce qui a été utilisée pour la génération de chemin est la **géométrie de construction de solides** (ou CSG) Bobrow (1985). Cependant, bien que contenant plus d'informations sur la géométrie d'une pièce que la représentation paramétrique, la représentation CSG est restreinte à modéliser

uniquement des pièces formées à partir d'opérations booléennes sur un nombre limité de solides élémentaires (sphère, parallélépipède, cône, etc.).

D'autres méthodes ont également été développées pour générer des chemins de couverture sans connaissances *a priori* de la géométrie de la surface. Par exemple, un système utilisant des capteurs montés sur un bras manipulateur tels qu'un scanner laser et une caméra afin d'identifier des géométries spécifiques (des nervures et des cavités) et générer automatiquement leur chemin de couverture associé (enregistré dans une librairie de procédures) pour des applications de peinture en spray a été proposé Pichler *et al.* (2002). Cependant, cette approche ne permet pas de générer des chemins adaptés quelle que soit la géométrie et la topologie des surfaces. Aussi, en vue d'automatiser la technique de contrôle non destructif par courants de Foucault, une stratégie de génération de chemins basée sur le contrôle de force a été proposée. En utilisant un bras manipulateur équipé de capteurs de force et de couple, la sonde EC est alors asservie en force et position afin de rester perpendiculaire et appliquer une pression constante sur la surface tout en balayant cette dernière Sattar et Brenner (2009). Toutefois, cette étude a montré que la sonde EC a des difficultés à maintenir un contact avec la surface pour un déplacement à une vitesse supérieure à $0,5 \text{ mm s}^{-1}$, résultant alors en pertes ou bruits de données de mesure.

Dans la section suivante, des techniques de génération de chemins de couverture sont présentées. Toutefois, afin de ne pas étendre cette revue de la littérature à toutes les techniques existantes de génération de chemins de couverture, seules celles identifiées comme les plus pertinentes pour atteindre l'objectif de recherche sont considérées. Autrement dit, les techniques qui sont présentées se limitent à celles utilisant un motif de couverture en zigzag car ce motif est adapté pour les trois approches de CND considérées. Ces techniques supposent également la connaissance *a priori* d'un modèle CAO de la surface à inspecter et dont la complexité de géométrie et de topologie sont telles que décrit dans les hypothèses de la recherche (cf. Section 1.3).

2.2.4 Techniques de génération de chemins en zigzag sur des surfaces tridimensionnelles

Parmi les différentes techniques de construction de chemins en zigzag sur des surfaces tridimensionnelles, trois sont principalement utilisées dans plusieurs domaines d'application tels que la déposition de matériel en spray et l'usinage Atkar *et al.* (2003) et Misra *et al.* (2005). Ces trois techniques sont illustrées sur la Figure 2.7.

La première technique utilise des **plans de coupe** ("section planes" en anglais). Cette méthode consiste à intersecter la surface par des plans de coupe régulièrement espacés selon une direction (cf. Figure 2.7(a)). Cette direction constitue la direction de balayage du chemin en zigzag. Une méthodologie basée sur cette technique est développée dans Sheng *et al.* (2000) à partir du maillage triangulaire de la surface.

La deuxième technique utilise des **courbes décalées** ("offset curves" en anglais). Elle consiste

à choisir une courbe de départ sur la surface, puis à déduire ses courbes adjacentes en décalant la première courbe le long de ses courbes orthogonales (cf. Figure 2.7(b)). Plus précisément, la méthodologie proposée dans Atkar *et al.* (2005) afin de générer des chemins de couverture adaptés à des opérations de dépôt de peinture, consiste à rechercher une courbe de départ optimale sur la surface en se basant sur des critères d'uniformité du dépôt et de minimisation du temps de cycle du chemin final. Les courbes décalées sont alors calculées dans un premier temps en sélectionnant des points particuliers le long de la courbe de départ puis en calculant les courbes orthogonales (appelées aussi "courbes d'indexation" dans cette méthodologie) à celle-ci sur la surface passant par chacun de ses points particuliers. Dans un deuxième temps, ces points particuliers sur la courbe de départ sont décalés d'une distance (optimale également en termes de temps de cycle et d'uniformité du dépôt) définie le long de leur courbe d'indexation associée. Les points ainsi décalés forment la courbe décalée à la courbe de départ, et les mêmes opérations sont effectuées pour déduire les passes adjacentes suivantes.

Enfin, la troisième technique utilise des **courbes isoparamétriques** ("isoparametric curves" en anglais). Les passes du chemin sont d'abord générés dans le plan paramétrique (u, v) le long d'un même paramètre, puis sont ensuite utilisées comme paramètres sur la surface (supposée paramétrique) pour obtenir les courbes isoparamétriques correspondantes. Ces courbes correspondent ainsi aux passes du chemin résultant sur la surface (cf. Figure 2.7(c)). Une méthodologie basée sur cette technique est brièvement rappelée dans Misra *et al.* (2005).

Bien que ces techniques permettent de construire des chemins de couverture sur des surfaces relativement planes, elles peuvent parfois conduire à des surfaces de géométrie plus complexe à des chemins inutilisables Atkar *et al.* (2009). Pour pouvoir appliquer ces techniques de construction de chemins sur des surfaces de géométrie et de topologie plus complexes que celles présentées sur la Figure 2.7, des étapes de segmentations successives ont été proposées pour représenter une surface complexe en plusieurs sous-surfaces de géométrie et de topologie plus simples Sheng *et al.* (2005); Atkar *et al.* (2009). Il devient alors plus aisé d'appliquer ces techniques de construction de chemins sur chacune de ces sous-surfaces.

Ces techniques ont déjà été utilisées dans plusieurs domaines d'application (déposition de matériel en spray, usinage, agriculture, etc.), mais celles-ci n'ont pas encore été testées pour des applications de CND telles que FPI, ECT ou IRT. De plus, ces techniques de génération de chemins sont adaptées principalement pour optimiser l'uniformité de recouvrement, minimiser le gaspillage de matière (dans le cas d'une application de déposition de matière), et minimiser le temps de cycle Atkar *et al.* (2009); Sheng *et al.* (2005); Feng et Li (2002), ce qui n'est pas considéré comme une priorité pour ces applications de CND.

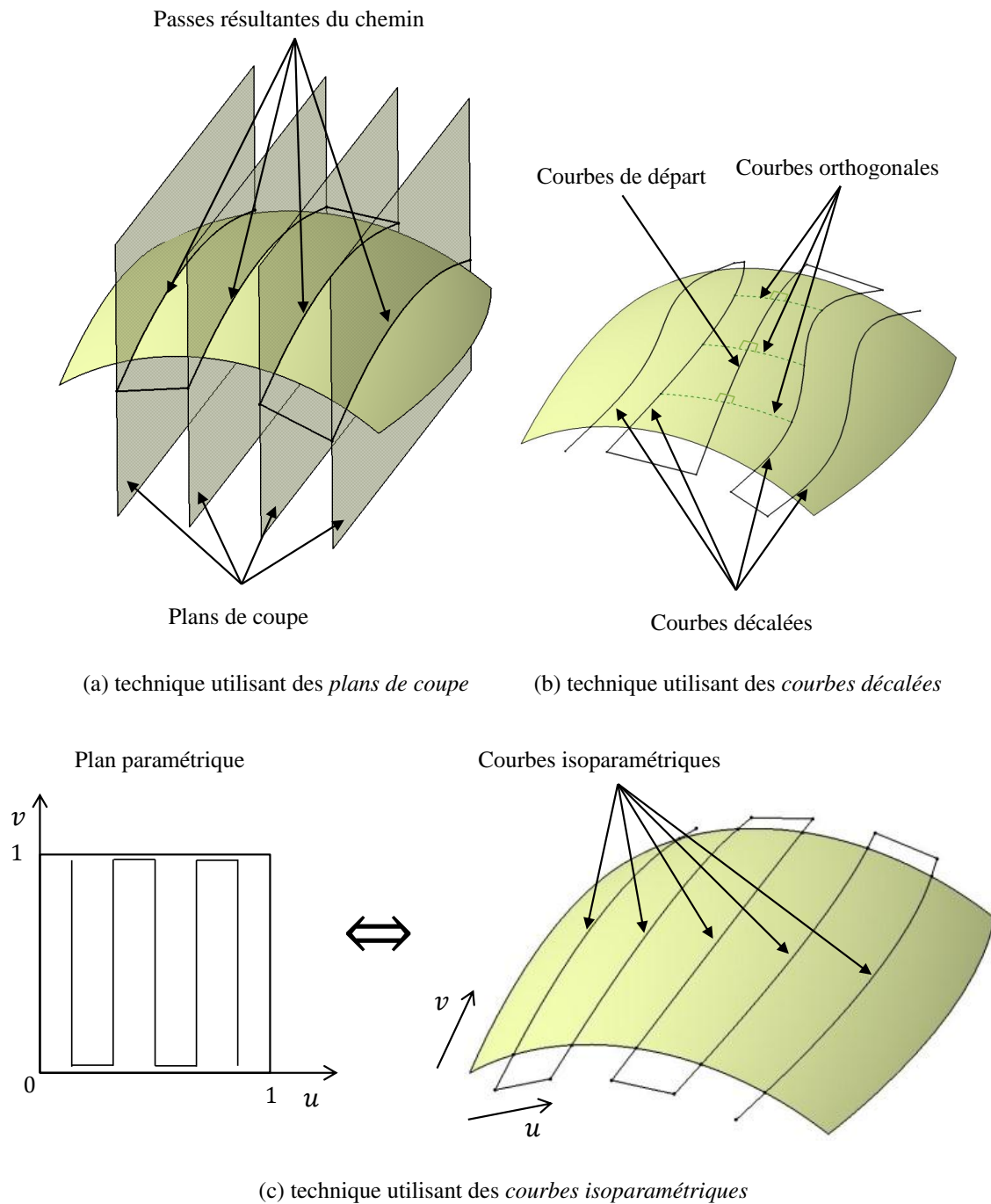


Figure 2.7 Trois techniques de génération de chemin en zigzag sur une surface 3D, adapté de Atkar *et al.* (2003) et Misra *et al.* (2005).

CHAPITRE 3

DÉMARCHE DE TRAVAIL

Avant d'étudier la manière de générer des chemins de couverture adaptés pour les techniques ECT, FPI et IRT, plusieurs choix ont d'abord été faits. En premier lieu, un motif de balayage en zigzag est utilisé car il constitue le motif le plus adapté aux techniques de CND considérées, et plus particulièrement pour l'inspection ECT. Comme discuté dans la Section 2.2.2 le choix du chemin de couverture dépend de l'application qu'on souhaite en faire. Par exemple, Sheng *et al.* (2005) ont proposé un chemin de couverture en spirale sur certaines surfaces dans le but d'améliorer l'uniformité de recouvrement. Dans le cas présent, l'intérêt d'utiliser le motif de chemin de couverture en zigzag pour ECT est que l'on assure que chaque point de la surface a été inspecté selon une même direction de balayage. Ainsi, il devient plus simple d'inspecter la surface dans la direction perpendiculaire et maximiser les chances de détecter toutes les fissures (cf. Section 1.1.2). Ensuite, des paramètres sont définis afin de modifier le chemin et l'adapter pour chaque situation (dimensions de la zone occupée par la sonde en contact avec la surface, résolution souhaitée pour l'inspection, prévention de collisions, etc.). De plus, la représentation du modèle CAO de la surface qui est considérée est celle utilisant un maillage triangulaire et le format de fichier de ce modèle est le format STL (pour STéréoLithographie). Comme décrit dans la Section 2.2.3, plusieurs approches de génération de chemins ont été proposées à partir de cette représentation et celle-ci n'impose pas une partition prédéfinie en plusieurs sous-surfaces comparativement à la représentation paramétrique. La représentation de la surface à l'aide d'un maillage permet alors plus de liberté pour la partitionner en plusieurs sous-surfaces par regroupement des facettes triangulaires, comme il sera vu au Chapitre 4. Enfin, l'automatisation de ECT a été considérée prioritaire devant les techniques FPI et IRT dans le cadre du projet MANU-418. La méthodologie développée se base alors sur les propriétés de chemin qu'une sonde différentielle à courants de Foucault (appelée ici sonde EC) suit habituellement lors d'une inspection manuelle, tout en permettant de rester utilisable pour les techniques FPI et IRT.

À partir de ces considérations et des hypothèses de recherche formulées à la Section 1.3, la technique de construction de chemins retenue est celle utilisant des plans de coupe (cf. Figure 2.7(a)), en raison de sa simplicité d'implémentation par rapport à l'autre méthode utilisant des courbes décalées. Comme il a été vu à la Section 2.2.4, cette technique fonctionne bien pour des surfaces relativement planes, mais ne convient plus pour des surfaces plus complexes telles que celle représentée sur la Figure 1.8. Pour faire face à cette difficulté, une stratégie "diviser pour régner" ("divide and conquer" en anglais) est utilisée. D'une manière générale, cette stratégie consiste à partitionner

un problème complexe en sous-problèmes plus simples. Dans le cas présent, cette stratégie sera employée pour partitionner la surface globale à inspecter en sous-surfaces ayant une géométrie et une topologie plus simples dans le but de pouvoir générer des chemins cohérents avec la méthode par plans de coupe. Adaptée des travaux de génération de chemins de couverture pour l'application de peinture en spray Sheng *et al.* (2005); Atkar *et al.* (2009), cette stratégie “diviser pour régner” est mise en pratique dans le présent travail de recherche par l'application successive de trois techniques de segmentation sur la surface triangulée pour la diviser en sous-surfaces de géométrie et topologie plus simples. Une fois la surface segmentée, les chemins sont alors construits sur chacune des sous-surfaces avec la méthode utilisant des plans de coupe. Ces chemins ne donnant jusqu'ici que les consignes de position de l'extrémité de l'instrument manipulé, l'orientation de ce dernier le long des chemins sera calculée par la suite. À des fins de vérification de la méthodologie proposée pour générer des chemins, des simulations et des tests expérimentaux seront menés.

L'article intitulé “Coverage path planning for eddy current inspection on complex aeronautical parts” inclus dans le Chapitre 4, détaille cette méthodologie de génération de chemin, incluant les étapes de segmentation, la construction des chemins par la méthode utilisant des plans de coupe, ainsi que des résultats sur une partie de la surface de la pièce représentée à la Figure 1.8. Cet article a été soumis au journal *Robotics and Computer Integrated Manufacturing* le 18 janvier 2013. Le Chapitre 5 apporte des éléments complémentaires à cette étude, tels que la méthode utilisée pour le calcul de l'orientation de l'outil en chacun des points du chemin, une autre méthodologie de génération de chemin de couverture adaptée pour l'inspection autour d'une indication dont la position et l'orientation sont connues à l'avance, une présentation de l'interface utilisateur du programme développé pour la génération de chemins, ainsi que des expérimentations sur un bras manipulateur en passant par des simulations sous le logiciel MotoSim.

CHAPITRE 4

ARTICLE : COVERAGE PATH PLANNING FOR EDDY CURRENT INSPECTION ON COMPLEX AERONAUTICAL PARTS

P. Olivieri, L. Birglen, X. Maldague, I. Mantegh

4.1 Abstract

The quality of aeronautical parts mainly relies on non-destructive testing (NDT) and eddy current (EC) testing is one of the most used NDT techniques in the aerospace industry. However, EC testing is still mainly performed by human operators and reliability as well as repeatability is not always guaranteed. To solve these issues, automating this NDT technique with a robotic system is investigated. In this paper, an EC probe equipped with a passive compliant system is assumed to be attached to the end-effector of a 6-DOF manipulator arm to carry on the inspection. Then, assuming that a 3D model of the inspected part is known a priori, a coverage path planning method using a zigzag (or rastering) pattern adapted to EC testing on aeronautical structures with a complex geometry is proposed. To reach this objective, the approach adopted in this work is to adapt existing coverage path planning techniques based on a “divide-and-conquer” strategy used for spraying applications to EC inspection. More precisely, three successive segmentations are applied to the surface to be inspected so that consistent rastering paths can be generated. Simulation results are shown for a complex part of an aeronautical structure to demonstrate the efficiency of the proposed method.

4.2 Introduction

In the aerospace industry, non-destructive inspections are critical tasks used to ensure the quality of manufactured parts. However, most of non-destructive testing techniques commonly used in this industry are currently performed by human operators. In addition to being repetitive and time-consuming, results heavily depend on each inspector’s skill which significantly impacts the reliability and repeatability of these inspections. To address these issues, automating NDT as much as possible is the most plausible solution.

Many NDT techniques exist but one of the most frequently used to inspect aeronautical parts is eddy current testing. This technique requires conductive materials and consists in moving a probe constituted of electrical coils over the inspected surfaces (usually in mild contact) in order to

generate eddy currents in the latter by a magnetic field. Thus, perturbations of this field result in an electric signal which can be used to detect surface and near-surface cracks. In an effort to automate this technique, it is proposed in this paper to handle the EC probe with a 6-DOF manipulator arm using an offline programming motion. Moreover, the EC probe is assumed spring-loaded along its main axis to deal with slight uncertainties in geometry and positioning. Then, to deploy such system in practice, it is first necessary to solve the *coverage path planning* problem in order to inspect all surfaces of interest without missing any spot.

This problem has been studied for many years, particularly for numerical control (NC) machining Dragomatz et Mann (1997) and material deposition applications such as spray painting Suh *et al.* (1991); Asakawa et Takeuchi (1997), thermal spraying Gadow *et al.* (2010) and spray forming Sheng *et al.* (2005). Coverage path planning is also a well known issue in other applications like cleaning De Carvalho *et al.* (1997), arable farming Jin et Tang (2011), demining Acar *et al.* (2003), etc.

Until now, few works have been led to automate EC inspection. Most of the existing automatic systems for this application are dedicated to a specific range of parts, especially these with a symmetrical axis (shafts, tubes, bearings). When a part has a more complex shape, a dedicated automatic system is often designed to inspect this particular shape Bangert (2009). In one of the most recent works on this topic, a 7-DOF robotic arm using position-force control strategies has been proposed to perform an EC inspection on surfaces with unknown geometry Sattar et Brenner (2009). However, though detection of defects are satisfying at low scanning speeds, they observed that the EC probe has difficulties to keep a contact with the surface at speeds greater than 0.5 mm s^{-1} , resulting in losses of data and, as a result, defects might go undetected. Moreover, though results are better than manual inspections under this speed, false alarms are still present. These limitations show that improvements still need to be brought.

In order to generate paths for EC inspection, it is proposed in this paper to adapt existing coverage path planning techniques for spraying applications. Indeed, paths for the latter applications present additional constraints compared with EC inspection (e.g. uniformity of material deposition, minimization of waste) but they are based on similarly usable zigzag patterns. Moreover, many studies have been published about path planning for spraying applications. Among these, several have proposed a framework to generate paths that, in addition to meeting coating constraints (uniformity, minimization of material waste), can also deal with complex 3D surfaces while reducing cycle-time process Sheng *et al.* (2005); Bi et Lang (2007); Atkar *et al.* (2009). Therefore, it is believed that paths for EC inspection can benefit the experience acquired throughout these works. In the next section, it is explained how some of the aforementioned frameworks have been adapted to EC inspection.

4.3 Outline of the proposed method

To automatically generate an inspection path, it is first assumed that the geometry of the surface to inspect is known, e.g. from a CAD model. Different approaches to compute coverage paths using such a model have been proposed in the literature. For instance, Tam (1999) has worked on coverage path planning using known parametric surfaces. Other authors have generated paths based on a triangular approximation of the surface geometry using the STL file format obtained from the CAD model. In Atkar *et al.* (2003) has been identified two methods to generate zigzag paths on a known surface: one using section planes, and another using offset curves, cf. Fig. 4.1 (adapted from Atkar *et al.* (2003)). In Sheng *et al.* (2000), section planes have also been chosen to construct the coverage paths. Although these methods work well for flat or relatively flat surfaces, problems arise for more complicated 3D shapes (e.g. surfaces with high curvatures or with holes). To address such limitations, some authors used a “divide-and-conquer” strategy: the global surface is partitioned into simpler patches so that the simpler methods to generate paths can be applied Atkar *et al.* (2009); Sheng *et al.* (2005).

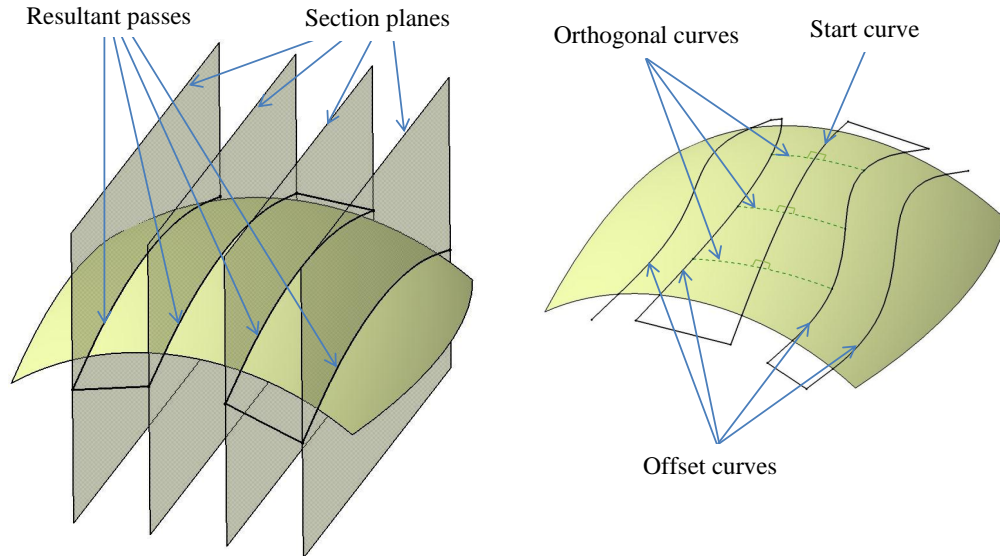


Figure 4.1 Construction of paths with section planes (left) and with offset curves (right), adapted from Atkar *et al.* (2003).

Inspired by these works, the general framework to generate zigzag paths for EC inspection as proposed in this paper is to first, partitioning the global STL surface into patches with three successive segmentation techniques based on surface geometry and topology, and then, generating paths on each patch by using the “section planes” method described in Section 4.7. This framework is illustrated in Fig. 4.2. Nevertheless, the main difference between coverage for material deposition

applications and inspection is that the latter does not need to meet uniformity constraints. Hence, some steps developed in Sheng *et al.* (2005) and Atkar *et al.* (2009) are adapted in order to build a complete coverage of the EC probe on complex 3D surfaces, without adding complexity to algorithms due to uniformity constraint. Furthermore, some improvements are proposed in this paper to extend the boustrophedon algorithm Choset et Pignon (1997), used to segment a 2D surface with convex holes inside based on topology, to more general types of surfaces (cf. Subsection 4.6.3).

The first segmentation, referred to as “watershed segmentation,” aims at partitioning the surface into regions separated by borders presenting local maxima of curvature (cf. Section 4.4). Thus, the new regions are expected to contain less variations of curvature, leading to more regular shapes and subsequently, more regular paths. However, based on variations of curvature, this segmentation does not guarantee a low absolute curvature inside these regions (for example, the watershed segmentation has no effects on a cylinder, whose curvature is constant everywhere), and some regions may not be flat enough to generate paths using section planes (cf. Section 4.7). Then, to ensure proper flatness, an additional segmentation based on normal directions of each triangular facet is applied to the previous regions to obtain a new segmentation with a maximum deviation angle (angle between two facets) lower than a fixed threshold value in each new region (cf. Section 4.5). These two successive segmentations ensure the partition of the surfaces into geometrically simple patches. However, patches with a complex topology (for example patches with holes) can still remain and result in paths leading to a long cycle-time or collisions between the EC probe and the part (cf. Subsection 4.6.1). Therefore, a final segmentation, called “Morse decomposition,” is applied to the patches to get topologically (and geometrically) simple subsurfaces (cf. Section 4.6).

The following sections detail these three segmentations, the path generation method, and the results obtained when applied to a structural part of an aircraft with a relatively complex shape.

4.4 Segmentation based on curvature: watershed segmentation

Most of the parts to be inspected are constituted of flat or almost flat surfaces connected by straight or round edges (fillets). In order to get regular paths, the whole surface must be partitioned into geometrically simple patches. Three methods have been experimented to this aim.

The first one is a “growth” method that consists in choosing at first a seed triangle to start a new patch (usually the triangle with the biggest area that is not yet attached to a patch). Then, each of its neighbor triangles are tested and attached or not to the patch according to the value of its *deviation angle* with its seed triangle. The deviation angle of a neighbor, noted θ , is defined as the tilt angle between its normal and that of the seed triangle. A neighbor triangle is accepted into the patch if its deviation angle is lower than a threshold value θ_{thresh} . Then, considering the accepted triangles (if any) as new seeds of the patch, the same computations are performed recursively on their respective

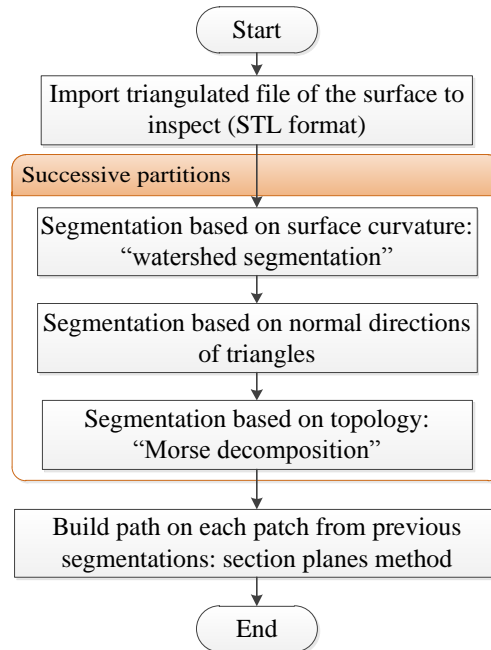


Figure 4.2 Flowchart of the method.

neighbors to accept them or not in the patch. This is repeated until no more neighbor triangles are accepted. Although this algorithm performs well with surfaces with straight edges, it gives poor results with surfaces limited by or including rounded edges.

The second method is nearly identical to the first one with the only difference being that, instead of computing the deviation angle of a neighbor triangle with its associated seed, it is now always computed with the same triangle, namely the original seed triangle of the patch (i.e., the first triangle that has created the patch). Contrary to the first method, the second one results generally in a valid segmentation on surfaces with rounded edges. However, it has been noticed by the authors during several tests that this method often leads to small isolated patches at an early stage as the threshold deviation is decreased. Also, like the previous segmentation method, results are dependent of the choice of the seed triangle starting the algorithm.

The two above methods have been proposed and tested by the authors but have shown issues to reach a satisfying segmentation of surfaces of interest. Finally, a third method exists, known as the “watershed segmentation.” Proposed in Atkar *et al.* (2009) to segment a surface with a complex geometry for spray painting applications, the idea was to build patches such that high curvature variations are located at the patch boundaries and consequently, keep lower variations inside. Indeed, resultant paths generated on such patches are expected to have low curvature variations, improving uniformity of paint, as well as reducing broad motions and large accelerations of

the robotic arm handling the spray gun. Although an automated inspection with an EC probe does not need to respect uniformity constraints, it can also benefit from regular paths avoiding large tool orientation changes since the latter is highly detrimental to the EC signal integrity.

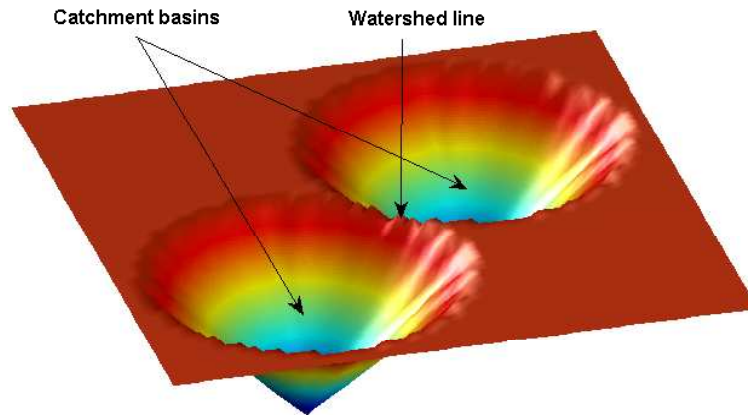


Figure 4.3 Principle of the watershed segmentation, adapted from Eddins (2002).

The watershed segmentation is originally an image processing technique to segment images into regions sharing similar properties. To illustrate its principle, let us imagine that each pixel is represented in three dimensions such that its color corresponds to an altitude computed by a *height function*. The map of this abstract representation is then composed of “catchment basins” separated by watershed lines (cf. Fig. 4.3). The analogy with hydrology is that a pixel will be associated to a region the same way as a drop of water will flow in its catchment basin. Generalized to 3D surface meshes Mangan et Whitaker (1999), the algorithm of watershed segmentation can be summarized as follows:

1. Evaluate the curvature (to which corresponds the height function) at each vertex of the mesh. As proposed in Mangan et Whitaker (1999), the curvature at each vertex is estimated by the norm of the covariance matrix of the normals attached to the neighbor triangles of the vertex.
2. Find vertices at a local minimum of curvature (i.e. vertices whose curvature is lower than that of all of their neighbors) and assign a label for each of these to identify new regions.
3. Find “flat” areas (i.e. groups of vertices with a *constant* curvature). Two types of flat areas are considered either as *flat minimum* when the curvature is a local minimum, or *flat plateaus* for the remaining flat areas that are not local minima of the curvature. Vertices at each flat minimum are also labeled and define a new region.
4. Descend all flat plateaus: find among adjacent vertices of each plateau that of lowest curvature and descend from this vertex to connected neighbors along the steepest descent, until a labeled vertex is encountered. Then, the same label is attributed to the vertices of the plateau

- (including those met on the way of steepest descent): the plateau is associated to the same region containing the encountered vertex.
5. Descend each remaining vertices until they encounter a labeled vertex. The same label is attributed to all vertices met through the way of steepest descent.
 6. Compute for each region previously obtained a measure called *watershed depth*. It is defined as the difference between the global minimum of curvature of a considered region and the minimum curvature among all vertices located at the region border. In other words, by representing each region as a catchment basin whose elevation at each point is defined by the corresponding curvature and assuming that one fills this virtual basin with water, one can imagine the watershed depth of this region as the maximum depth beyond which the water overflows.
 7. Merge each region whose watershed depth is lower than a threshold value (noted $depth_{thresh}$) with a neighbor region that shares the boundary vertex of the former with a minimum curvature.

The elements introduced in the watershed algorithm are illustrated in 2D in Fig. 4.4. In this figure, considering that the watershed depth of region 2 is lower than the fixed threshold value, region 2 will be merged with region 1. The interest of this last merging step is to limit the formation of too many small regions, which often occurs after having identified all regions associated to a local minimum.

Finally, after implementing this segmentation, results are satisfying for both types of surfaces considered before (with straight or rounded edges). Additionally, the formation of patches does not depend on the choice of a seed triangle anymore, which gives to watershed segmentation a less random nature. This latter segmentation method based on curvature is subsequently adopted in this work. However, a limitation of this method is that the segmentation is based on curvature variations. Thus, absolute curvature and the global geometry of resultant patches are not controlled. For instance, the watershed segmentation is useless with a cylinder because the curvature is the same for all vertices. These particular surfaces with a constant curvature remain an issue as it could lead to a discontinuous path with the path generation method using section planes (as explained later in Section 4.7), which is the objective of the next technique detailed below.

4.5 Segmentation based on surface normals

Very similar to the two first methods discussed in the previous section, this segmentation ensures that patches have a minimal flatness such that they can be cut by section planes to generate paths. The method consists in growing regions of a patch previously obtained with the watershed

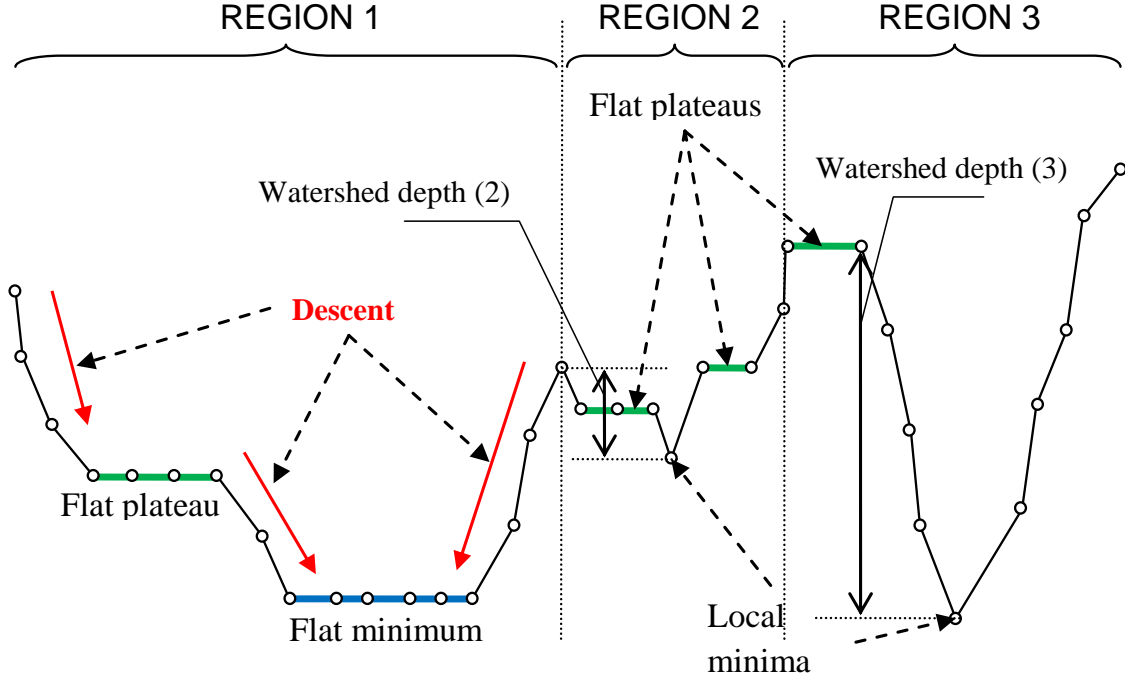


Figure 4.4 Illustration of the watershed segmentation algorithm.

segmentation by starting from a seed triangle, and accepting each neighbor triangle whose maximum deviation angle (among all deviation angles formed between its normal and these of triangles belonging to the current growing region) is lower than a threshold value, noted θ_{thresh} . Calling $\theta(T_k, T_i)$ the deviation angle between the triangles T_k and T_i , a neighbor triangle T_k is accepted in the patch when the following condition Eq. (4.1) is respected:

$$\max_{T_i \in patch} \theta(T_k, T_i) \leq \theta_{thresh}, \quad (4.1)$$

The result of this segmentation on a cylinder is shown in Fig. 4.5, with a threshold value $\theta_{thresh} = 120^\circ$.

