



Analyse qualitative de robots

Romain Benoit

► To cite this version:

Romain Benoit. Analyse qualitative de robots. Robotique [cs.RO]. Université d'Angers, 2017. Français. NNT : 2017ANGE0034 . tel-01760305

HAL Id: tel-01760305

<https://theses.hal.science/tel-01760305>

Submitted on 6 Apr 2018

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

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

Thèse de Doctorat

Romain BENOIT

*Mémoire présenté en vue de l'obtention du
grade de Docteur de l'Université d'Angers
sous le sceau de l'Université Bretagne Loire*

École doctorale : Sciences et Technologies de l'Information, Mathématiques

Discipline : Automatique, productique

Spécialité : Automatique et informatique appliquées

Unité de recherche : Laboratoire Angevin de Recherche en Ingénierie des Systèmes (LARIS EA7315)

Soutenue le 16 novembre 2017

Thèse N° : 112461

Analyse qualitative de robots

JURY

Présidente :	Marie-Françoise ROY , Professeur Emérite, Université de Renne 1
Rapporteurs :	Luc JAULIN , Professeur des Universités, ENSIETA Brest Fabrice ROUILLIER , Directeur de recherche, INRIA
Examineurs :	Guillaume MOROZ , Chargé de recherche, INRIA Michel COSTE , Professeur Emérite, Université de Renne 1
Invité :	Damien CHABLAT , Directeur de recherche CNRS, LS2N
Directeur de Thèse :	Philippe WENGER , Directeur de recherche CNRS, LS2N
Co-encadrants de Thèse :	Nicolas DELANOUE , Maitre de conférences, Université d'Angers Sébastien LAGRANGE , Maitre de conférences, Université d'Angers

Remerciements

Je souhaite d'abord remercier ma famille, pour tellement de choses qu'il vaut mieux dire : "pour tout, au risque d'inclure quelques remerciements de trop".

Mes remerciements dans le domaine de la recherche vont vers mon directeur de thèse et mes encadrants de thèse pour leurs connaissances, conseils et humanités qui ont posés les fondations des développements détaillés dans ce mémoire que vous tenez entre les mains ou affichez sur votre écran.

Je souhaite remercier tout aussi chaleureusement toutes les autres personnes avec qui j'ai interagi au cours de cette thèse, qu'ils soient chercheurs, techniciens, secrétaires ou même personnels de cantine à l'université de Mihono.

A vous les gens qui vivent et œuvrent dans l'ombre pour que la recherche éclaire la société, je dédie cette composition.

Table des matières

Table des matières.....	2
Table des Illustrations	4
Chapitre 1 Introduction.....	6
1.1 Introduction condensée.....	6
1.2 Motivations et domaines d'application	7
1.3 Concepts considérés	7
1.4 Travaux développés.....	7
Chapitre 2 Etat de l'art	8
2.1 Contexte robotique : Etude des lieux singuliers.....	8
2.1.1. Concepts principaux.....	8
2.1.2. Lieux singuliers et postures.....	9
2.1.3. Robot cuspidal et changement non singulier de posture.....	10
2.2 Application générique et contrôle des dérivées.....	14
2.2.1. Espace des jets et jets d'applications	14
2.2.2. Topologies sur $\mathcal{C}^\infty(X, Y)$ et sous-ensembles résiduels	16
2.2.3. Transversalité et généricité - Théorème de Thom	17
2.2.4. Extension des définitions et théorèmes aux multi-jets.....	18
2.3 Analyse par Intervalles	19
2.3.1. Principe et définitions.....	19
2.3.2. Application au calcul numérique informatique	21
2.3.3. Applications en robotique.....	23
2.3.4. Algorithmes élémentaires	25
2.4 Introduction à la problématique de classification.....	29
Chapitre 3 Préliminaires.....	30
3.1 Equivalence et invariant : une introduction	30
3.1.1. Concepts formels d'Equivalence et d'Invariants	30
3.1.2. Propriété fondamentale de l'équivalence et introduction à la définition d'un invariant pour des fonctions de $\mathcal{D} \subseteq \mathbb{R}$ dans $\mathbb{R}$	32
3.2 Détection de lieu caractéristique et construction d'invariant	32
3.2.1. Lieu singulier caractéristique et Classes de singularités	32
3.2.2. Invariant hybride entre équivalence et conjugaison	35
3.2.3. Cheminement intuitif vers $\mathcal{K}_0$, pour $f : \mathcal{D}_f \rightarrow \mathbb{R}$ où $\mathcal{D}_f \subseteq \mathbb{R}$	35
Chapitre 4 Caractérisation et Détection des singularités	42
4.1 Introduction : la Théorie et la Pratique	42
4.2 Méthodologie générale et Application en dimension 1	42
4.2.1. Caractérisation Théorique	42
4.2.2. Détection Pratique.....	44
4.2.3. Conclusion relative aux méthodes en dimension 1	46
4.3 Application en dimension 2	47
4.3.1. Caractérisation Théorique	47
4.3.2. Détection Pratique (et performances des algorithmes)	53
4.3.3. Conclusion relative aux méthodes en dimension 2	62
4.4 Application en dimension 3	63
4.4.1. Etude théorique	63
4.4.2. Détection Pratique.....	71
4.4.3. Conclusions et perspectives de l'étude en dimension 3	80

4.5	Conclusions relatives à ce chapitre et à ses perspectives	82
4.5.1.	Méthodologie générale	82
4.5.2.	Utilisation de l'Analyse par Intervalles	82
Chapitre 5	Classification globale	83
5.1	Contribution vis-à-vis de la robotique	84
5.2	Plan du chapitre	85
5.3	Résumé en images de la construction de l'abstraction	85
5.4	Bases théoriques	90
5.4.1.	Caractérisation et Classification (remise en situation)	90
5.4.2.	Triangulations et complexes simpliciaux abstraits	94
5.5	Construction d'une abstraction de f	96
5.5.1.	Etape 0 : Isoler les points et ensembles critiques	96
5.5.2.	Etape 1 : Construire la stratification labellisée tracée	97
5.5.3.	Etape 2 : Créer l'abstraction	97
5.5.4.	Etape 3 (optionnelle) : Réalisation	103
5.5.5.	Application à une fonction géométrique d'un robot 3-RPR	103
5.6	Application à la robotique	105
5.6.1.	Notations, modèles et conventions	105
5.6.2.	Exemples de comportements abstraits	106
5.7	Mouvements internes et abstraction	115
5.7.1.	Concepts de Mouvement interne et d'Eclatement	115
5.7.2.	Lien entre mouvements internes et éclatement	116
5.7.3.	Quatrilation	117
5.7.4.	Construction de la quatrilation d'une fonction	118
5.7.5.	Quatrilation c_f décomposée à l'aide d'une abstraction c_{f_s}	119
5.8	Manipulateurs 3R avec mouvements internes	119
5.8.1.	Comportement 4 : « Mouvements internes isolés »	119
5.8.2.	Comportement 5 : « Mouvements internes non isolés »	122
5.9	Conclusion du chapitre et futurs travaux possibles	126
Chapitre 6	Conclusion du mémoire	128
Annexes	130	
A.1	Expressions formelles des fonctions décrivant les robots 3R	130
A.2	Appendices au Chapitre 5	131
A.2.1.	Conventions choisies pour les figures	131
A.2.2.	Liste des abstractions présentées	132
A.3	Nomenclature	136
A.4	Reproductions plus uniformisées des algorithmes	139
Bibliographie	146	

Table des Illustrations

Figure 1 : Point cusp $f(C)$ et point cusp articulaire C d'une fonction géométrique f	10
Figure 2 : Point cusp et trajectoire $(\gamma, f(\gamma))$ induisant un changement non singulier de posture	11
Figure 3 : Schéma cinématique d'un robot 3R (avec $\theta_1 = 0$) - orthogonal si $\beta_2 = \beta_3 = 0$	12
Figure 4 : Deux ensembles (en rouge et en bleu) compatibles avec un même encadrement (en noir) ..	24
Figure 5 : Algorithmes $SIVIA_{f,V,E,\varepsilon}$ (rouge et noir) et $IVIA_{f,E,\varepsilon}$ (bleu et noir)	26
Figure 6 : Un algorithme $NpI_{f,E,\varepsilon}$	28
Figure 7 : Lorsque ce diagramme commute f et g sont équivalentes.....	30
Figure 8 : Lorsque ce diagramme commute f et g sont conjuguées.....	31
Tableau 1 : Tableau de variation de la fonction $f : x \mapsto x^2 - x$, pour $\mathfrak{D}_f = [-2; 2]$ et $\mathfrak{D}_f = \mathbb{R}$	36
Tableau 2 : Tableau de variation simple de $f : x \mapsto x^2 - x$	37
Tableau 3 : Esquisse de $f : x \mapsto x^2 - x$	38
Tableau 4 : Tableau de variation détaillé de $f : x \mapsto x^3 - x$, définie sur $\mathfrak{D}_f = \mathbb{R}$	40
Tableau 5 : Esquisse de $f : x \mapsto x^3 - x$	40
Tableau 6 : Encrage de $f : x \mapsto x^3 - x$	40
Figure 9 : Arbre des $\mathcal{S}_v$ pour une fonction entre espaces de dimensions 1	43
Figure 10 : Arbre des $\mathcal{S}_v$ pour une fonction entre espaces de dimensions 2	47
Figure 11 : Schéma cinématique d'un manipulateur 3R.....	50
Table 1 : Exemples de manipulateurs robotiques étudiés (ND : Non Déterminé).....	58
Table 2 : Performances des algorithmes pour les manipulateurs robotiques de la Table 1 (NA : l'algorithme n'a pas abouti et on ne peut donc pas conclure)	58
Figure 12 : Interprétations géométriques du modèle volumique de solide oblong $\mathcal{O}(\mathbb{Y}, \mathfrak{r})$	61
Figure 13 : Arbre des $\mathcal{S}_v$ pour une fonction entre espaces de dimensions 3	63
Figure 14 : Vues en projection de S_f et de $f(S_f)$ et (en rouge) de $\mathcal{S}_{1,1}(f)$ et de $f(\mathcal{S}_{1,1}(f))$ où : $f = f_{b,m,p,q}$ avec $b = m = 2$ et $p = q = 1$	70
Figure 15 : Algorithme encadrant S_f (génériquement $S_f = \mathcal{S}_1(f)$).....	72
Figure 16 : Algorithme encadrant $\mathcal{S}_{1,1}(f)$	73
Figure 17 : Algorithme encadrant $\mathcal{S}_{1,1,1}(f)$	74
Figure 18 : Vues en projection, de S_{f_0} et $\mathcal{S}_{1,1}(f_0)$ (en rouge), avec un rendu, par la méthode POVRAY, des recouvrements de S_{f_0} (pavés en bleu et en rouge) et de $\mathcal{S}_{1,1}(f_0)$ (pavés en rouge) fournis par l'exécution de référence.	77
Table 3 : Evolution des performances en fonction des précisions choisies.....	78
Table 4 : Evolution des performances en fonction de l'imprécision sur les paramètres constitutifs.	79
Figure 19 : Schéma de principe et paramètres géométriques d'un manipulateur parallèle 3-RPR.....	81
Figure 20 : Diagramme commutatif d'objets caractérisant f	86
Figure 21 : Schéma cinématique d'un manipulateur 3R orthogonal.	86
Figure 22 : Partitions des espaces articulaire et opérationnel selon la fonction f , décrivant le manipulateur représenté à la Figure 21.....	87
Figure 23 : Triangulation des partitions de la Figure 22.....	88
Figure 24 : Représentation géométrique de l'abstraction du manipulateur exemple de la Figure 21	89

Figure 25 : Deux fonctions géométriques g_1 et g_2 distinctes et de contours apparents homéomorphes. g_1 est la fonction géométrique d'un robot série 3R et g_2 est la fonction géométrique d'un robot parallèle 3-RPR. g_1 et g_2 diffèrent par les pré-images de leurs contours apparents.	91
Figure 26 : Diagramme commutatif détaillé d'objets caractérisant f	94
Figure 27 : Plan du processus de construction d'une abstraction c_f de f	96
Figure 28 : (Réalisation de) l'abstraction d'un disque topologique i_1 de bords labélisés	99
Figure 29 : Méthode algorithmique de construction des triplets de labels pour abstraire un cylindre.	100
Figure 30 : (Réalisation de) l'abstraction d'un cylindre topologique E de bords labélisés	101
Figure 31 : Stratification tracée $\mathfrak{s}(g_2)$ de la fonction g_2 de la Figure 25	103
Figure 32 : Triangulation $ c_{g_2} $, réalisation géométrique de l'abstraction de g_2 , issue de la Figure 25	104
Figure 33 : Schéma cinématique d'un manipulateur 3R orthogonal dans sa position d'origine.	105
Figure 34 : Stratification tracée labélisée $\mathcal{L}(\mathfrak{s}(f_1))$ de l'exemple pour le comportement 1	107
Figure 35 : Triangulation du comportement 1	109
Figure 36 : Stratification tracée labélisée $\mathcal{L}(\mathfrak{s}(f_2))$ de l'exemple pour le comportement 2	110
Figure 37 : Triangulation du comportement 2	111
Figure 38 : Partie articulaire $\mathcal{L}(\mathfrak{s}_a(f_3))$ de $\mathcal{L}(\mathfrak{s}(f_3))$ illustrant le comportement 3	112
Figure 39 : Partie opérationnelle $\mathcal{L}(\mathfrak{s}_p(f_3))$ de $\mathcal{L}(\mathfrak{s}(f_3))$ illustrant le comportement 3	113
Figure 40 : Triangulation du Comportement 3	114
Figure 41 : Eclatement $\mathfrak{E}$ et sa fonction d'éclatement $\mathfrak{e}$, localement équivalente à $(x, y) \mapsto (x, xy)$	116
Figure 42 : Description combinatoire locale d'une fonction avec mouvement interne	117
Figure 43 : Stratification tracée labélisée $\mathcal{L}(\mathfrak{s}(f_4))$ de l'exemple pour le comportement 4	120
Figure 44 : Quatrilation du comportement 4	121
Figure 45 : Partie articulaire $\mathcal{L}(\mathfrak{s}_a(f_5))$ de $\mathcal{L}(\mathfrak{s}(f_5))$ illustrant le comportement 5	123
Figure 46 : Partie opérationnelle $\mathcal{L}(\mathfrak{s}_p(f_5))$ de $\mathcal{L}(\mathfrak{s}(f_5))$ illustrant le comportement 5	123
Figure 47 : Quatrilation du comportement 5	125
Figure 48 : Schéma général d'un système considéré dans ce mémoire	128
Figure 49 : Présentation plus uniformisée de l'algorithme $\text{SIVIA}_{f,V,E,\varepsilon}$ de la Figure 5	139
Figure 50 : Présentation plus uniformisée de l'algorithme $\text{IVIA}_{f,E,\varepsilon}$ de la Figure 5	140
Figure 51 : Présentation plus uniformisée de l'algorithme $\text{NpI}_{f,E,\varepsilon}$ de la Figure 6	141
Figure 52 : Présentation plus uniformisée de l'algorithme d'encadrement de S_f de la Figure 15	142
Figure 53 : Présentation plus uniformisée de l'algorithme, encadrant $S_{1,1}(f)$, de la Figure 16	143
Figure 54 : Présentation plus uniformisée de l'algorithme, encadrant $S_{1,1,1}(f)$, de la Figure 17	144
Figure 55 : Présentation plus uniformisée de l'algorithme de la Figure 29	145

Chapitre 1 Introduction

1.1 Introduction condensée

Le terme « Analyse qualitative » dans le titre de cette thèse se réfère à notre problématique générale qui consiste à *caractériser et à classifier des fonctions* en s'appuyant sur des méthodes d'analyse numérique explicites, faisant appel à des propriétés mathématiques formelles.