Thus, by applying successively the watershed segmentation and this segmentation based on normal directions, resultant patches of the whole surface are ensured to be “geometrically simple” in a sense that patches do not contain large variations of curvature, and are flat enough such that the method of path generation by section planes can be applied without resulting into discontinuous paths (cf. Section 4.7). However, these segmentations do not deal with patches with a more complex topology, resulting sometimes in unexpected paths on these particular patches. A final segmentation is therefore performed to address this issue, namely the “Morse decomposition.”

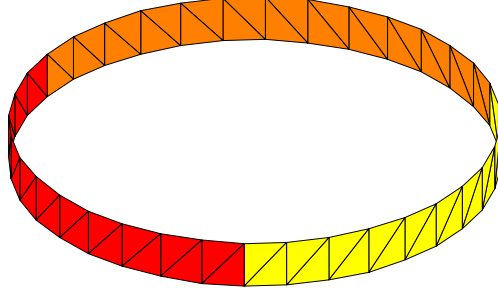


Figure 4.5 Three patches formed using the segmentation based on surface normals, $\theta_{thresh} = 120^\circ$.

4.6 Segmentation based on topology: the Morse decomposition

4.6.1 Rationale

After applying the previous segmentations, topologically complex patches can still remain inappropriately separated (e.g. patches with holes). Keeping such patches in this state can reduce the quality of the coverage path. Indeed, in material deposition applications, passing over holes increase the waste of material and increase cycle-time Atkar *et al.* (2009). In our case, with an EC probe in contact with surfaces, these particular patches are even more critical. It can not only extend cycle-time process of inspection (Fig. 4.6), but also result in collisions (Fig. 4.7). Indeed, holes in a patch can remain after the watershed segmentation, as shown with the yellow patches in Fig. 4.6 and 4.7. Then, passing over a hole in a patch can lead to the probe falling in the latter and if the path is continued by the manipulator, to breaking the probe (Fig. 4.7).

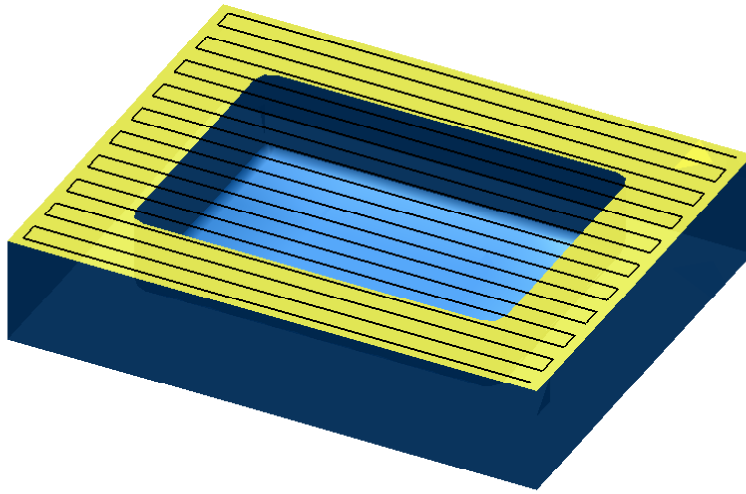


Figure 4.6 Path over a hole resulting in a long cycle-time.

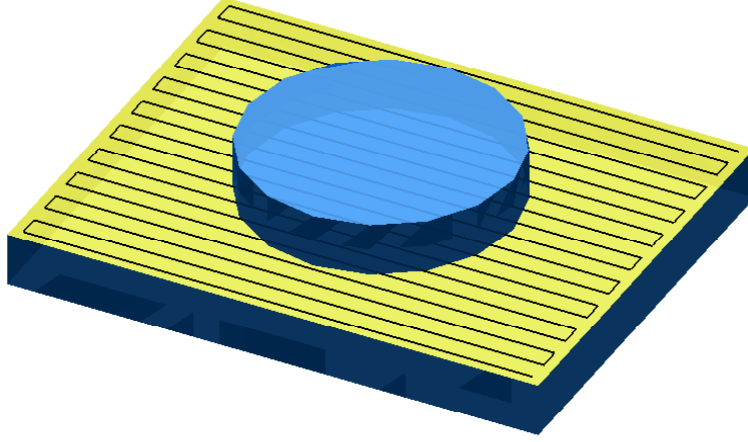


Figure 4.7 Path over a hole resulting in collisions.

Therefore, inspired again from Atkar *et al.* (2009), a Morse decomposition is performed on the previously obtained patches to get “topologically simpler” ones.

4.6.2 Morse Decomposition

The Morse decomposition defined in Acar *et al.* (2002) is part of the *exact cellular decomposition* family. In robotics, such family of decomposition is used to partition a robot’s free configuration space into a set of regions called cells such that the union of these cells fills exactly the original set without intersecting among themselves and such that they have a specific topology for path planning purposes. For instance, the *trapezoidal decomposition* is an exact cellular decomposition that is often used to plan a path between two robot’s configurations of the free configuration space. What characterizes Morse decompositions is that new cells are topologically simple enough (with no holes), such that back and forth motions along a sweep pattern defined by a Morse function (often a zigzag pattern of straight lines) can be performed.

Let us consider a configuration space \mathcal{CS} in the plane with a set of obstacles $\mathcal{O} = \bigcup_i \mathcal{O}_i$ defined with smooth boundaries in this configuration space, and its free configuration space $\mathcal{FCS} = \mathcal{CS} \setminus \mathcal{O}$. Let us also define a real-valued function referred to as the “slice function” and noted $h : \mathbb{R}^2 \rightarrow \mathbb{R}$ such that $\forall (x_1, x_2) \in \mathbb{R}^2, h(x_1, x_2) = x_1$, i.e. the projection along the horizontal axis (cf. Fig. 4.8). Then, a slice is the set denoted $\mathcal{CS}_\lambda = \{(x_1, x_2) \in \mathcal{CS} \mid h(x_1, x_2) = \lambda\}$. With these definitions, the new cells of the Morse decomposition of \mathcal{FCS} are built at *critical points*¹ of $h|_{\partial\mathcal{FCS}}$, the slice function restricted to the boundary of the free configuration space.

It was shown in Acar *et al.* (2002) that the critical points of $h|_{\partial\mathcal{FCS}}$ can be obtained when the

1. As defined in Acar *et al.* (2002), a point $p \in \mathbb{R}^m$ of a real-valued function $h : \mathbb{R}^m \rightarrow \mathbb{R}$ is a critical point of h if $\frac{\partial h}{\partial x_1}(p) = \dots = \frac{\partial h}{\partial x_m}(p) = 0$.

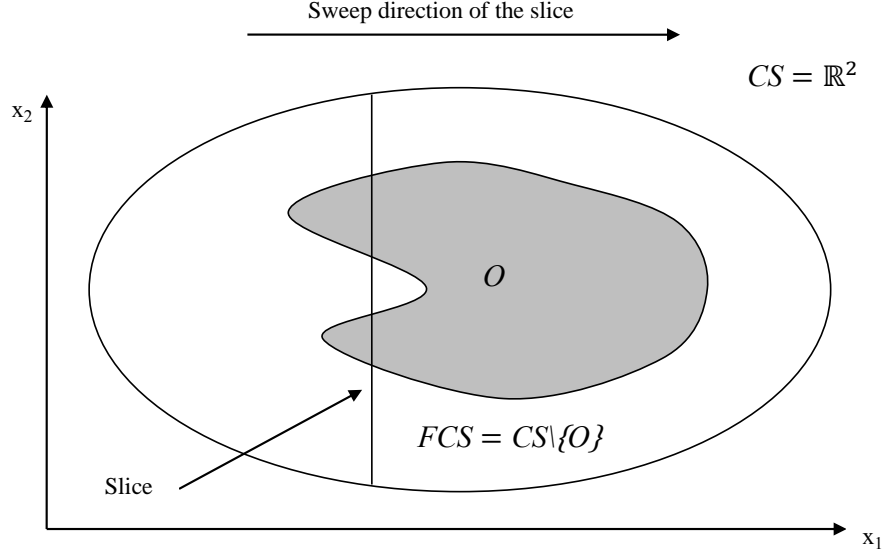


Figure 4.8 The region is decomposed in new cells by sweeping a “slice,” adapted from Acar *et al.* (2002).

slice is tangent to the obstacle boundary, as depicted in Fig. 4.9. Mathematically, the gradient of the slice function is colinear to the gradient of the obstacle boundary at these particular points. Also, assuming that $h|_{\partial \mathcal{FCS}}$ is a Morse function, i.e. that all of its critical points are non-degenerate², and that there is only one critical point per slice, it was proven that the connectivity of the slice remains the same between two adjacent critical points. Hence, considering an interval denoted \mathcal{FCS}_λ^j of the slice \mathcal{CS}_λ in the free configuration space, Morse theory ensures that there will be neither severing nor merging transformations of this interval between two adjacent critical points. Then, back and forth motion along this interval can easily be performed. As shown in Fig. 4.9, the boundaries of new cells are defined at these intervals containing critical points. Note that this decomposition can be extended for a n -dimensional configuration space.

So far, smooth obstacle boundaries were considered to be in a position to use the Morse theory. However, in practice, obstacle boundaries are not necessarily smooth, which is the case when the surface is approximated with triangles. To extend the properties of the Morse decomposition on non-smooth boundaries, it was assumed in Acar *et al.* (2002) that all the boundaries (smooth and non-smooth) are such that they ensure the slice function to be Lipschitz³. In this more general case, a critical point of a non-smooth boundary is identified whenever the gradient of the slice function is contained within the convex hull of the Clarke’s generalized gradient defined at this point. The

2. Again, as defined in Acar *et al.* (2002), a critical point $p \in \mathbb{R}^m$ is non-degenerate if and only if its Hessian $\left(\frac{\partial^2 h}{\partial x_i \partial x_j}(p)\right)$ is non-singular.

3. A function h is Lipschitz at x , if for all y in the neighborhood of x , $|h(x) - h(y)| \leq K|x - y|$, where $K > 0$.

reader may refer to Acar *et al.* (2002) for details to compute the generalized gradient at non-smooth boundaries. Note that this method to find critical points is also valid for smooth boundaries because the generalized gradient is equal to the usual gradient in that case.

In the following discussion, the considered surfaces are supposed to have non-smooth boundaries such that the condition to use the Morse decomposition is always respected. Moreover, to extend this decomposition to surfaces embedded in \mathbb{R}^3 , it is also assumed that the projection function $\Pi_{\mathbf{n}_a}$ on a plane of direction \mathbf{n}_a (the *average normal* of the meshed surface defined in Sheng *et al.* (2000)) is a bijection from the surface to the projected surface on this plane (this assumption will be justified further in this subsection).

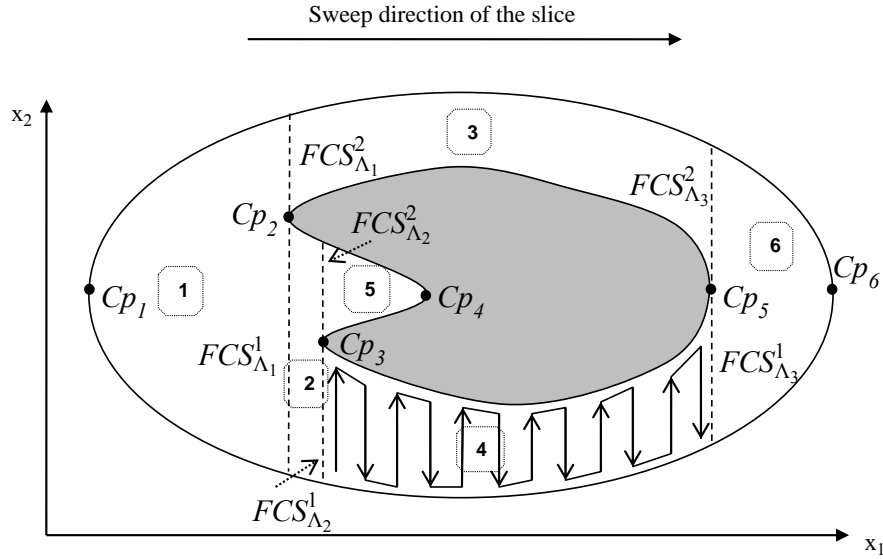


Figure 4.9 The Morse decomposition in a planar case leads here to six new cells (critical points are represented by black dots), adapted from Acar *et al.* (2002).

Then, to implement the Morse decomposition for this type of surface, the boustrophedon algorithm Choset et Pignon (1997) is used and modified.

4.6.3 Extension of the Boustrophedon Algorithm

Applied originally on polygonal and convex boundaries in the plane, the boustrophedon algorithm defines an *Event* structure associated to each vertex of the boundaries. These events contain the necessary information to construct the new cells and four types of event are considered (cf. Fig. 4.10):

- IN event (or *opening* event): closes the current cell and opens two new ones;
- OUT event (or *closing* event): closes the two current cells and opens a new one;

- FLOOR event: event associated to a vertex defining the lower boundary of a cell;
- CEILING event: event associated to a vertex defining the upper boundary of a cell.

To establish a relationship with the Morse decomposition, the IN and OUT events correspond to critical points associated to the slice function h previously defined. However, in the boustrophedon decomposition, the IN (respectively OUT) event of an obstacle is defined at the left-most (respectively right-most) vertex of the polygon. With these definitions, the boustrophedon algorithm works well as soon as the surfaces defined by polygonal boundaries are convex. Indeed, this algorithm cannot be applied in a more general case as shown in Fig. 4.11 insofar as the paths of resultant patches can still cross the boundaries of such obstacles and lead to collision issues similar to what observed in Fig. 4.7.

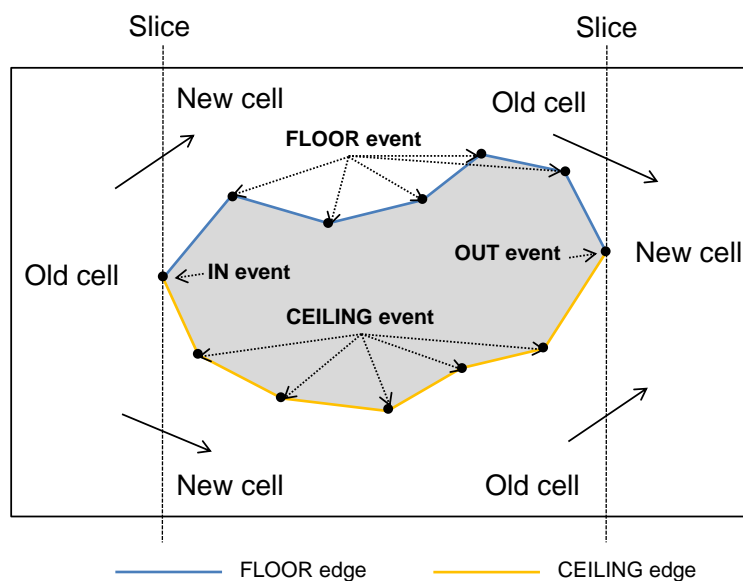


Figure 4.10 Boustrophedon decomposition in a planar case Choset et Pignon (1997).

After the development of the Morse decomposition, an algorithm that addresses this issue has been proposed by Acar et Choset (2002). Although this algorithm is suitable for sensor-based mobile robots moving in unknown planar environments, it can be inefficient for a case when the geometry of the surface is known *a priori*.

Then, similarly to the work presented in Atkar *et al.* (2009), the IN and OUT events are redefined at critical points. Furthermore, in order to take into account the possible non-convexity of boundaries, these definitions are extended by considering two types of IN and OUT events (the definitions of FLOOR and CEILING events are not changed):

- IN₁ event: closes the current cell and opens two new ones;
- OUT₁ event: closes the two current cells and opens a new one;

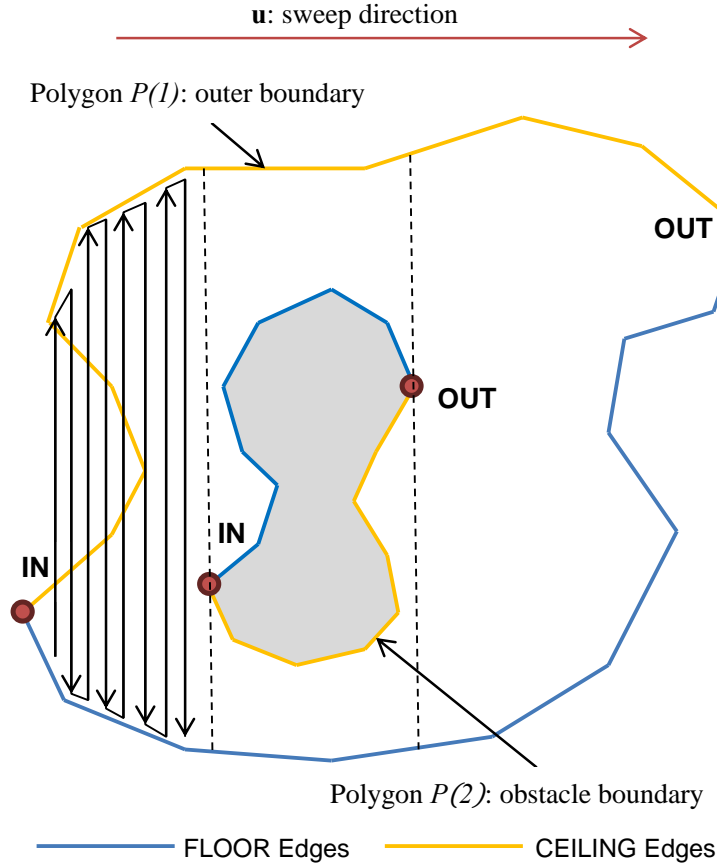


Figure 4.11 Boustrophedon decomposition on non-convex obstacles.

- IN_2 event: only opens a new cell;
- OUT_2 event: only closes the current cell.

In the following discussion will be shown how to identify these new events in order to extend the boustrophedon algorithm to non-convex boundaries assuming that the geometry of the surface is known *a priori*.

Let us at first consider the planar case. As suggested by Choset et Pignon (1997) to start the boustrophedon algorithm, a list of polygon(s) describing the boundaries of the surface to cover is assumed known before starting the decomposition. Also, it is assumed that the vertices of each boundary are arranged in counterclockwise order. In order to localize these events, some notations are introduced (illustrated in Fig. 4.12). Considering a vertex G at the boundary of a polygon, its previous and following vertices are denoted respectively A and B (according to the counterclockwise order). Then, by testing the sign of $\sin(\mathbf{AG}, \mathbf{GB})$, and the signs of the scalar products $\mathbf{AG}^T \mathbf{u}$ and $\mathbf{GB}^T \mathbf{u}$, where \mathbf{u} is the unitary vector of the sweep direction, it is possible to identify the different types of critical points introduced before. In Fig. 4.12, vectors \mathbf{n}_{AG} and \mathbf{n}_{GB} are the

outer-pointing normals of the obstacle boundary defined respectively at segments $[AG]$ and $[GB]$. The critical points are listed in Tables 4.1 and 4.2 where they are classified by their respective properties.

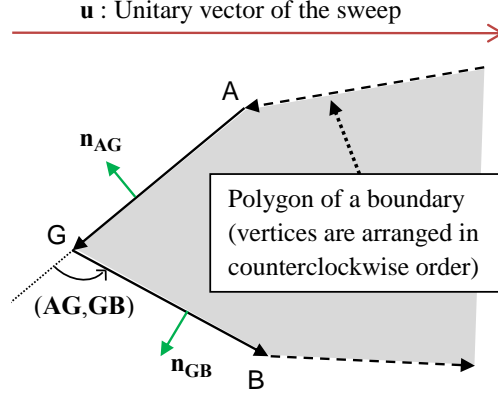


Figure 4.12 Notations used for the search of critical points.

In the same way as the boustrophedon algorithm, each vertex of each polygon is analyzed in counterclockwise order. After having found critical points using the new definitions, FLOOR events of the outer boundary are then associated to all vertices comprised between an IN event (type 1 or 2) and the next OUT event (type 1 or 2). Conversely, FLOOR events of inner boundaries are associated to all vertices located between an OUT event (type 1 or 2) and the next IN event (type 1 or 2). Remaining vertices that are neither a critical point nor associated to a FLOOR event are deduced as CEILING events.

After modifying this part of the initial boustrophedon algorithm and using the *cell* structures with the pointers described in Choset et Pignon (1997), the resultant decomposition of complicated surface such as the one in Fig. 4.11 is obtained (cf. Fig. 4.13).

So far the decomposition was considered only in the planar case. But in reality, surfaces to be inspected are not necessarily planar. Thus, it is now assumed that the surfaces are three-dimensional with geometries such that the projection function $\Pi_{\mathbf{n}_a}$ on a plane of direction \mathbf{n}_a is a bijection from the surface to its projected surface. To justify this assumption for free-form surfaces embedded in \mathbb{R}^3 , let us consider the slice $\mathcal{CS}_\lambda = \{(x_1, x_2, x_3) \in \mathcal{CS} \mid h(x_1, x_2, x_3) = \lambda\}$. The slice function h is defined as follows: $h : \mathbb{R}^3 \rightarrow \mathbb{R}$ such that $\forall \mathbf{x} = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}^T \in \mathbb{R}^3$, $h(\mathbf{x}) = \mathbf{x}^T \mathbf{u}$. This represents the projection along the unit vector \mathbf{u} where \mathbf{u} describes a sweep direction orthogonal to \mathbf{n}_a .

As it will be explained in Section 4.7 about the path generation method, the directions \mathbf{n}_a and \mathbf{u} correspond respectively to the **FRONT** and **RIGHT** directions of the surface (cf. Fig. 4.14). Additionally, \mathcal{CS}_λ corresponds to the intersection between the surface \mathcal{FCS} and the section plane $\mathcal{P}_\lambda = \{(x_1, x_2, x_3) \in \mathbb{R}^3 \mid h(x_1, x_2, x_3) = \lambda\}$. In the cases where $\Pi_{\mathbf{n}_a} : \mathcal{FCS} \rightarrow \Pi_{\mathbf{n}_a}(\mathcal{FCS})$ is not bijective,

tive, the borders of the new cells stemming from the Morse decomposition become discontinuous and thus, ambiguous.

Then, to extend the validity of the Morse decomposition on 3D free-form surfaces, it is necessary to suppose that the projection function $\Pi_{\mathbf{n}_a}$ is a bijection from the surface to its projected surface, which is the case after having applied the previous segmentation based on the surface normals if $\theta_{thresh} < 180^\circ$. However, unlike in the planar case, it is not guaranteed that the distance between two adjacent passes can be kept exactly constant anymore.

Table 4.1 Identification of critical points of the outer boundary $P(1)$.

Type of critical points	IN_1	IN_2	OUT_1	OUT_2
Illustration of the critical point				
$(\mathbf{AG}^T \mathbf{u} < 0) \text{ AND } (\mathbf{GB}^T \mathbf{u} > 0)$	✓	✓	✗	✗
$(\mathbf{AG}^T \mathbf{u} > 0) \text{ AND } (\mathbf{GB}^T \mathbf{u} < 0)$	✗	✗	✓	✓
$\sin(\mathbf{AG}, \mathbf{GB}) > 0$	✗	✓	✗	✓
$\sin(\mathbf{AG}, \mathbf{GB}) < 0$	✓	✗	✓	✗

4.7 Path generation

Among the two cited techniques to generate zigzag paths from a CAD model (Fig. 4.1), using section planes is chosen here. After some experimentations it was found that path results are satisfying and look “intuitive” in most cases.

To do this, the algorithm used is taken from Sheng *et al.* (2000). To compute a path on a meshed surface with triangular elements, a rectangular bounding box enclosing the surface is built (cf. Fig. 4.14). The three main directions of the box, noted **RIGHT**, **FRONT** and **TOP**, define a right-handed orthonormal base. In Sheng *et al.* (2000), the **FRONT** direction was defined as “the opposite of the average normal” \mathbf{n}_a of the surface, and the **RIGHT** direction as the sweep direction of the covering path. In this work, the **RIGHT** direction was computed such that the **FRONT** rectangle of the bounding box has the minimum area. However in our work, instead of computing the sweep direction, the latter is let for the user to specify as a parameter noted *sweep_dir* (Fig. 4.15).

Table 4.2 Identification of critical points of the inner boundaries $P(j)$, $j \geq 2$.

Type of critical points	IN_1	IN_2	OUT_1	OUT_2
Illustration of the critical point				
$(\mathbf{AG}^T \mathbf{u} < 0)$ AND $(\mathbf{GB}^T \mathbf{u} > 0)$	✓	✓	✗	✗
$(\mathbf{AG}^T \mathbf{u} > 0)$ AND $(\mathbf{GB}^T \mathbf{u} < 0)$	✗	✗	✓	✓
$\sin(\mathbf{AG}, \mathbf{GB}) > 0$	✓	✗	✓	✗
$\sin(\mathbf{AG}, \mathbf{GB}) < 0$	✗	✓	✗	✓

The **TOP** direction is deduced to complete a right-handed orthonormal base. Subsequently, cutting planes of **RIGHT** direction are offset along the surface and the actual points of the path are computed as the intersection between the planes and the triangles' edges of the mesh. Finally, the intersection points are sorted at each section plane either along the **TOP** or $-\mathbf{TOP}$ direction to reproduce each pass of the zigzag path. The offset distance between two consecutive planes, noted d , is assumed constant and constitutes the *inter-pass distance* parameter between each pass of the path. The last two path parameters are described in Fig. 4.15: *offset_edge* and *Os_d* (for “overspray distance”).

However, it is important to use this method with section planes on surfaces that respect certain geometric constraints, namely they must be relatively flat. Quantitatively, it is required that each point of the projected surface on a plane of direction **FRONT** is the associated image of one and only one point of the original surface by the projection function on the **FRONT** plane. Indeed, when it is not the case, intersected points at each section plane, once sorted along the **TOP** direction, leads to an inconsistent path (Fig. 4.16).

4.8 Results

In this last section, some results of the path generation method described in the previous sections are presented. This paper focuses on the inspection of aeronautical parts with complex shapes, such as the example illustrated in Fig. 4.17, the local surface selected for inspection is highlighted in yellow.

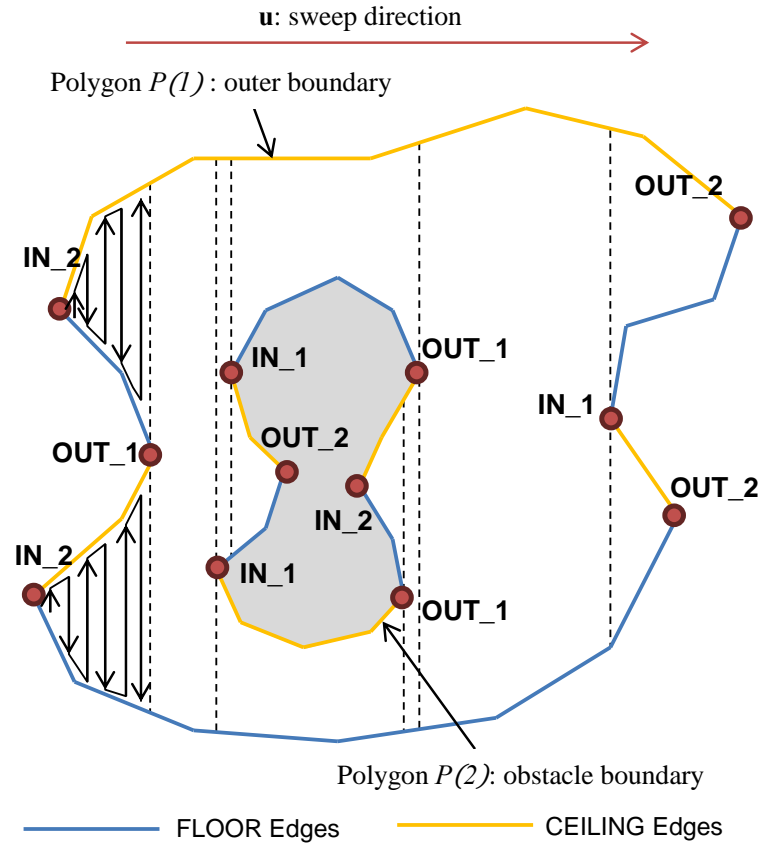


Figure 4.13 Modified boustrophedon decomposition applied on non-convex obstacles.

Results obtained by applying successively the three segmentations according to the methodology presented in the flowchart of Fig. 4.2 are shown in Fig. 4.18, 4.19 and 4.20. After applying the first segmentation, the selected surface is partitioned into patches, mostly separated by straight edges and fillets (Fig. 4.18). However, at this stage the small cylindrical surface in the hole is still considered as a single patch. As seen in the path generation Section 4.7, this geometry can result in inconsistent paths. Then, the second segmentation ensures that the maximum deviation angle of each patch is lower than 100 degrees (Fig. 4.19). Now, if the partition after the second segmentation is considered (cf. Fig. 4.19), one can see that the generated path on the orange patch (the largest one) would lead to passes colliding with the part, as shown in Fig. 4.7. Then, the last segmentation (Fig. 4.20) guarantees that all patches can be covered by back-and-forth motions in a fixed sweep direction, without crossing its borders. Finally, a result of paths generated on each patch is displayed in Fig. 4.21. Though results seem satisfying on most of the surface, it can be seen that paths on fillets are not what is expected for an eddy current testing yet. It should be noted that the parameters of the two first segmentations are tuned until resultant patches are satisfying, in the sense that a tradeoff has been found to reach the best quality of patches (flatness, few curvature

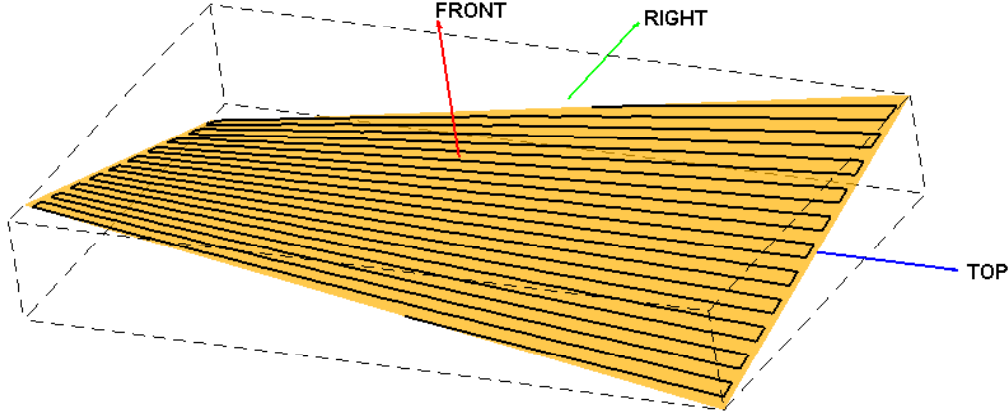


Figure 4.14 Bounding box of the surface and resultant path with $sweep_dir = 0^\circ$, $d = 5mm$, $Os_d = -2mm$ and $offset_edge = 2mm$.

variations inside) while limiting their number. Indeed, usually the more parameters are imposed on the segmentation to improve patch quality (decreasing of $depth_{thresh}$ and θ_{thresh}), the more patches are generated, increasing the time of operations.

4.9 Conclusion

Many aeronautical parts have complex shapes and generating zigzag paths on their surfaces is a far from simple task. Assuming that the geometry of the part is known with a STL file, the solution proposed here first consists in applying successive segmentations to the target surface in order to get patches with a controlled geometry and topology, and then generating paths on each patch using section planes.

The first segmentation (watershed), is based on curvature and partitions the surface into regions such that all local maxima of curvature are located at their borders. Then, a merging step of these regions is performed to define final patches by tuning a parameter that controls the minimum “depth” of curvature. However, by detecting only variations of curvature to segment the surface, this method does not ensure that patches have low absolute curvatures: some may have high absolute curvatures, leading generally to paths with large tool orientation changes (or even inconsistent paths) as well as important deviations of the distance between two consecutive passes. A solution is proposed by applying a second segmentation on each patch previously computed, based on normal directions of triangular facets. More precisely, it controls the flatness of each patch by limiting their maximum deviation angle to a fixed value. Then, a third segmentation is applied, called Morse decomposition, to get resultant patches such that back and forth motions along a relative sweep direction can be performed to cover it entirely without crossing their borders. Once definitive patches

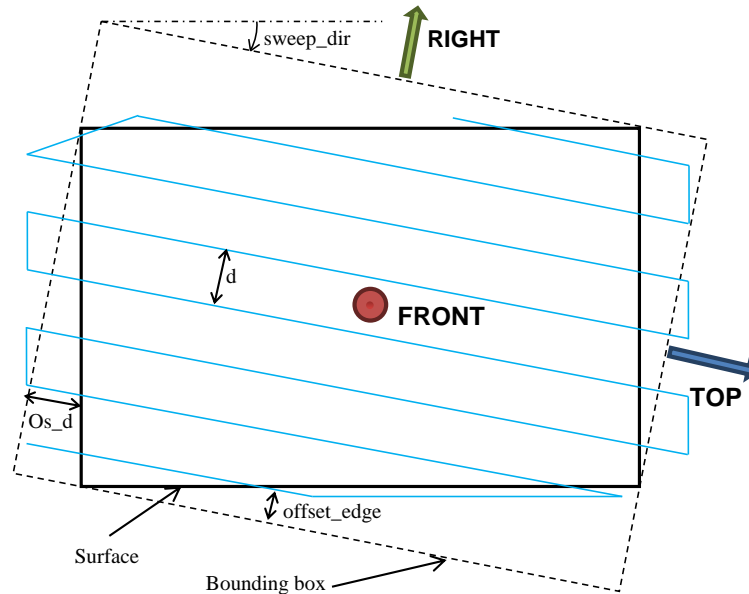


Figure 4.15 Parameters of a path.

are obtained, paths are generated by using section planes offset along a sweep direction with a fixed interval distance.

Results of this methodology on a complex aeronautical part such as the one displayed in Fig. 4.17 give mostly consistent paths for an eddy current inspection. However, limitations appear in regions such as fillets, where paths are not suitable for this application. Another issue is that changes of distance between two adjacent passes, appearing particularly in curved areas, are not evaluated quantitatively, but only by the user's appreciation.

4.10 Acknowledgments

This work was supported by the organizations Natural Sciences and Engineering Research Council of Canada (NSERC) and Consortium de recherche et d'innovation en aérospatiale au Québec (CRIAQ). The authors also gratefully thank their industrial partners L-3 Communications, Pratt & Whitney Canada and NRC for their financial support and their useful suggestions throughout this work.

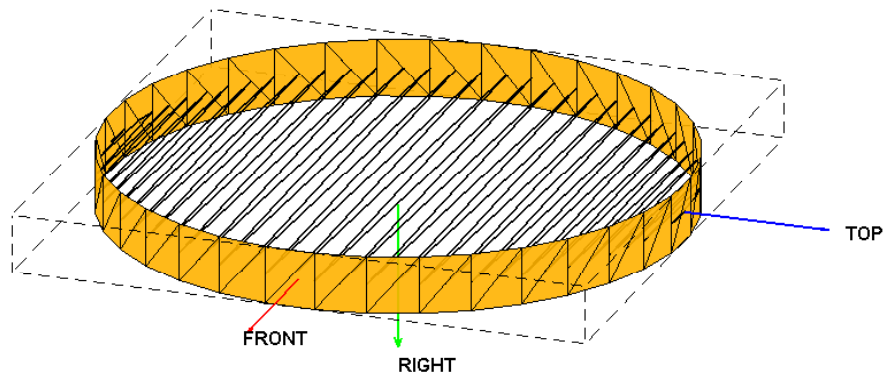


Figure 4.16 Limitation of the method with section planes.

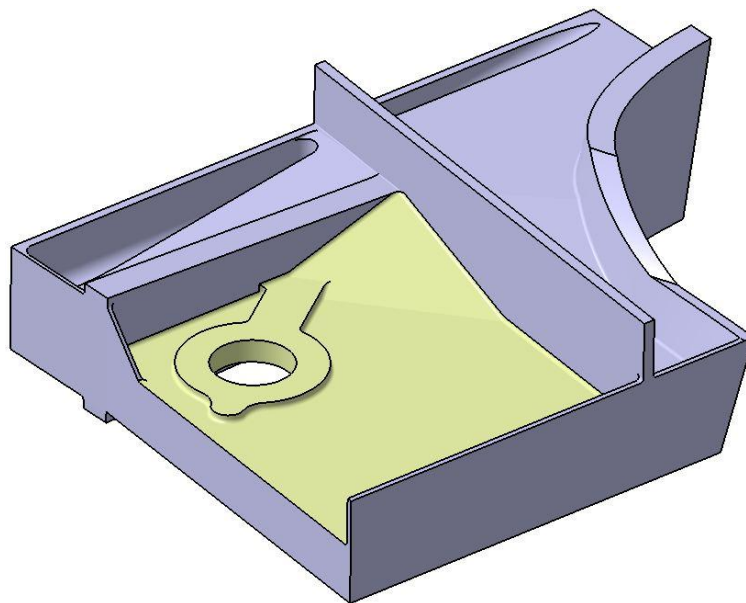


Figure 4.17 Aeronautic structural part. The light yellow surface is selected here for an inspection with eddy currents.

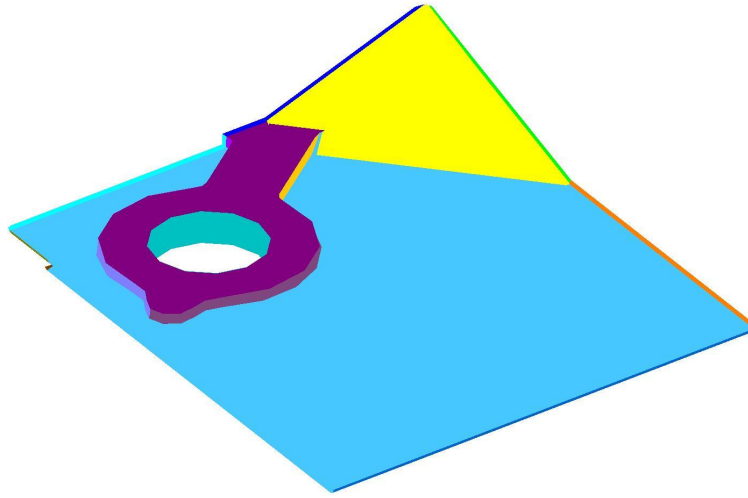


Figure 4.18 First segmentation: watershed segmentation with $depth_{thresh} = 0.1$.

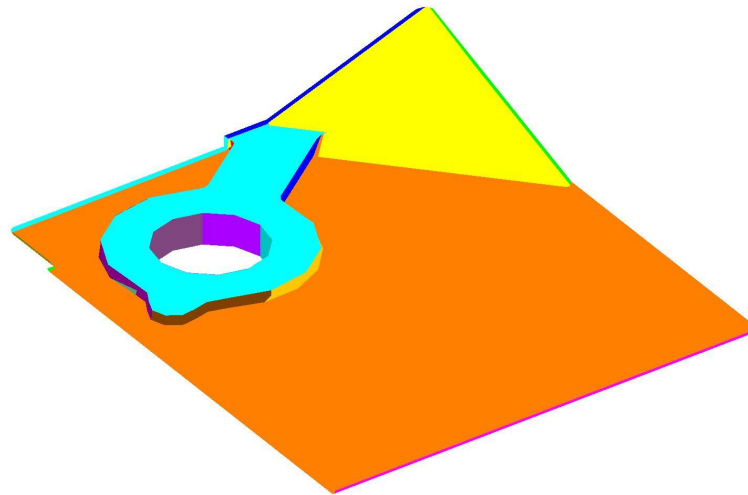


Figure 4.19 Second segmentation: based on surface normals with $\theta_{thresh} = 100^\circ$.

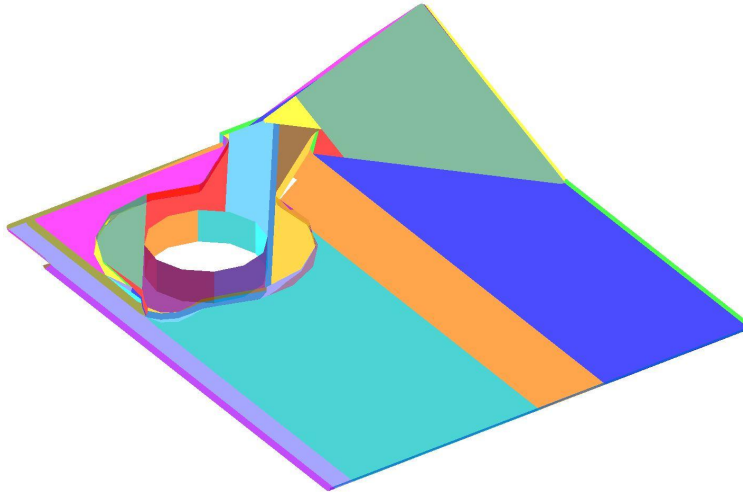


Figure 4.20 Third segmentation: the Morse decomposition.

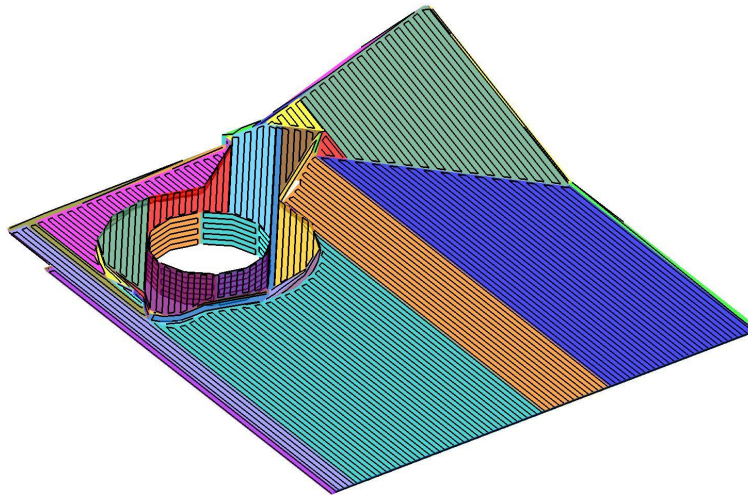


Figure 4.21 Path result with $sweep_dir = 0^\circ$, $d = 2mm$, $Os_d = -1mm$ and $offset_edge = 0mm$.

CHAPITRE 5

ASPECTS MÉTHODOLOGIQUES ET RÉSULTATS COMPLÉMENTAIRES

Dans le chapitre précédent, les chemins générés décrivent la position (x, y, z) dans l'espace que les instruments de mesure devront suivre, mais l'orientation de ces derniers n'a pas été abordée. Cette dernière information est importante, en particulier pour l'application de la technique ECT qui nécessite que l'orientation de la sonde soit toujours normale à la surface de la pièce (cf. Section 1.1.2). Dans ce chapitre, la première section apporte quelques précisions sur une étape préliminaire de la segmentation watershed qui n'a pas été discutée au Chapitre 4. La deuxième section explique la méthode de calcul de l'orientation le long du chemin. Ensuite, une autre méthode de génération de chemin de couverture est proposée dans la section suivante pour l'inspection autour d'une indication¹ dont la position et l'orientation sont supposées connues *a priori*. L'interface utilisateur du programme de génération de chemins est également présentée dans la quatrième section. Enfin, des résultats de simulations des trajectoires d'un bras manipulateur à l'aide du logiciel MotoSim, ainsi que des résultats de tests expérimentaux dans une cellule robotisée sont présentés dans les Sections 5 et 6.

5.1 Remaillage lors de la segmentation “watershed”

Comme expliqué dans le Chapitre 4, la première segmentation appliquée au maillage, appelée segmentation “watershed”, rassemble des facettes triangulaires pour partitionner la surface en plusieurs patches contenant chacun un sommet ayant un minimum local de courbure et dont les frontières correspondent aux zones ayant des maxima locaux de courbure. En d'autres termes, il est attendu après cette segmentation que les patches formés contiennent les zones de faibles courbures (relatives) et repoussent aux frontières les zones de fortes courbures (relatives).

Cependant, un inconvénient du maillage triangulaire est que certaines régions relativement planes ne contiennent pas suffisamment de sommets pour que la segmentation watershed puisse détecter un sommet ayant un minimum local de courbure Mangan et Whitaker (1999). Pour illustrer ce phénomène, considérons le maillage de la figure Figure 5.1. On constate que chacune de faces est décrite uniquement par quatre sommets correspondant aux coins du parallélépipède. La valeur de courbure pour ces sommets sera alors élevée et aucun sommet ayant un minimum local de courbure, représentatif de chacune des faces, ne sera détecté. Cette segmentation ne va donc pas

1. Une indication est un terme utilisé avec les techniques de ressuage (FPI par exemple) pour désigner une trace visible de pénétrant après application du révélateur. Elles sont décrites en termes d'indications linéaires (la longueur est supérieure ou égale à trois fois la largeur) ou non linéaires dans les autres cas Chemin et Dubosc (2010).

segmenter la surface du parallélépipède en ses différentes faces, ce qui aurait été souhaitable. Un raffinement du maillage dont la méthode est présentée dans Mangan et Whitaker (1999) est alors effectué. Le résultat du raffinement sur cet exemple est présenté sur la Figure 5.2.

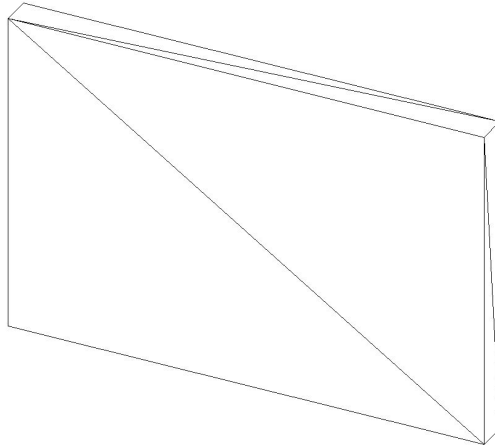


Figure 5.1 Surface triangulée d'un parallélépipède rectangle. Le maillage n'a pas été raffiné dans ce cas-ci.

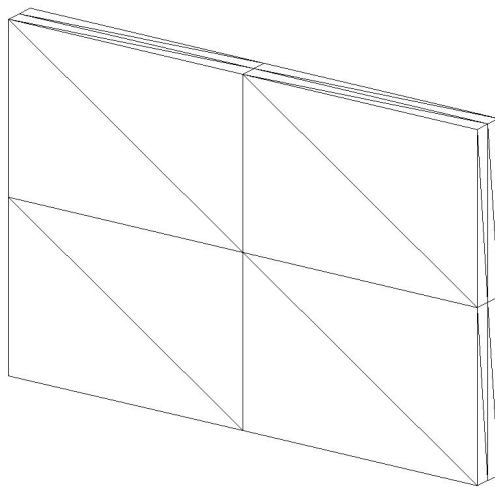


Figure 5.2 Surface remaillée. Un sommet est à présent défini au milieu de chacune des faces.

5.2 Calcul de l'orientation de l'outil

Jusqu'à présent, le chemin de la sonde est uniquement décrit en termes de position (x, y, z) dans l'espace. Afin de respecter la contrainte qui consiste à maintenir la sonde EC normale à la surface en chaque point du chemin, une méthode de calcul de cette normale est proposée. Ensuite, l'orientation

du référentiel de la sonde (supposée différentielle, cf. Section 1.3) est déduite de manière à ce que l'alignement de ses deux bobines soit toujours parallèle à la direction de son déplacement afin de maximiser l'amplitude du signal de la sonde devant un défaut (cf. Figure 1.4). Enfin, un calcul de cinématique inverse est effectué sur ces vecteurs d'orientation afin de décrire cette orientation selon la convention d'orientation "X-Y-Z fixed angles" lisible par le modèle du bras manipulateur utilisé durant la phase expérimentale (cf. Section 5.5).

5.2.1 Calcul de l'orientation du référentiel de la sonde EC en chaque point du chemin

À partir de la représentation au format STL de la surface triangulée, la normale unitaire de chacune des facettes triangulaires est connue. On rappelle également que chaque point du chemin est calculé comme l'intersection d'un des côtés d'un triangle avec un plan de coupe (cf. Chapitre 4).

Notons M un point du chemin et $[Pt_1 Pt_2]$ le côté d'un triangle passant par ce point. Il existe alors un scalaire $t_M \in [0, 1]$ tel que :

$$M = Pt_1 + t_M(Pt_2 - Pt_1) \quad (5.1)$$

Notant respectivement \mathbf{n}_M , \mathbf{n}_{Pt_1} et \mathbf{n}_{Pt_2} les normales unitaires des points M , Pt_1 et Pt_2 , l'approximation faite sur le calcul de la normale au point M est de considérer que **la normale évolue linéairement** entre les points Pt_1 et Pt_2 . On obtiendrait donc :

$$\mathbf{n}_M = \mathbf{n}_{Pt_1} + t_M(\mathbf{n}_{Pt_2} - \mathbf{n}_{Pt_1}) \quad (5.2)$$

Comme les normales sont toujours considérées unitaires dans les raisonnements, la normale est finalement calculée selon la relation :

$$\mathbf{n}_M = \frac{\mathbf{n}_{Pt_1} + t_M(\mathbf{n}_{Pt_2} - \mathbf{n}_{Pt_1})}{\|\mathbf{n}_{Pt_1} + t_M(\mathbf{n}_{Pt_2} - \mathbf{n}_{Pt_1})\|} \quad (5.3)$$

De plus, **la normale de chacun des points du maillage triangulaire est approximée comme la moyenne des normales des triangles voisins au point considéré**. Soit Pt un point du maillage et V l'ensemble de ses triangles voisins. Notons également \mathbf{n}_T la normale unitaire associée au triangle T . La normale \mathbf{n}_{Pt} associée au point Pt est :

$$\mathbf{n}_{Pt} = \frac{\frac{1}{\dim(V)} \sum_{T \in V} \mathbf{n}_T}{\left\| \frac{1}{\dim(V)} \sum_{T \in V} \mathbf{n}_T \right\|} = \frac{\sum_{T \in V} \mathbf{n}_T}{\|\sum_{T \in V} \mathbf{n}_T\|} \quad (5.4)$$

Pour chaque passe du chemin, il est probable que son point de début (et/ou de fin) se situe à l'intérieur d'un triangle et non sur un de ses côtés pour respecter une distance Os_d (paramètre de chemin défini au Chapitre 4) avec la frontière de la surface. De la même manière qu'entre

chaque point du côté d'un triangle, la normale est supposée évoluer linéairement entre deux points d'intersection consécutifs définissant la passe.

Une fois la normale de la surface calculée pour chacun des points du chemin, l'orientation du référentiel de la sonde en un point M noté $(\mathbf{n}^M, \mathbf{o}^M, \mathbf{a}^M)$ est définie de la manière suivante :

1. Choisir le vecteur d'approche \mathbf{a}^M comme l'opposé de la normale à la surface, i.e. $\mathbf{a}^M = -\mathbf{n}_M$.
2. Choisir le vecteur d'orientation \mathbf{o}^M orthogonal au vecteur d'approche \mathbf{a}^M et à la direction de balayage, de vecteur directeur noté **RIGHT**. Ce vecteur \mathbf{o}^M correspond alors à la direction de déplacement de la sonde. En supposant initialement que ce vecteur d'orientation est défini parallèle à l'alignement des deux bobines de la sonde différentielle, la condition que cet alignement des bobines soit toujours orienté selon la direction de déplacement sera alors respectée (cf. Figure 1.4). Après normalisation, le vecteur d'orientation est calculé par la relation suivante :

$$\mathbf{o}^M = \frac{\mathbf{a}^M \times \mathbf{RIGHT}}{\|\mathbf{a}^M \times \mathbf{RIGHT}\|} \quad (5.5)$$

3. Enfin, le vecteur normal \mathbf{n}^M est déduit pour compléter le système d'axe orthonormé direct du référentiel de la sonde, i.e. $\mathbf{n}^M = \mathbf{o}^M \times \mathbf{a}^M$.

Un exemple illustrant les référentiels de la sonde en chaque point du chemin est présentée sur la Figure 5.3.

5.2.2 Orientation décrite avec la convention “X-Y-Z fixed angles”

Le robot qui sera utilisé pour les tests expérimentaux est le MotoMan SV3X-L. D'une manière générale, chaque robot utilise une convention pour décrire l'orientation de son effecteur. Dans le cas du modèle MotoMan SV3X-L, la convention d'orientation utilisée est la convention “X-Y-Z fixed angles”. Selon cette convention Craig (2004), l'orientation d'un référentiel B par rapport à un référentiel A est décrite en appliquant successivement au référentiel A trois rotations autour de ses axes **fixes** (et non par rapport à ses axes transformés par rotation comme pour les angles d'Euler) dans cet ordre : une rotation d'angle R_x autour de l'axe X , puis une rotation d'angle R_y autour de l'axe Y , et enfin une rotation d'angle R_z autour de l'axe Z . En d'autres termes, l'orientation d'un référentiel B par rapport à un référentiel A est décrit directement par le système de coordonnées (R_x, R_y, R_z) utilisant cette convention d'orientation.

Selon cette convention, l'orientation de la sonde EC, définie précédemment par rapport au référentiel absolu de la surface à l'aide du système d'axes $(\mathbf{n}, \mathbf{o}, \mathbf{a})$ en chaque point du chemin, peut alors être décrit par un triplet de coordonnées (R_x, R_y, R_z) . Connaissant les vecteurs du référentiel

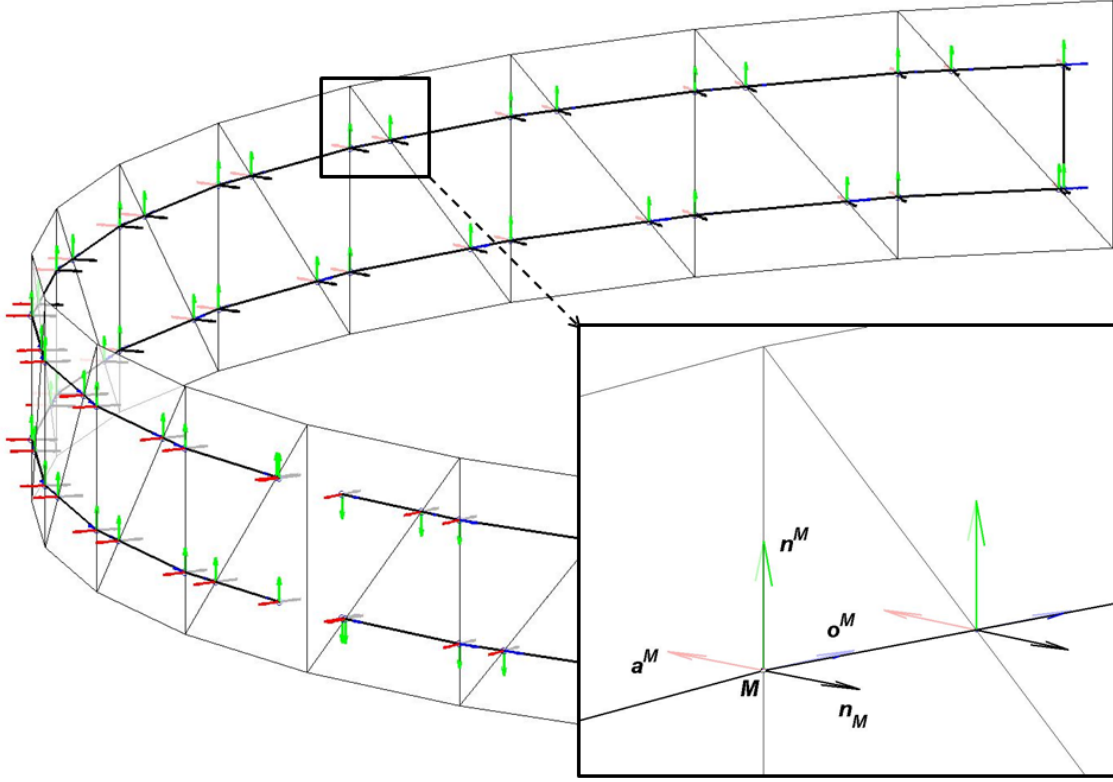


Figure 5.3 Représentations des référentiels de la sonde en chaque point du chemin.

de la sonde $\mathbf{n} = (n_x, n_y, n_z)^T$, $\mathbf{o} = (o_x, o_y, o_z)^T$ et $\mathbf{a} = (a_x, a_y, a_z)^T$, les angles de rotation (R_x, R_y, R_z) associées sont alors calculés par² :

Cas 1 : $|n_z| < 1$

$$\begin{cases} R_z = \text{atan2}(n_y, n_x) \\ R_y = \text{atan2}(-n_z, \cos(R_z)n_x + \sin(R_z)n_y) \\ R_x = \text{atan2}(\sin(R_z)a_x - \cos(R_z)a_y, -\sin(R_z)o_x + \cos(R_z)o_y) \end{cases} \quad (5.6)$$

Remarque : Une autre solution de la cinématique inverse serait de calculer dans un premier temps $R_z = \text{atan2}(-n_y, -n_x)$, puis déduire les angles R_y et R_x à partir des mêmes Équations (5.6) ci-dessus. Cependant, c'est la première solution qui est également adoptée dans la description de l'orientation du robot MotoMan MotoMan (2001), donc cette deuxième solution est écartée.

2. La fonction atan2 est une fonction à deux arguments et correspond à une variante de la fonction atan . Pour tout couple $(x, y) \neq (0, 0)$, atan2 associe la valeur $\text{atan2}(x, y) \in]-\pi; \pi]$ correspondant à l'angle en radians formé entre la partie positive de l'abscisse x d'un plan, et le point de ce plan aux coordonnées (x, y) .

Cas 2 : $n_z = -1$

$$\begin{cases} R_z = 0^\circ \text{ et } R_y = 90^\circ \text{ (convention usuelle)} \\ R_x = \text{atan2}(o_x, o_y) \end{cases} \quad (5.7)$$

Cas 3 : $n_z = 1$

$$\begin{cases} R_z = 0^\circ \text{ et } R_y = -90^\circ \text{ (convention usuelle)} \\ R_x = -\text{atan2}(o_x, o_y) \end{cases} \quad (5.8)$$

5.3 Génération de chemins autour d'une indication de coordonnées connues

Dans cette section, une autre méthode de génération de chemins est proposée. La première méthode de génération de chemins (cf. Chapitre 4) permettait d'inspecter une surface dans sa globalité. Les chemins générés dans cette partie permet une inspection plus locale de la surface, mais plus raffinée. Plus précisément, ces chemins d'inspection sont spécifiques à la technique ECT. Étant donné la sensibilité élevée de cette technique pour la détection de fissures en surfaces et légèrement sous la surface, et considérant les faibles dimensions de la surface de couverture de la sonde EC, il a été décidé dans ce projet de pouvoir générer des chemins autour d'une **indication**, en supposant que la position et l'orientation de celle-ci soient connues *a priori*. Une indication est un terme utilisé dans des techniques de contrôle non destructif par ressuage et correspond aux traces visibles de pénétrant sur la surface après application du révélateur. Ces traces peuvent représenter aussi bien la localisation de défauts débouchants pertinents, que celle de défauts non pertinents (rugosité, marque d'outil, tâche d'huile, etc.). Les indications sont classées généralement dans deux catégories Chemin et Dubosc (2010) : les indications linéaires (leur longueur est supérieure ou égale à trois fois leur largeur), et les indications non linéaires (celles dont leur longueur est inférieure à trois fois leur largeur). La Figure 1.1(d) représente une inspection visuelle sous éclairage UV d'une surface après application de la technique FPI. Les traces vert-fluorescentes correspondent aux traces de pénétrant ayant ressue en surface après application du révélateur, et constituent donc des indications. De plus, connaissant l'orientation de la fissure, la direction de balayage de la sonde EC sera calculée pour être toujours perpendiculaire à la fissure et ainsi maximiser la sensibilité de la sonde (cf. Figure 1.4).

Pour générer ce type de chemins, une approche similaire à la méthode détaillée dans le Chapitre 4 est utilisée. Bien qu'il n'y ait aucune étape de segmentation dans cette approche, la technique de génération de chemins basée sur des plans de coupe est encore utilisée. Cependant, ces plans de coupe ne sont plus espacés uniformément le long d'une même direction de balayage, mais le long

de la ligne brisée définie par les points représentatifs de l'indication. De plus, afin de générer des chemins perpendiculaires à la direction de l'indication, la direction de chacun des plans de coupe est définie pour que chacun de ces plans soit normal à la ligne brisée de l'indication. La Figure 5.4 illustre cette nouvelle approche.

De même que pour la génération de chemin sur une surface globale, des paramètres de chemin sont définis pour l'inspection autour d'une indication (cf. Figure 5.5). Ils sont les suivants :

- d_Ind : distance (en mm) entre deux passes consécutives du chemin de balayage,
- W_Ind : largeur (en mm) du balayage. Cette largeur correspond à l'amplitude du balayage par rapport à la ligne de l'indication,
- $offset_Ind$: distance (en mm) séparant la première passe par rapport à l'extrémité de l'indication, ainsi que la dernière passe vis-à-vis de l'autre extrémité.

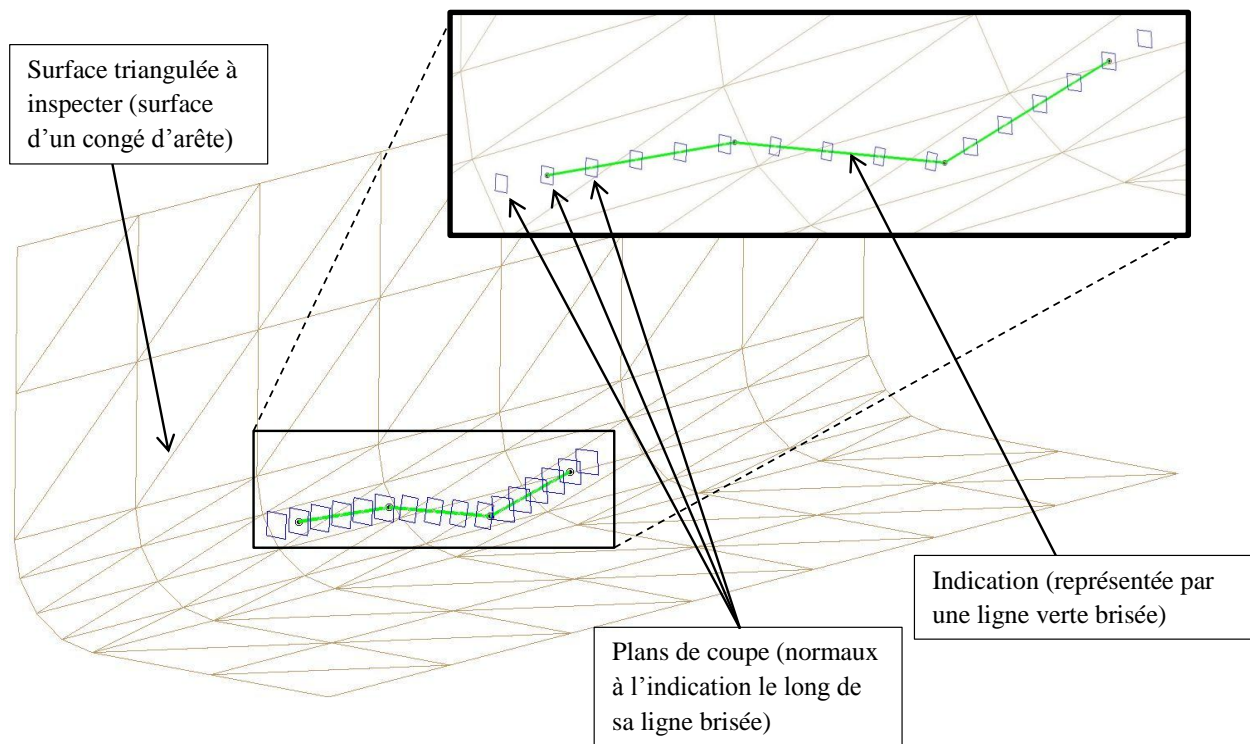


Figure 5.4 Méthode de génération de chemin autour d'une indication par plans de coupe régulièrement espacés et normaux le long de la ligne brisée représentant l'indication.

Ensuite, pour chaque plan de coupe, les points d'intersection entre ce plan et les côtés des triangles de la surface maillée sont calculés, et sélectionnés pour respecter la largeur de balayage W_Ind . Enfin, les points sélectionnés sont ordonnés pour former le balayage en zigzag. Un résultat de la génération de chemin à partir du programme est présenté Figure 5.6.

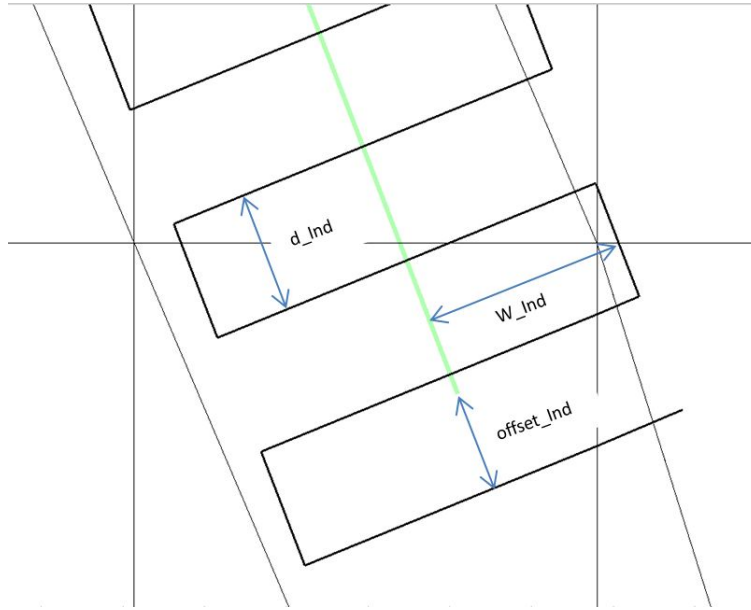


Figure 5.5 Paramètres du chemin de couverture autour d’une indication.