Nous nous proposons d'appliquer cette problématique générale à des fonctions, au moins *génériques*, et à des fonctions décrivant le comportement de manipulateurs robotiques, de préférence novateurs.

Notez que le terme « générique » sera couramment utilisé et qu'il caractérise (selon des spécifications données par le contexte) le fait d'être usuel ou usuellement applicable.

Je propose de suivre une ligne directrice claire et de se concentrer sur les explications théoriques, plutôt que de multiplier des exemples caractéristiques afin d'éviter des extrapolations malheureuses. Dans un même ordre d'idée, je ne présente pas les travaux développés de façon chronologique, mais de façon thématique pour s'affranchir des redites.

Ce mémoire se décompose en :

1. Une introduction posant les bases du contenu de ce mémoire (Vous êtes ici.)
2. Une partie d'état de l'art qui pose les bases des outils que nous possédons :
 - a. Les notions de robotique nécessaires et le concept d'étude des lieux singuliers
 - b. La notion de généricité (i.e. « suffisamment général »)
 - c. La méthodologie de l'Analyse par Intervalles, choisie pour nos études numériques
3. Des préliminaires qui introduisent les concepts plus avancés dont nous ferons usage :
 - a. Concept d'équivalence et d'invariant
 - b. Concept d'ensemble caractéristique d'un invariant
 - c. Construction d'un représentant ou d'un descripteur d'un invariant
4. Un chapitre sur la caractérisation et la détection d'ensembles d'intérêt pour la classification :
 - a. Une méthode générale de caractérisation théorique et de détection pratique
 - b. Application de la méthode pour des fonctions entre espaces de dimensions 1 à 3
 - c. Introduction de l'application de la méthode pour des robots 3R (dimensions 2)
5. Un chapitre sur la classification globale et la construction d'un descripteur de comportement
 - a. Mise au point sur l'interprétation, en robotique, des concepts développés
 - b. Concept combinatoire de complexe simplicial et de triangulation
 - c. Méthode de construction d'une abstraction, un descripteur de l'invariant considéré
 - d. Illustrations sur des exemples de robots (« 3R orthogonaux »)
 - e. Concept de mouvement interne et description combinatoire de ceux ci
6. Mes conclusions sur ce mémoire et sur les travaux qui y sont relatifs
7. Les annexes au mémoire regroupant les compléments au Chapitre 5 et la nomenclature ainsi que des reproductions, plus uniformisées, des organigrammes pour les algorithmes proposés au cours de ce mémoire.

A la fin de certaines sections, vous trouverez des paragraphes en **gris** qui ne font pas partie des développements de cette thèse, à proprement parler. Ces paragraphes sont constitués de mes réflexions et de mes interrogations contribuant à l'ouverture et à la compréhension vis à vis des parties que ces paragraphes concluent.

La version numérique de ce rapport comporte des liens actifs. La première lettre de chaque expression, portant un de ces liens, est écrite en majuscule mettant ainsi en évidence l'existence du lien concerné.

Je vous souhaite d'apprécier la lecture de ce mémoire et les développements scientifiques qu'il élabore.

1.2 Motivations et domaines d'application

La classification (de fonctions) est une problématique générale motivée par un très grand nombre de sous-problématiques complémentaires.

Cette thèse aborde la classification à partir des considérations et des intérêts liés à la robotique. Toutefois, les développements avancés dans ce mémoire peuvent contribuer à tout domaine connexe à la problématique de classification de fonctions.

(On citera par exemple, l'étude des systèmes par l'intermédiaire de fonctions de transition.)

En général, deux considérations apparaissent motiver la classification :

- Deux fonctions dans la même classe peuvent être analysées par une unique étude.
- Distinguer plusieurs classes permet de séparer des comportements différents et permet ainsi de détecter certains changements de comportement.

Notre approche s'inscrit dans ces deux considérations. En effet, nous nous proposons de définir des classes de fonctions, décrivant des robots. Chaque classe regroupera des fonctions auxquelles de mêmes études, issues du domaine robotique, sont intrinsèquement applicables.

Notre objectif principal est donc, comme usuellement dans les problématiques de classification, de définir et de construire un objet discriminant pour la relation d'équivalence considérée :

« égales à des difféomorphismes près sur les espaces sources et buts des fonctions ».

1.3 Concepts considérés

Nos travaux s'appuient sur 3 éléments de base qui sont, dans l'ordre :

1. Les connaissances, concepts et résultats dans les domaines de la Robotique
2. Les concepts mathématiques liés à l'étude des singularités
3. Les méthodes numériques et informatiques de résolution d'équations

Le Chapitre 2 sera consacré à la présentation et à la mise en place des connaissances et résultats fondamentaux liés à ces trois éléments de base, dans le même ordre que ci-dessus.

Nous suivons ainsi le principe de tout état de l'art, au sens où les résultats présentés ne sont pas des découvertes de l'auteur. Toutefois, la formulation de ces résultats a été adaptée afin de correspondre à la façon de raisonner et au style de rédaction choisis.

Les éléments principaux du Chapitre 2, à retenir, sont les concepts élémentaires de Robotique, combinés aux résultats connus sur l'étude des singularités des robots. L'état de l'art est complété par la présentation rigoureuse du concept de *généricité* qui justifie la considération d'un ensemble restreint de singularités, au sens où une quantité résiduelle de fonctions ne présentent que ces singularités. Enfin, ce chapitre est agrémenté d'une introduction générale au concept de l'Analyse par Intervalles, qui est la méthodologie que nous privilégions pour l'implémentation des méthodes développées.

Afin de parfaire la mise en place des travaux, que nous présentons dans ce mémoire, nous proposons, dans le Chapitre 3, de définir formellement le concept d'équivalence vis-à-vis duquel nous classifions les fonctions, ainsi que les concepts qui y sont liés. Nous définissons ainsi la notion d'invariant et de représentant d'un invariant. Ces préliminaires introduisent aussi la notion d'ensembles caractéristiques et la progression vers la définition d'un objet décrivant le comportement d'une fonction.

1.4 Travaux développés

Les travaux développés s'appuyant fortement sur les notions introduites dans le Chapitre 2 et le Chapitre 3, il serait prématuré de détailler ceux-ci dès à présent.

Toutefois, nous précisons dès à présent que ceux-ci se classent, eux-mêmes, selon deux catégories. La première contribution, développée dans le Chapitre 4, consiste à détecter les ensembles caractéristiques vis-à-vis de l'équivalence $\sim$ considérée. L'applicabilité de la méthodologie proposée est vérifiée pour des fonctions entre espaces de dimensions topologiques entre 1 et 3. La méthodologie proposée est aussi appliquée aux fonctions décrivant un certain type de robots que sont les robots 3R.

La seconde contribution de cette thèse s'oriente vers *la classification globale des fonctions*.

La finalité de ces travaux, présentés dans le Chapitre 5, est de proposer un objet, décrivant le comportement d'une fonction d'intérêt et s'approchant d'une représentation d'un invariant.

Chapitre 2 Etat de l'art

L'aspect premier de cette étude est de caractériser et de classifier des manipulateurs robotiques industriels, à l'aide d'outils mathématiques, d'algorithmes et de programmes informatiques. De ce fait, les travaux présentés s'appuient sur les résultats et travaux, antérieurs et en cours, dans ces mêmes domaines.

2.1 Contexte robotique : Etude des lieux singuliers

2.1.1. Concepts principaux

Nous ne présenterons pas, dans ce document, les concepts, méthodologies et termes les plus basiques de robotique, que nous supposons maîtrisés. Le lecteur à la recherche de détails sur ces notions pourra se référer à un ouvrage de référence ou pédagogique sur la robotique tel que (Bop, 2010).

Nous rappelons, toutefois, ici les concepts principaux de robotiques sur lesquels s'appuieront les travaux détaillés dans ce document.

Définition 2.1.1.1 (Espace articulaire)

Espace vectoriel des variables articulaires du robot (décrivant l'état des liaisons actives du robot).

Définition 2.1.1.2 (Espace opérationnel)

Espace vectoriel des variables opérationnelles (variables de position et d'orientation de l'effecteur).

Définition 2.1.1.3 (Espace de travail)

Sous-espace, de l'espace opérationnel, des variables opérationnelles accessibles par l'effecteur.

Définition 2.1.1.4 (Espace des configurations)

Une configuration d'un robot désigne un couple formé par un vecteur de l'espace articulaire et un vecteur de l'espace de travail.

L'espace des configurations est l'ensemble des configurations admises par le robot.

Proposition 2.1.1.1 (Contraintes sur la configuration)

La géométrie d'un robot et les types de liaisons entre solides permettent de définir un système de contraintes sur les configurations admissibles par le robot.

Définition 2.1.1.5 (Fonction géométrique)

Un robot admet une *fonction géométrique* sur une partie A' de son espace articulaire si pour tout point de A' , le robot admet une unique position et orientation de son effecteur. La fonction géométrique sur A' associe, à un point de A' , la position et l'orientation de l'effecteur.

Remarque 2.1.1.1 (Espace articulaire et opérationnel vus comme espaces vectoriels)

Nous interprétons :

- Les variables articulaires sous forme de leurs commandes moteurs :
 - Chaque translation induite par une glissière est représentée par un élément de $\mathbb{R}$.
 - Pour chaque liaison pivot, la rotation totale, est représentée par un élément de $\mathbb{R}$, plutôt que par l'angle formé vis-à-vis de la rotation nominale.
- Les variables opérationnelles sous forme d'un vecteur « vivant dans $\mathbb{R}^6$ » :
 - Les trois variables de position définissent un vecteur de $\mathbb{R}^3$.
 - L'orientation de l'effecteur est caractérisée par le vecteur unitaire $u \in \mathbb{R}^3$ associé à l'axe principal du solide portant l'effecteur final.

Cette interprétation nous permet d'identifier, trivialement, chaque fonction géométrique à une fonction entre espaces vectoriels réels.

Définition 2.1.1.6 (Robot série)

Un robot dont chaque solide est relié au plus à deux autres solides et tel que la base et l'effecteur du robot soient chacun relié à un seul solide.

Proposition 2.1.1.2 (Fonction géométrique d'un robot série)

Un robot série admet une fonction géométrique sur tout son espace articulaire.

2.1.2. Lieux singuliers et postures

Soient, pour le robot considéré :

- A , son espace articulaire
- $\mathcal{P}$, son espace opérationnel
- W , son espace de travail
- $C \subseteq A \times W$, son espace de configuration
- C_s , l'ensemble des contraintes sur la configuration

Définition 2.1.2.1 (Posture)

L'ensemble des postures possibles, sur $A' \subseteq A$, de $p \in W$ est $\{q \in A' \mid C_s(q, p) \text{ sont vérifiées}\}$.

Les postures sont les *solutions géométriques inverses*, si une fonction géométrique est définie sur A' .

Définition 2.1.2.2 (Trajectoire)

Une fonction $Tr: \mathbb{R} \supset [a, b] \ni t \mapsto (q_t, p_t)$, continue, telle que $\forall t \in [a, b], C_s(q_t, p_t)$ sont vérifiées.

Définition alternative : Une courbe dans l'espace des configurations du robot.

Définition 2.1.2.3 (Changement de posture)

Lorsqu'une trajectoire $(q_t, p_t)_{t \in [a, b]}$ vérifie :

$$\exists (t_1, t_2) \in [a, b]^2 \text{ tel que } t_1 \neq t_2, p_{t_1} = p_{t_2} \text{ et } q_{t_1} \neq q_{t_2}$$

Proposition 2.1.2.1

Pour un robot série de fonction géométrique f , isoler des sous-espaces $A' \subseteq A$ et $W' \subseteq W$ tels que $\forall p \in W', \#\{q \in A' \mid f(q) = p\} = 1$ (où $\#E$ est le nombre d'éléments dans l'ensemble E) est nécessaire pour définir, localement, sur W' vers A' , un inverse à f et concevoir directement des trajectoires du robot dans des sous-espaces de W .