5.4 Programme de génération de chemins

Dans ce travail de recherche, un programme accompagné d’une interface graphique (GUI) a été développé afin de simplifier la procédure de génération de chemins aux utilisateurs et leur laisser la liberté de modifier les paramètres de chemin. Ce programme comprend les deux méthodes de génération de chemin : celle appliquée pour l’inspection d’une surface globale (cf. Chapitre 4), et celle appliquée pour l’inspection autour d’une indication (cf. Section 5.3). L’architecture générale du programme est présentée en Figure 5.7. Quelle que soit la méthode utilisée dans ce programme, une surface maillée dans un fichier STL doit d’abord être importée pour pouvoir générer des chemins. De plus, dans le cas de la génération de chemins autour d’une indication, un fichier Excel (version 2003) contenant les coordonnées spatiales de cette indication (exprimées dans le référentiel absolu de la surface) doit également être importé pour que le programme puisse effectuer le calcul du chemin.

Une fois que l’utilisateur est satisfait des chemins générés après réglage des paramètres, ces chemins peuvent ensuite être exportés. Dans ce programme, les chemins peuvent être exportés soit dans un fichier Excel (version 2003), soit dans un fichier texte (extension “.txt”). Dans les deux cas, les coordonnées sont présentées sous forme de tableau de valeurs dont les trois premières colonnes correspondent respectivement aux trois composantes de position (x, y, z) et trois autres colonnes qui correspondent respectivement aux trois composantes de l’orientation de l’outil (R_x, R_y, R_z) calculées selon la convention “X-Y-Z fixed angles” (cf. Section 5.2.2). Ce format de présentation dans les fichiers de données des chemins est volontairement choisi pour être facilement utilisable par le

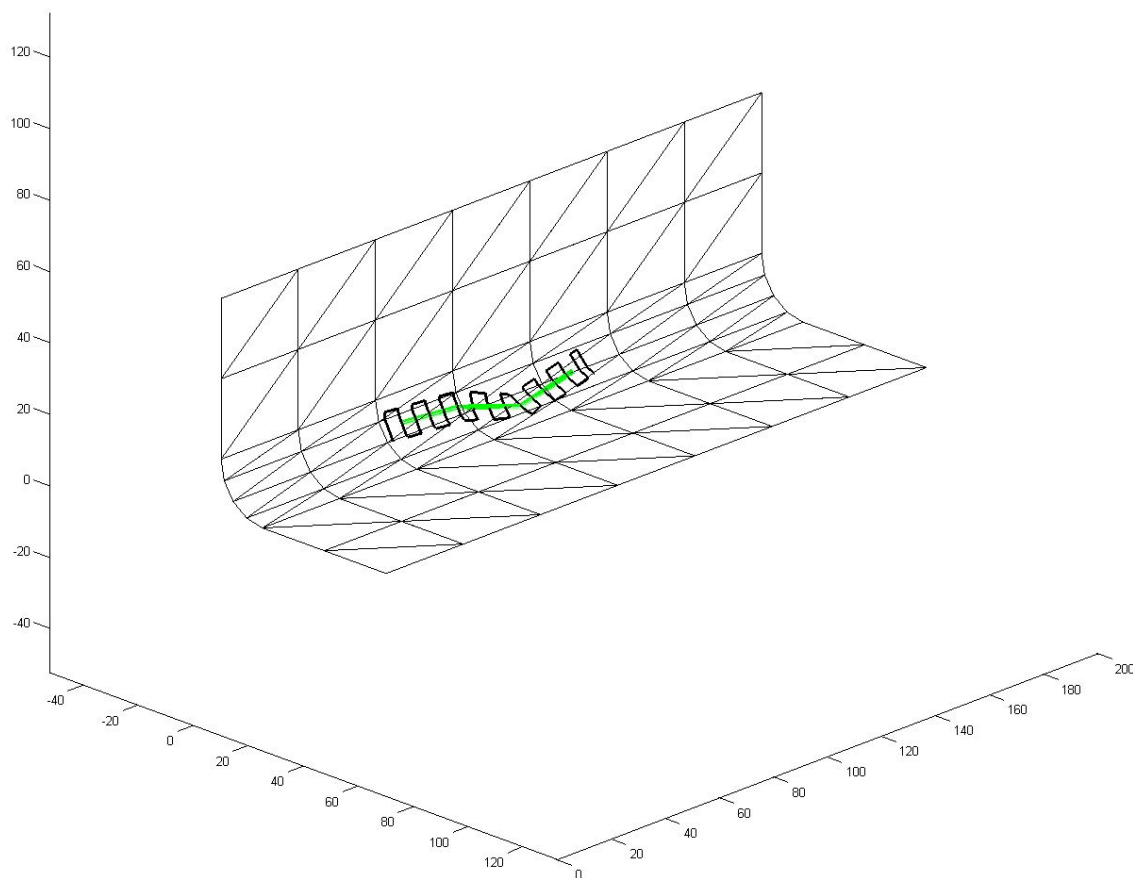


Figure 5.6 Chemin d'inspection autour d'une indication avec les valeurs de paramètres suivants : $d_Ind = 5mm$, $W_Ind = 4mm$ et $offset_Ind = 5mm$.

logiciel de simulation de robots MotoSim. L'utilisation de ce logiciel pour simuler les mouvements du robot le long des chemins calculés précédemment fera justement l'objet de la prochaine section.

Les interfaces des deux méthodes de génération de chemin sont présentées sur les Figures 5.8 et 5.9. Le lecteur intéressé pourra trouver plus de détails sur la présentation du programme dans son manuel utilisateur (cf. Annexe A).

5.5 Simulations d'un bras manipulateur sous le logiciel MotoSim

À partir des coordonnées de position et d'orientation des chemins de couverture exportées dans une table Excel, celles-ci peuvent ensuite être utilisées dans le logiciel MotoSim pour simuler les trajectoires robot associées à ces chemins.

Il est rappelé qu'une trajectoire correspond à un chemin paramétré en fonction du temps. Ainsi, les dérivées première et seconde d'une trajectoire correspondent respectivement à la vitesse et à l'accélération du déplacement le long du chemin qui lui est associé.

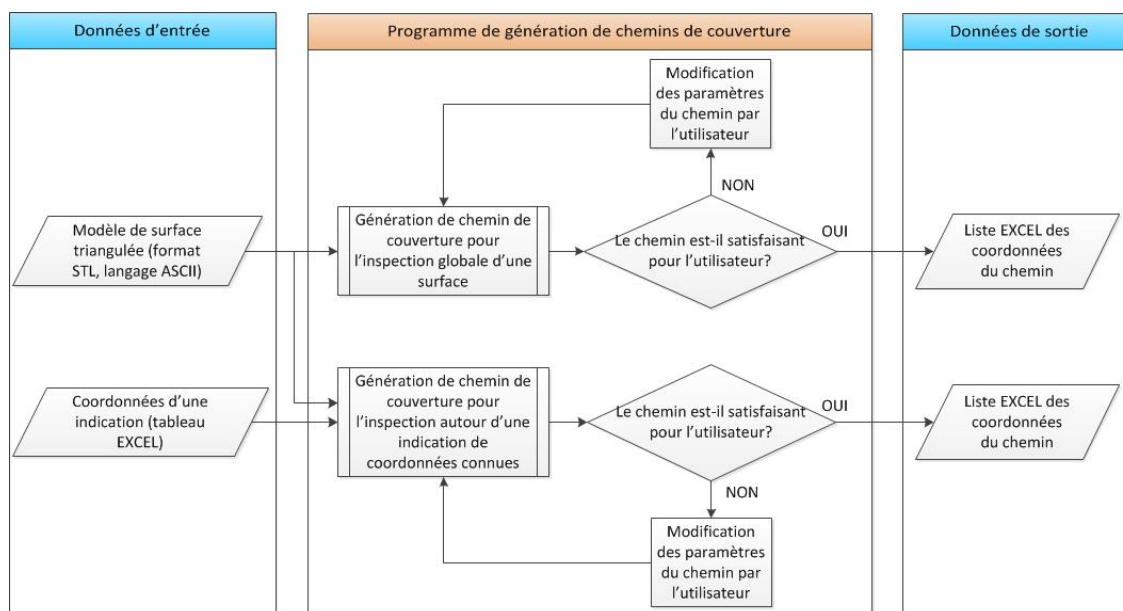


Figure 5.7 Architecture globale du programme de génération de chemins de couverture.

En plus de simuler des trajectoires robot, l'utilisation de MotoSim permet également de définir n'importe quelle géométrie d'outil. Des géométries d'outil d'inspection adaptés pour le type de surface à inspecter (congés, surfaces planes, etc.) ou pour faciliter l'accessibilité pour son inspection (dans des trous par exemple), peuvent alors être ajoutés dans le modèle de simulation. Les configurations du robot seront alors recalculées automatiquement en conséquence. De plus, ce logiciel de simulation a l'avantage d'informer l'utilisateur sur la présence éventuelle de collisions, ou simplement si la position désirée n'est pas atteignable par le robot.

Sur la Figure 5.10 est représenté un résultat de simulation de la trajectoire sous MotoSim avec le robot MotoMan SV3X-L.

5.6 Tests expérimentaux

Une fois que la trajectoire du robot est jugée satisfaisante pour l'utilisateur (le déplacement simulé correspond au déplacement attendu du robot et aucune collision n'est observée) dans la simulation précédente avec le logiciel MotoSim, des tests expérimentaux peuvent ensuite être menés.

Pour cela, les données de la trajectoire simulée avec MotoSim, enregistrées dans un fichier ".jbi" (cf. Figure 5.11), sont transférées au contrôleur (modèle XRC 2001) du robot MotoMan (modèle SV3XL). Ensuite, à l'aide du boîtier d'apprentissage associé au contrôleur, l'utilisateur peut ordonner l'exécution de cette trajectoire par le robot.

Plusieurs tests avec ce modèle de robot ont été effectués pour suivre des chemins d'inspection aussi bien sur des surfaces planes, des surfaces de courbures non nulles, ou encore autour d'une

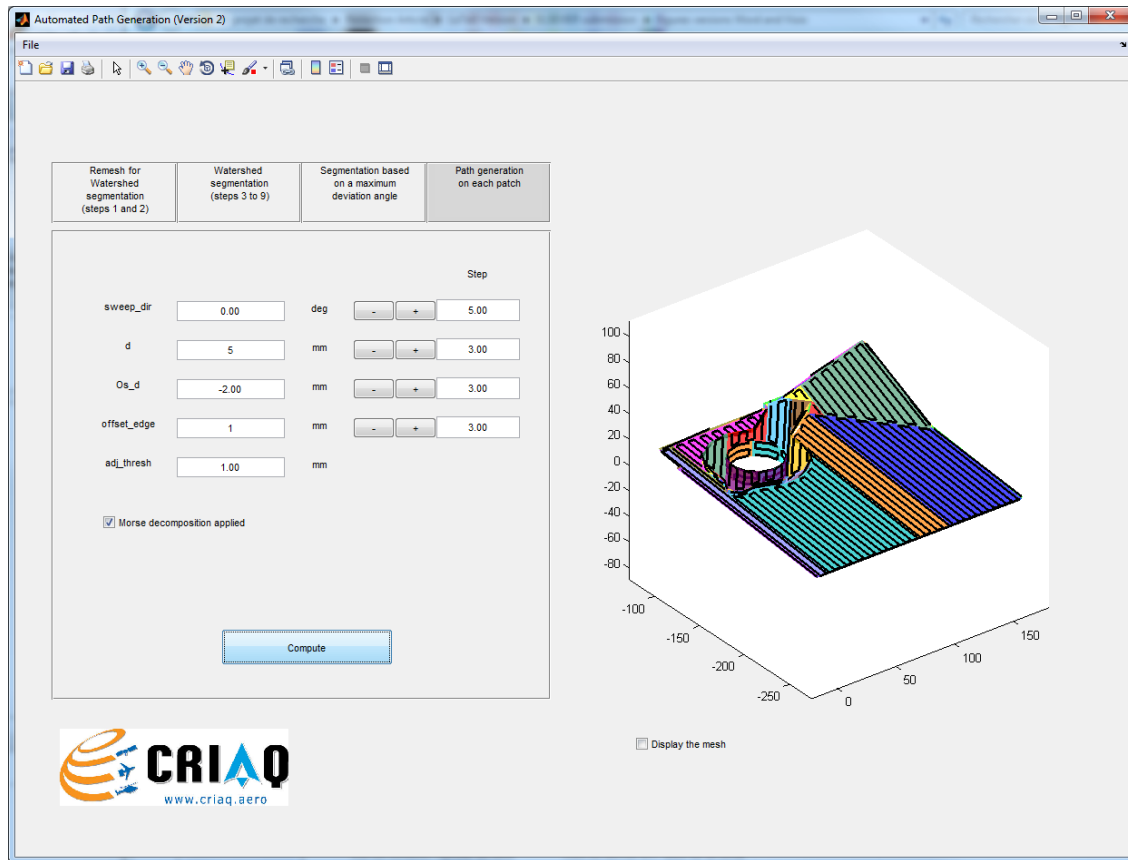


Figure 5.8 Interface graphique pour la génération de chemins sur une surface globale.

indication dans un congé d'arête de 2mm de rayon. Une appréciation visuelle des mouvements du robot le long de ces chemins permet d'affirmer à premier abord la validité des chemins calculés. Toutefois, ces tests sont encore préliminaires avant de pouvoir définitivement conclure sur la validité de la méthode de génération développée dans ce projet de recherche. En effet, dans ces tests, les surfaces à inspecter ne sont pas considérées (le robot inspecte une surface "dans le vide") et une maquette simulant visuellement une sonde EC est montée sur la bride d'outil du robot au lieu d'une véritable sonde (cf. Figure 5.12). Des tests utilisant une vraie sonde EC (et non une maquette) ainsi que des surfaces dont on connaît la localisation des fissures suite à une inspection manuelle, permettront de comparer et d'évaluer rigoureusement l'efficacité de cette méthode de génération de chemins par rapport à l'inspection manuelle.

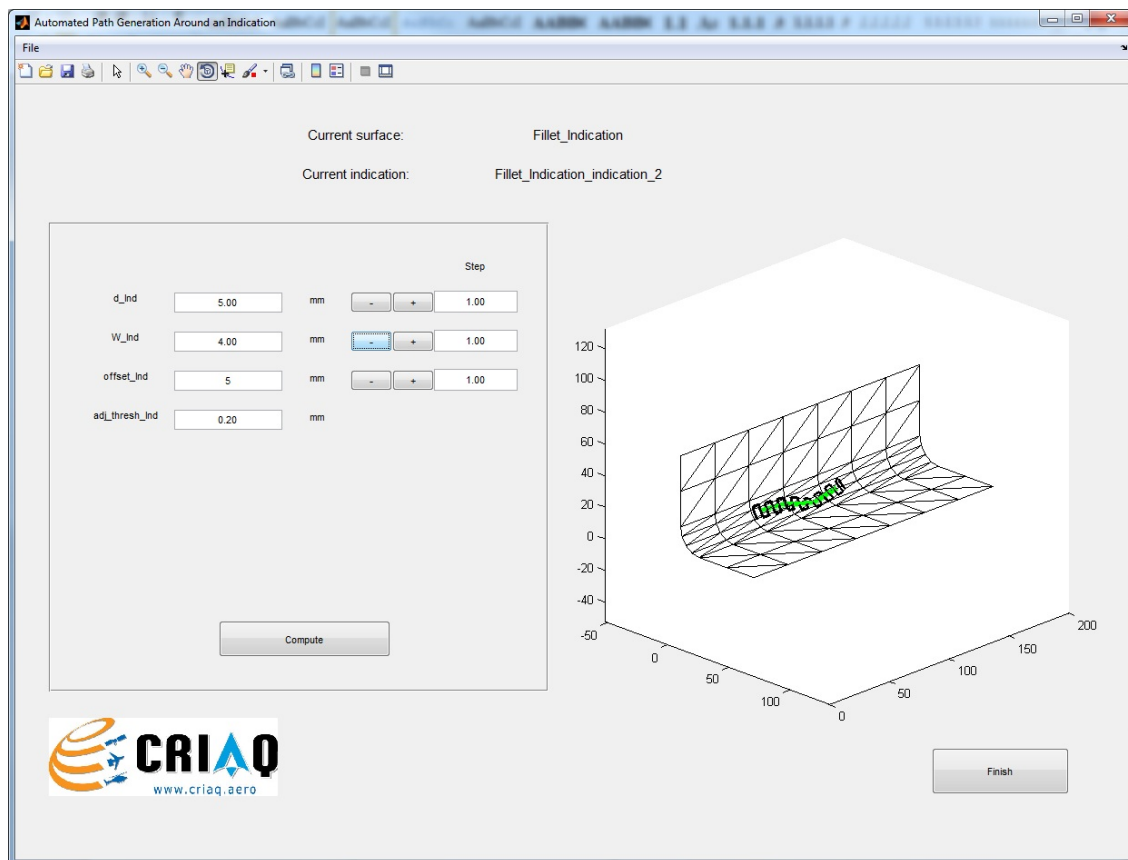


Figure 5.9 Interface graphique pour la génération de chemins autour d'une indication.

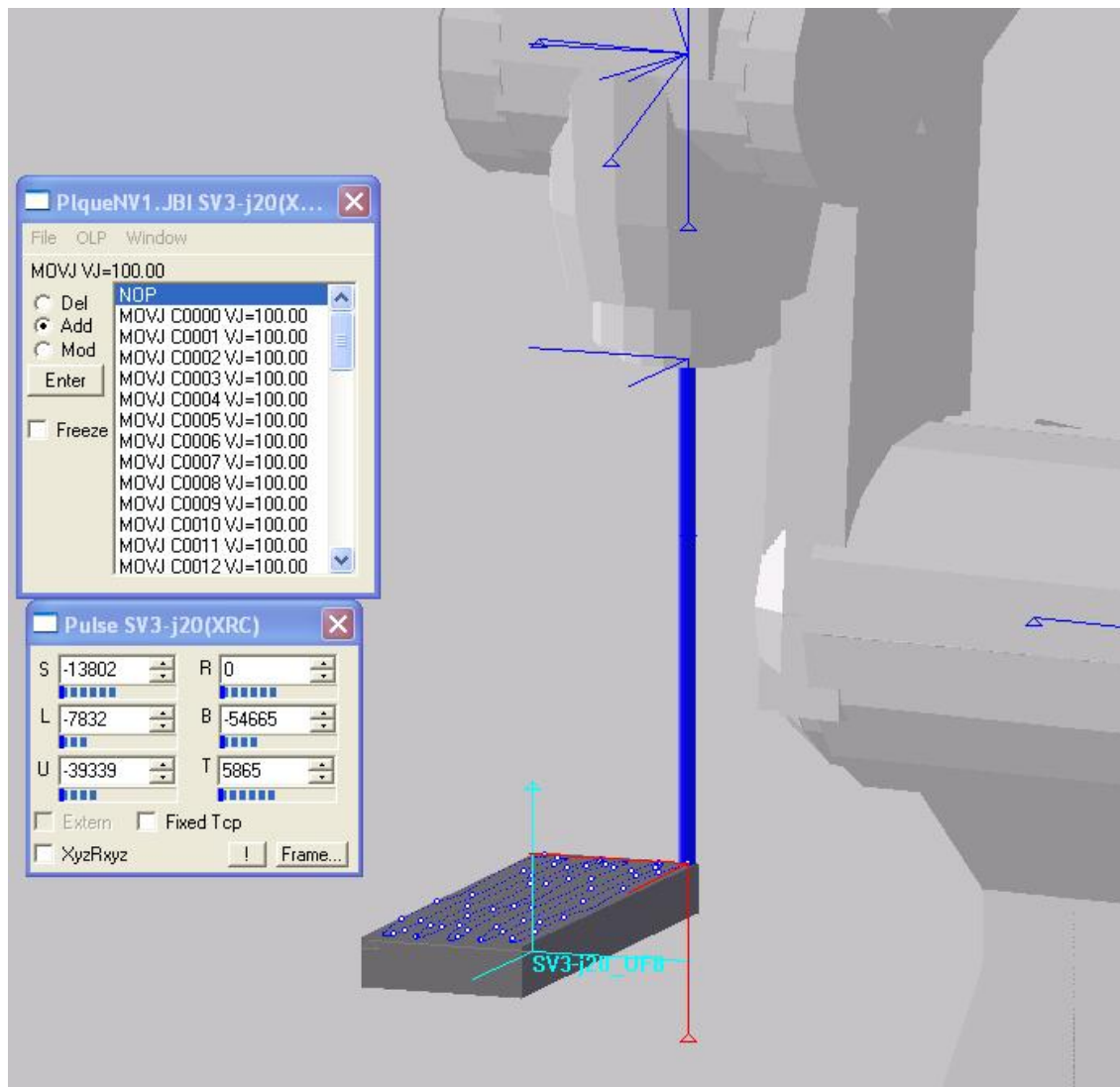
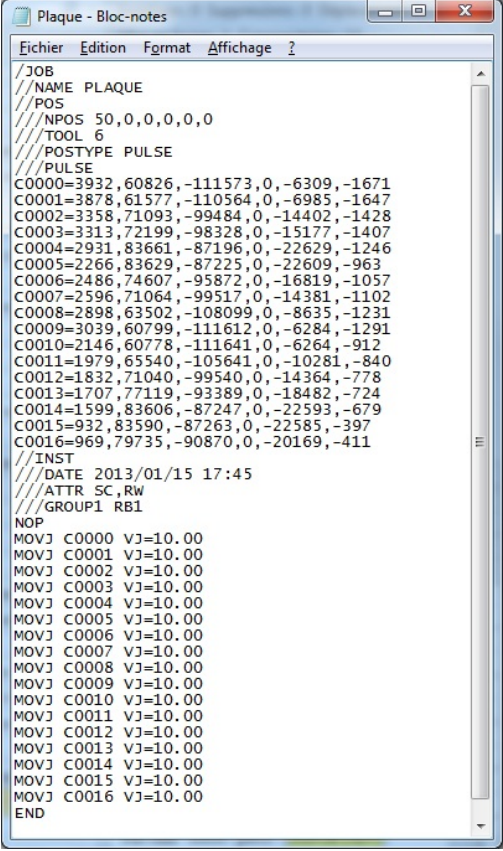


Figure 5.10 Simulation sous le logiciel MotoSim d'une trajectoire sur une face d'une plaque.



```

JOB
//NAME PLAQUE
//POS
//NPOS 50,0,0,0,0,0
//TOOL 6
//POSTYPE PULSE
//PULSE
C0000=3932,60826,-111573,0,-6309,-1671
C0001=3878,61577,-110564,0,-6985,-1647
C0002=3358,71093,-99484,0,-14402,-1428
C0003=3313,72199,-98328,0,-15177,-1407
C0004=2931,83661,-87196,0,-22629,-1246
C0005=2266,83629,-87225,0,-22609,-963
C0006=2486,74607,-95872,0,-16819,-1057
C0007=2596,71064,-99517,0,-14381,-1102
C0008=2898,63502,-108099,0,-8635,-1231
C0009=3039,60799,-111612,0,-6284,-1291
C0010=2146,60778,-111641,0,-6264,-912
C0011=1979,65540,-105641,0,-10281,-840
C0012=1832,71040,-99540,0,-14364,-778
C0013=1707,77119,-93389,0,-18482,-724
C0014=1599,83606,-87247,0,-22593,-679
C0015=932,83590,-87263,0,-22585,-397
C0016=969,79735,-90870,0,-20169,-411
//INST
//DATE 2013/01/15 17:45
//ATTR SC,RW
//GROUP1 RB1
NOP
MOVJ C0000 VJ=10.00
MOVJ C0001 VJ=10.00
MOVJ C0002 VJ=10.00
MOVJ C0003 VJ=10.00
MOVJ C0004 VJ=10.00
MOVJ C0005 VJ=10.00
MOVJ C0006 VJ=10.00
MOVJ C0007 VJ=10.00
MOVJ C0008 VJ=10.00
MOVJ C0009 VJ=10.00
MOVJ C0010 VJ=10.00
MOVJ C0011 VJ=10.00
MOVJ C0012 VJ=10.00
MOVJ C0013 VJ=10.00
MOVJ C0014 VJ=10.00
MOVJ C0015 VJ=10.00
MOVJ C0016 VJ=10.00
END

```

Figure 5.11 Fichier “Plaque.jbi” produit par MotoSim et contenant les informations de la trajectoire. Dans ce fichier sont indiqués, entre autres, l’outil utilisé, les configurations du robot (16 configurations pour cette trajectoire) exprimées en pulse le long de la trajectoire ainsi que la vitesse associée à chacune des configurations.



Figure 5.12 Test expérimental avec le robot MotoMan SV3XL.

CHAPITRE 6

DISCUSSION GÉNÉRALE

Dans ce chapitre, les limitations du programme de génération de chemins ainsi que les améliorations futures recommandées sont discutées.

6.1 Limitations du programme de génération de chemins

Une première limitation de ce logiciel concerne la segmentation watershed. En effet, un paramètre doit être réglé afin de trouver un compromis entre une segmentation minimale de la surface au niveau des zones de fortes variations de courbure et un nombre limité de patchs non pertinents. Ces patchs sont considérés non pertinents lorsqu'ils sont formés par des sommets ayant un minimum local de courbure dû à de faibles erreurs numériques dans le calcul des courbures (cf. Section 4.4). Aussi, plus la surface est complexe et de grande dimension, plus ce compromis sera difficile à obtenir.

Un autre inconvénient est que, dans le programme développé, il n'est pas possible d'attribuer des paramètres de chemin différents pour chacun des patchs issus de la segmentation de la surface globale. Les patchs d'une surface ont donc toujours les mêmes paramètres de chemin et une modification de ces paramètres sera alors appliquée à tous ces patchs.

Une solution recommandée pour contourner les limitations discutées précédemment serait, au lieu d'exporter la surface globale à inspecter dans un seul fichier de maillage (format STL), d'exporter plutôt chacune de ses surfaces paramétriques constituant la surface globale dans des fichiers de maillage séparés. Un exemple de surface paramétrique composant une surface plus large est illustré Figure 6.1. En effet, en appliquant la génération de chemins sur des surfaces paramétriques, celles-ci sont *souvent* plus petites avec une géométrie régulière, et sont représentatives des attributs de la surface (congés, surfaces planes, surfaces cylindriques, etc.). Les fichiers de maillage des surfaces paramétriques étant indépendants les uns des autres, des paramètres de chemin peuvent alors être adaptés pour chacune d'elles en fonction de l'attribut qu'elles représentent.

En revanche, cette solution nécessite plus d'efforts de la part de l'utilisateur pour exporter chacune des surfaces paramétriques. D'une certaine manière, il s'agit de segmenter "manuellement", à partir du logiciel de CAO, la surface en plusieurs surfaces paramétriques avant d'appliquer la génération de chemins (incluant ses propres étapes de segmentations) sur chacune de celles-ci. Le lecteur intéressé trouvera plus de détails sur cette méthode d'exportation dans le manuel utilisateur en Annexe A.

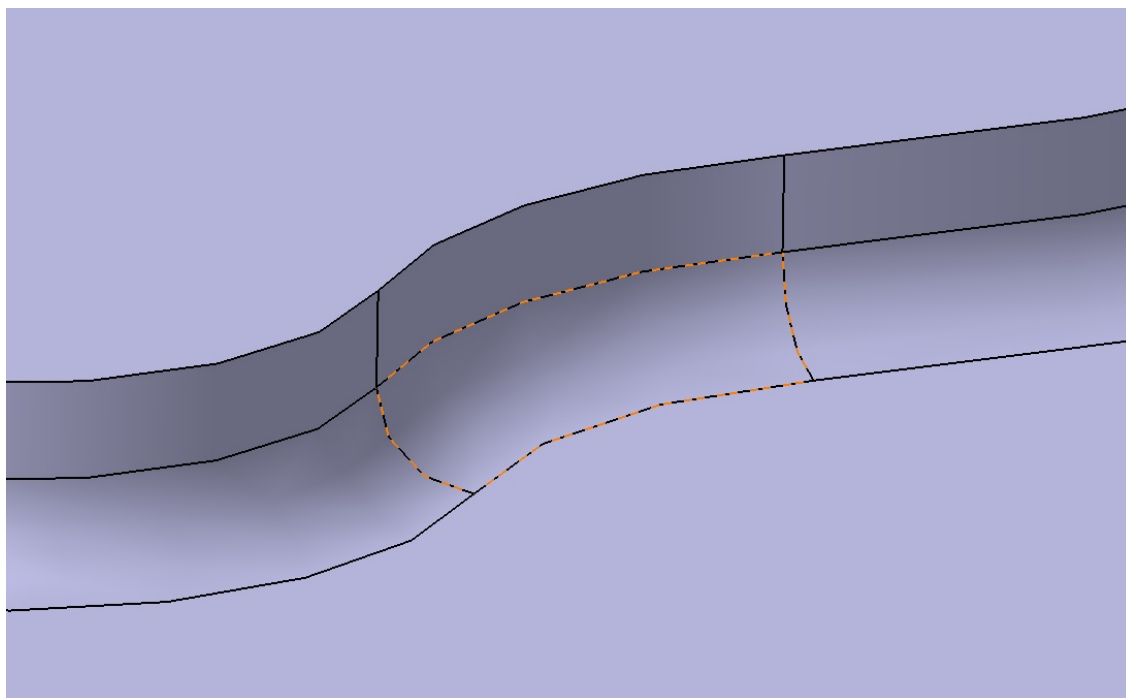


Figure 6.1 Illustration d'une surface paramétrique constituant la surface d'une pièce. Les frontières de la surface paramétrique sélectionnée sont marquées en pointillés orange.

Une autre limitation concerne la méthode de génération utilisant des plans de coupe. Bien que cette méthode soit rapide à implémenter dans le programme par rapport à celle utilisant des courbes décalées (cf. Figure 2.7), elle ne garantit pas nécessairement une distance égale entre chaque passe, en particulier si la surface a une courbure élevée (cf. Figure 6.2). Cet effet peut affecter les résultats d'inspection dans le cas où la distance entre deux passes adjacentes est supérieure à la distance définie entre les plans de coupe, ce qui peut conduire à des zones de la surface non couvertes par l'outil d'inspection (sonde EC par exemple). Un moyen de limiter cette augmentation serait de segmenter davantage la surface en diminuant son angle de déviation maximal. Une réduction de la distance entre les plans est également un moyen de limiter cet effet, à défaut de générer aussi des passes plus rapprochées que nécessaire dans des zones où le plan tangent local à la surface devient de plus en plus perpendiculaire aux plans de coupe. De plus, le rapprochement des plans de coupe ne permet pas de garantir que la distance maximale désirée entre deux passes adjacentes est respectée.

6.2 Améliorations futures

Afin de s'affranchir des limitations mentionnées précédemment pour la méthode utilisant des plans de coupe (i.e. des augmentations non contrôlées de la distance entre les passes sur des surfaces

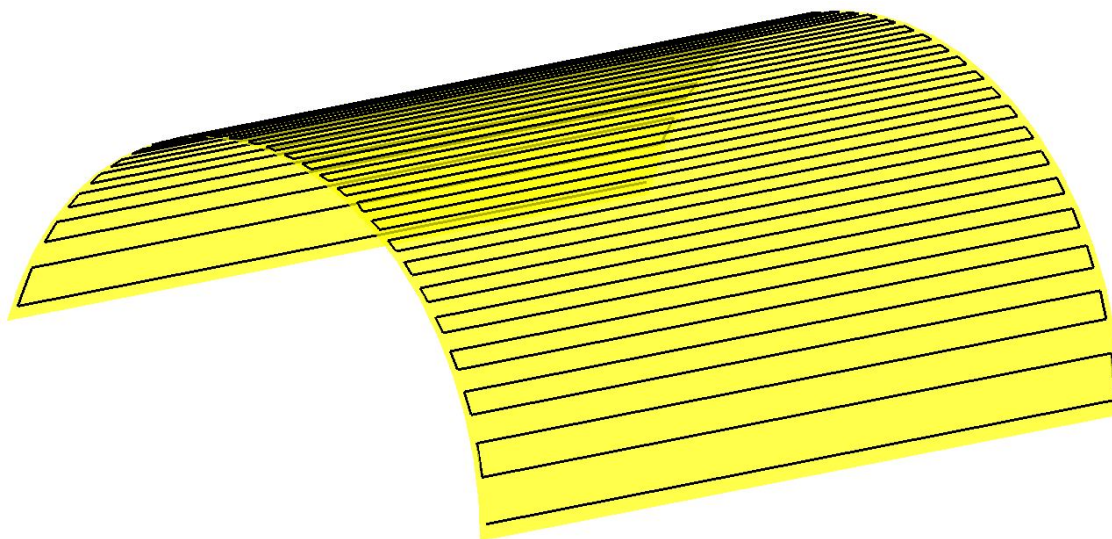


Figure 6.2 Limitation de la méthode des plans de coupe. La distance entre les passes adjacentes peut augmenter significativement, en particulier sur des surfaces de courbure élevées.

de courbure élevée), la méthode utilisant des courbes décalées peut être envisagée pour assurer une distance égale entre les passes adjacentes. À défaut de changer de méthode de génération de chemins pour conserver celle utilisant des plans de coupe, une autre solution serait de développer une méthode de calcul qui permette, à partir de la géométrie de la surface et de la direction de balayage, de déduire la distance suffisante entre les plans de coupe pour que la distance entre les passes adjacentes soit toujours inférieure à une valeur maximale fixée à l'avance. Ainsi, cela permettrait d'assurer qu'aucune zone ne soit oubliée durant l'inspection. Une autre approche pour limiter les augmentations de distance non contrôlées entre les passes serait de sur-segmenter la surface en diminuant la valeur maximale des angles de déviation de chaque passe, au détriment toutefois d'un temps de cycle long.

La vérification rigoureuse des chemins générés avec la méthodologie développée dans ce travail de recherche est importante pour améliorer la qualité d'inspection et éviter des dommages sur les outils d'inspection ou sur la cellule robotisée. Une étude devra donc être menée par la suite afin de contrôler avec précision si les chemins générés répondent bien aux critères des techniques de CND considérées. Par exemple, pour vérifier si la sonde EC reste toujours “juste en contact” et normale à la surface en tout point du chemin, une méthode serait de développer un outil “test” muni de capteurs mesurant l'écart de position et d'orientation de l'outil entre sa trajectoire réelle et sa trajectoire théorique. Un exemple d'outil test est représenté sur la Figure 6.3.

Enfin, une autre amélioration dont pourrait bénéficier le programme de génération de chemin développé, serait d'ajouter une étape qui permette de relier les différents chemins sur chaque patch. De plus, cette intégration des différents chemins en un seul chemin pourrait être optimisée pour

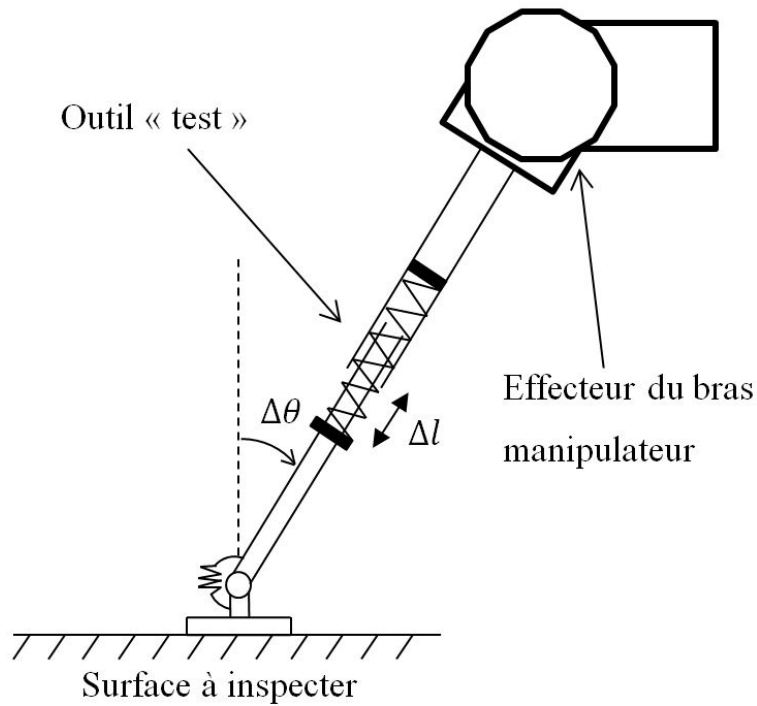


Figure 6.3 Exemple d'outil "test" mesurant l'erreur de positionnement et d'orientation dans le cas d'une inspection ECT.

réduire le temps du cycle de la sonde pour effectuer l'inspection de la surface globale. Par exemple, une étude a été proposée dans Tewolde et Sheng (2008) pour rechercher le parcours le plus court liant les extrémités de chemin de chacun des patches à l'aide d'un algorithme génétique ou encore d'un algorithme d'optimisation s'inspirant des colonies de fourmis.

CHAPITRE 7

CONCLUSION

L'objectif du projet MANU-418 dans lequel s'inscrit ce travail de recherche est de développer un système permettant d'automatiser, à l'aide d'un bras manipulateur à 6-DOF, trois techniques de CND fréquemment utilisées dans l'industrie aéronautique pour l'inspection de défauts en surface ou légèrement sous la surface : ECT, FPI et IRT. L'objectif spécifique qu'il est proposé d'atteindre dans ce travail de recherche est de développer une méthodologie ainsi qu'un outil logiciel pour la génération de chemins de couverture adaptés aux techniques d'inspection FPI, IRT et plus particulièrement ECT, sur des surfaces complexes de structures aéronautiques typiques. Jusqu'à présent, peu de recherches ont été menées afin d'automatiser ces techniques de contrôle non destructif sur des surfaces complexes sans aucune symétrie de révolution (cf. Section 2.1.2).

Avant de développer une méthodologie, des hypothèses de géométrie et de topologie des surfaces considérées sont formulées. De plus, il est supposé qu'on dispose à l'avance d'un modèle 3D de la surface à inspecter (cf. Section 1.3). La première étape de cette méthodologie consiste à mailler la surface avec des facettes triangulaires. Ce maillage servira alors de support à la génération de chemins. Ensuite, compte tenu de la complexité des surfaces considérées, une stratégie "diviser pour régner" est employée afin de simplifier la surface globale en patches plus "simples". En pratique, cette stratégie est mise en œuvre en appliquant trois segmentations différentes successivement sur la surface maillée afin de subdiviser celle-ci en patches de géométrie et de topologie plus simples. Ayant identifié le motif de chemins de couverture en zigzag comme étant le plus pertinent pour la réalisation de l'objectif, les chemins sont finalement générés sur chacun de ces patches en intersectant leurs facettes triangulaires avec des plans de coupe décalés à égale distance les uns par rapport aux autres. La méthodologie ainsi développée sera utilisée pour la génération de chemins de couverture sur toute la surface à inspecter (cf. Chapitre 4). Aussi, une autre méthodologie est étudiée dans ce travail de recherche pour générer des chemins autour d'une indication (par exemple une trace de pénétrant sur une surface après ressuage, révélant alors la présence éventuelle d'un défaut débouchant) dont la position et l'orientation sont supposées connues *a priori*. De manière similaire à la première méthodologie, les chemins sont générés perpendiculairement autour de cette indication en intersectant les facettes triangulaires à l'aide de plans espacés à distance égale et normaux le long de la ligne brisée représentative de l'indication (cf. Section 5.3). Ensuite, pour les deux méthodologies présentées, une fois les positions de l'outil définies le long d'un chemin, l'orientation de l'outil est calculée afin de respecter les contraintes d'une inspection ECT : l'axe de la sonde EC doit en tout temps être normal à la surface, et l'alignement de ses deux bobines doit

toujours être orienté le long de sa direction de déplacement (cf. Section 5.2). Aussi, afin d’adapter les chemins ainsi générés pour les techniques FPI et IRT, un paramètre de distance entre la surface et l’extrémité de l’outil d’inspection est défini (le contact correspond donc à une valeur nulle de ce paramètre).

Parallèlement, un outil logiciel muni d’une interface utilisateur a été développé dans l’environnement MATLAB afin de générer des chemins basés sur ces méthodologies (cf. Section 5.4). Un ensemble de paramètres de chemin sont proposés et modifiables par l’utilisateur pour que celui-ci puisse générer les chemins désirés (distance entre deux passes adjacentes, direction de balayage, etc.). Une fois les chemins calculés, une liste de coordonnées des positions et orientations des chemins sont exportés dans une table EXCEL pour pouvoir être utilisée par la suite par un robot. À partir des données de chemins exportés, des simulations de trajectoires (chemins exprimés par rapport au temps) d’un robot MotoMan (SV3X-L) sont effectuées avec le logiciel MotoSim (cf. Section 5.5). Après validation des trajectoires dans ce logiciel (absence de collisions, positions atteignables par le robot, etc.), des instructions sont finalement générées pour tester expérimentalement ces trajectoires sur le modèle réel du robot utilisé dans la simulation. Quelques tests expérimentaux des trajectoires du bras robotisé ont pu être effectués (cf. Section 5.6).

Les premières simulations et tests expérimentaux des chemins ainsi générés correspondent visuellement aux mouvements attendus pour les techniques d’inspection considérées, en particulier pour la technique ECT. Néanmoins, il conviendra à court terme de valider cette méthodologie avec davantage de tests expérimentaux. Par exemple, un outil “test” pourrait être conçu dans le but de vérifier son positionnement et son orientation vis-à-vis d’une surface complexe typique d’une structure aérospatiale, au regard des contraintes fixées pour les techniques de CND considérées. De plus, une limitation notable de cette méthodologie est le fait que la distance peut varier entre les passes adjacentes, effet intrinsèque de la méthode par plans de coupe sur des surfaces de courbure élevée. Une amélioration possible serait donc d’utiliser une autre méthode de génération de chemins telle que celle utilisant des courbes décalées (cf. Chapitre 6).

Sur le plus long terme, l’utilité de cette méthodologie repose sur le fait qu’elle peut s’adapter sur n’importe quel type de maillage, à condition que celui-ci ne soit pas disjoint. Par exemple, une surface inconnue ou ayant des incertitudes de géométrie pourrait être analysée dans un premier temps à l’aide d’un scanner 3D pour obtenir un maillage triangulaire de cette surface. Ce maillage pourrait ensuite être utilisé par le programme développé pour générer des chemins de couverture. Néanmoins, étant donné la taille généralement petite des triangles et les nombreuses irrégularités du maillage qui peuvent résulter du scan, il est très probable que le programme ne soit pas, pour l’instant, suffisamment robuste pour générer des chemins sur de telles surfaces. Notons également que cette méthodologie pourrait aussi bien être adaptée à la technique d’inspection par ultrasons dont les contraintes de chemins sont semblables à la technique ECT. Toutefois, un contrôle en force

devra être pris en compte pour cette technique car une pression constante doit être appliquée par l'instrument d'inspection (transducteur) sur la surface durant l'inspection.

RÉFÉRENCES

- ACAR, E. U. et CHOSET, H. (2002). Sensor-based coverage of unknown environments : incremental construction of Morse decompositions. *Int J Rob Res*, 21, 345–366.
- ACAR, E. U., CHOSET, H., RIZZI, A. A., ATKAR, P. N. et HULL, D. (2002). Morse decomposition for coverage tasks. *Int J Rob Res*, 21, 331–344.
- ACAR, E. U., CHOSET, H., ZHANG, Y. et SCHERVISH, M. (2003). Path planning for robotic demining : robust sensor-based coverage of unstructured environments and probabilistic methods. *Int J Rob Res*, 22, 441–466.
- ADAIR, T. L., WEHENER, D. H., KINDREW, M. G., WINTER, H. I. et MACCRACKEN, B. (1998). Automated fluorescent penetrant inspection (FPI) system is triple A. *AUTOTESTCON '98. IEEE Systems Readiness Technology Conference, 1998 IEEE*. Salt Lake City, UT, 498 – 529.
- ANTONIO, J. K., RAMABHADRAN, R. et LING, T. . L. (1997). A framework for optimal trajectory planning for automated spray coating. *International Journal of Robotics and Automation*, 12.
- ASAKAWA, N. et TAKEUCHI, Y. (1997). Teachingless spray-painting of sculptured surface by an industrial robot. *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on (IEEE Int Conf Robot Autom)*. vol. 3, 1875–1879.
- ATKAR, P. N., CHOSET, H. et RIZZI, A. A. (2003). Towards optimal coverage of 2-dimensional surfaces embedded in \mathbb{R}^3 : choice of start curve. *Intelligent Robots and Systems, 2003. Proceedings. 2003 IEEE/RSJ International Conference on (Rep U S)*. vol. 4, 3581–3587.
- ATKAR, P. N., CONNER, D. C., GREENFIELD, A., CHOSET, H. et RIZZI, A. A. (2009). Hierarchical segmentation of piecewise pseudoextruded surfaces for uniform coverage. *IEEE Trans Autom Sci Eng*, 6, 107–120.
- ATKAR, P. N., GREENFIELD, A., CONNER, D. C., CHOSET, H. et RIZZI, A. A. (2005). Uniform coverage of automotive surface patches. *The International Journal of Robotics Research*, 24, 883–898.
- ATMOSPHERES, S. (2013). Fluorescent penetrant inspection services. Consulté en Janvier 2013.
- BANGERT, M. (2009). Tech showcase : Eddy current today. Web site. Last accessed December 13th, 2012.
- BI, Z. M. et LANG, S. Y. T. (2007). A framework for CAD- and sensor-based robotic coating automation. *IEEE Trans Industr Inform*, 3, 84–91.

- BOBROW, J. E. (1985). NC machine tool path generation from CSG part representations. *Computer-Aided Design*, 17.
- BORUCKI, J. S. et JORDAN, G. (1989). *Nondestructive Evaluation and Quality Control*, ASM Handbook, vol. 17, chapitre Liquid Penetrant Inspection. 78–80.
- BUCKLEY, J. M. (2013). Basic eddy current testing, pt 1.
- CASTENADO, C. I. (2005). *Quantitative subsurface defect evaluation by pulsed phase thermography : depth retrieval with the phase*. Thèse de doctorat, Université Laval.
- CHEMIN, P. et DUBOSC, P. (2010). Dpcnews 020 – la levée de doute. Site Web de Ressuage Magnétoscopie DPC.
- CHEN, H., FUHLBRIGGE, T. et LI, X. (2009). A review of CAD-based robot path planning for spray painting. *Industrial Robot : An International Journal*, 36, 45–50.
- CHOSSET, H. et PIGNON, P. (1997). Coverage path planning : the boustrophedon cellular decomposition. *Proceedings of the International Conference on Field and Service Robotics*. Canberra, Australia.
- CRAIG, J. J. (2004). *Introduction to Robotics : Mechanics and Control (3rd Edition)*. Pearson Education International.
- CURRENT, J. R. et SCHILLING, D. A. (1989). Technical notes, the covering salesman problem. *Transportation Science*, 3.
- DE CARVALHO, R. N., VIDAL, H. A., VIEIRA, P. et RIBEIRO, M. I. (1997). Complete coverage path planning and guidance for cleaning robots. *Industrial Electronics, 1997. ISIE '97., Proceedings of the IEEE International Symposium on*. vol. 2, 677–682.
- DRAGOMATZ, D. et MANN, S. (1997). A classified bibliography of literature on NC milling path generation. *Comput Aided Des*, 29, 239–247.
- EDDINS, S. (2002). Matlab news & notes, the watershed transform. Web site. Last accessed December 13th, 2012.
- FENG, H. . Y. et LI, H. (2002). Constant scallop-height tool path generation for three-axis sculptured surface machining. *Computer-Aided Design*, 34, 647–654.
- GADOW, R., CANDEL, A. et FLORISTÁN, M. (2010). Optimized robot trajectory generation for thermal spraying operations and high quality coatings on free-form surfaces. *Surf Coat Technol*, 205, 1074–1079.
- JIN, J. et TANG, L. (2011). Coverage path planning on three-dimensional terrain for arable farming. *Journal of Field Robotics*, 28, 424–440.

- LUO, B., LIEBENBERG, B., RAYMONT, J. et SANTOSPIRITO, S. P. (2011). Automatic thermographic image defect detection of composites. *SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics, 80130S–1–80130S–9.
- MANGAN, A. et WHITAKER, R. (1999). Partitioning 3D surface meshes using watershed segmentation. *IEEE Trans Vis Comput Graph*, 5, 308–321.
- MARPOSS (2013). M110, automatic measuring machine for inspection of shafts (crankshafts, camshafts, gear shafts). www.marposs.com.
- MISRA, D., SUNDARARAJAN, V. et WRIGHT, P. K. (2005). Zig-zag tool path generation for sculptured surface finishing. *Dimacs Series in Discrete Mathematics and Theoretical Computer Science*. vol. 67, 265–280.
- MOTOMAN (2001). *MotoSim User's Manual (6.4)*, chapitre 6 – Converting Matrix to AXIS6 Data. 61.
- NDT EDUCATION RESOURCE CENTER (2013a). Basic processing steps of a liquid penetrant inspection.
- NDT EDUCATION RESOURCE CENTER (2013b). Introduction to thermal testing.
- NDT EDUCATION RESOURCE CENTER (2013c). Probes - mode of operation.
- PICHLER, A., VINCZE, M., ANDERSEN, H., MADSEN, O. et HAÜSLER, K. (2002). A method for automatic spray painting of unknown parts. *Proceedings of the 2002 IEEE International Conference on Robotics & Automation*. Washington, DC, 444–449.
- PIRES, J. N. (2007). *Industrial Robots Programming, Building Applications for the Factories of the Future*, chapitre Introduction to the Industrial Robotics World. 27–29.
- PROCECO (2013). Automated florescent penetrant inspection (FPI) – immersion systems.
- ROHMANN GMBH (2008). Eddy current test instruments and systems product catalogue.
- SATTAR, T. (2013). Mechatronics, robotics and non-destructive testing research centre.
- SATTAR, T. P. et BRENNER, A.-A. (2009). Robotic system for inspection of test objects with unknown geometry using NDT methods. *Ind Rob*, 36, 340–343.
- SCHMIDT, T., DUTTA, S. et ULLMANN, T. (2012). Production integrated NDT by means of automated thermography. *11th International Conference on Quantitative InfraRed Thermography (QIRT)*. Naples.
- SHENG, W., CHEN, H., XI, N. et CHEN, Y. (2005). Tool path planning for compound surfaces in spray forming processes. *IEEE Trans Autom Sci Eng*, 2, 240–249.
- SHENG, W., XI, N., SONG, M., CHEN, Y. et MACNEILLE, P. (2000). Automated CAD-guided robot path planning for spray painting of compound surfaces. *Intelligent Robots and Systems, 2000. Proceedings. 2000 IEEE/RSJ International Conference on (Rep U S)*. vol. 3, 1918–1923.

SPICER, J. M. et OSIANDER, R. (2002). *Nondestructive Evaluation ; Theory, Techniques, and Applications*, CRC Press, chapitre Active Thermography.

STARMAN, S. et MATZ, V. (2012). Automated system for crack detection using infrared thermographic testing. *4th International CANDU In-service Inspection Workshop and NDT in Canada 2012 Conference*. Toronto.

STARMANS ELECTRONICS LTD. (2013). Products, infrared testing. www.starmans.net.

SUH, S.-H., WOO, I.-K. et NOH, S.-K. (1991). Development of an automatic trajectory planning system (ATPS) for spray painting robots. *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on (IEEE Int Conf Robot Autom)*. vol. 3, 1948–1955.

TAM, H.-Y. (1999). Toward the uniform coverage of surfaces by scanning curves. *Comput Aided Des*, 31, 585–596.

TEWOLDE, G. S. et SHENG, W. (2008). Robot path integration in manufacturing processes : Genetic algorithm versus ant colony optimization. *IEEE Transactions on Systems, Man, and Cybernetics–Part A : systems and humans*, 38, 278–287.

VETTERLEIN, T., GEORGI, S., TIEDE, WAGENER, M. et RONGEN, H. (2006). Automated dye penetrant systems with process control and documentation in the aerospace industry. *ECNDT*. Forschungszentrum Jülich, Germany.

WU, Y. . H. et HSIAO, C. . C. (2002). Reliability assessment of automated eddy current system for turbine blades. *The e-Journal of Nondestructive Testing*, 7.

ANNEXE A**MANUEL UTILISATEUR****Programme de génération de chemins de couverture
pour des applications de contrôle non-destructif**

TABLE DES MATIÈRES

TABLE DES MATIÈRES	75
LISTE DES FIGURES	76
CHAPITRE 1 PRÉSENTATION GÉNÉRALE DU PROGRAMME	78
CHAPITRE 2 PRÉPARATION DES DONNÉES D'ENTRÉE	80
2.1 Préparation de la surface triangulée au format STL	80
2.1.1 Exportation de la surface globale	80
2.1.2 Exportation des sous-surfaces de la surface globale	83
2.2 Préparation des données d'une indication	85
CHAPITRE 3 UTILISATION DU PROGRAMME DE GÉNÉRATION DE CHEMINS DE COUVERTURE	90
3.1 Génération de chemins pour l'inspection globale d'une surface	90
3.1.1 Importation de la surface	92
3.1.2 Remaillage de la surface et segmentation "watershed"	92
3.1.3 Segmentation basée sur un angle de déviation maximal	95
3.1.4 Segmentation topologique basée sur des fonctions de Morse et généra- tion de chemins de couverture de la surface	96
3.1.5 Calcul de l'orientation et exportation du chemin	98
3.2 Génération de chemins pour l'inspection autour d'une indication de coordon- nées connues	99
3.2.1 Importation d'une indication	99
3.2.2 Génération de chemins de couverture autour d'une indication	101
3.2.3 Exportation du chemin	102
CHAPITRE 4 DONNÉES DE SORTIE ET LEUR UTILISATION FUTURE	105
CHAPITRE 5 VOCABULAIRE	109
Bibliographie	110

LISTE DES FIGURES

Figure 1.1	Architecture globale du programme de génération de chemins de couverture.	78
Figure 2.1	Exemple de modèle 3D d'une pièce à inspecter.	81
Figure 2.2	Sélection des surfaces de l'extraction multiple. Dans l'arborescence, l'encadré rouge indique l'extraction multiple de la surface globale à inspecter.	82
Figure 2.3	L'exportation d'une surface disjointe peut empêcher le programme de génération de chemins de fonctionner correctement.	83
Figure 2.4	Des portions surface non sélectionnées à l'intérieur de la surface à inspecter peuvent résulter en des chemins non désirés.	84
Figure 2.5	Facettisation de la surface extraite.	85
Figure 2.6	Exportation de la surface maillée au format STL (langage ASCII). . . .	86
Figure 2.7	Illustration d'une sous-surface. La sous-surface sélectionnée en rouge est bordée par des arêtes mortes.	87
Figure 2.8	Extraction des sous-surfaces de la surface globale. Dans l'arborescence, l'encadré rouge indique les différents éléments extraits.	87
Figure 2.9	Inspection sous un éclairage ultraviolet des traces de pénétrant fluorescent (en vert) après ressuage. Extrait de [1].	88
Figure 2.10	Points représentatifs d'une indication (représentée ici par une ligne blanche). Ces points doivent être ordonnés le long de l'indication. . . .	88
Figure 2.11	Liste des coordonnées des points représentatifs d'une indication.	89
Figure 3.1	Fenêtre de démarrage du programme.	90
Figure 3.2	Création automatiques des fichiers de types MAT et FIG après calcul de chaque étape de la procédure de génération de chemins.	91
Figure 3.3	Fenêtre après importation de la surface STL.	93
Figure 3.4	Surface triangulée d'un parallélépipède rectangle. Le maillage n'a pas été raffiné dans ce cas-ci.	94
Figure 3.5	Surface remaillée. Un sommet est à présent défini au milieu de chacune des faces.	95
Figure 3.6	Étape de remaillage du programme.	96
Figure 3.7	Segmentation watershed de la surface pour $depth_thresh = 0, 1$	97
Figure 3.8	Segmentation basée sur un angle maximal de déviation $ang_dev_thresh = 100^\circ$	98
Figure 3.9	Paramètres des chemins de couverture d'une surface.	99
Figure 3.10	Génération des chemins de couverture de la surface. La segmentation par fonction de Morse a été effectuée et les paramètres de chemin sont $sweep_dir = 0^\circ$, $d = 5mm$, $Os_d = -2mm$, $offset_edge = 1mm$ et $adj_thresh = 1mm$	100

Figure 3.11	Génération des chemins de couverture de la surface. La segmentation par fonction de Morse n'a pas été effectuée et les paramètres de chemin sont $sweep_dir = 0^\circ$, $d = 5mm$, $Os_d = -2mm$, $offset_edge = 1mm$ et $adj_thresh = 1mm$	101
Figure 3.12	Représentation d'une indication (en vert clair sur cette figure) dans l'environnement Catia.	102
Figure 3.13	Représentation d'une indication dans le programme.	103
Figure 3.14	Paramètres du chemin de couverture autour d'une indication.	104
Figure 4.1	tableau EXCEL des chemins de couverture d'une surface. Chacun des patches ont leur propre chemin de couverture. Les trois premières colonnes correspondent aux coordonnées de position (x, y, z) et les trois dernières colonnes correspondent aux coordonnées d'orientation (R_x, R_y, R_z) .106	106
Figure 4.2	tableau EXCEL du chemin de couverture autour d'une indication. Un seul chemin continu est généré autour de l'indication. Les trois premières colonnes correspondent aux coordonnées de position (x, y, z) et les trois dernières colonnes correspondent aux coordonnées d'orientation (R_x, R_y, R_z)	107
Figure 4.3	Importation d'un chemin dans le logiciel MotoSim. Extrait de [4]. . . .	108

CHAPITRE 1

PRÉSENTATION GÉNÉRALE DU PROGRAMME

Ce programme de génération de chemins de couverture adaptés pour le contrôle non destructif propose deux options :

- génération de chemins de couverture pour l'inspection globale d'une surface,
- génération de chemins de couverture pour l'inspection autour d'une indication de coordonnées connues *a priori*.

Comme décrit dans la Figure 1.1, ce programme accepte deux données d'entrées :

- Un modèle de surface représenté par un maillage triangulaire. Le format de fichier accepté est le **format stéréolithographie (STL)** et son contenu doit être dans le **langage ASCII** (le langage binaire n'est pas reconnu par le programme). Cette donnée est obligatoire pour procéder à une génération de chemin.
- Un modèle d'indication décrit par une liste ordonnée de ses points caractéristiques dans un tableau EXCEL. Le format de fichier accepté est le format XLS (Excel 2003). Cette donnée est nécessaire pour la génération de chemins autour d'une indication, mais ne l'est pas pour générer des chemins d'inspection globale de la surface.

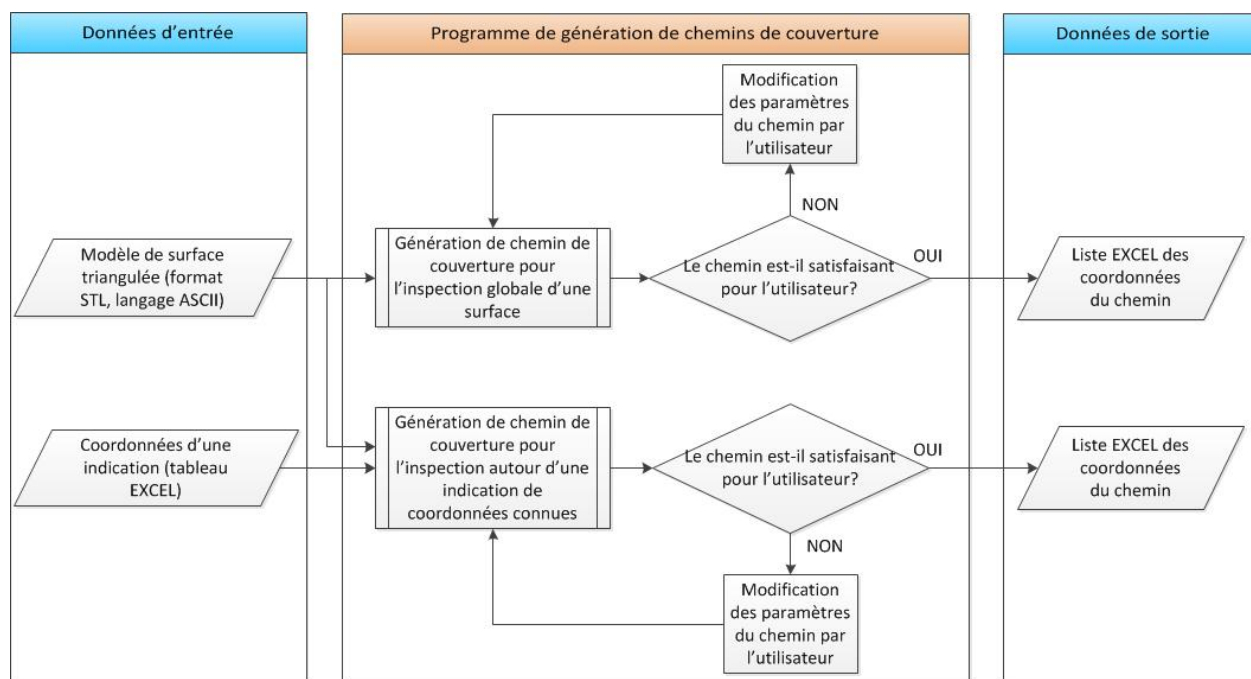


Figure 1.1 Architecture globale du programme de génération de chemins de couverture.

À partir de ces données d'entrée, l'utilisateur a la possibilité de générer un chemin d'inspection globale de la surface importée et/ou de générer un chemin autour d'une indication importée (si c'est le cas). Pour les deux options, des paramètres de chemin par défaut sont

proposés, mais si l'utilisateur n'est pas satisfait du résultat visuel du chemin, il peut générer de nouveau des chemins en changeant directement les paramètres. Une fois que l'utilisateur est satisfait du résultat, il peut alors exporter les chemins générés. Ces chemins seront automatiquement exportés dans un tableau EXCEL (Excel 2003), contenant la liste ordonnée des coordonnées (x, y, z) des points du chemin ainsi que les coordonnées d'orientation associées (R_x, R_y, R_z) .

La section 2 présente une méthode pour préparer les données d'entrée du programme. La section 3 détaille ensuite les différentes fonctions de ce logiciel. Enfin, la section 4 décrit les données de sortie.

CHAPITRE 2

PRÉPARATION DES DONNÉES D'ENTRÉE

Dans cette section, des méthodes sont suggérées pour préparer les deux types de données d'entrée que le programme utilise : un modèle de la surface approché par un maillage triangulaire (format STL avec langage ASCII) et un modèle d'une indication décrit par une liste de ses points caractéristiques.

2.1 Préparation de la surface triangulée au format STL

Dans cette sous-section, deux procédures sont suggérées pour préparer la surface à inspecter dans un fichier au format STL. Cette préparation peut être faite dans un logiciel de conception assistée par ordinateur (CAO). Dans ce manuel, le logiciel utilisé est CATIA V5R20. Une méthode d'exportation similaire pourrait être appliquée dans d'autres logiciels de CAO, mais il est important que le fichier STL ait une présentation similaire à la Figure 2.6 (le détail de l'analyse du fichier STL par le programme se trouve dans la fonction “rndread.m” de ce programme). Dans les deux procédures, il est supposé que le modèle de la pièce que l'on souhaite inspecter soit disponible. À titre d'exemple, le modèle 3D considéré dans cette section est le suivant (cf. Figure 2.1) et la surface choisie pour l'inspection est représentée en couleur claire :

2.1.1 Exportation de la surface globale

La procédure ci-dessous explique comment exporter dans CATIA (V5R20) la surface à inspecter dans un unique fichier STL.

1. Extraire les surfaces à inspecter : dans l'atelier **Shape/Generative Shape Design**, utiliser la commande **Extraction multiple** . Sélectionner toutes les portions de la surface à inspecter (cf. Figure 2.2).

Remarque 1 : Il est important durant cette phase de sélectionner des surfaces qui ne sont pas disjointes (cf. Figure 2.3). Autrement, le programme risque de ne pas fonctionner. De même, si des portions de surfaces à l'intérieur de la surface à inspecter sont oubliées durant cette extraction (cf. Figure 2.4), les résultats de la génération de chemin peuvent en être affectés.

Remarque 2 : Si la sélection des surfaces en vert ne s'affiche pas correctement, régler la précision d'affichage en 3D dans **Outils/Options.../Général/Affichage** puis onglet “**Performances**”, et diminuer la précision fixe à la valeur 0,15.

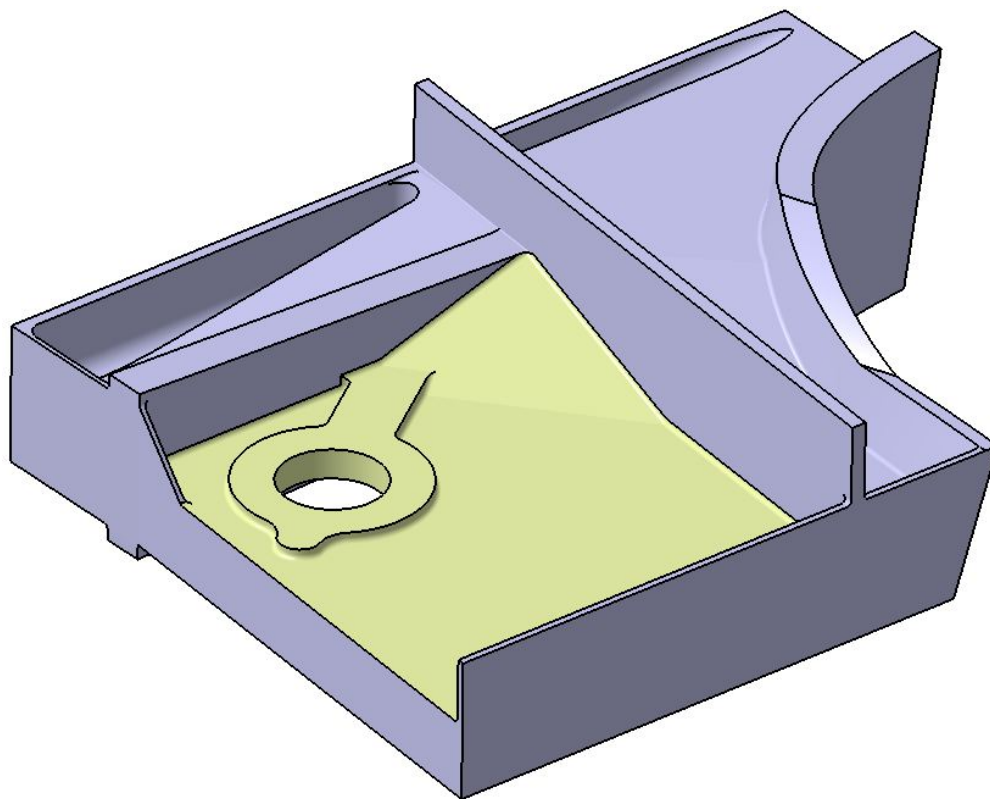


Figure 2.1 Exemple de modèle 3D d'une pièce à inspecter.