Dans le cadre d'une stratégie de conception où l'on souhaite que le robot puisse automatiquement calculer des trajectoires selon la tâche à effectuer, on peut se proposer de construire une fonction qui associe, à un point de l'espace de travail, un point unique de l'espace articulaire, de telle sorte que le robot respecte cette fonction, sous certaines conditions.

Définition 2.1.2.4 (Lieu singulier d'une fonction)

Le lieu singulier S_f d'une fonction $f : A \rightarrow \mathcal{P} \supseteq W$, différentiable sur tout A est défini par :

$$S_f = \{z \in A \mid Df(z) \text{ n'est pas de rang plein}\}$$

où $Df(z)$ est la matrice associée à la différentielle, df_z , de f , en z

Définition 2.1.2.5 (Lieu singulier d'un robot défini par une fonction)

Le lieu singulier d'un robot, qui est défini, sur tout A , par une fonction différentiable $f : A \rightarrow \mathcal{P}$, est S_f

Remarque 2.1.2.1 (Assertion fausse)

La conjecture que "Un changement de posture ne peut se produire que si la trajectoire traverse le lieu singulier du robot." est fausse.

En effet, certains robots présentent des changements de postures pour des trajectoires ne traversant pas le lieu singulier du robot.

Si la conjecture de la Remarque 2.1.2.1 était vraie, alors, pour un robot série, il suffirait de construire une *fonction géométrique inverse* pour chacun des sous-espaces, de l'espace articulaire, délimités par le lieu singulier, afin de diriger le robot, directement dans l'espace de travail, tant que l'on reste dans l'un de ces sous-espaces.

2.1.3. Robot cuspidal et changement non singulier de posture

2.1.3.1. Notion de cuspidalité

Définition 2.1.3.1 (Points cusps)

Point cusp :

Point de l'espace de travail admettant 3 solutions géométriques inverses confondues ($\Rightarrow$ *singulière*).

Point cusp *articulaire* :

Solution géométrique inverse triple d'un point cusp.

Définition 2.1.3.2 (Robot série cuspidal)

Un robot série cuspidal est un robot série qui présente au moins un cusp.

Ce terme désigne aussi, *par extension* un robot admettant un changement non singulier de posture, du fait que les robots avec cette particularité présentent souvent, en pratique, au moins un cusp.

Remarque 2.1.3.1 (Lien avec un cusp mathématique)

Un point cusp d'un robot série est un point cusp, au sens mathématique, dans au moins une section plane de l'espace de travail.

Remarque 2.1.3.2 (Contre image non singulière d'un cusp)

Un cusp présente usuellement une contre image non singulière, dans l'espace articulaire, en plus de sa contre image triple singulière. Cette propriété est illustrée à la Figure 1. Cette contre image régulière, de l'image du lieu singulier, au voisinage du cusp opérationnel, décrit un cusp mathématique.

Pour de plus amples détails, il est possible de se référer à (Corvez, 2005)

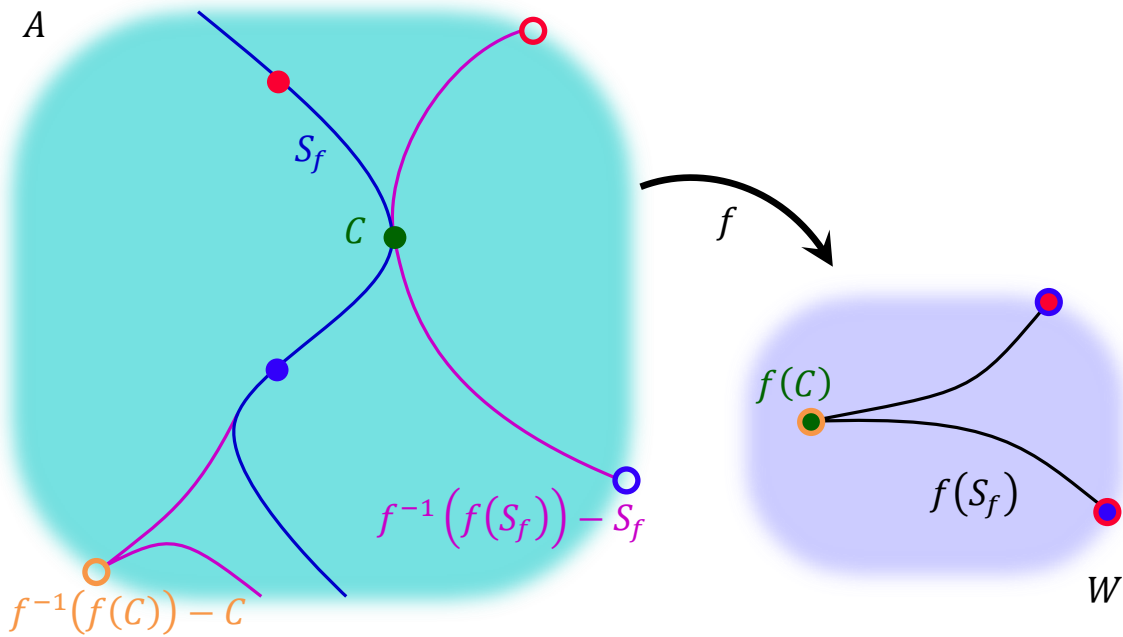


Figure 1 : Point cusp $f(C)$ et point cusp articulaire C d'une fonction géométrique f

2.1.3.2. Repères chronologiques

La chronologie menant à la Remarque 2.1.2.1, que nous détaillons ici est résumée depuis celle présentée dans (Wenger, 2007), les exemples venant étayer cette remarque étant des robots cuspidaux.

Les premiers travaux faisant état de changements de posture non singuliers ont été publiés en 1988. Citons premièrement l'article (Parenti, et al., 1988) mettant en évidence, numériquement, des trajectoires non singulières induisant des changements de postures, pour deux robots 6R. La thèse (Burdick, 1988) traite principalement des robots redondants et met en lumière de possibles changements de posture non-singulier. (Burdick, 1988) présente plusieurs exemples de robots 3R cuspidaux ainsi que des conditions générales sur les robots qui seraient cuspidaux et non cuspidaux. Cette propriété de *cuspidalité* ne fut toutefois pas étudiée plus en détail avant 1992 où elle fut confirmée et décrite, plus formellement, par P. Wenger. On retrouvera cette étude, reprise et améliorée, dans (Wenger, 2004). D'autres résultats sur la *cuspidalité* furent obtenus depuis 1992, notamment la façon de définir une trajectoire engendrant un changement non singulier de posture. On se référera pour cela à l'article (El Omri, et al., 1995).

Le fait majeur est qu'une trajectoire induisant un changement non singulier de posture est réalisable, au voisinage d'un point cusp, en entourant ce dernier dans l'espace de travail, tout en ne traversant pas le lieu singulier dans l'espace articulaire.

Une telle trajectoire $(\gamma, f(\gamma))$ traversera deux fois la contre image du lieu singulier, privée du lieu singulier lui-même, de manière à contenir deux des pré-images de l'un des points "à l'intérieur" de la zone définie par le cusp dans une coupe de l'espace de travail, comme représenté à la Figure 2.

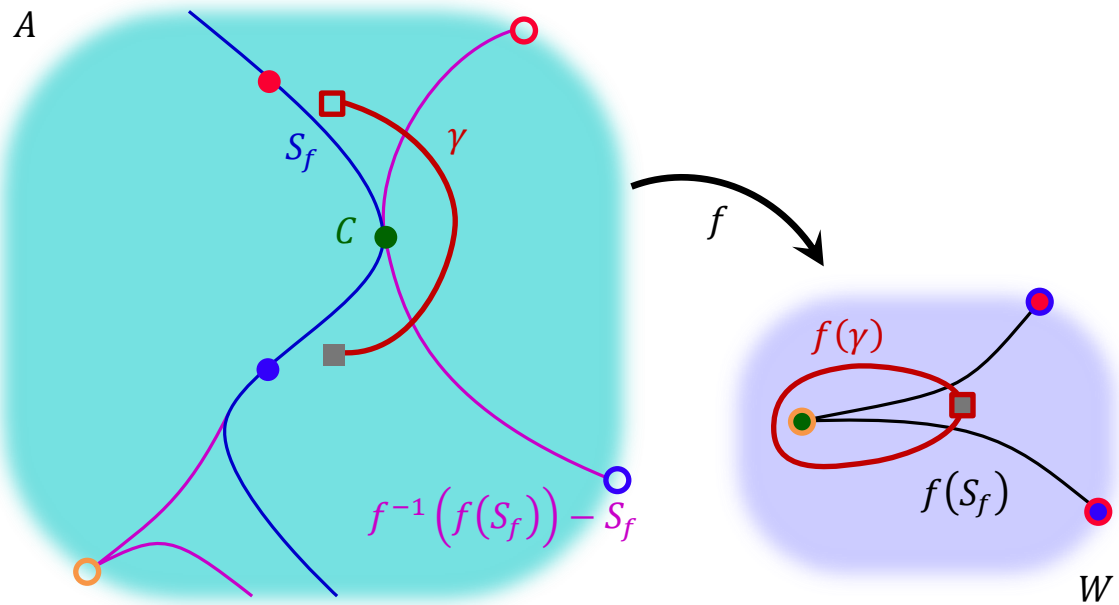


Figure 2 : Point cusp et trajectoire $(\gamma, f(\gamma))$ induisant un changement non singulier de posture

La notion de point cusp et de racine triple avait déjà été abordée dans certains travaux, tels que la thèse (Tsai, 1990), où ce point singulier spécifique est répertorié parmi les points singuliers stables des robots 3R généraux, sans que la propriété de cuspidalité qu'il induit ne soit mise en lumière.

Afin de visualiser la structure d'un robot 3R, son schéma cinématique est proposé à la Figure 3.

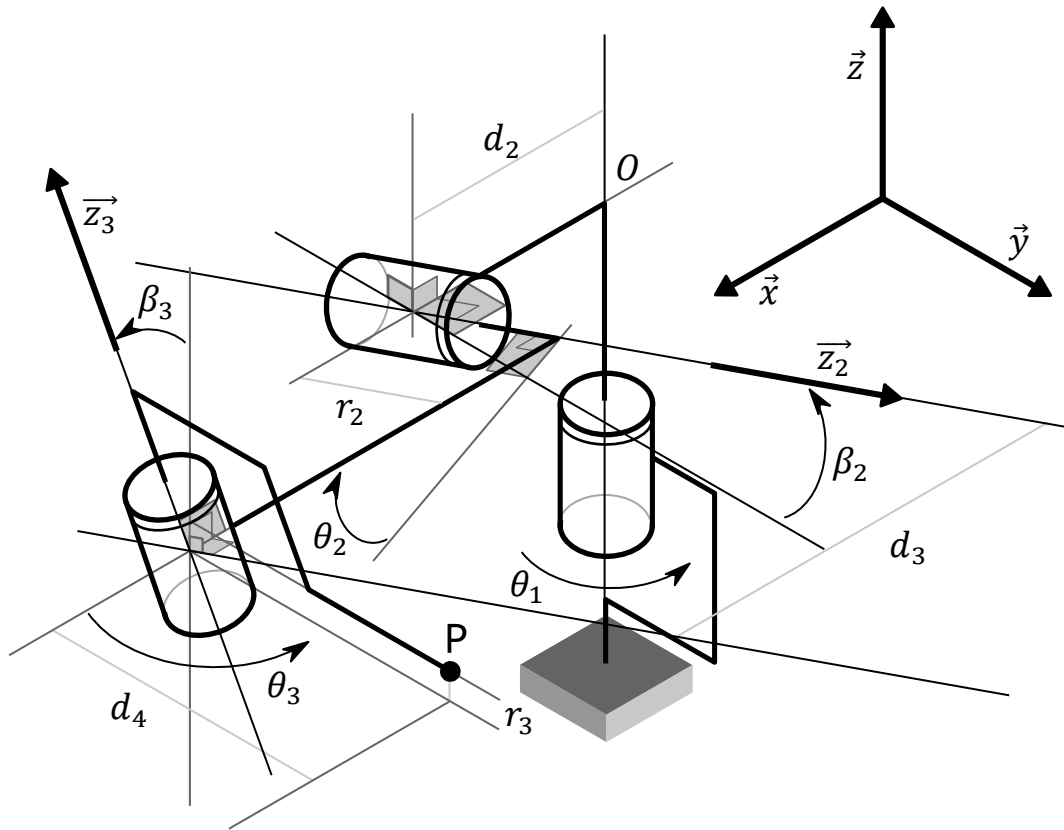


Figure 3 : Schéma cinématique d'un robot 3R (avec $\theta_1 = 0$) - orthogonal si $\beta_2 = \beta_3 = 0$

La présence d'un point cusp n'est toutefois pas démontrée comme étant une condition nécessaire, en général, pour qu'un changement non singulier de posture se produise. Cette nécessité n'a, pour l'instant, été établie que dans le cas des robots 3R orthogonaux, dans la thèse (Baili, 2004). La présence d'un cusp dans l'espace de travail d'un robot n'en est pas moins une condition suffisante à un possible changement non singulier de posture.

De ce fait, on pourra retrouver, dans (Burdick, 1995) certaines conditions, énoncées par J.W. Burdick, sur les paramètres de DH (« Denavit et Hartenberg ») d'un robot pour que celui-ci soit non cuspidal (i.e. il ne présente pas de changement non singuliers de posture). (Wenger, 1997) présente aussi certaines conditions supplémentaires, moins intuitives, énoncées par P. Wenger, qui impliquent la non cuspidalité d'un manipulateur.

2.1.3.3. Robots parallèles et changement de mode d'assemblage

Définition 2.1.3.3 (Robot parallèle (Merlet, 1997))

Un robot dont l'un des solides est relié à plus de deux autres solides, ou tel que la base ou l'effecteur soit relié à plus d'un solide.

Définition 2.1.3.4 (Mode d'assemblage)

L'ensemble des modes d'assemblage, sur $W' \subseteq W$, de $\rho \in A$ est $\{p \in W' | Cs(\rho, p) \text{ sont vérifiées}\}$.

Définition 2.1.3.5 (Changement de mode d'assemblage)

Lorsqu'une trajectoire $(\rho_t, p_t)_{t \in [a, b]}$ vérifie :

$$\exists (t_1, t_2) \in [a, b]^2 \text{ tel que } t_1 \neq t_2, p_{t_1} \neq p_{t_2} \text{ et } \rho_{t_1} = \rho_{t_2}$$

La notion de point cusp apparait de façon similaire pour les robots parallèles. En effet, la présence d'un point cusp, dans une section plane de l'espace articulaire, induit un possible changement non singulier de mode d'assemblage, du robot, au voisinage de ce point.

Il est à noter que ce point cusp est un point d'un autre type que les points cusp apparaissant pour les robots séries.

Il n'a pas été démontré que la présence d'un tel point est une condition nécessaire et suffisante de changement non singulier de mode d'assemblage, pour au moins un sous-ensemble de robots.

Il est d'ailleurs vérifié que la présence d'un tel point n'est pas globalement nécessaire, puisque de récents développements ont montré que les changements non singuliers de mode d'assemblage pouvaient être observés sans la présence de point cusp, pour certains robots parallèles comme cela est détaillé dans l'article (Coste, et al., 2014).

En conclusion :

Le comportement cuspidal est une découverte assez récente. L'utilisation et la compréhension de ce comportement pourraient contribuer à la conception de robots d'architecture novatrice et capable d'une plus grande adaptabilité, en proposant la possibilité d'accéder à plusieurs configurations sans s'approcher des configurations singulières qui peuvent engendrer des instabilités.

2.2 Application générique et contrôle des dérivées

Au cours de ce *document* nous considérerons très fréquemment des applications et fonctions *génériques*. De ce fait, nous allons préciser, auparavant le sens que nous donnons à cette propriété.

Pour cela, nous devons présenter quelques notions et résultats (énoncés par *René Thom*) sur la notion d'intersection transverse ainsi que les constructions de base permettant de manipuler cette notion de transversalité.

Les énoncés présentés dans cette section sont issus de (Golubitsky, et al., 1986). Nous renvoyons donc à cette référence, le lecteur qui voudrait retrouver les démonstrations des résultats avancés.

Désignations

- Dans tout ce document, afin d'éviter les confusions avec les **variétés algébriques**, précisons que le terme *variété* sera utilisé comme synonyme du terme anglais *manifold* et désignera une *variété différentielle*.
- Nous utiliserons le symbole $\pitchfork$ pour désigner la transversalité entre deux objets. La notion d'intersection transverse sera définie plus loin dans cette section.
- *lisse* et C^∞ sont synonymes de *infinitement différentiable*.
- Nous utiliserons des *multi-indices* α pour manipuler plus aisément des dérivées partielles multiples en écrivant, pour $\alpha = (k_j)_{j \in J}$, où $\forall j \in J, k_j \in \mathbb{N}^*$:

$$|\alpha| = \#J \in \mathbb{N} \text{ et } \partial x_\alpha = \prod_{j \in J} \partial x_{k_j}, \text{ d'où } \frac{\partial^{|\alpha|} f}{\partial x_\alpha}(p) = \frac{\partial^{|\alpha|} f}{\prod_{j \in J} \partial x_{k_j}}(p)$$

- On note $\mathcal{T}_M(E)$, ou plus simplement $\mathcal{T}_M E$, l'espace tangent à l'espace E , au point M .

2.2.1. Espace des jets et jets d'applications

Définition 2.2.1.1 (Espace des jets : construction, terminologie)

Soit X et Y deux variétés de classe C^∞ , $p \in X$ et $(f, g) \in (C^\infty(X, Y))^2$

1. f et g ont un contact d'ordre 0, en p , si $\exists q \in Y, f(p) = g(p) = q$
2. f et g ont un contact d'ordre 1, en p , si f et g ont un contact d'ordre 0, en p et si $(df)_p = (dg)_p$ (en tant qu'applications, entre espaces tangents, de $\mathcal{T}_p(X)$ dans $\mathcal{T}_p(Y)$).
3. f et g ont un contact d'ordre k (pour $k \in \mathbb{N}^*$) en p , si df et dg ont un contact d'ordre $(k-1)$ en tout point de $\mathcal{T}_p(X)$.
4. « $f \sim_{k,p,q} g$ » signifiera « f et g ont un contact d'ordre k en p et $f(p) = g(p) = q$ »
5. $J^k(X, Y)_{p,q}$ désigne l'ensemble des classes des applications $f: X \rightarrow Y$ telles que $f(p) = q$, selon l'équivalence $\sim_{k,p,q}$.
6. Les k -jet de X dans Y sont les éléments de $J^k(X, Y) = \bigcup_{(p,q) \in X \times Y} J^k(X, Y)_{p,q}$.

Proposition 2.2.1.1 (Issue de la Définition 2.2.1.1)

La Définition 2.2.1.1 induit, pour les espaces des jets, que :

1. $\forall (p, q) \in X \times Y, J^0(X, Y)_{p,q}$ est restreint à une seule classe (non vide, contenant toutes les applications f telles que $f(p) = q$).
 $J^0(X, Y)$ peut donc s'identifier à $X \times Y$.
2. Si X est un ouvert de $\mathbb{R}^n$, $Y = \mathbb{R}^m$ et $(f, g) \in (C^\infty(X, Y))^2$.
Alors $f \sim_{k,p} g$ ($\Leftrightarrow f \sim_{k,p,q} g$ pour une image q quelconque) ssi toute dérivée partielle, d'ordre inférieur ou égal à k , d'une composante f_i , de f , en p , est égale à la même dérivée partielle de la même composante g_i , de g .

$$\text{Autrement dit : } f \sim_{k,p} g \Leftrightarrow \forall \alpha \text{ tel que } |\alpha| \leq k, \frac{\partial^{|\alpha|} f_i}{\partial x_\alpha}(p) = \frac{\partial^{|\alpha|} g_i}{\partial x_\alpha}(p)$$

3. Si X est un ouvert de $\mathbb{R}^n$, $Y = \mathbb{R}^m$ et $(f, g) \in (C^\infty(X, Y))^2$.
Alors $f \sim_{k,p} g$ ssi $\forall l \leq k$, les développements de Taylor de f et g à l'ordre l sont égaux.

Preuve 2.2.1.1 (De la Proposition 2.2.1.1)

Nous démontrons d'abord (1) et que (2) $\Leftrightarrow$ (3). Nous démontrons alors (2) et (3) en démontrant (2).

(1) : Relativement immédiat, considérant le fait que toutes les applications de $J^0(X, Y)_{p,q}$ sont des applications f telles que $f(p) = q$ et que $f \sim_{0,p,q} g \Leftrightarrow f(p) = g(p) = q \Leftrightarrow (f, g) \in (J^0(X, Y)_{p,q})^2$. De plus $J^0(X, Y)_{p,q}$ n'est pas vide car on peut considérer l'application $c_q : X \ni x \mapsto q \in Y$. $J^0(X, Y)$ étant la réunion sur $X \times Y$ d'ensembles de classes, contenant chacun une classe unique, cet ensemble s'identifie canoniquement à son index $X \times Y$.

(2) $\Leftrightarrow$ (3) :

$\Rightarrow$: Se prouve par la construction du développement de Taylor à partir des dérivées partielles d'ordres inférieurs ou égaux à l'ordre du développement.

$\Leftarrow$: En identifiant les coefficients du développement de Taylor, on peut retrouver les dérivées partielles d'ordres inférieurs ou égaux à l'ordre du développement de Taylor

(2) : On démontre ce résultat, pour $f, g : U \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$, en p , par récurrence :

Initialisation : Un contact d'ordre 1 en p équivaut au fait que $(df)_p = (dg)_p$, ce qui implique l'égalité des dérivées partielles de f et de g d'ordre 1.

(Notons qu'un contact d'ordre 0 en p implique $f(p) = g(p)$, ce qui correspond à une égalité des dérivées partielles d'ordre 0.)

Récurrence : Supposons que l'on a le résultat (2) pour les ordres strictement inférieurs à k .

L'application df_p est définie comme l'application : $df_p : \mathfrak{U}_p U = \mathbb{R}^n \rightarrow \mathbb{R}^m = \mathfrak{U}_{f(p)} \mathbb{R}^m$ telle que :

$$df_p : y \mapsto \left(\sum_{j=1}^n \frac{\partial f_i}{\partial x_j}(p) y_j \right)_{i \in \llbracket 1; m \rrbracket} \quad (a)$$

L'expression de l'application dg_p s'obtient en remplaçant f par g dans l'équation (a), en remarquant que $f \sim_{k,p} g$ (avec $k \geq 0$) $\Rightarrow f(p) = g(p)$ et $dg_p : \mathfrak{U}_p U = \mathbb{R}^n \rightarrow \mathbb{R}^m = \mathfrak{U}_{g(p)} \mathbb{R}^m = \mathfrak{U}_{f(p)} \mathbb{R}^m$.

$f \sim_{k,p} g \Rightarrow \forall v \in \mathbb{R}^n, (df_p) \sim_{(k-1),v} (dg_p)$.

Par hypothèse de récurrence, on a donc :

$$\forall i \in \llbracket 1; m \rrbracket, \forall v \in \mathbb{R}^n, \forall \alpha \text{ multi-indice tel que } |\alpha| \leq (k-1), \frac{\partial^{|\alpha|} \bar{f}_i}{\partial x_\alpha}(p, v) = \frac{\partial^{|\alpha|} \bar{g}_i}{\partial x_\alpha}(p, v) \quad (b)$$

$\forall j \in \llbracket 1; n \rrbracket$, considérons les vecteurs $v_j = (\delta_{l,j})_{l \in \llbracket 1; n \rrbracket}$ où $\delta_{l,j}$ vérifie $\delta_{l,j} = 0$ si $l \neq j$ et $\delta_{j,j} = 1$.

En considérant ces vecteurs v_j , l'équation (b) implique :

$$\forall j \in \llbracket 1; n \rrbracket, \forall i \in \llbracket 1; m \rrbracket, \forall \alpha \text{ multi-indice tel que } |\alpha| \leq (k-1) \frac{\partial^{|\alpha|+1} f_i}{\partial x_\alpha \partial x_j}(p) = \frac{\partial^{|\alpha|+1} g_i}{\partial x_\alpha \partial x_j}(p) \quad (c)$$

L'équation (c) implique que (2) est vraie jusqu'au rang k , du fait que les équations précédentes décrivent toutes les dérivées partielles d'ordres inférieurs ou égaux à k .

La Définition 2.2.1.1 nous permet de définir le jet d'une application $j^k(f)$:

Définition 2.2.1.2 (Jet d'une application)

Soit $f : X \rightarrow Y$, une application lisse ($\mathcal{C}^\infty$). On définit le k -jet de f comme l'application :

$$j^k f : X \ni p \mapsto [f]_{J^k(X,Y)_{p,f(p)}} \in J^k(X, Y)$$

Proposition 2.2.1.2 (Propriétés des jets d'applications)

A partir du point (3) de la Proposition 2.2.1.1, on peut affirmer que $j^k f(x)$ est une autre façon de décrire le développement de Taylor à l'ordre k , de f en x .

$j^k f$ est une application lisse (i.e. *infinitement différentiable*).

2.2.2. Topologies sur $\mathcal{C}^\infty(X, Y)$ et sous-ensembles résiduels

Définition 2.2.2.1 (Ensemble résiduel)

Soit $(E, \mathcal{T}_E)$, un espace topologique (*i.e.* l'espace E , muni de la topologie, sur $E, \mathcal{T}_E$).

V est un sous-ensemble résiduel de E , selon la topologie $\mathcal{T}_E$, si il peut être décrit par une intersection dénombrable d'ensembles, chacun dense selon $\mathcal{T}_E$.

Définition 2.2.2.2 (Application générique)

Une application f est générique si elle appartient à un ensemble d'applications F , préalablement défini, qui est au moins *résiduel* pour l'espace topologique $(E, \mathcal{T}_E)$ considéré.

L'objectif principal de cette sous-section est de poser les bases permettant de comprendre correctement la Proposition 2.2.2.1. Cette proposition nous permettra de prouver que des ensembles d'applications considérées sont des ensembles d'applications *génériques* et ceci, uniquement en vérifiant qu'un nombre « suffisant » de conditions sur leurs dérivées ne sont pas vérifiées.

La notion de *généricité* formalise la « généralité » d'un ensemble d'applications, motivant l'intérêt de méthodes et de constructions optimisées vis-à-vis de ces applications « standard ».

La Proposition 2.2.2.1 est issue de (Golubitsky, et al., 1986) et est, plus particulièrement, une reformulation des Théorèmes 4.4 et 4.9 du chapitre 2, section 4, de cet ouvrage.

La démonstration de la Proposition 2.2.2.1 fait appel à la notion de *transversalité*, que nous détaillerons, en même temps que cette démonstration dans la sous-section 2.2.3.

Proposition 2.2.2.1 (Résidualité de fonctions qui ne satisfont pas un système R)

Soient X et Y , deux variétés et $f : X \rightarrow Y$, une application.

Soit R , un système d'équations sur un espace des jets $J^k(X, Y)$ des applications $\{f : X \rightarrow Y\}$.

(On notera que le jet $j^k f$ d'une application f est une nouvelle application, qui formalise l'ensemble des applications formé par f et ses dérivées d'ordre au plus k .)

Désignons par $\text{codim}(R)$, le nombre d'équations fonctionnellement indépendantes du système R .

Soit R_f , le système d'équations sur les points de X , induit par le système R , lorsque appliqué à f .

Si X et Y sont deux variétés et si $\dim(X) < \text{codim}(R)$,

alors $\{f : X \rightarrow Y \mid R_f \text{ n'a pas de racines}\}$ est un ensemble résiduel pour la topologie $\mathcal{C}^\infty$ de Whitney.