Remarque 3 : Afin de ne pas oublier certaines portions de surface, il est recommandé de visualiser le modèle dans un **rendu réaliste avec arête**.

2. Facettisation de la surface extraite : dans l'atelier **Usinage/STL Prototypage Rapide**, cliquer sur la commande **Facettisation** ("Tessellation" en anglais), sélectionner la surface extraite et régler les paramètres du maillage (la flèche et le pas du maillage). Cliquer sur OK.

Remarque 1 : Si cet atelier n'est pas disponible, une alternative possible est d'aller dans l'atelier **Shape/Shape Sculptor** et d'utiliser la commande **Facettiser** ("Tessellate" en anglais) qui est identique à celle de l'atelier **STL Prototypage Rapide**.

Remarque 2 : En visualisant dans CATIA avec un **rendu filaire**, il sera plus simple de contrôler à l'œil la taille du maillage.

Remarque 3 : Selon la précision souhaitée du maillage, il est possible de diminuer la flèche afin que le maillage respecte davantage la géométrie de la surface. Néanmoins, il est recommandé de ne pas mailler la surface avec trop de facettes car plus le nombre de facettes est élevé, plus le temps de calcul du programme de génération de chemins

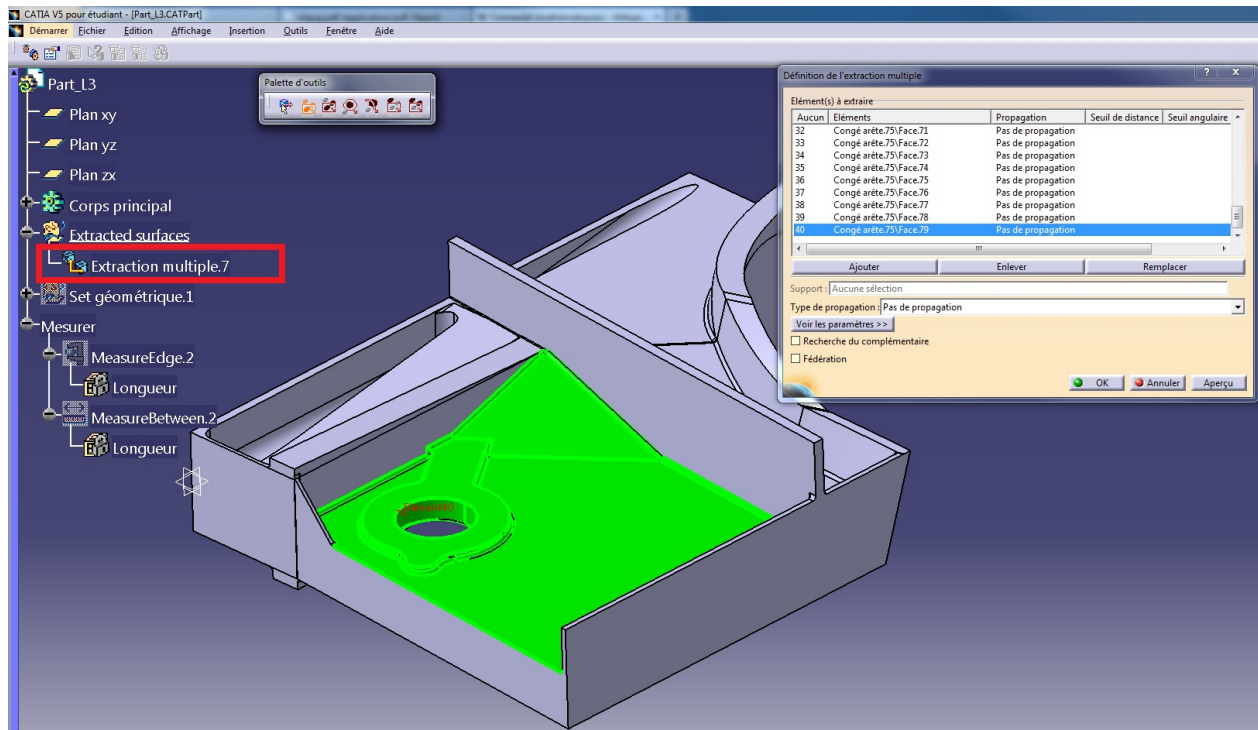


Figure 2.2 Sélection des surfaces de l'extraction multiple. Dans l'arborescence, l'encadré rouge indique l'extraction multiple de la surface globale à inspecter.

augmente. Pour vérifier le nombre de facettes (triangles) du maillage, cliquer sur la commande **Informations**. Une surface triangulée initialement par environ 500 facettes est une quantité acceptable vis-à-vis du temps de calcul pour la génération de chemins sur celle-ci.

Remarque 4 : Il est important de vérifier que les normales des facettes triangulaires soient orientées **vers l'extérieur de la pièce**. En effet, l'orientation des normales est utilisée dans le calcul de l'orientation de l'outil du robot en chacun des points du chemin.

3. Enregistrer le maillage au format STL : **cacher** dans un premier temps tous les éléments de l'arborescence, **excepté le maillage**. Ensuite, cliquer sur **File/Enregistrer sous...**, sélectionner le format STL dans le type de fichiers, attribuer un nom et cliquer sur OK. Une fois enregistré, le fichier STL ouvert dans un éditeur de texte devrait ressembler à la Figure 2.6 ci-dessous.
4. Dans le dossier "Surfaces" du répertoire du programme de génération de chemins, créer un dossier portant exactement le même nom que le fichier STL précédemment enregistré. Puis copier et coller le fichier STL dans ce nouveau dossier. Il sera alors plus facile pour l'utilisateur d'importer la surface dans le programme et permettra d'éviter toute confusion lors de l'importation d'une indication (cf. Section 2.2).

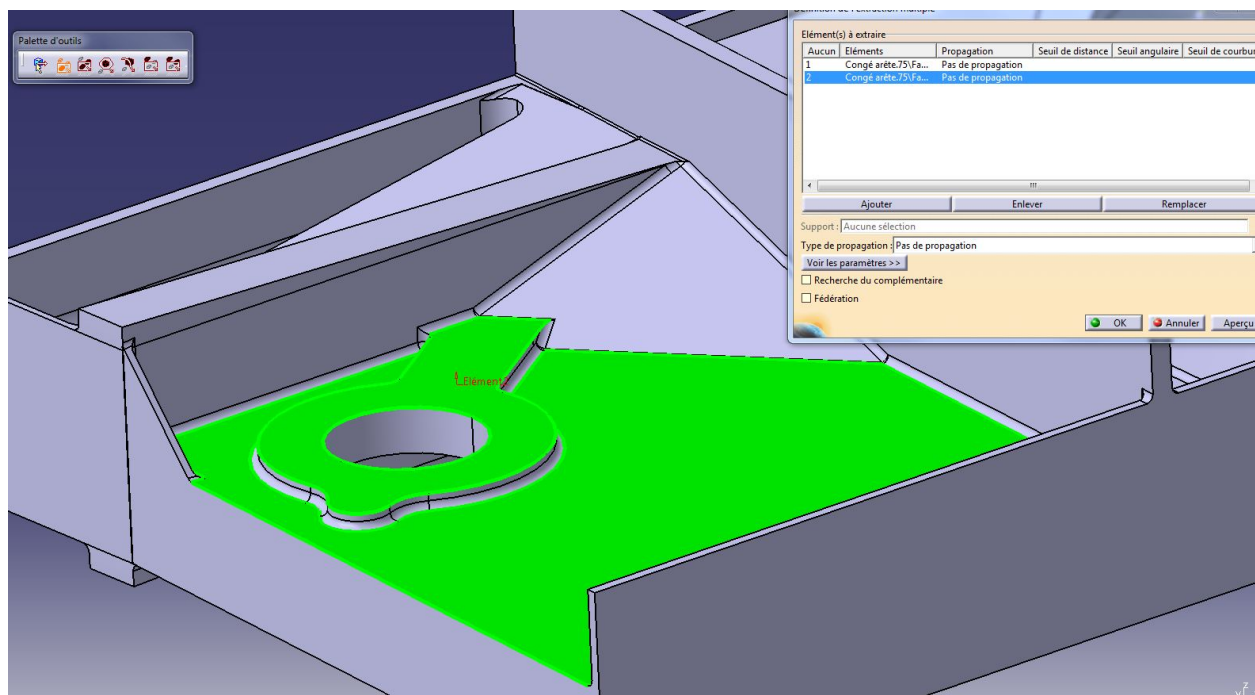


Figure 2.3 L'exportation d'une surface disjointe peut empêcher le programme de génération de chemins de fonctionner correctement.

Remarque 1 : Il est important de cacher tous les composants à l'exception du maillage avant enregistrement au format STL. Sinon, d'autres composantes (autres surfaces de la pièce par exemple) seront maillées automatiquement et enregistrées dans ce même fichier STL.

Remarque 2 : Ne pas utiliser la commande **Export en fichier STL** dans l'atelier **Usinage/STL Prototypage Rapide** (ou **Exporter** dans l'atelier **Shape/Shape Sculptor**) pour enregistrer la surface maillée au format STL. En effet, les types de fichier STL proposés sont soit en langage binaire (*.stl), soit en langage ASCII, *mais différent du modèle tel que présenté Figure 2.6*. Le programme de génération de chemins analyse uniquement des types de fichiers STL présentés tels que celui de la Figure 2.6 (cf. fonction "rndread.m" du programme).

2.1.2 Exportation des sous-surfaces de la surface globale

À la différence de la première méthode qui consiste à exporter la surface globale dans un fichier STL unique, la deuxième méthode présentée ici consiste à exporter séparément des sous-surfaces de la surface globale dans des fichiers STL indépendants. On appelle ici *sous-surfaces* toutes surfaces paramétriques composant une surface plus globale. Ces sous-surfaces sont limitées par des arêtes vives et/ou *mortes* (cf. Figure 2.7).

La première étape de cette deuxième méthode est similaire à celle de la première méthode, mais elle consiste à extraire les sous-surfaces désirées dans des éléments séparés (cf.

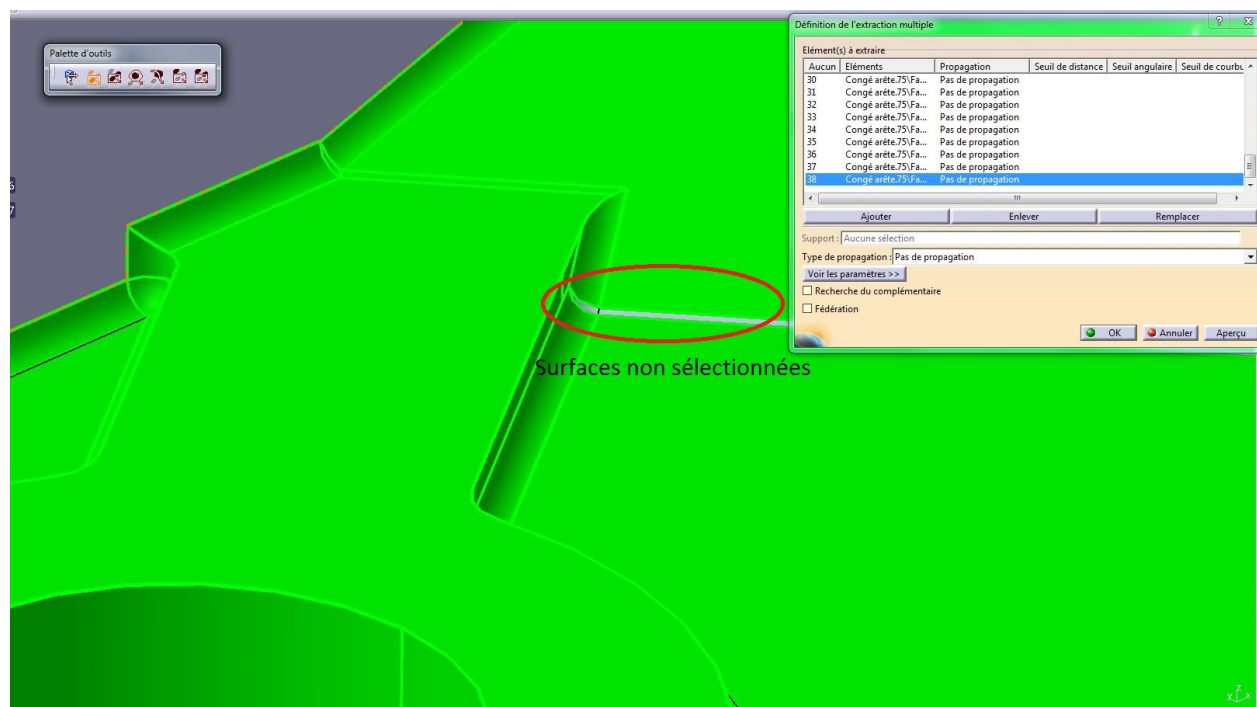


Figure 2.4 Des portions surface non sélectionnées à l'intérieur de la surface à inspecter peuvent résulter en des chemins non désirés.

Figure 2.8). Une autre méthode consiste à exporter des portions de la surface globale dans des fichiers STL séparés, et non la surface globale même directement dans un seul fichier STL.

Remarque 1 : La commande **extraction multiple** peut être utilisée pour extraire dans un seul élément plusieurs petites sous-surfaces qui sont jugées appropriées pour être rassemblées dans un même fichier STL (cf. Figure 2.8).

Remarque 2 : De la même manière que dans la première méthode, chaque élément extrait à l'aide de l'extraction multiple doit être connexe (non disjoints).

Ensuite, pour chacun des éléments extraits, les étapes 2, 3 et 4 de la première méthode sont appliquées.

Remarque 3 : Chaque élément de surface extrait doit avoir sa propre facettisation. Au moment de l'enregistrement, cacher tous les composants excepté la facettisation que l'on souhaite exporter.

Cette méthode est plus fastidieuse pour l'utilisateur. Néanmoins, elle permet plus de possibilités pour la génération de chemin par rapport à la première méthode, car chacune des portions de la surface maillée exportée dans un fichier STL séparé aura ses propres paramètres

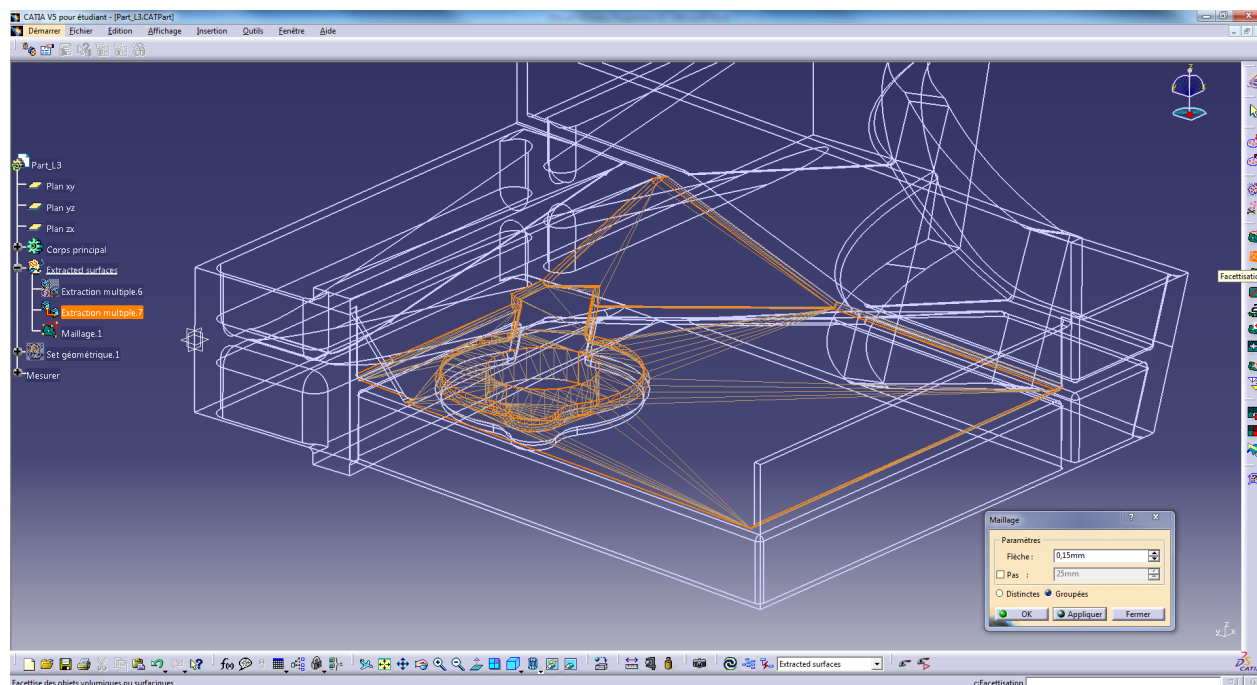


Figure 2.5 Facettisation de la surface extraite.

de chemin (orientation du balayage, distance entre deux passes consécutives). Il sera alors plus simple pour l'utilisateur de générer les chemins désirés.

De plus, l'algorithme de génération de chemin étant appliqué séparément sur des surfaces plus petites que la surface globale considérée dans la première méthode, le temps de calcul des chemins pour chacune des surfaces est réduit (moins de facettes triangulaires sont traitées).

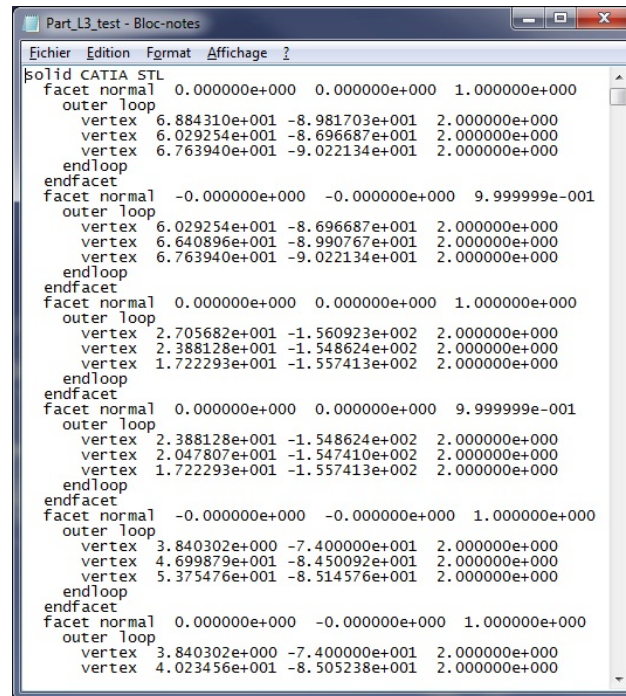
2.2 Préparation des données d'une indication

Cette section présente comment modéliser une indication qui pourra être utilisé par la suite dans le programme de génération de chemins.

Une *indication* est un terme utilisé dans des techniques de contrôle non destructif par ressuage (*Fluorescent Penetrant Inspection*, par exemple) et correspond aux traces visibles de pénétrant sur la surface après application du révélateur. Ces traces peuvent représenter aussi bien la localisation de défauts débouchants pertinents (fissures ou autres “vides” formés à la surface surface), que celle de défauts non pertinents (rugosité, marque d'outil, tâche d'huile, etc.).

Les fissures révèlent souvent des indications linéaires. Dans ce programme de génération de chemins, une indication sera représentée par une **liste ordonnée des points représentant cette indication**. Plus il y a de points pour décrire une indication, plus celle-ci sera représentée avec précision dans le programme. La Figure 2.10 montre un exemple d'indication ainsi que des points représentatifs de celle-ci.

Remarque 1 : Une indication doit être représentée par au moins deux points distincts.



```

Solid CATIA STL
facet normal 0.000000e+000 0.000000e+000 1.000000e+000
  outer loop
    vertex 6.884310e+001 -8.981703e+001 2.000000e+000
    vertex 6.029254e+001 -8.696687e+001 2.000000e+000
    vertex 6.763940e+001 -9.022134e+001 2.000000e+000
  endloop
endfacet
facet normal -0.000000e+000 -0.000000e+000 9.999999e-001
  outer loop
    vertex 6.029254e+001 -8.696687e+001 2.000000e+000
    vertex 6.640896e+001 -8.990767e+001 2.000000e+000
    vertex 6.763940e+001 -9.022134e+001 2.000000e+000
  endloop
endfacet
facet normal 0.000000e+000 0.000000e+000 1.000000e+000
  outer loop
    vertex 2.705682e+001 -1.560923e+002 2.000000e+000
    vertex 2.388128e+001 -1.548624e+002 2.000000e+000
    vertex 1.722293e+001 -1.557413e+002 2.000000e+000
  endloop
endfacet
facet normal 0.000000e+000 0.000000e+000 9.999999e-001
  outer loop
    vertex 2.388128e+001 -1.548624e+002 2.000000e+000
    vertex 2.047807e+001 -1.547410e+002 2.000000e+000
    vertex 1.722293e+001 -1.557413e+002 2.000000e+000
  endloop
endfacet
facet normal -0.000000e+000 -0.000000e+000 1.000000e+000
  outer loop
    vertex 3.840302e+000 -7.400000e+001 2.000000e+000
    vertex 4.699879e+001 -8.450092e+001 2.000000e+000
    vertex 5.375476e+001 -8.514576e+001 2.000000e+000
  endloop
endfacet
facet normal 0.000000e+000 -0.000000e+000 1.000000e+000
  outer loop
    vertex 3.840302e+000 -7.400000e+001 2.000000e+000
    vertex 4.023456e+001 -8.505238e+001 2.000000e+000
  endloop

```

Figure 2.6 Exportation de la surface maillée au format STL (langage ASCII).

Remarque 2 : Les positions (x, y, z) des points représentatifs de l'indication doivent être exprimées par rapport au **référentiel absolu de la surface**.

Cette liste de points doit être présentée dans un tableau EXCEL (Excel 2003) tel que représenté Figure 2.11. Chaque ligne contenant des valeurs numériques correspondent aux coordonnées (x, y, z) d'un point de l'indication. Il faut noter aussi que l'ordre des lignes correspond à l'ordre dans lesquels les points sont placés les uns par rapport aux autres (plus de détails sont fournis dans la fonction de la librairie Matlab "xlsread").

Ensuite, afin d'éviter toute confusion, enregistrer cette table EXCEL (Excel 2003) dans le dossier de la surface correspondante (Répertoire du programme/Surfaces/NomSurfaceCorrespondante) et adopter un nom pour ce fichier du type "NomSurfaceCorrespondante_indication_NomeroIndication.xls".

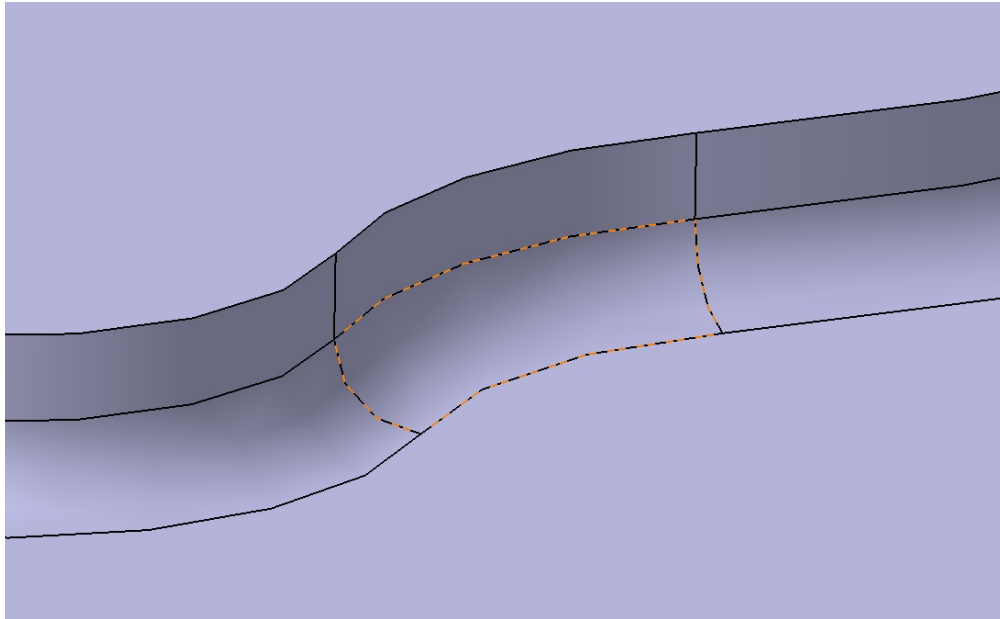


Figure 2.7 Illustration d'une sous-surface. La sous-surface sélectionnée en rouge est bordée par des arêtes mortes.

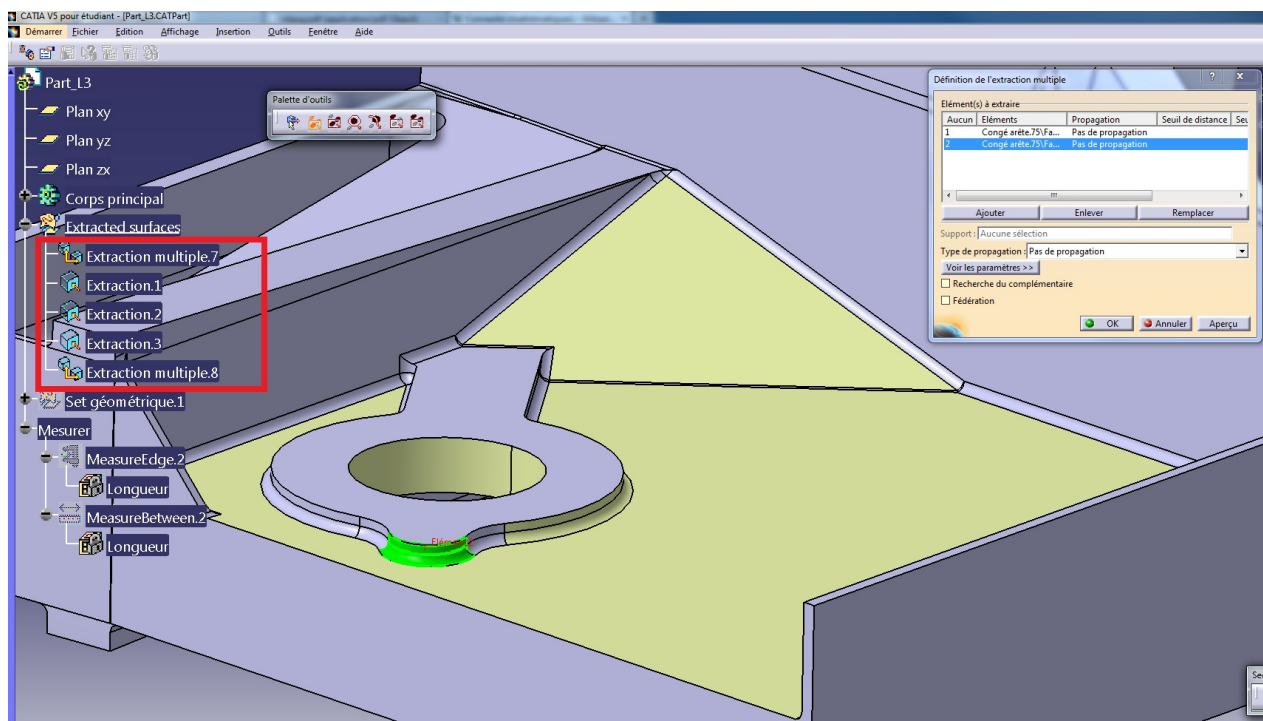


Figure 2.8 Extraction des sous-surfaces de la surface globale. Dans l'arborescence, l'encadré rouge indique les différents éléments extraits.



Figure 2.9 Inspection sous un éclairage ultraviolet des traces de pénétrant fluorescent (en vert) après ressuage. Extrait de [1].

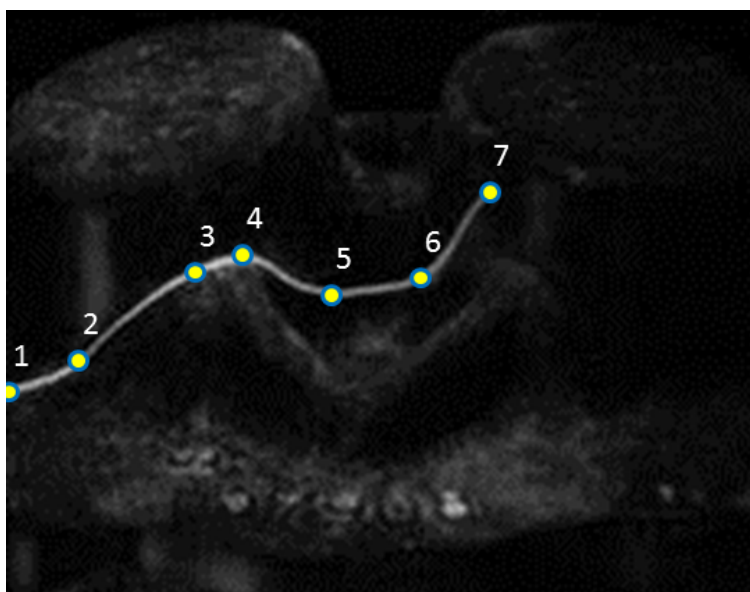
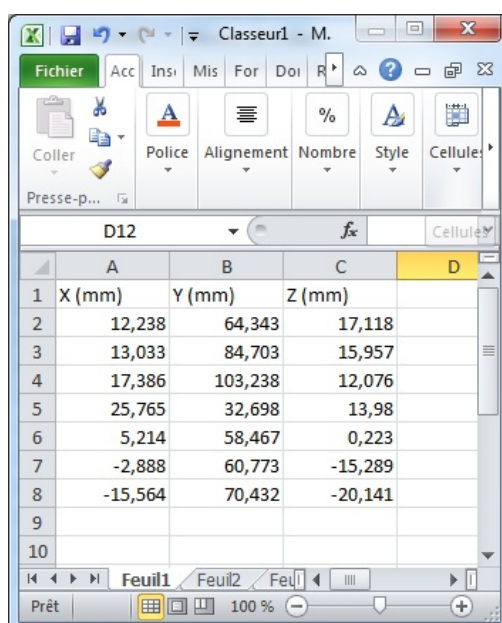


Figure 2.10 Points représentatifs d'une indication (représentée ici par une ligne blanche). Ces points doivent être ordonnés le long de l'indication.



	A	B	C	D
1	X (mm)	Y (mm)	Z (mm)	
2	12,238	64,343	17,118	
3	13,033	84,703	15,957	
4	17,386	103,238	12,076	
5	25,765	32,698	13,98	
6	5,214	58,467	0,223	
7	-2,888	60,773	-15,289	
8	-15,564	70,432	-20,141	
9				
10				

Figure 2.11 Liste des coordonnées des points représentatifs d'une indication.

CHAPITRE 3

UTILISATION DU PROGRAMME DE GÉNÉRATION DE CHEMINS DE COUVERTURE

Il est supposé à présent que le fichier STL de la surface (ou sous-surface) à inspecter existe et est placé dans le dossier correspondant de cette surface (cf. Section 2.1, étape 4). Pour démarrer le programme, il faut exécuter le script Matlab “START_PROGRAM”. La fenêtre Figure 3.1 apparaît alors.

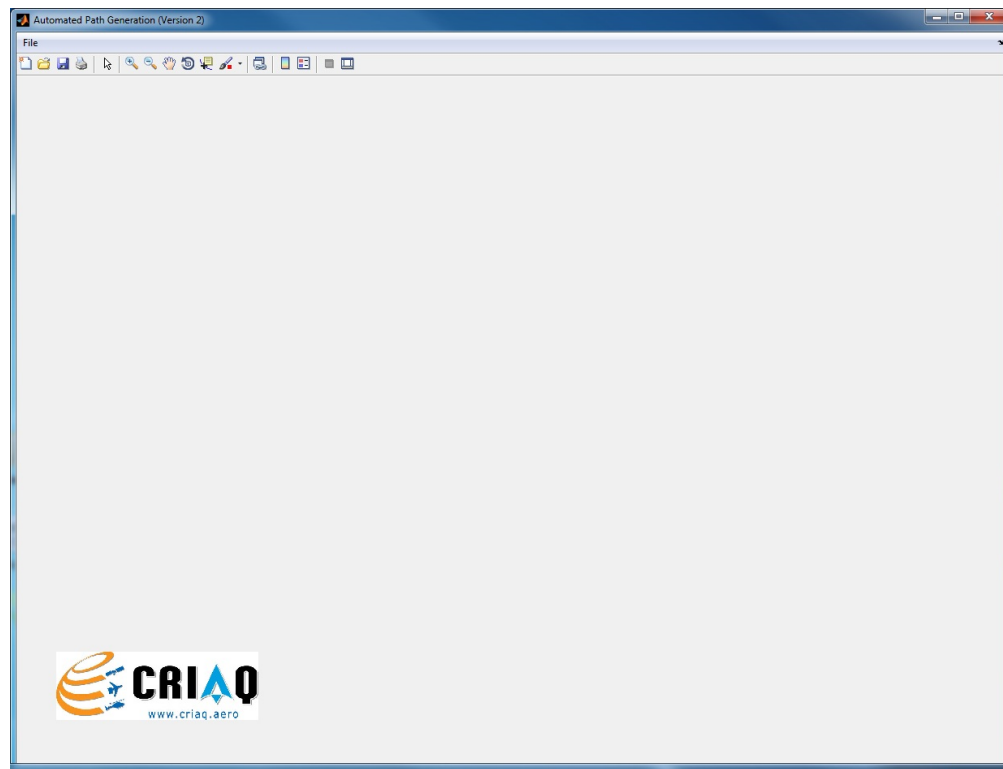


Figure 3.1 Fenêtre de démarrage du programme.