Pour que la Proposition 2.2.2.1 soit totalement compréhensible, il est nécessaire que la topologie $\mathcal{C}^\infty$ de Whitney soit elle-même définie.

Par souci de clarté, nous formulons d'abord une définition simple de la topologie $\mathcal{C}^\infty$ de Whitney, indépendamment de la notion de jet d'application.

Définition 2.2.2.3 (Topologie $\mathcal{C}^\infty$ de Whitney : $\mathcal{T}^\infty(X, Y)$)

Soit l'ensemble $\mathcal{C}^\infty(X, Y)$ des applications indéfiniment différentiables (*i.e. lisses*) de X vers Y .

Quel que soient les espaces topologiques X et Y , la topologie $\mathcal{C}^\infty$ de Whitney, notée $\mathcal{T}^\infty(X, Y)$ (*qui peut être associée à $\mathcal{C}^\infty(X, Y)$, pour en faire un espace topologique*)

est la topologie engendrée par les ouverts définis comme :

un ensemble d'applications $f \in \mathcal{C}^\infty(X, Y)$ telles que, il existe un ouvert U (de dimension appropriée) et $\exists k \in \mathbb{N}$, tel que les images de X , par les différentielles d'ordre k de f sont incluses dans l'ouvert U .

Nous allons maintenant détailler comment $\mathcal{T}^\infty(X, Y)$ peut être explicitement construite, en s'appuyant alors sur la notion de jet, omise dans la Définition 2.2.2.3.

Définition 2.2.2.4 (Topologies $\mathcal{T}^k(X, Y)$ (*i.e. $\mathcal{C}^k$ de Whitney*) : construction)

Soit X et Y , deux variétés lisses et $k \in \mathbb{N}$

Soit $U_k \subseteq J^k(X, Y)$ et $\mathcal{J}(U_k) = \{f \in \mathcal{C}^\infty(X, Y) \mid j^k(X) \subseteq U_k\}$

$\mathcal{T}^k(X, Y)$, la topologie $\mathcal{C}^\infty$ de Whitney, sur $\mathcal{C}^\infty(X, Y)$, est la topologie engendrée par :

$$\{\mathcal{J}(U_k) \mid U_k \text{ est un ouvert de } J^k(X, Y)\}$$

$\mathcal{T}^\infty(X, Y)$ peut alors être définie comme la topologie engendrée par :

$$\mathcal{U} = \bigcup_{k \in \mathbb{N}} \mathcal{U}_k \text{ où } \mathcal{U}_k \text{ est l'ensemble des ouverts de } \mathcal{T}^k(X, Y)$$

On notera que la Définition 2.2.2.4 est une retranscription de la méthode de construction de $\mathcal{T}^\infty(X, Y)$, telle que présentée à la section 3 du chapitre 2 de (Golubitsky, et al., 1986).

2.2.3. Transversalité et genericité - Théorème de Thom

Définition 2.2.3.1 (Transversalité)

Soient X et Y , deux variétés de classe $\mathcal{C}^\infty$

Soient $x \in X$ et W , une sous-variété de Y ainsi que $f \in \mathcal{C}^\infty(X, Y)$

$f \pitchfork_x W$ (*i.e. f intersecte transversalement W , en x*) si l'une des deux propriétés suivantes est vérifiée :

- $f(x) \notin W$
- $f(x) \in W$ et $\mathbb{T}_{f(x)}Y = \mathbb{T}_{f(x)}W + (df)_x(\mathbb{T}_xX)$

Si $A \subseteq X$, $f \pitchfork_A W$ (*i.e. f intersecte transversalement W , en A*) si $\forall x \in A, f \pitchfork_x W$

Par extension, lorsque $A \subseteq W$, on note $f \pitchfork_A W$ si $\forall x \in X \mid f(x) \in A, f \pitchfork_x W$

Enfin, si $f \pitchfork_X W$, on notera $f \pitchfork W$ (*i.e. f intersecte transversalement W*)

Les propriétés de la transversalité qui suivent nous permettront de prouver la Proposition 2.2.2.1.

Proposition 2.2.3.1 (Transversalité et codimension)

Soient X et Y , deux variétés de classe $\mathcal{C}^\infty$ et W , une sous-variété de Y ainsi que $f \in \mathcal{C}^\infty(X, Y)$

Si $f \pitchfork W$ Alors $f^{-1}(W)$ est une sous-variété de X et $\text{codim}(f^{-1}(W)) = \text{codim}(W)$

Conséquences

Si $\dim(X) = \text{codim}(W)$ alors $f^{-1}(W)$ est un ensemble de points isolés.

Si $\dim(X) < \text{codim}(W)$ alors $f^{-1}(W) = \emptyset$

La Proposition 2.2.3.1 nous permettra, en premier lieu, de conclure quant à la nature des racines d'une application et quant à la sous-variété qu'elles forment, en nous basant sur sa dimension.

Toutefois, l'intérêt de la Proposition 2.2.3.1 dévoile toute son envergure en combinaison avec la Proposition 2.2.3.2, ci-après.

Proposition 2.2.3.2 (Théorème de transversalité de Thom)

Soient X et Y , deux variétés de classe $\mathcal{C}^\infty$ et Z , une sous-variété de $J^k(X, Y)$

$\uparrow_Z = \{f \in \mathcal{C}^\infty(X, Y) \mid j^k f \notin Z\}$ est un ensemble résiduel pour $\mathcal{T}^\infty(X, Y)$

De plus, si Z est fermée, alors $\uparrow_Z$ est ouvert.

La Proposition 2.2.3.1 et la Proposition 2.2.3.2 combinées impliquent trivialement le Corollaire 2.2.3.1 qui implique la Proposition 2.2.2.1 lorsque Z est une variété définie par un système R .

Corollaire 2.2.3.1

Soient X et Y , deux variétés de classe $\mathcal{C}^\infty$

Soit Z , une sous-variété de $J^k(X, Y)$, telle que $\dim(X) < \text{codim}(Z)$

Alors $j^k f \notin Z \Leftrightarrow j^k f \cap Z = \emptyset$ et $\{f \in \mathcal{C}^\infty(X, Y) \mid j^k f \cap Z = \emptyset\}$ est résiduel (pour $\mathcal{T}^\infty(X, Y)$).

On notera toutefois que la Proposition 2.2.3.2 permet de définir une généricité plus large où la transversalité n'est pas restreinte au cas trivial de la non intersection, selon la définition :

Définition 2.2.3.2 (Application générique selon une contrainte/variété)

Une application $f \in \mathcal{C}^\infty(X, Y)$ est générique selon la sous-variété Z de $J^k(X, Y)$, si $j^k f \notin Z$

(Alternativement f est générique selon la contrainte dont l'espace complémentaire est défini par Z .)

2.2.4. Extension des définitions et théorèmes aux multi-jets

Au regard des définitions sur le jet d'application et des propositions avancées jusqu'ici, on ne peut analyser que des conditions locales sur les applications et leurs différentielles.

Il est toutefois possible d'étendre les propositions énoncées afin de considérer des conditions, définies sur un nombre fini de points distincts.

Pour ce faire, il est nécessaire de définir :

Définition 2.2.4.1 (Multi-jet)

Soient :

- $f : X \rightarrow Y$, où X et Y sont des variétés
- $\mathfrak{J}^m(X) = \{(x_i)_{i \in \llbracket 1; m \rrbracket} \mid \exists (i, j) \in \llbracket 1; m \rrbracket^2, i \neq j \text{ et } x_i = x_j\} \subset X^m$, où $\llbracket a; b \rrbracket = [a, b] \cap \mathbb{Z}$
- $\mathfrak{K}_m(X) = X^m - \mathfrak{J}^m(X)$

Le m -multi- k -jet de f , $j_m^k(f)$ est le k -jet de l'application $\mathfrak{K}_m(f) : \mathfrak{K}_m(X) \rightarrow Y^m$ définie par :

$$\forall i \in \llbracket 1; m \rrbracket, \forall x \in X, (\mathfrak{K}_m(f))_i(x) = f(x)$$

Autrement dit : $j_m^k(f) = j^k(\mathfrak{K}_m(f))$

On peut alors étendre tous les résultats, que l'on a déjà énoncés pour les jets, à la notion de multi-jets, notamment le Corollaire 2.2.3.1, en rappelant que la dimension de $\mathfrak{K}_m(X)$ est la même que celle de X^m , et est donc égale à $m \times \dim(X)$

Remarque 2.2.4.1 (Théorème de Thom pour les multi-jets)

Le théorème de Thom s'applique aussi à $\mathfrak{K}_m(f)$, en tant qu'application, ce qui permet de déduire certaines résidualités, par transversalité, pour $\mathcal{T}^\infty(\mathfrak{K}_m(X), Y^m)$.

Toutefois, on peut aussi déduire que les applications f , telles que $\mathfrak{K}_m(f)$ vérifie une condition de transversalité appropriée, forment une partie résiduelle pour $\mathcal{T}^\infty(X, Y)$.

2.3 Analyse par Intervalles

2.3.1. Principe et definitions

L'Analyse par Intervalles est un concept de calcul, opérant sur des intervalles (i.e. des ensembles connexes) encadrant des valeurs plutôt que sur des valeurs isolées. L'objectif de cette méthode réside principalement dans son application au calcul numérique, de par sa capacité à garantir que des valeurs appartiennent à un intervalle dont les bornes peuvent être représentées *exactement* par un ordinateur. (Jaulin, et al., 2001) et (Moore, 1996) constituent des ouvrages de référence sur le concept et les bases de l'Analyse par Intervalles.

Toutefois, on pourrait résumer ce concept par l'affirmation :

L'Analyse par Intervalles est une méthode de calcul simultané des valeurs et des erreurs.

Définition 2.3.1.1 (Espace des intervalles $\mathbb{I}E$ de E)

Soit un espace topologique $(E, \mathcal{T}_E)$ (i.e. l'espace E , muni de la topologie, sur E , $\mathcal{T}_E$).

L'espace des intervalles de $(E, \mathcal{T}_E)$ est l'ensemble des éléments connexes de $\mathcal{T}_E$.

Nous désignerons cet ensemble par $\mathbb{I}_{\mathcal{T}_E}$ et par $\mathbb{I}E$ ou $\mathbb{I}(E)$ lorsque la topologie est sans ambiguïté.

Définition 2.3.1.2 (Sous-espace de $\mathbb{I}E$ recouvrant E)

$\Omega \subseteq \mathbb{I}E$ recouvre E si $\forall z \in E, \exists I \in \Omega$ tel que $z \in I$

(Ceci implique que l'on a alors $\bigcup_{I \in \Omega} I \supseteq E$)

Définition 2.3.1.3 (Fonction de taille et Taille d'un intervalle)

Une fonction $w_\Omega : \Omega \rightarrow \overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$ est une fonction de taille sur Ω si :

$$\forall (I, J) \in \Omega^2, \quad I \subseteq J \Rightarrow w_\Omega(I) \leq w_\Omega(J)$$

$w_\Omega(I)$ est alors la taille de I selon w_Ω

Si Ω et w_Ω sont sans ambiguïté, on définira « la taille de I » comme étant $w(I) = w_\Omega(I)$

Définition 2.3.1.4 (Fonction d'inclusion de f)

Une fonction d'inclusion $[f]_{\Omega_A; \Omega_p} : \Omega_A \rightarrow \Omega_p$ de $f : A \rightarrow \mathbb{P}$ vérifie :

- Ω_A recouvre A et Ω_p recouvre $f(A)$ (et, usuellement, recouvre $\mathbb{P}$)
- $\forall I \in \Omega_A, [f]_{\Omega_A; \Omega_p}(I) \supseteq f(I)$

On qualifiera alors f de *fonction naturelle de* $[f]_{\Omega_A; \Omega_p}$.

Par simplicité d'écriture, on posera $[f] = [f]_{\Omega_A; \Omega_p}$ lorsque (Ω_A, Ω_p) est fixé ou quelconque.

Définition 2.3.1.5 (Intervalle minimal d'un ensemble)

Soit $V \subseteq E$ et $\Omega \subseteq \mathbb{I}E$, un intervalle minimal de V dans Ω , noté $\langle V \rangle_\Omega$ vérifie :

$$V \subseteq \langle V \rangle_\Omega \in \Omega \quad \text{et} \quad \forall I \in \Omega, \quad (V \subseteq I \subseteq \langle V \rangle_\Omega) \Rightarrow I = \langle V \rangle_\Omega$$

Notons que si Ω est stable par l'intersection et qu'un intervalle $\langle V \rangle_\Omega$ existe, alors $\langle V \rangle_\Omega$ est unique et

$$\langle V \rangle_\Omega = \bigcap_{V \subseteq J \in \Omega} J$$

Définition 2.3.1.6 (Fonction d'inclusion minimale de f)

Une fonction d'inclusion $\langle f \rangle_{\Omega_A; \Omega_p} : \Omega_A \rightarrow \Omega_p$ de $f : A \rightarrow \mathbb{P}$ est minimale si elle vérifie :

$$\forall I \in \Omega_A, \forall J \in \Omega_p, \quad \left(f(I) \subseteq J \subseteq \langle f \rangle_{\Omega_A; \Omega_p}(I) \right) \Rightarrow J = \langle f \rangle_{\Omega_A; \Omega_p}(I)$$

Selon la Définition 2.3.1.5, on peut aussi définir une fonction d'inclusion minimale $\langle f \rangle_{\Omega_A; \Omega_p}$ de f par

$$\forall I \in \Omega_A, \quad \langle f \rangle_{\Omega_A; \Omega_p}(I) = \langle f(I) \rangle_{\Omega_p}$$

Bien que l'Analyse par Intervalles fait sens pour toute fonction d'inclusion, on considèrera, sauf mention explicite du contraire que la fonction d'inclusion $[f]_{\Omega_A; \Omega_P}$ est choisie de façon à ce que :

- $[f]_{\Omega_A; \Omega_P}$ s'approche d'une fonction d'inclusion minimale.
- Selon sa définition : « $[f]_{\Omega_A; \Omega_P}$ tend vers f lorsque (Ω_A, Ω_P) tend vers $(\mathbb{I}A, \mathbb{I}P)$ ».

« S'approcher d'une fonction d'inclusion minimale » veut aussi dire « Définir $[f]_{\Omega_A; \Omega_P}$ de telle sorte que les intervalles retournés sont construits pour être les plus petits possibles, au sens de l'inclusion ». Toutefois quoi que ce premier objectif fait intuitivement sens, il est nécessaire de formellement définir ce que l'on entend par « $[f]_{\Omega_A; \Omega_P}$ tend vers f lorsque (Ω_A, Ω_P) tend vers $(\mathbb{I}A, \mathbb{I}P)$ ».

Définition 2.3.1.7 (Fonction d'inclusion $\mathbb{I}f$, extension par intervalles de f)

Une extension par intervalles de $f : A \rightarrow P$ est une fonction d'inclusion $\mathbb{I}f : \mathbb{I}A \rightarrow \mathbb{I}P$ définie par :

$$\forall I \in \mathbb{I}A, \quad \mathbb{I}f(I) = \langle f(I) \rangle_{\mathbb{I}P}$$

(i.e. $\mathbb{I}f(I)$ est un intervalle minimal de $f(I)$ dans $\mathbb{I}P$ et $\mathbb{I}f$ est une fonction d'inclusion minimale.)

On notera que l'existence d'une telle fonction d'inclusion impose $\forall I \in \mathbb{I}A, \exists J \in \mathbb{I}P$ tel que $f(I) \subseteq J$. Ceci est toujours vrai lorsque P est lui-même connexe.

De plus, si $\mathbb{I}P$ est stable par l'intersection ou, plus généralement, si $\mathbb{I}f$ est unique, alors $\mathbb{I}f$ vérifie :

$$\forall I \in \mathbb{I}A, \quad \mathbb{I}f(I) = \bigcap_{f(I) \subseteq J \in \mathbb{I}P} J$$

Définition 2.3.1.8 (fonction d'inclusion étagée f)

En termes colloquiaux, il s'agit d'une fonction d'inclusion $[f]_{\Omega_A; \Omega_P}$ « qui converge vers $\mathbb{I}f$ ».

Formellement, $[f]_{\Omega_A; \Omega_P}$ étagée f si :

Il existe une famille de fonctions d'inclusions $([f]_{\Omega_a; \Omega_p})_{(\Omega_a, \Omega_p) \subseteq \mathbb{I}A \times \mathbb{I}P}$ telle que :

- $\forall ((\Omega_a, \Omega_p), (\Omega_s, \Omega_t)) \subseteq (\mathbb{I}A \times \mathbb{I}P)^2, \forall I \in \Omega_a, \forall J \in \Omega_s :$

$$I \subseteq J \Rightarrow \forall V = \bigcup_{k \in K} E_k \text{ tel que } [f]_{\Omega_s; \Omega_t}(J) \subseteq V, \text{ on a } [f]_{\Omega_a; \Omega_p}(I) \subseteq V$$
- $\lim_{\Omega_p \rightarrow \mathbb{I}P} [f]_{\Omega_a; \Omega_p} = \langle f \rangle_{\Omega_a; \mathbb{I}P} : \Omega_a \ni I \mapsto \langle f(I) \rangle_{\mathbb{I}P}$
- $\lim_{\Omega_a \rightarrow \mathbb{I}A} \langle f \rangle_{\Omega_a; \mathbb{I}P} = \mathbb{I}f$

On dira alors que $[f]_{\Omega_A; \Omega_P}$ étagée f , par l'intermédiaire de $F = ([f]_{\Omega_a; \Omega_p})_{(\Omega_a, \Omega_p) \subseteq \mathbb{I}A \times \mathbb{I}P}$.

De plus, comme $\forall (\Omega_a, \Omega_p) \subseteq \mathbb{I}A \times \mathbb{I}P, [f]_{\Omega_a; \Omega_p}$ étagée f , par l'intermédiaire de F , on dira alors, par extension, que « $F = ([f]_{\Omega_a; \Omega_p})_{(\Omega_a, \Omega_p) \subseteq \mathbb{I}A \times \mathbb{I}P}$ étagée f ».

Remarque 2.3.1.1 (Compléments sur l'étagage de f)

On a $\langle f \rangle_{\Omega_a; \mathbb{I}P} : \Omega_a \ni I \mapsto \bigcap_{f(I) \subseteq J \in \mathbb{I}P} J$ si $\forall I \in \Omega_a, \{J \in \mathbb{I}P \mid f(I) \subseteq J\}$ est stable par l'intersection.

En effet, par définition de $\mathbb{I}P$, si l'intersection conserve la connexité, pour tout K tel que $V \subseteq K \in \mathbb{I}P$ alors le plus petit élément, pour l'inclusion, de $\mathbb{I}P$ contenant V est l'élément connexe $(\bigcap_{V \subseteq K \in \mathbb{I}P} K)$.

Par définition, l'existence d'une fonction d'inclusion étagée f , impose donc l'existence de $\mathbb{I}f$.

D'un point de vue pratique, le fait que $[f]_{\Omega_A; \Omega_P}$ étagée f signifie principalement que $[f]_{\Omega_A; \Omega_P}$ peut être « affinée en même temps que Ω_A et Ω_P » en la substituant par une fonction d'inclusion cohérente, ceci jusqu'à la proximité choisie de $\mathbb{I}f$.

Le fait que $[f]_{\Omega_A; \Omega_P}$ étagée f justifie l'identification usuelle de $[f]_{\Omega_A; \Omega_P}$ à la fonction naturelle f .

Cette identification fait sens puisque $[f]_{\Omega_A; \Omega_P}$ constitue alors une représentation cohérente de f , lorsque l'on se restreint aux intervalles de Ω_A et de Ω_P .

2.3.2. Application au calcul numérique informatique

Il est impossible à un ordinateur d'indexer, et donc de coder, simultanément, la valeur numérique de tous les nombres réels. De ce fait, le calcul numérique par ordinateur ne peut retourner la valeur numérique exacte d'un résultat que pour un nombre fini de réels. On note alors que le résultat retourné peut être majoré ou minoré par des valeurs représentables par l'ordinateur.

L'encadrement de tout réel par des valeurs représentables est ce qui motive l'Analyse par Intervalles.

2.3.2.1. Les ensembles d'intervalles considérés

En pratique, on étudiera chaque espace $\mathbb{R}^k$, où $k \in \mathbb{N}$, par l'intermédiaire d'un ensemble d'intervalles $\Omega \subseteq \mathbb{IR}^k$ et on considèrera, usuellement, dans nos applications :

- Soit $\mathbb{IR}$, l'ensemble des éléments de $\mathbb{R}$, dont la valeur numérique est codée par l'index utilisé.
- Soit $\mathbb{IR} = \{ \{a, b\} \mid (\{, \}) \in \{[,]\}^2, (a, b) \in (\mathbb{IR} \cup \{-\infty, +\infty\}) \text{ et } \{a, b\} \subseteq \mathbb{R} \}$
(i.e. $\mathbb{IR}$ est l'ensemble des intervalles de $\mathbb{R}$ dont les bornes appartiennent à $\mathbb{IR} \cup \{-\infty, +\infty\}$.)
- On pose $\Omega = (\mathbb{IR})^k$

On remarquera que Ω vérifie alors :

- Ω recouvre $\mathbb{R}^k$ et est stable par l'intersection.
- On peut donc représenter tout point x de $\mathbb{R}^k$ par $\bigcap_{x \in J \in \Omega} J$ qui est ainsi défini.

2.3.2.2. La fonction de taille utilisée

Considérons, donc, $\Omega = (\mathbb{IR})^k$, tel que défini précédemment, au § 2.3.2.1.

Nous proposons de définir la fonction de taille w utilisée, à partir d'une norme sur $\mathbb{R}^k$:

- Soit $m : \mathbb{R}^k \rightarrow \mathbb{R}$, la norme choisie sur $\mathbb{R}^k$
- Soit $\mathfrak{X} : (\mathbb{IR})^k \rightarrow \mathbb{R}^k$, la fonction « différence des bornes » définie par :

$$\mathfrak{X}\left(\prod_{i \in \llbracket 1; n \rrbracket} \{a_i, b_i\}\right) = (b_i - a_i)_{i \in \llbracket 1; n \rrbracket} \text{ où } \forall i \in \llbracket 1; n \rrbracket, (\{, \}) \in \{[,]\}^2 \text{ et } (a_i, b_i) \in \mathbb{R}^2$$

- On pose alors $\forall I \in (\mathbb{IR})^k, w(I) = m(\mathfrak{X}(I))$

Cette définition ne s'applique pas directement à un intervalle I admettant des bornes infinies. L'extension proposée de la définition, à un tel intervalle, se base sur le remplacement de « ∞ » par une variable l dans l'expression de I , afin d'obtenir un intervalle $J(l) \in \mathbb{IR}^k$ tel que $\lim_{l \rightarrow +\infty} J(l) = I$.

On obtient alors $w(J(l)) = m(\mathfrak{X}(J(l)))$ et on peut ainsi poser $w(I) = \lim_{l \rightarrow +\infty} m(\mathfrak{X}(J(l)))$

2.3.2.3. Les fonctions d'inclusion d'intérêt

On substitue donc à $f : \mathbb{R}^m \supseteq I_f \rightarrow V \subseteq \mathbb{R}^n$, une fonction d'inclusion $[f]_{\Omega_A; \Omega_p}$ telle que :

- Ω_A recouvre I_f et est stable par l'intersection
- Ω_p recouvre $V \supseteq f(I_f)$ et est stable par l'intersection
- Tout point x de I_f est représenté par $\bigcap_{x \in J \in \Omega_A} J$
- Tout point y de V est représenté par $\bigcap_{y \in J \in \Omega_p} J$

La fonction $[f] = [f]_{\Omega_A; \Omega_p}$ utilisée, parmi celles qui suivent le modèle précédent reste toutefois libre.

On choisira usuellement $[f]$ minimale ou étayant f .

Toutefois, ce choix est sans impact sur la validité des algorithmes basés sur l'Analyse par Intervalles. Ce choix est principalement motivé par son influence, a priori bénéfique, sur les performances des algorithmes, qui sont optimales pour $\mathbb{I}f$, l'extension par intervalles de la fonction d'intérêt f .

Les ensembles Ω_A et Ω_p peuvent être théoriquement choisis distincts pour chaque fonction d'intérêt f .

Toutefois, sauf mention contraire, on utilise les ensembles d'intervalles tels que définis au § 2.3.2.1.

On notera tout de même que l'utilisation pratique d'intervalles plus généraux que des « boîtes » ou d'index d'entiers variables pourrait permettre de définir des algorithmes plus optimisés. Une telle « Analyse par Intervalles plus générale » est qualifiée d'interprétation abstraite.

Définition 2.3.2.1 Fonction d'inclusion naturelle

Soit $\mathcal{F}$, une famille de fonctions et $g : \mathcal{F} \ni g \mapsto g(g)$, une fonction d'inclusion de g « axiomatique ». Une fonction d'inclusion naturelle $[f]_{\mathcal{F},g}$ de f , selon $\mathcal{F}$ et g est définie par une famille d'applications

$$f_{\mathcal{F},g} = \{f_v \mid v \in \mathbb{N}^k \text{ où } k \in \mathbb{N}\}$$

La famille $f_{\mathcal{F},g}$ est définie de telle sorte que :

- $\forall k \in \mathbb{N}, \forall v \in \mathbb{N}^k, f_v \in \mathcal{F} \text{ ou } \text{Im}(f_v) = \emptyset$
- Soient :
 - $\pi : (v_j)_{j \in \llbracket 1; k \rrbracket} \mapsto (v_j)_{j \in \llbracket 1; k-1 \rrbracket}$
 - $\forall k \in \mathbb{N}, \forall v \in \mathbb{N}^k, f_{v_{\text{lv}}}$ est définie itérativement par :
 - $f_{v_{\text{lv}}} = f_v(u)$, où u est le vecteur des éléments, retournés par $\{f_w \mid v = \pi(w)\}$, ordonnés dans l'ordre croissant de la valeur de la dernière composante de w .
 - $\forall n \in \mathbb{N} - \{0\}, \forall k \in \mathbb{N}, \forall v \in \mathbb{N}^k, f_{v_{\text{lv}n}}$ est définie, par récurrence, par :
 - **Initialisation :**
 $f_{v_{\text{lv}1}} = f_v(u)$, où u est le vecteur des éléments retournés par $\{f_w \mid v = \pi(w)\}$, ordonnés dans l'ordre croissant de la valeur de la dernière composante de w .
 - **Récurrence :**
 $\forall n \in \mathbb{N} - \{0,1\}, f_{v_{\text{lv}n}} = f_v(u)$, où u est le vecteur des éléments retournés par $\{f_{w_{\text{lv}(n-1)}} \mid v = \pi(w)\}$, ordonnés dans l'ordre croissant de la valeur de w_{k+1} .

Alors :

- $f = f_{\emptyset_{\text{lv}}}$
- $\forall n \in \mathbb{N} - \{0\}, \forall k \in \mathbb{N}, \forall v \in \mathbb{N}^k, f_{v_{\text{lv}n}} \notin \mathcal{F}$

L'expression de la fonction d'inclusion naturelle $[f]_{\mathcal{F},g}$ s'obtient en remplaçant, $\forall k \in \mathbb{N}, \forall v \in \mathbb{N}^k$, dans l'expression de $f_{\emptyset_{\text{lv}}}$, la fonction f_v par sa fonction d'inclusion associée $g(f_v)$.

Remarque 2.3.2.1 (A propos de la Définition 2.3.2.1 de la fonction d'inclusion naturelle)

Le choix de proposer une définition générale et formellement explicite est motivé par :

- la possibilité d'étendre les raisonnements au cadre de l'interprétation abstraite.
- la perspective d'une expertise, a priori des performances de détection des points d'intérêt, grâce à la connaissance des propriétés des fonctions d'inclusion mises en œuvre.