3.1 Génération de chemins pour l’inspection globale d’une surface

Cette section explique la procédure à suivre pour générer des chemins sur une surface à inspecter. Cette procédure s’effectue en 5 étapes :

1. Importation de la surface.
2. Remaillage de la surface et segmentation basée sur la courbure locale de la surface : segmentation “watershed”.

3. Segmentation basée sur la topologie de la surface utilisant des fonctions de Morse (segmentation optionnelle).
4. Génération des chemins.
5. Calcul de l'orientation de l'outil et exportation du chemin.

Noter que, pour chacune de ces étapes, des fichiers Matlab *.MAT et *.FIG associés sont automatiquement créés dans les répertoires respectifs “simulation_results/NomSurfaceCorrespondante/MAT_files” et “simulation_results/NomSurfaceCorrespondante/MAT_files”. Si ces répertoires n'existent pas déjà, ils seront aussi créés automatiquement (cf. Figure 3.2).

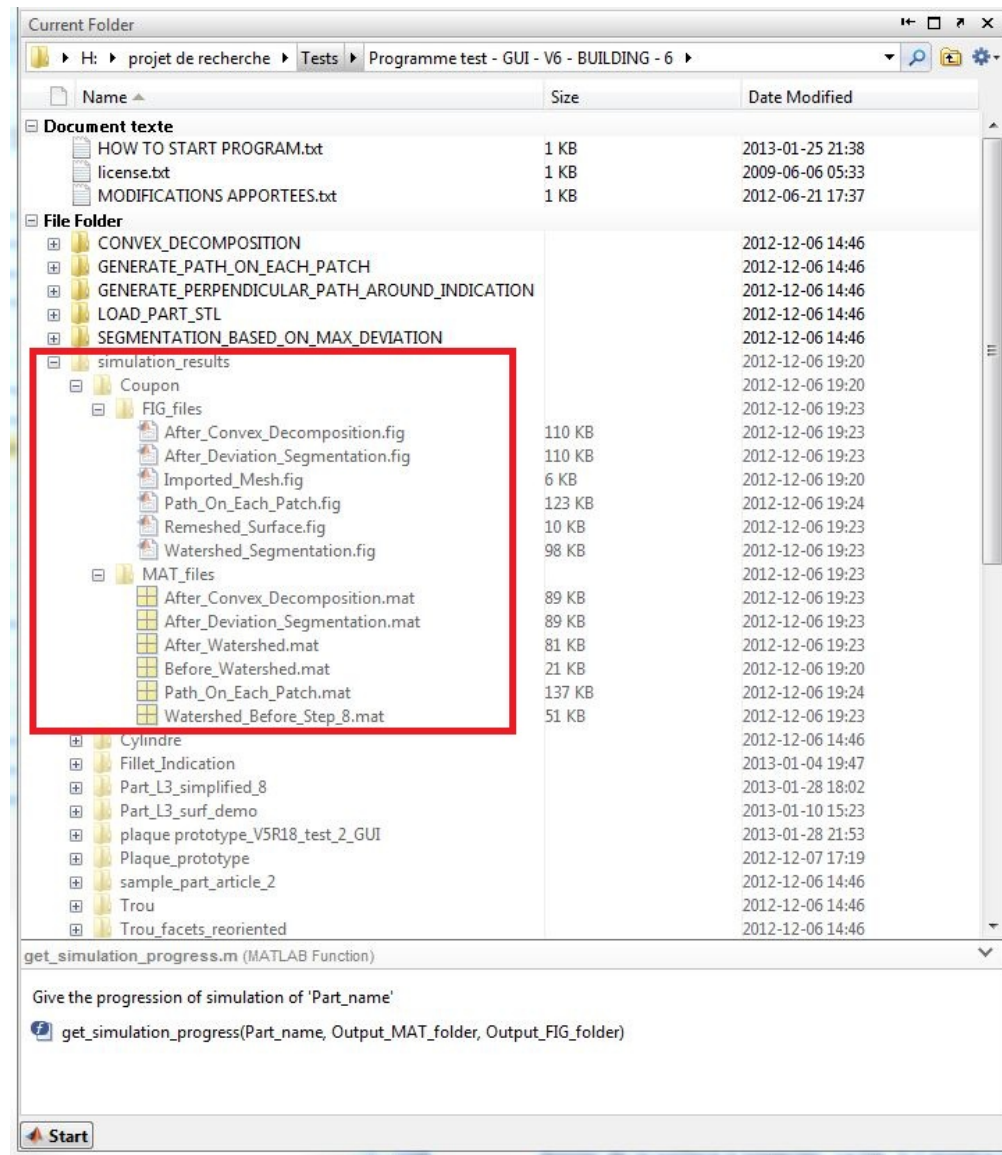


Figure 3.2 Création automatiques des fichiers de types MAT et FIG après calcul de chaque étape de la procédure de génération de chemins.

Pour que le programme ne renvoie pas d’erreur, il est important d’effectuer ces étapes dans cet ordre, sans manquer une étape. Néanmoins, il est possible pour l’utilisateur de recommencer une étape déjà effectuée pour la tester avec d’autres paramètres. Un message d’avertissement va alors prévenir l’utilisateur que ce “retour en arrière” va impliquer la suppression des fichiers de type *.MAT et *.FIG associés à toutes les étapes qui suivent l’étape que l’utilisateur veut recalculer (les précédents fichiers de cette étape seront également écrasés).

3.1.1 Importation de la surface

Pour importer une surface, cliquer sur le menu **File/Import STL file...** et sélectionner le fichier STL dans le dossier de la surface à inspecter. La Figure 3.3 montre la fenêtre obtenue (pour le cas de la surface considérée Figure 2.1). Sur cette figure, quatre onglets sont disposés pour l’application des prochaines étapes en vue de générer les chemins de couverture. Il est possible de manipuler la surface avec les outils habituels de visualisation de Matlab, tels que le zoom, la rotation et le déplacement latéral de la surface.

Astuce : Pour revenir rapidement au point de vue par défaut de la surface, double-cliquez sur cette surface.

Une fois l’importation effectuée, l’étape de remaillage du premier onglet peut être abordée, objet du paragraphe suivant.

3.1.2 Remaillage de la surface et segmentation “watershed”

Cette étape a lieu dans le premier onglet.

Contexte de l’étape de remaillage

La première segmentation appelée segmentation “watershed”, a pour objectif de “découper” la surface en plusieurs patchs dont les lignes de coupe sont formées au niveau des zones ayant des maxima locaux de courbure. Cette segmentation calcule dans un premier temps la valeur de courbure¹ en chacun des sommets du maillage et identifie ceux ayant un minimum local de courbure. Ensuite, les sommets restants sont regroupés autour du sommet ayant un minimum local de courbure et se trouvant sur le chemin de plus grande pente (en termes de valeur de courbure).

Cependant, un inconvénient du maillage triangulaire est que certaines régions relativement planes ne contiennent pas suffisamment de sommets pour que la segmentation watershed puisse détecter un sommet ayant un minimum local de courbure [3]. Pour illustrer ce phénomène, considérons le maillage de la figure Figure 3.4. On constate que chacune des faces est décrite uniquement par quatre sommets correspondant aux coins du parallélépipède. La valeur de courbure pour ces sommets sera alors élevée et aucun sommet ayant un minimum local de courbure, représentatif de chacune des faces, ne sera détecté. Cette segmentation ne

1. Plus précisément, il s’agit d’une valeur représentative de la courbure entre 0 et 1. Son calcul est détaillé dans [3].

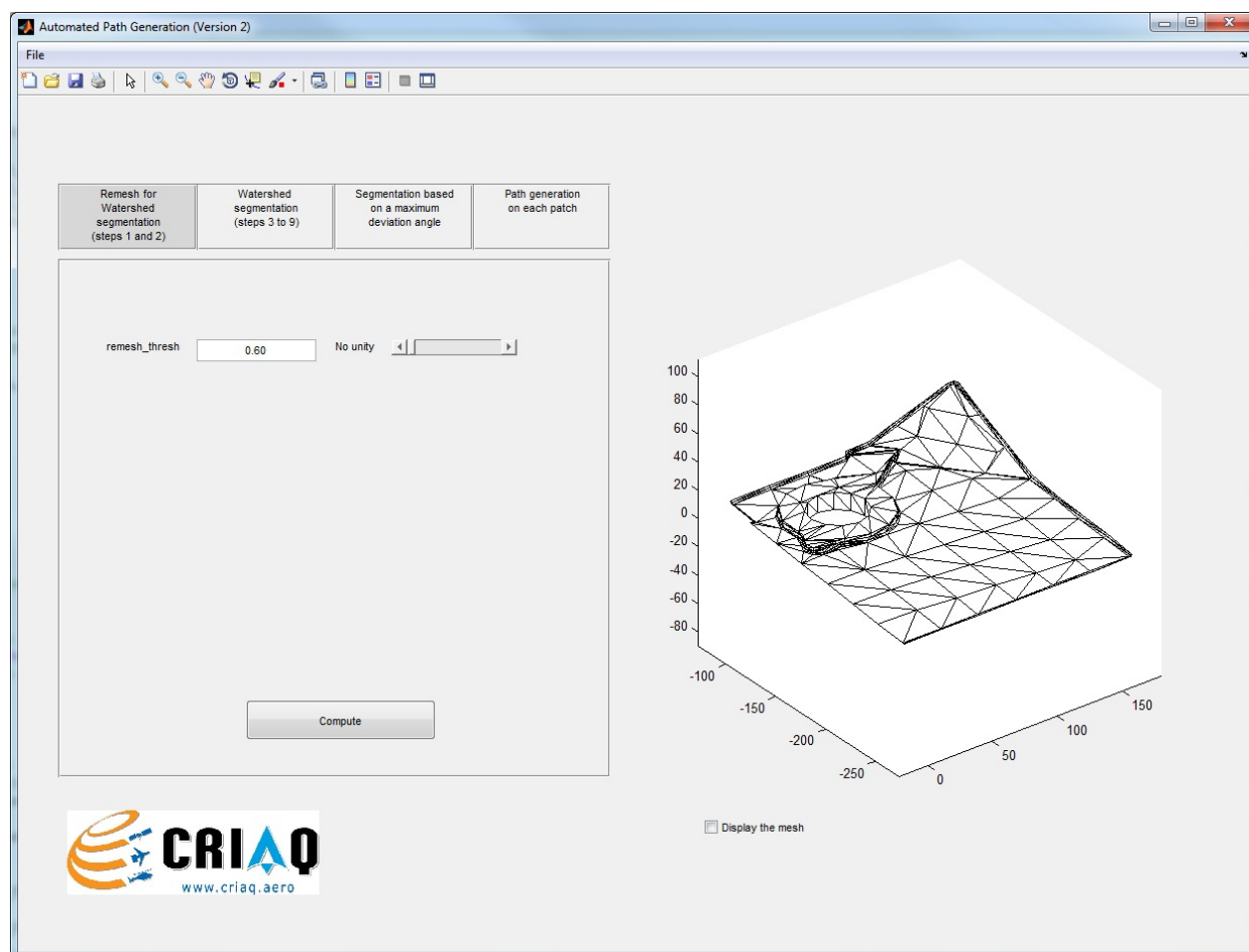


Figure 3.3 Fenêtre après importation de la surface STL.

va donc pas segmenter la surface du parallélépipède en ses différentes faces, ce qui aurait été souhaité. Un raffinement du maillage dont la méthode est présentée dans [3] est alors effectué. Le résultat du raffinement sur cet exemple est présenté sur la Figure 3.5.

Application du remaillage dans le programme

Les zones qui nécessitent un remaillage sont généralement celles ayant des courbures élevées. Afin de ne pas remailler tous les triangles de la surface et surcharger inutilement le nombre de facettes, seuls les triangles contenant au moins deux sommets de courbure supérieure à une valeur seuil seront remaillés. Cette valeur seuil **comprise entre 0 et 1** est laissée comme paramètre, nommé *remesh_thresh*, pour que l'utilisateur puisse raffiner le maillage autant que nécessaire (dépendamment des résultats de segmentation obtenus par la suite). Une fois le paramètre réglé, cliquer sur Compute. Le résultat obtenu est illustré Figure 3.6.

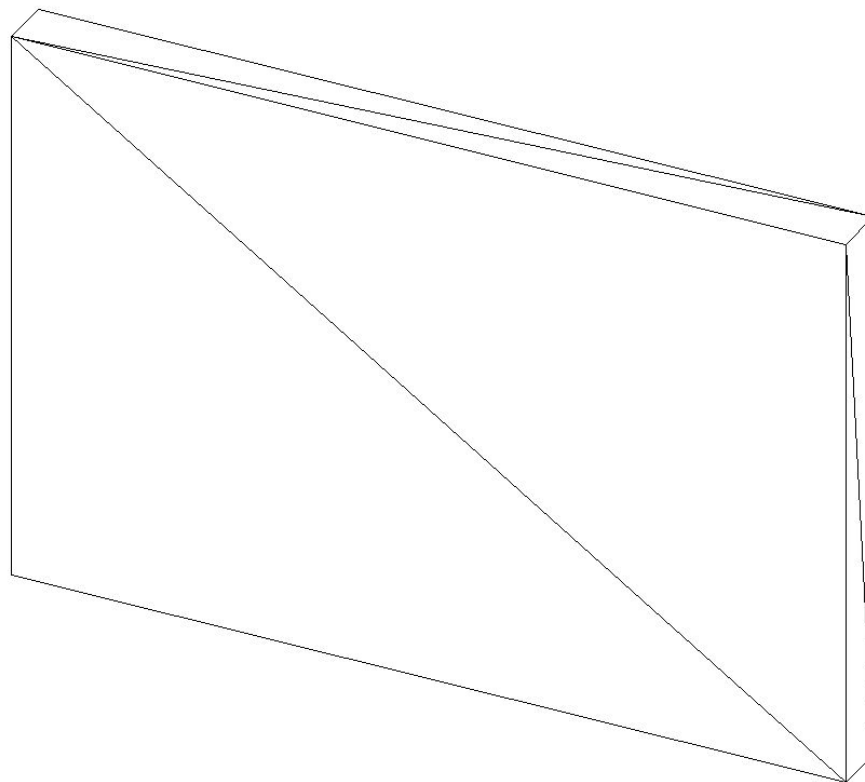


Figure 3.4 Surface triangulée d'un parallélépipède rectangle. Le maillage n'a pas été raffiné dans ce cas-ci.

Application de la segmentation watershed dans le programme

Cliquer ensuite sur le deuxième onglet pour appliquer la segmentation watershed. Un paramètre nommé *depth_thresh* **compris entre 0 et 1** peut alors être modifié. Un défaut de cette segmentation est que celle-ci est sensible à n'importe quelle variation de courbure : aussitôt qu'un minimum local est détecté, une nouvelle région est créée. Un nombre trop important de sous-régions peut alors en résulter. Afin de limiter ce phénomène, une étape de fusion de ces petites sous-régions (appelé aussi patches) est effectuée à la fin de la segmentation. Plus précisément, pour chacun de ces patches, une valeur de "profondeur" (*depth*) est calculée et correspond à l'écart entre sa valeur maximale de courbure et sa valeur minimale. Si cette valeur est inférieure au seuil fixé par le paramètre *depth_thresh*, ce patch sera alors fusionné avec un de ses voisins. Plus de détails peuvent être trouvés dans [3]. Régler ce paramètre et cliquer sur Compute. Un résultat est illustré Figure 3.7 pour une valeur de ce paramètre fixée à 0, 1.

Remarque : une option est disponible au bas de la figure (checkbox : "Display the mesh") pour afficher ou cacher le maillage de la surface pour contrôler davantage la géométrie de la surface durant les étapes de segmentations et de générations de chemin.

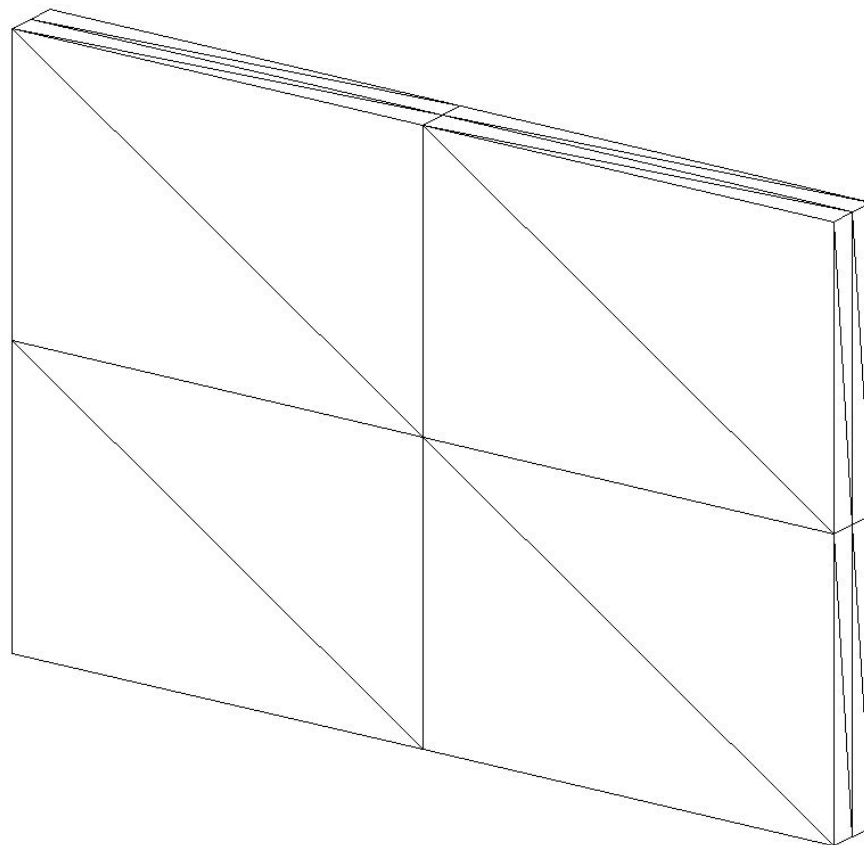


Figure 3.5 Surface remaillée. Un sommet est à présent défini au milieu de chacune des faces.

3.1.3 Segmentation basée sur un angle de déviation maximal

Cliquer sur le troisième onglet pour appliquer la deuxième segmentation. Cette segmentation s'appliquera directement sur les patches formés lors de la première segmentation. Cette segmentation a pour but de partitionner des surfaces dont l'angle de déviation maximale est trop élevé. On appelle ici **angle de déviation** entre deux facettes triangulaires l'angle formé entre leurs normales. L'angle de déviation maximum d'une surface correspond à l'angle de déviation maximum parmi tous les couples de ses facettes triangulaires. Cette segmentation permet donc d'avoir un contrôle sur la planéité des sous-régions ainsi obtenues. Régler le seuil de l'angle de déviation maximal, nommée *ang_dev_thresh* en degrés **compris entre 0 et 180 degrés**. Ce paramètre permet de contrôler l'angle de déviation maximale des sous-régions résultantes de cette segmentation. Cliquer ensuite sur Compute pour effectuer cette segmentation.

Le résultat obtenu pour une valeur de $ang_dev_thresh = 100^\circ$ est illustré Figure 3.8.

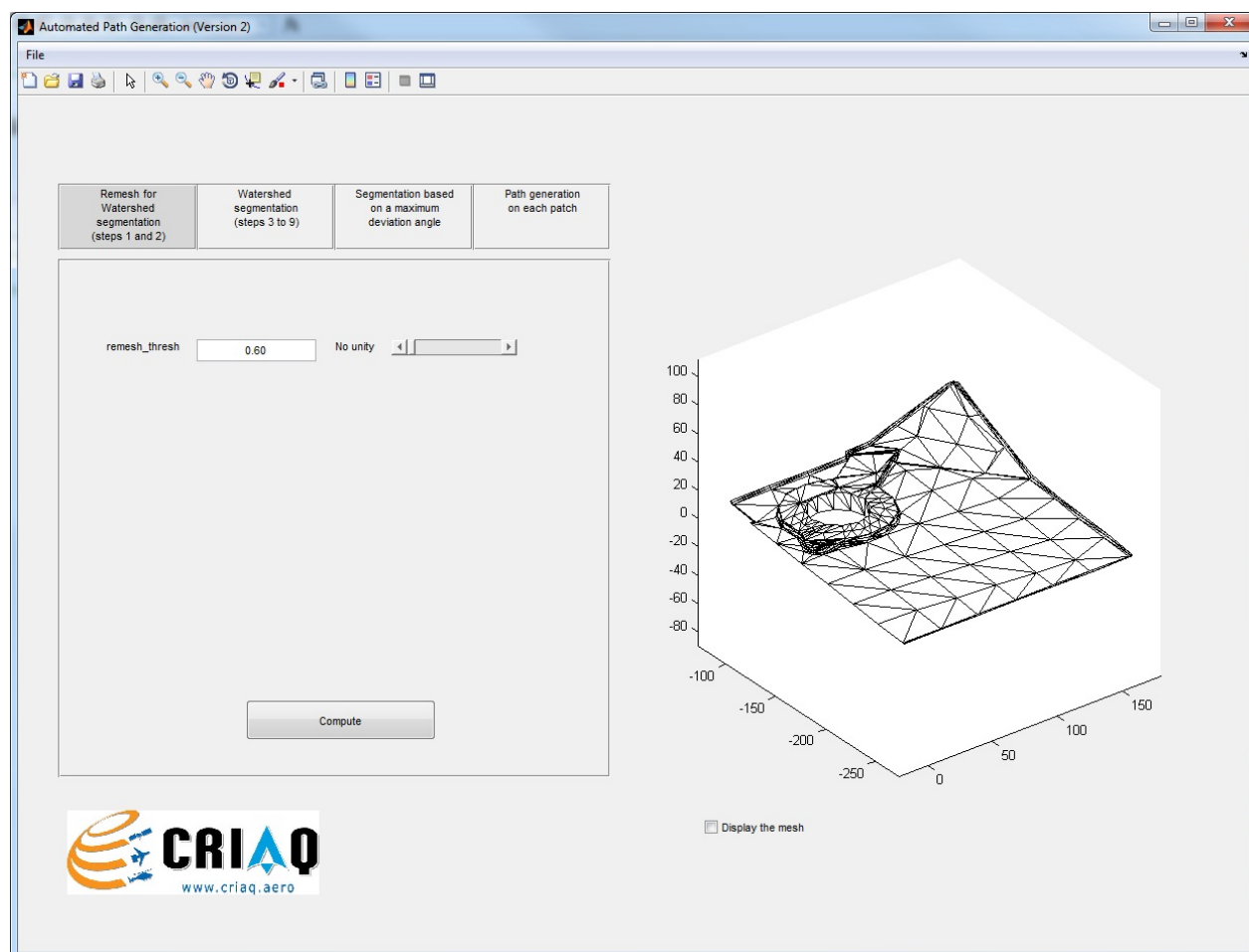


Figure 3.6 Étape de remaillage du programme.

3.1.4 Segmentation topologique basée sur des fonctions de Morse et génération de chemins de couverture de la surface

Cliquer sur le dernier onglet. La segmentation basée sur des fonctions de Morse est optionnelle. Si l'utilisateur souhaite appliquer cette dernière segmentation, cocher l'option "Morse decomposition applied". Cette segmentation permet de simplifier la topologie des patches issus de la deuxième segmentation et éviter de générer des chemins qui passent au travers de trous ou de bordures non convexes des patches.

Dans cet onglet, l'utilisateur peut maintenant générer les chemins sur la surface. Les paramètres de chemin sur chaque patch sont illustrés Figure 3.9. Une fois ces paramètres réglés, cliquer sur Compute.

Remarque 1 : le dernier paramètre de chemin non mentionné dans la Figure 3.9, *adj_thresh*, correspond au seuil de distance minimum que doit respecter deux points consécutifs du chemin le long de chaque passe du balayage. Un point dont la distance entre celui-ci et le point suivant le long d'une passe, est inférieure au seuil *adj_thresh* (en mm), sera supprimé du che-

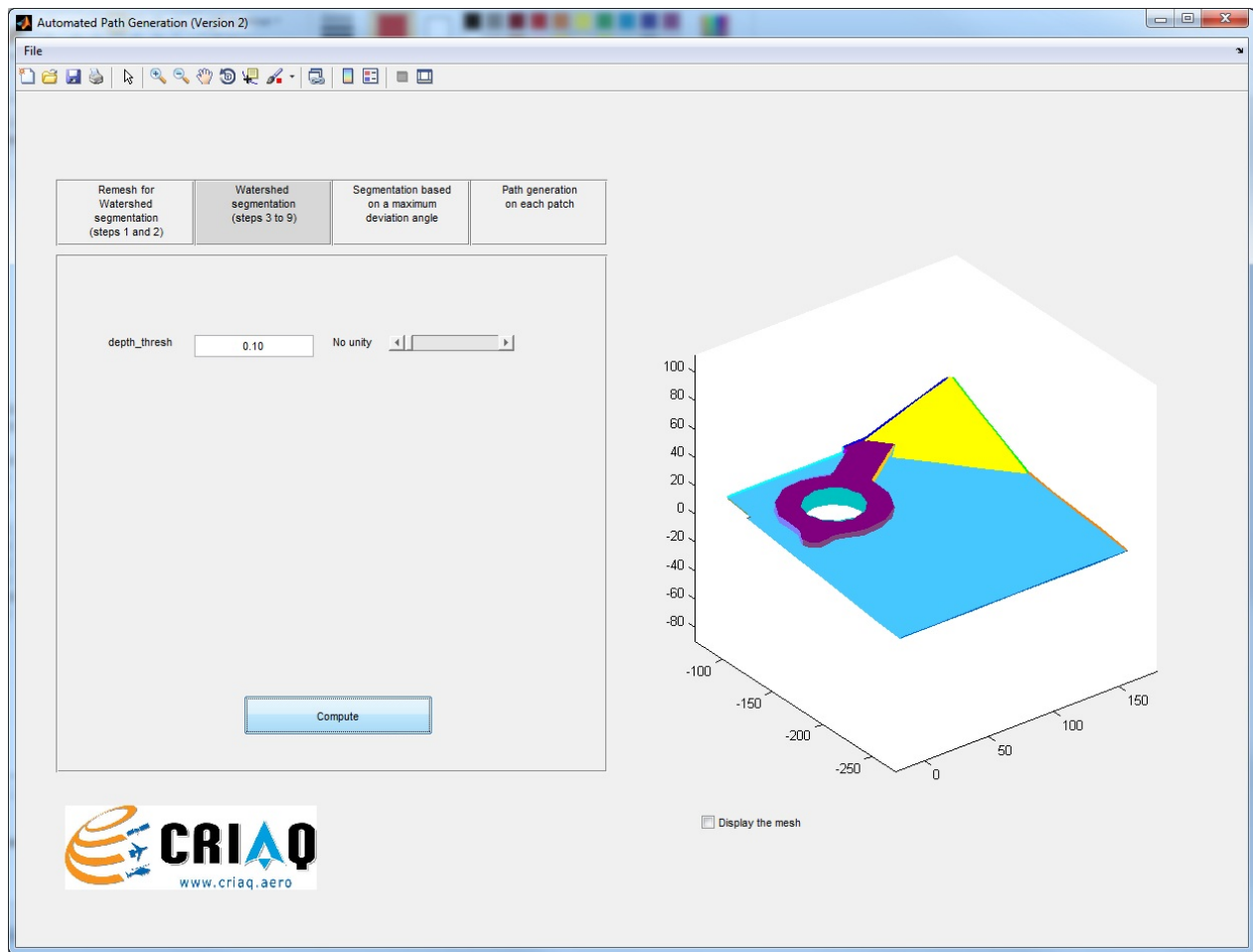


Figure 3.7 Segmentation watershed de la surface pour $depth_thresh = 0,1$.

min final. L'utilité de ce paramètre est de supprimer des points trop rapprochés qui peuvent surcharger l'information du chemin sans apporter de précision pour autant.

Remarque 2 : les paramètres de chemin doivent respecter les limites suivantes :

- *sweep_dir* : aucune contrainte (normalement entre -180° et $+180^\circ$),
- *d* : doit être **strictement positif**,
- *Os_d* : aucune contrainte (peut être positif ou négatif),
- *offset_edge* : doit être **positif ou nul**,
- *adj_thresh* : doit être **positif ou nul**.

Remarque 3 : selon le nombre de triangles de chacun des patches, la dernière segmentation peut nécessiter plus de temps de calcul que les segmentations précédentes.

Une fois les paramètres de chemin réglés, cliquer sur Compute. La Figure 3.10 montre le résultat de la génération de chemin après application de la dernière segmentation utilisant des fonctions de Morse. La Figure 3.11 montre le même résultat sans l'application de cette

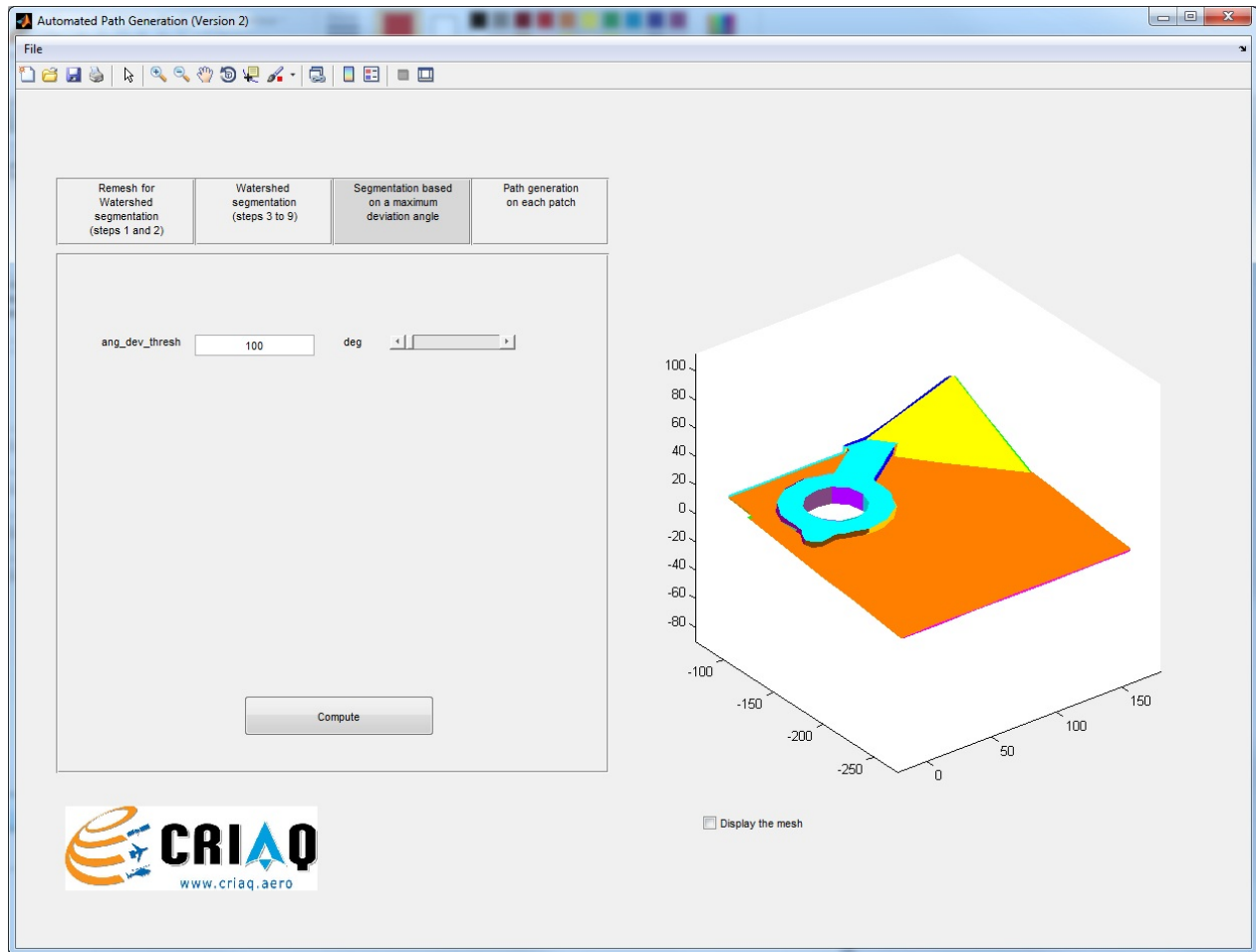


Figure 3.8 Segmentation basée sur un angle maximal de déviation $ang_dev_thresh = 100^\circ$.

dernière segmentation.