La définition proposée s'interprète, de façon colloquiale, comme l'écriture de la fonction f sous forme d'une composition de fonctions g_k suivie d'un remplacement de chacune de ces fonctions par $[g_k]$.

La formulation proposée prend en compte la nécessité de considérer une composition « multiple ».

De plus, la structure mise en œuvre s'avère similaire à la structure que nous considérerons pour classifier les niveaux de singularités, dans la suite de ce mémoire.

Enfin, l'existence de $[f]_{\mathcal{F},g}$ justifie le concept de l'association automatique d'une fonction d'inclusion à chaque fonction d'intérêt f .

On notera que, par définition $\forall f \in \mathcal{F}, [f]_{\mathcal{F},g} = g(f)$.

De même, la définition proposée n'implique pas, en général, pour f quelconque, l'unicité de $[f]_{\mathcal{F},g}$.

De plus, l'existence de $[f]_{\mathcal{F},g}$ n'est pas assurée, pour une fonction f quelconque.

Toutefois, par soucis d'exhaustivité, nous remarquerons que l'existence de la décomposition $f_{\emptyset_{\text{lv}}}$ implique usuellement que $\exists m \in \mathbb{N}$ tel que $(\forall k \in \mathbb{N} - \llbracket 0; m \rrbracket, \exists! v \in \mathbb{N}^k \text{ tel que } \text{Im}(f_v) \neq \emptyset)$.

En pratique, si $\text{Id} \in \mathcal{F}$, on vérifie que $\exists m \in \mathbb{N}$ tel que $(\forall k \in \mathbb{N} - \llbracket 0; m \rrbracket, \exists! v \in \mathbb{N}^k \text{ tel que } f_v = \text{Id})$.

Dans nos applications pratiques, $\mathcal{F} = \{\text{Id}, x \mapsto a \in \mathbb{R}, +, \times, x \mapsto x^m \text{ où } m \in \mathbb{N}, \cos, \sin, \exp\}$.

L'application g associera alors à chaque fonction de $\mathcal{F}$ sa fonction d'inclusion naturelle « canonique », pour l'Analyse par Intervalles.

L'Analyse par Intervalles fournit un cadre pertinent pour le calcul numérique informatique.

En effet, l'utilisation de fonctions d'inclusions assure que les intervalles décrivant l'image, de tout ensemble, par une fonction d'intérêt, encadrent, de façon garantie, l'image formelle de cet ensemble.

2.3.3. Applications en robotique

D'après la sous-section 2.3.2, il apparaît naturel que l'Analyse par Intervalles puisse être utilisée pour beaucoup d'applications en robotiques, tel que présenté dans (Merlet, 2011).

Cette recherche numérique garantie présente, en effet les intérêts :

- de pouvoir s'appliquer à la résolution des problèmes géométriques directs et inverses (i.e. la recherche des positions et des orientations admissibles d'un robot pour une commande articulaire et, respectivement, la recherche des commandes articulaires admissibles d'un robot pour une position et une orientation de celui-ci).
- d'être applicable aux manipulateurs robotiques parallèles tout comme aux robots séries.
- de retourner un encadrement de toutes les solutions du problème géométrique considéré.

Citons aussi l'application qui nous concerne directement dans ce mémoire : *l'analyse des singularités*.

L'analyse des singularités consiste, en effet, à trouver les points singuliers de la fonction géométrique du manipulateur robotique étudié.

Pour isoler ces points singuliers, une approche générale est considérée et consiste à subdiviser et à réduire la taille des intervalles d'étude contenant potentiellement des points singuliers d'intérêt. L'aspect pratique de cette approche est, initialement, motivé par le fait que les points recherchés peuvent être définis comme étant des racines d'une équation (ou d'un système d'équation) s'écrivant « $f = 0$ ». Ainsi, tout intervalle I tel que $0 \notin f(I)$, ne contient aucun point recherché. Complémentairement, si un intervalle I peut contenir un point d'intérêt, un opérateur sur les intervalles peut être considéré afin de réduire celui-ci à une liste d'intervalles inclus dans ce dernier, contenant toutes les racines de f incluses dans l'intervalle initial I . Les intervalles issus de ces subdivisions sont alors traités à nouveau jusqu'à une condition d'arrêt.

Une illustration de ce processus est explicitée dans les articles (Bohigas, et al., 2012) et (Bohigas, et al., 2013), en application à l'encadrement des points singuliers de manipulateurs.

Ce processus est synergique avec l'Analyse par Intervalles de par sa propension à opérer sur des intervalles et de par son objectif d'encadrer des points d'intérêt.

On notera maintenant, comme le montre la Figure 4, que l'encadrement des lieux d'intérêt n'est pas toujours suffisant pour maîtriser la nature et les propriétés de ces lieux singuliers. En effet, certaines propriétés locales, telles que l'auto intersection, ne peuvent être détectées par un encadrement seul. *Afin de maîtriser au mieux la nature des points et des lieux d'intérêt, il apparaît donc nécessaire de caractériser et de détecter certaines propriétés locales des lieux formés par les points d'intérêt.*

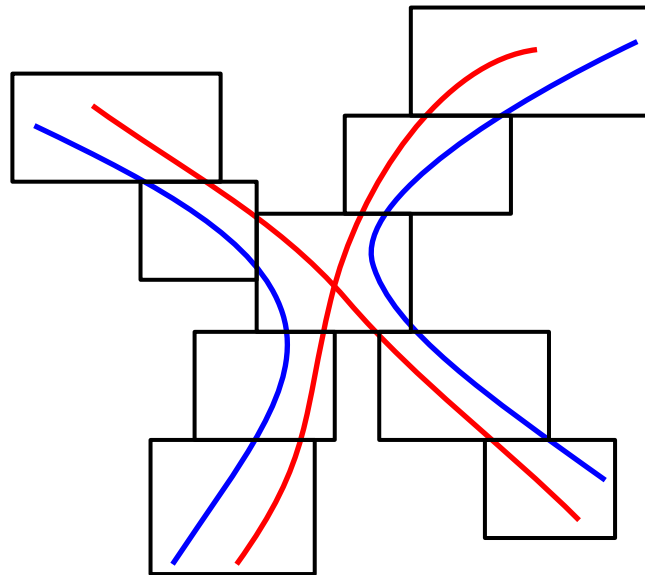


Figure 4 : Deux ensembles (en rouge et en bleu) compatibles avec un même encadrement (en noir)

Notre objectif dans ce mémoire sera donc, premièrement, de donner un sens aux propriétés locales que présentent usuellement les positions singulières des manipulateurs robotiques. Ceci nous conduira à caractériser, en cascade, des ensembles de points d'intérêt inclus les uns dans les autres.

Cette caractérisation est illustrée, dans ce mémoire par une détection numérique s'appuyant sur l'Analyse par Intervalles. L'étude de cette caractérisation sera l'objet du Chapitre 4 et trouve ses bases dans les concepts introduits dans le Chapitre 3.

Notons, toutefois, que la détection de ces points d'intérêt, de spécificités croissantes, s'appuie sur deux algorithmes associés à une stratégie d'encadrement absolu et à une stratégie de détection garantie.

Ces deux algorithmes sont « SIVIA » et « la méthode de Newton par Intervalles ». Respectivement, SIVIA fournit un encadrement des points d'intérêt et la méthode de Newton par Intervalles retourne un encadrement des points d'intérêt repérant des intervalles encadrant des points isolés parmi eux.

Le détail de ces algorithmes élémentaires, proposé au § 2.3.4, conclut cette section sur l'Analyse par Intervalles.

Notre objectif général est donc de savoir reconnaître des ensembles de points d'intérêt qui s'avèrent être caractéristiques des manipulateurs robotiques étudiés. Puis, nous nous proposons de classifier plusieurs fonctions en comparant les ensembles d'intérêt isolés pour chacune de celles-ci.

En pratique, l'utilisation de l'Analyse par Intervalles permet de s'assurer d'extraire un encadrement total des points d'intérêt recherchés. Cet encadrement est obtenu sans imprécision numérique et permet de détecter des points d'intérêt isolés.

2.3.4. Algorithmes élémentaires

2.3.4.1. (Set) Inversion Via Interval Analysis (SIVIA et IVIA)

La méthode SIVIA est un algorithme de recouvrement élémentaire de l'Analyse par Intervalles.

Tout comme l'indique son nom, cette méthode a pour finalité de caractériser la pré-image d'un ensemble, à l'aide de la théorie de l'Analyse par Intervalles.

Les paramètres en entrée d'une méthode SIVIA sont :

- Une fonction d'inclusion $[f]$ (assimilée à sa fonction naturelle d'intérêt f)
- Un ensemble V (usuellement un intervalle) dont on cherche la pré-image $f^{-1}(V)$
- Un ensemble E d'intervalles dans lequel on recherche $f^{-1}(V)$
- Une limite de taille ε

Par la suite, $\text{SIVIA}_{f,V,E,\varepsilon}$ désignera l'algorithme SIVIA admettant les paramètres précédents en entrée. (De cette manière, il sera plus simple de garder à l'esprit les paramètres définissant l'algorithme.)

Le principe du fonctionnement de l'algorithme $\text{SIVIA}_{f,V,E,\varepsilon}$ est de subdiviser les intervalles dans E de façon à obtenir un ensemble d'intervalles $E_{f,V,\varepsilon}$ vérifiant $\forall J \in E, \forall x \in J, \exists I \in E_{f,V,\varepsilon}$ tel que $x \in I$ et tel que $\forall I \in E_{f,V,\varepsilon}$, l'une des trois propriétés suivante est vérifiée :

- $f(I) \subseteq V$
- $f(I) \cap V = \emptyset$
- $w(I) \leq \varepsilon$

On a donc que $E_{f,V,\varepsilon} = E_{f,V,\varepsilon,\subseteq} \cup E_{f,V,\varepsilon,\emptyset} \cup E_{f,V,\varepsilon,?}$ où :

- $E_{f,V,\varepsilon,\subseteq} = \{I \in E_{f,V,\varepsilon} \mid f(I) \subseteq V\}$
- $E_{f,V,\varepsilon,\emptyset} = \{I \in E_{f,V,\varepsilon} \mid f(I) \cap V = \emptyset\}$
- $E_{f,V,\varepsilon,?} = \{I \in E_{f,V,\varepsilon} \mid I \notin (E_{f,V,\varepsilon,\subseteq} \cup E_{f,V,\varepsilon,\emptyset}) \subseteq \{I \in E_{f,V,\varepsilon} \mid w(I) \leq \varepsilon\}$

L'algorithme $\text{SIVIA}_{f,V,E,\varepsilon}$ retourne $E_{f,V,\varepsilon}$ sous forme de ses sous-ensembles $E_{f,V,\varepsilon,\subseteq}$, $E_{f,V,\varepsilon,?}$ et $E_{f,V,\varepsilon,\emptyset}$.

De ce fait la méthode $\text{SIVIA}_{f,V,E,\varepsilon}$ permet de déduire un encadrement de $f^{-1}(V)$ puisque :

$$\{x \in I \mid I \in E_{f,V,\varepsilon,\subseteq}\} \subseteq f^{-1}(V) \subseteq \{x \in I \mid I \in (E_{f,V,\varepsilon,\subseteq} \cup E_{f,V,\varepsilon,?})\}$$

L'algorithme IVIA : un algorithme SIVIA spécialisé

Dans ce mémoire nous utiliserons plus particulièrement l'algorithme que nous baptisons IVIA.

L'algorithme $\text{IVIA}_{f,E,\varepsilon}$ est une version remaniée de l'algorithme $\text{SIVIA}_{f,\{0\},E,\varepsilon}$.

Cet algorithme spécialisé $\text{IVIA}_{f,E,\varepsilon}$ cherche à optimiser la méthode $\text{SIVIA}_{f,\{0\},E,\varepsilon}$ en s'appuyant sur les particularités que présente l'inversion de l'ensemble $\{0\}$. L'algorithme $\text{SIVIA}_{f,\{0\},E,\varepsilon}$ retourne, en effet, usuellement et en pratique, $E_{f,\{0\},\varepsilon,\subseteq} = \emptyset$. De ce fait, on se propose de rechercher uniquement l'ensemble d'intervalles $E_{f,\varepsilon,\supseteq} = E_{f,\{0\},\varepsilon,\supseteq} = \{I \in E_{f,\varepsilon} \mid 0 \in f(I)\} = E_{f,\{0\},\varepsilon} - E_{f,\{0\},\varepsilon,\emptyset}$.

En conclusion, $\text{IVIA}_{f,E,\varepsilon}$ suit le même principe que $\text{SIVIA}_{f,V,E,\varepsilon}$ mais ne retourne qu'un seul ensemble d'intervalles $E_{f,\varepsilon,\supseteq} = E_{f,\{0\},\varepsilon,\subseteq} \cup E_{f,\{0\},\varepsilon,?}$ qui vérifie $(E \cap f^{-1}(\{0\})) \subseteq \{x \in I \mid I \in E_{f,\varepsilon,\supseteq}\}$.

De ce fait, nous présentons simultanément les algorithmes $\text{SIVIA}_{f,V,E,\varepsilon}$ et $\text{IVIA}_{f,E,\varepsilon}$ à la Figure 5, pour souligner leur similarité. En pratique les algorithmes utilisent des listes plutôt que des ensembles.

Remarque 2.3.4.1 (Compléments et implémentation des algorithmes)

1. On notera que, par construction, $\forall I \in E_{f,V,\varepsilon}, \exists J \in E$ tel que $I \subseteq J$.
2. En pratique, il n'est pas nécessaire de conserver et de retourner l'ensemble $E_{f,V,\varepsilon,\emptyset}$ si l'objectif de l'implémentation est seulement d'encadrer $f^{-1}(V)$. Cette omission de $E_{f,V,\varepsilon,\emptyset}$ permet notamment de limiter la mémoire nécessaire à l'exécution de l'algorithme en dimensions élevées.
3. Par extension, si R est le système $f = 0$, on pose $\text{IVIA}_{R,E,\varepsilon} = \text{IVIA}_{f,E,\varepsilon}$ et $E_{R,\varepsilon,\supseteq} = E_{f,\varepsilon,\supseteq}$.