3.1.5 Calcul de l'orientation et exportation du chemin

Une fois que le chemin désiré a été calculé, l'utilisateur peut enfin exporter le chemin. Pour ce faire, cliquer sur le menu **File/Export the path** et choisir le nom du tableau Excel à exporter.

Remarque 1 : Le nom par défaut proposé est du type “NomSurfaceCorrespondante_Paths.xls”. Afin d’éviter toute confusion sur la location du fichier, le répertoire par défaut pour l’enregistrement de ce tableau correspond à celui de la surface (Répertoire du programme/Surfaces/NomSurfaceCorrespondante).

Remarque 2 : La liste des points du chemin peut également être exportée au format texte. Cependant, l’exportation dans un fichier EXCEL (Excel 2003) est recommandé si l’utilisation future de ce chemin est de l’importer dans le logiciel de simulation de robots MotoSim.

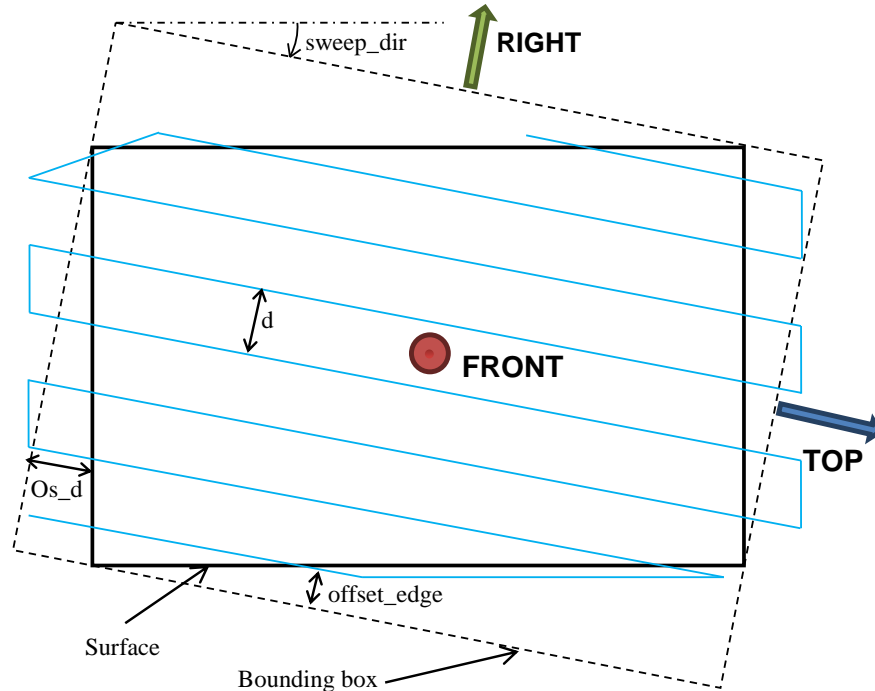


Figure 3.9 Paramètres des chemins de couverture d’une surface.

Remarque 3 : Au cours de l’exportation, les orientations de l’outil sont automatiquement calculées et transformées en coordonnées d’orientation (R_x, R_y, R_z) respectant la convention d’orientation “X-Y-Z fixed angles”. L’utilisateur peut obtenir plus d’informations sur cette convention d’orientation dans [2].

3.2 Génération de chemins pour l’inspection autour d’une indication de coordonnées connues

3.2.1 Importation d’une indication

Il est supposé dans cette sous-section que le tableau EXCEL (Excel 2003) de l’indication à inspecter existe et est placé dans le dossier de sa surface correspondante (cf. Section 2.2).

Remarque 1 : Avant d’importer une indication, il est important de **d’abord importer la surface STL** sur laquelle cette indication apparaît **et d’appliquer la génération de chemins sur cette surface**. La raison est que la méthode de génération de chemins de couverture autour d’une indication utilise les données des patches issues des étapes de segmentation (la segmentation utilisant des fonctions de Morse n’est pas obligatoire).

Remarque 2 : Dans le cas où l’utilisateur aurait généré des chemins de couverture sur la surface globale en utilisant la méthode d’exportation des sous-surfaces (cf. Section 2.1.2), il ne sera pas possible de générer un chemin de couverture autour d’une indication appartenant à différentes sous-surfaces. Néanmoins, si l’indication est entièrement à l’intérieur d’une même

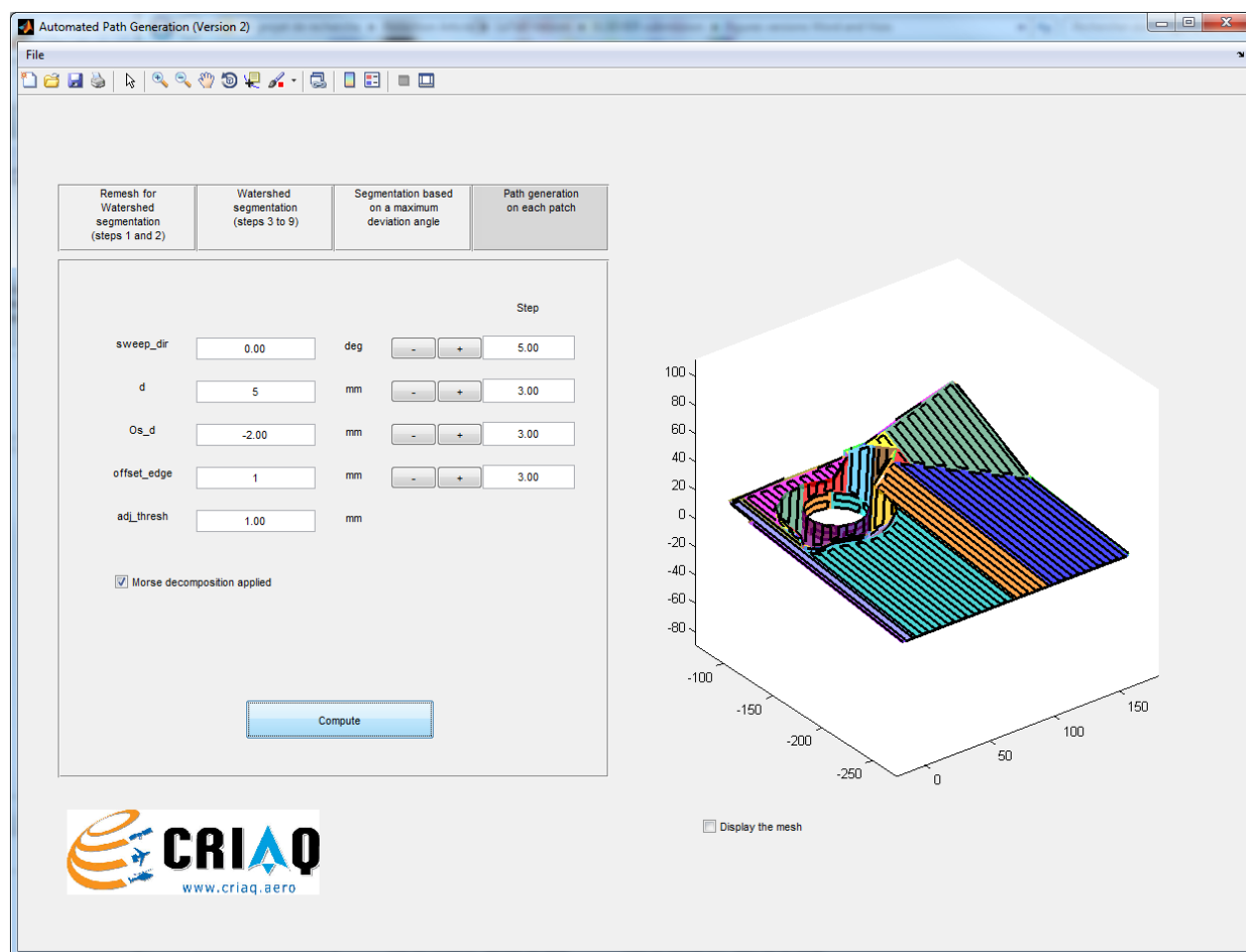


Figure 3.10 Génération des chemins de couverture de la surface. La segmentation par fonction de Morse a été effectuée et les paramètres de chemin sont $sweep_dir = 0^\circ$, $d = 5mm$, $Os_d = -2mm$, $offset_edge = 1mm$ et $adj_thresh = 1mm$.

sous-surface, l'utilisation de cette sous-surface suffira pour générer des chemins autour de l'indication. En revanche, dans le cas où une indication déborde sur plusieurs sous-surfaces, il est nécessaire d'exporter au moins toutes les sous-surfaces contenant l'indication *dans un même fichier STL* (utilisation de la commande **extraction multiple** dans CATIA, cf. Section 2.1.1).

Pour importer une indication, cliquer sur le menu **File/Import an indication...** et sélectionner le fichier EXCEL de l'indication préalablement placé dans le dossier de sa surface correspondante.

Une nouvelle fenêtre apparaît et l'indication est représentée alors comme une ligne épaisse verte sur la surface maillée. La Figure 3.13 montre un exemple de la nouvelle figure.

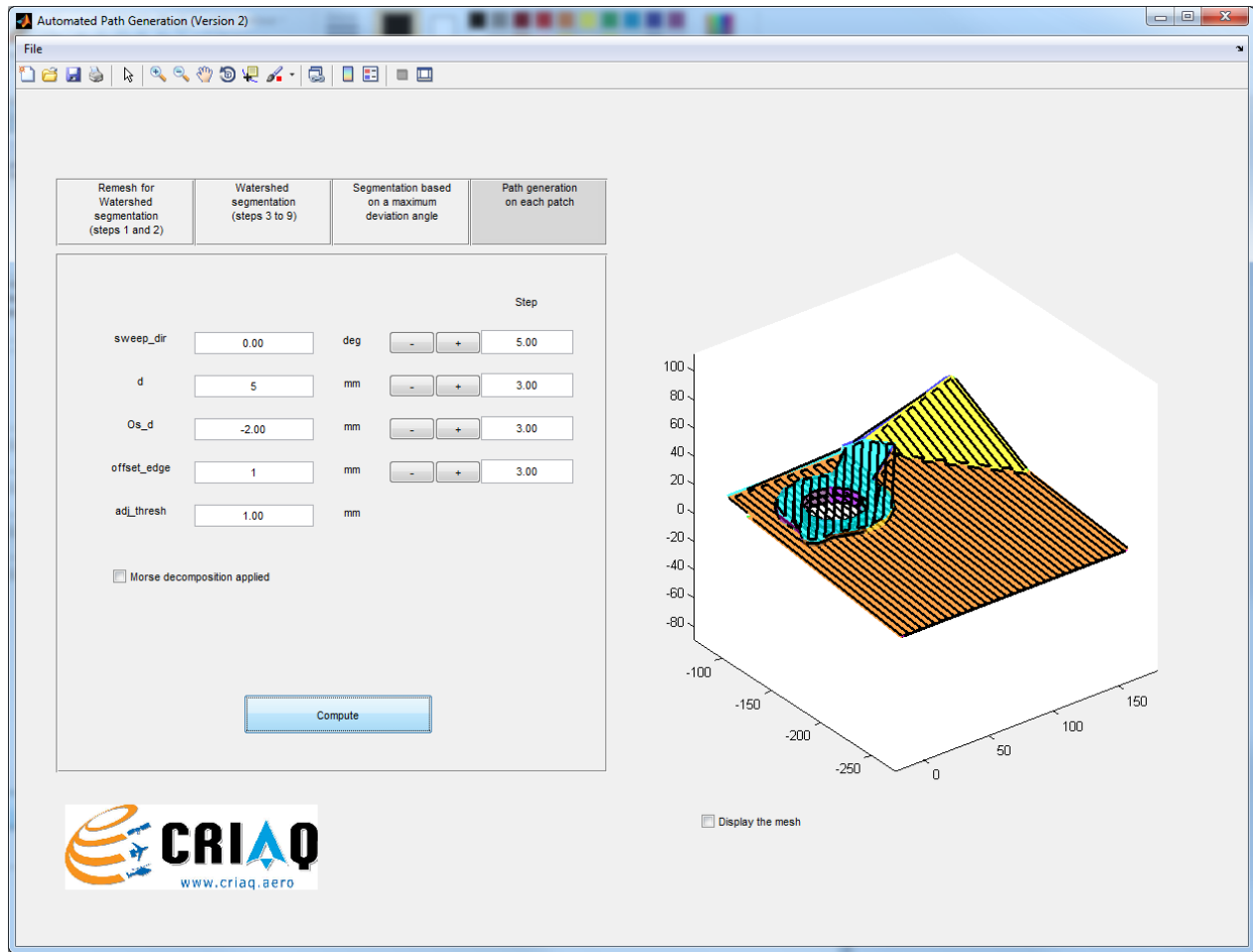


Figure 3.11 Génération des chemins de couverture de la surface. La segmentation par fonction de Morse n'a pas été effectuée et les paramètres de chemin sont $sweep_dir = 0^\circ$, $d = 5mm$, $Os_d = -2mm$, $offset_edge = 1mm$ et $adj_thresh = 1mm$.

3.2.2 Génération de chemins de couverture autour d'une indication

Une fois l'indication importée, l'utilisateur peut ensuite générer un chemin autour de celle-ci. Pour maximiser la sensibilité de mesure de la sonde à courant de Foucault devant l'indication, sa direction de balayage est calculée de manière à ce que celle-ci soit toujours perpendiculaire à cette indication.

Remarque 1 : De même que dans la remarque 1 de la Section 3.1.4, le paramètre nommé adj_thresh_Ind , correspond au seuil de distance minimum que doit respecter deux points consécutifs du chemin le long de chaque passe du balayage. Un point dont la distance entre celui-ci et le point suivant le long d'une passe, est inférieure au seuil adj_thresh_Ind (en mm), sera supprimé du chemin final. L'utilité de ce paramètre est de supprimer des points trop rapprochés qui peuvent surcharger l'information du chemin sans apporter de précision pour autant.

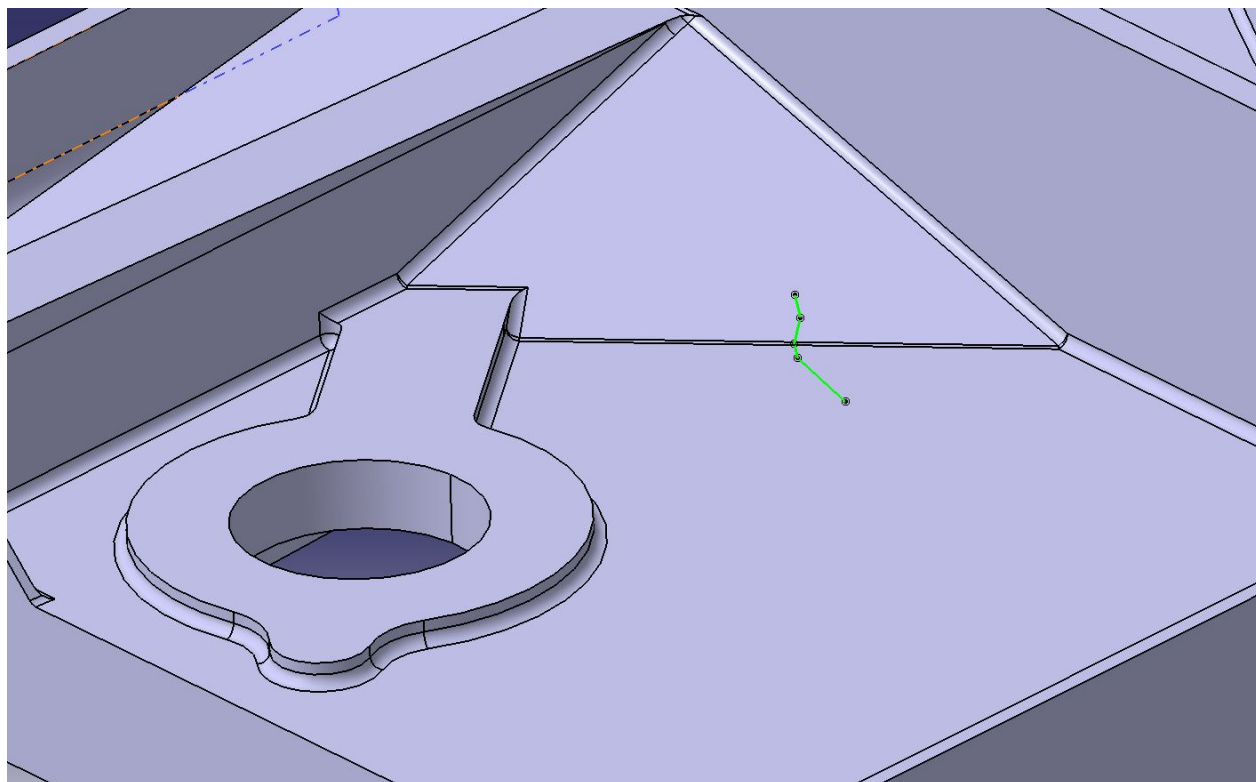


Figure 3.12 Représentation d'une indication (en vert clair sur cette figure) dans l'environnement Catia.

Remarque 2 : les paramètres de chemin doivent respecter les limites suivantes :

- d_Ind : doit être **strictement positif**,
- W_Ind : doit être **positif ou nul**,
- $offset_Ind$: doit être **positif ou nul**,
- adj_thresh_Ind : doit être **positif ou nul**.

3.2.3 Exportation du chemin

Une fois que le chemin désiré a été calculé, l'utilisateur peut enfin exporter le chemin. Pour ce faire, cliquer dans la nouvelle fenêtre de génération de chemin autour d'une indication sur le menu **File/Export the path around an indication** et choisir le nom du tableau Excel à exporter.

Remarque 1 : Le nom par défaut proposé est du type "NomSurfaceCorrespondante_indication_NomeroIndication_Paths.xls". Afin d'éviter toute confusion sur la location du fichier, le répertoire par défaut pour l'enregistrement de ce tableau correspond à celui de la surface (Répertoire du programme/Surfaces/NomSurfaceCorrespondante).

Remarque 2 : La liste des points du chemin peut également être exportée au format texte. Cependant, l'exportation dans un fichier EXCEL (Excel 2003) est recommandé si l'utilisation

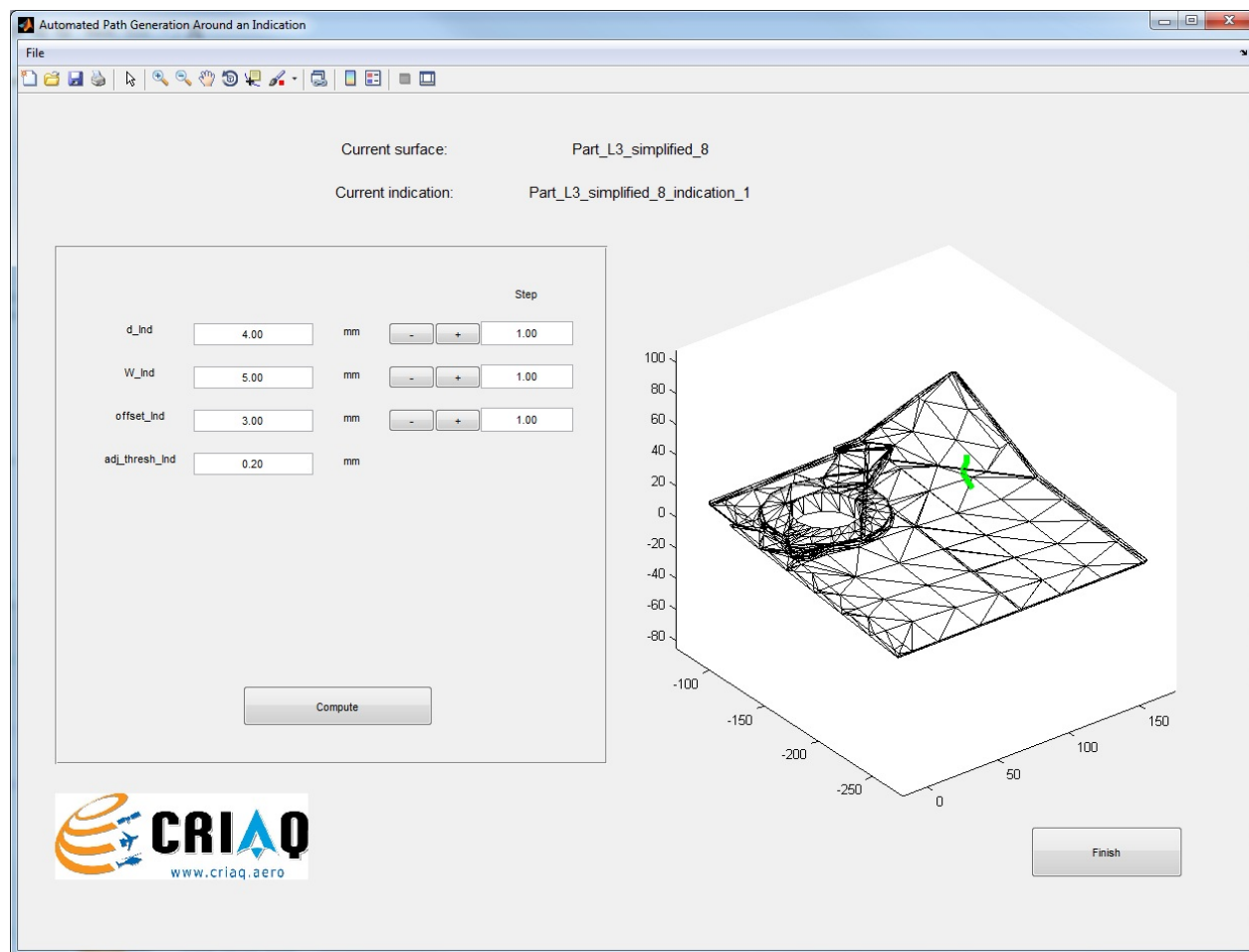


Figure 3.13 Représentation d'une indication dans le programme.

future de ce chemin est de l'importer dans le logiciel de simulation de robots MotoSim.

Remarque 3 : Au cours de l'exportation, les orientations de l'outil sont automatiquement calculées et transformées en coordonnées d'orientation (R_x, R_y, R_z) respectant la convention d'orientation "X-Y-Z fixed angles". L'utilisateur peut obtenir plus d'informations sur cette convention d'orientation dans [2].

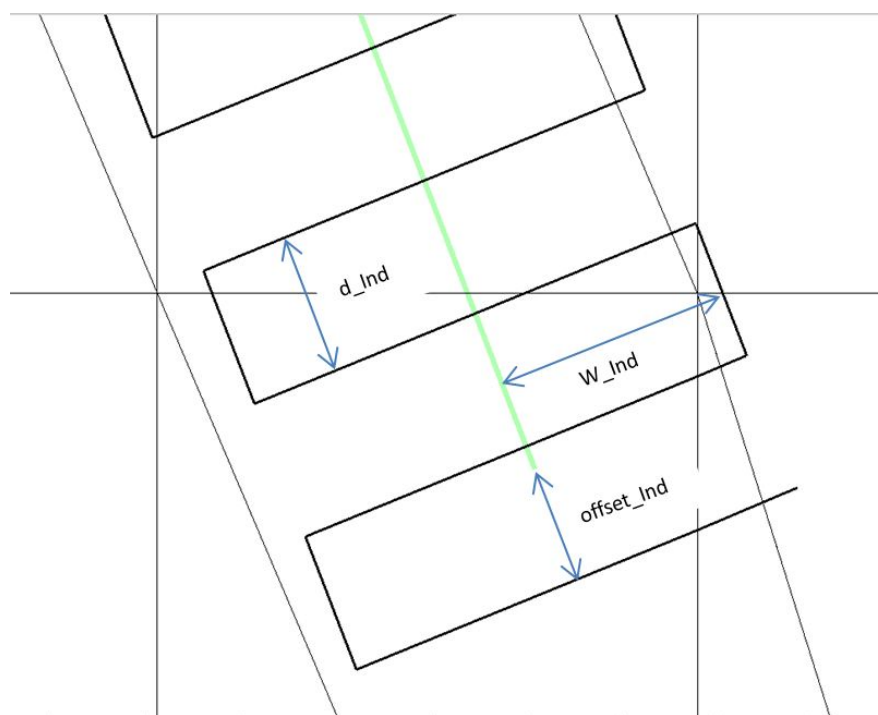


Figure 3.14 Paramètres du chemin de couverture autour d'une indication.

CHAPITRE 4

DONNÉES DE SORTIE ET LEUR UTILISATION FUTURE

Les données de sortie des chemins d'inspection d'une surface globale et d'inspection autour d'une indication sont illustrées respectivement en Figure 4.1 et 4.2.

Utilisation future dans le logiciel de simulation de robots MotoSim (Version 6.4) :

À partir de ces données de sortie, il est ensuite possible de les utiliser dans le logiciel de simulation de robots MotoSim (version 6.4). Une fonctionnalité dans ce logiciel est de pouvoir importer des positions robot à partir de données d'un fichier Excel, à condition que le format respecte celui qui est présenté en Figure 4.3.

	A	B	C	D	E	F	G	Barre de formule
1	X	Y	Z	Rx	Ry	Rz		
2								
3	Patch(1)							
4								
5	-73	22,5	15	180	0	0		
6	-67,5	22,5	15	180	0	0		
7	0	22,5	15	180	0	0		
8	7,5	22,5	15	180	0	0		
9	73	22,5	15	180	0	0		
10	73	17,5	15	180	0	0		
11	22,5	17,5	15	180	0	0		
12	0	17,5	15	180	0	0		
13	-52,5	17,5	15	180	0	0		
14	-73	17,5	15	180	0	0		
15	-73	12,5	15	180	0	0		
16	-37,5	12,5	15	180	0	0		
17	0	12,5	15	180	0	0		
18								
19								
20	Patch(2)							
21								
22	-75	-23	2,5	90	0	90		
23	-75	-8,33333333	2,5	90	0	90		
24	-75	0	2,5	90	0	90		
25	-75	16,6666667	2,5	90	0	90		
26	-75	23	2,5	90	0	90		
27	-75	23	7,5	90	0	90		
28	-75	0	7,5	90	0	90		
29	-75	-23	7,5	90	0	90		
30	-75	-23	12,5	90	0	90		
31	-75	-16,666667	12,5	90	0	90		
32	-75	0	12,5	90	0	90		
33	-75	8,33333333	12,5	90	0	90		
34	-75	23	12,5	90	0	90		
35								
36								
37	Patch(3)							
38								
39	-73	-25	12,5	-90	0	0		
40	-25	-25	12,5	-90	0	0		
41	0	-25	12,5	-90	0	0		
42	50	-25	12,5	-90	0	0		
43	73	-25	12,5	-90	0	0		
44	73	-25	7,5	-90	0	0		
45	0	-25	7,5	-90	0	0		
46	-73	-25	7,5	-90	0	0		
47	-73	-25	2,5	-90	0	0		
48								

Figure 4.1 tableau EXCEL des chemin de couverture d'une surface. Chacun des patches ont leur propre chemin de couverture. Les trois premières colonnes correspondent aux coordonnées de position (x, y, z) et les trois dernières colonnes correspondent aux coordonnées d'orientation (R_x, R_y, R_z).

	A	B	C	D	E	F	G	H
1	X	Y	Z	Rx	Ry	Rz		
2								
3	Fillet_Indication_indication_1							
4								
5	15,3570067	57,0124326	13,7495388	-176,732364	-40,4920916	174,975241		
6	14,4030799	57,0860272	14,3869321	-176,129078	-44,8250196	174,51732		
7	13,9425077	57,14286	15,0682082	-175,648015	-47,9209694	174,145695		
8	11,14181	57,491236	19,25975	-170,748309	-67,1868521	169,978482		
9	10,544174	57,6858999	22,2642602	-165,141714	-75,85306	164,698476		
10	10,5968457	61,6782848	21,9994627	-165,894971	-75,0721063	165,422965		
11	10,9653561	61,5582524	20,1468409	-169,558029	-69,7358306	168,885731		
12	11,14181	61,5007773	19,25975	-170,748309	-67,186852	169,978482		
13	13,9370138	61,1530847	15,0764305	-175,641892	-47,9584828	174,140923		
14	14,3934	61,0963153	14,3934	-176,122648	-44,8684034	174,512394		
15	15,5806017	61,0047238	13,6001375	-176,869724	-39,4283496	175,078536		
16	15,791824	64,9979696	13,4590033	-176,995658	-38,4259492	175,173021		
17	14,3934	65,1058567	14,3934	-176,122648	-44,8684033	174,512394		
18	13,474153	65,2202009	15,7691508	-175,093268	-51,1403907	173,708658		
19	11,14181	65,5103187	19,25975	-170,748309	-67,1868518	169,978482		
20	10,8030082	65,6206743	20,963017	-168,189086	-72,0948838	167,606081		
21	10,6495174	65,6706698	21,7346652	-166,560168	-74,3103945	166,05996		
22	10,702189	69,6630548	21,4698676	-167,15059	-73,5697568	166,622828		
23	11,14181	69,51986	19,25975	-170,748309	-67,1868517	169,978482		
24	13,0112922	69,287317	16,4618711	-174,47131	-54,3513207	173,207157		
25	14,3934	69,115398	14,3934	-176,122648	-44,8684032	174,512394		
26	16,0030464	68,9912153	13,317869	-177,115725	-37,4451826	175,262954		
27	16,1794334	72,9871486	13,2000111	-177,209817	-36,6588807	175,333364		
28	14,3934	73,1249394	14,3934	-176,122648	-44,8684031	174,512394		
29	12,5484315	73,3544332	17,1545915	-173,759661	-57,5706025	172,618436		
30	11,14181	73,5294014	19,25975	-170,74831	-67,1868515	169,978482		
31	10,7427421	73,5500513	21,2500557	-167,584074	-73,081507	167,03514		

Figure 4.2 tableau EXCEL du chemin de couverture autour d'une indication. Un seul chemin continu est généré autour de l'indication. Les trois premières colonnes correspondent aux coordonnées de position (x, y, z) et les trois dernières colonnes correspondent aux coordonnées d'orientation (R_x, R_y, R_z).

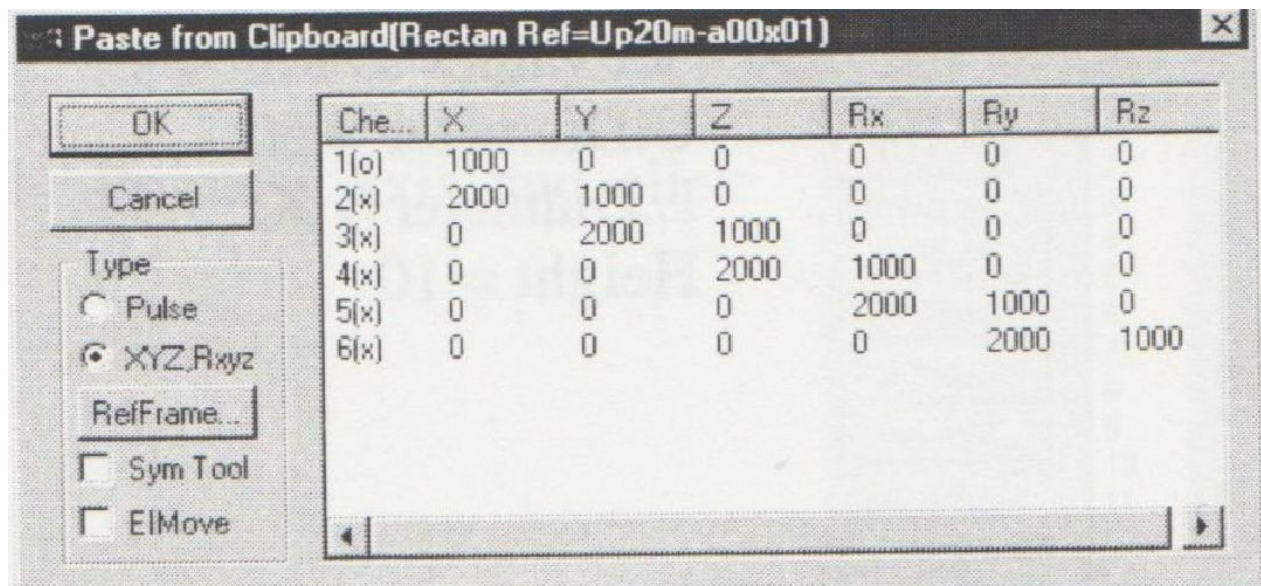


Figure 4.3 Importation d'un chemin dans le logiciel MotoSim. Extrait de [4].

CHAPITRE 5

VOCABULAIRE

Indication : Une indication est un terme utilisé dans des techniques de contrôle non destructif par ressuage (*Fluorescent Penetrant Inspection*, par exemple) et correspond aux traces visibles de pénétrant sur la surface après application du révélateur. Ces traces peuvent représenter aussi bien la localisation de défauts débouchants pertinents, que celle de défauts non pertinents (rugosité, marque d’outil, tâche d’huile, etc.). Un exemple est illustré Figure 2.9.

ECT : Eddy Current Testing. En français, cet acronyme signifie “inspection par courants de Foucault”.

Bibliographie

- [1] Solar Atmospheres. Fluorescent penetrant inspection services.
- [2] John J Craig. *Introduction to Robotics : Mechanics and Control (3rd Edition)*. Pearson Education International, 2004.
- [3] A. P. Mangan and R. T. Whitaker. Partitioning 3d surface meshes using watershed segmentation. *IEEE Transactions on Visualization and Computer Graphics*, 5(4), 1999.
- [4] MotoMan. *MotoSim User's Manual (6.4)*.