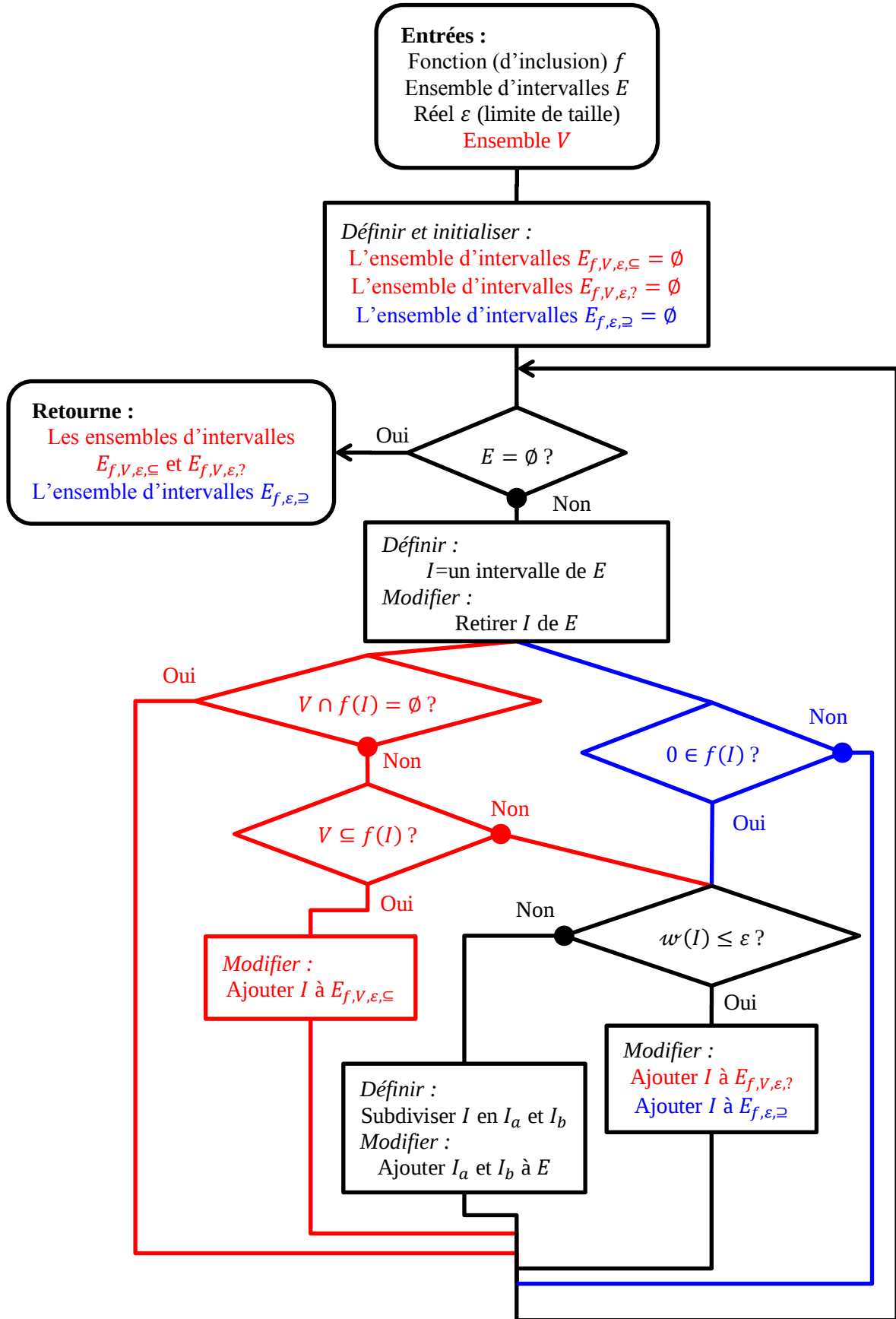


Figure 5 : Algorithmes SIVIA $_{f,V,E,\varepsilon}$ (rouge et noir) et IVIA $_{f,E,\varepsilon}$ (bleu et noir)

2.3.4.2. Méthode de Newton par Intervalles

Tout comme les algorithmes SIVIA et IVIA, l'objectif de la méthode de Newton par Intervalles, abrégée par « NpI », est de détecter et d'encadrer des ensembles d'intérêt.

La comparaison entre cette méthode et les algorithmes précédents se résume par les points suivants :

1. Tout comme IVIA, NpI est dédiée à la détection des racines d'une fonction d'intérêt f .
2. L'applicabilité de NpI sur un intervalle requiert que f soit différentiable sur celui-ci.
3. Sur certains intervalles, NpI peut garantir l'existence et l'unicité d'une racine d'intérêt isolée.

Plus particulièrement, la méthode NpI est basée sur l'étude d'un opérateur, sur les intervalles, N_f .

Soit un système carré d'équations décrit par $f = 0$.

L'opérateur de Newton par Intervalles N_f , associé à la fonction f est défini par :

$$N_f : I \mapsto \bar{x}(I) - (Df(I))^{-1} \times f(\bar{x}(I)) \text{ où } I \text{ est un intervalle et où } \bar{x}(I) \in I \quad (d)$$

Dans l'assertion (d) :

- $Df(I)$ est une matrice d'intervalles encadrant les matrices de $\{Df(z) \mid z \in I\}$, au sens où :
 $\forall z \in I, \quad \forall (i, j) \in \mathbb{N}^2, \quad (Df(z)_{i,j} \text{ existe} \Leftrightarrow Df(z)_{i,j} \text{ existe}) \text{ et } (Df(z)_{i,j} \in Df(I)_{i,j})$
- L'opérateur $\mathcal{M} \mapsto (\mathcal{M})^{-1}$ désigne l'opérateur d'inversion de matrice.
- L'application $\bar{x}$ est quelconque. Toutefois, deux définitions usuelles de $\bar{x}$ sont :
 - $\bar{x} : I \mapsto$ le milieu de I (i.e. un élément assimilable à un centre de gravité de I)
 - $\bar{x} : I \mapsto$ un élément au hasard de I

La méthode NpI est basée sur le fait que la relation topologique entre I et $N_f(I)$ est liée à la présence d'une racine de f dans I , selon les propriétés suivantes :

- Si $N_f(I) \subset I$ alors $(\exists! x \in I \text{ tel que } f(x) = 0)$.
- Si $N_f(I) \cap I = \emptyset$ alors $(\nexists x \in I \text{ tel que } f(x) = 0)$.
- Si $N_f(I) \cap I \neq \emptyset$ alors $(\nexists x \in I - (N_f(I) \cap I) \text{ tel que } f(x) = 0)$

L'algorithme de Newton par Intervalles, tel que défini dans (Neumaier, 1990) est donc construit à partir de ces trois propriétés de l'opérateur de Newton par Intervalles. Similairement aux méthodes d'encadrement SIVIA et IVIA, le principe de l'algorithme NpI est de subdiviser les intervalles d'un ensemble initial E d'intervalles de recherche de façon à extraire un ensemble d'intervalles $\mathfrak{R}_{E,f,\varepsilon}$ tel que $\forall I \in \mathfrak{R}_{E,f,\varepsilon}$, l'une des deux propriétés suivante est vérifiée :

- $0 \in f(I)$
- $w(I) \leq \varepsilon$

De plus, $\exists \mathfrak{R}_{E,f,\varepsilon} \subseteq \mathfrak{R}_{E,f,\varepsilon}$ tel que $(\forall I \in \mathfrak{R}_{E,f,\varepsilon}, \exists! x \in I \text{ tel que } f(x) = 0)$

Tout comme pour l'algorithme IVIA_{f,E,ε}, chaque algorithme de Newton par Intervalles sera désigné par NpI_{f,E,ε} où f, E et ε sont les trois paramètres le définissant.

Le diagramme d'une implémentation de l'algorithme NpI_{f,E,ε} est donné à la Figure 6.

Remarque 2.3.4.2 (Compléments et implémentation de l'algorithme)

1. Relativement à l'assertion (d) :
 - a. Lorsque nous implémentons la méthode NpI_{f,E,ε} l'opération $\mathcal{M} \mapsto (\mathcal{M})^{-1}$ consiste, en pratique, à appliquer la formule de l'inversion de matrice à la matrice $\mathcal{M}$.
 - b. Par souci d'exhaustivité, notons qu'il est possible de remplacer $(Df(I))^{-1} \times f(\bar{x}(I))$ par tout ensemble $\Xi(I, f, \bar{x}(I))$ contenant $\{w \mid \exists \mathcal{M} \in Df(I) \text{ tel que } \mathcal{M}w = f(\bar{x}(I))\}$.
L'expression de $\Xi(I, f, \bar{x}(I))$ peut être optimisée selon la fonction d'intérêt f étudiée.
2. La méthode NpI_{f,E,ε} proposée peut ne pas détecter une racine z de f si :
 - a. $Df(z)$ n'est pas inversible
 - b. ε est trop grand pour empêcher $(\exists (z, y) \in I^2, f(z) = f(y) = 0 \text{ et } w(I) \leq \varepsilon)$
3. NpI_{f,E,ε} peut seulement garantir l'existence de racines ponctuelles isolées de f

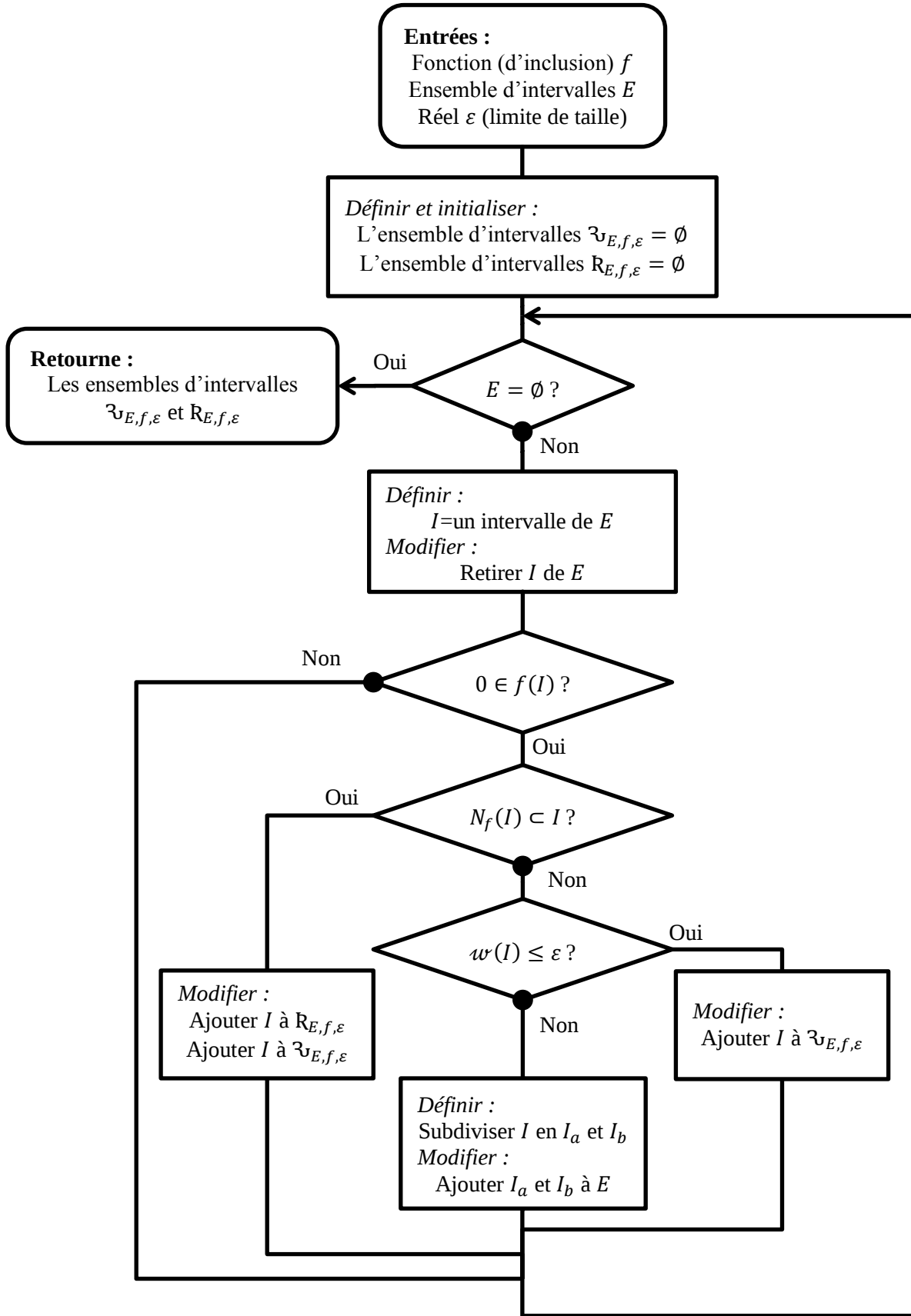


Figure 6 : Un algorithme $\text{NpI}_{f,E,\varepsilon}$

2.4 Introduction à la problématique de classification

La classification d'objets est une problématique générale, dont le concept de base est simple, mais dont l'exécution peut être complexe.

Classifier des objets consiste à, simultanément :

- Regrouper les objets compatibles selon une condition
- Séparer les objets incompatibles selon la même condition.

La condition considérée est qualifiée de relation d'équivalence pour la classification réalisée.

Bien que les deux objectifs de toute classification soient complémentaires, on note, en pratique, qu'il est, en général, impossible de complètement réaliser chacun d'eux.

La problématique de la classification implique ainsi la constitution de tests performants de compatibilité et d'incompatibilité avec la relation d'équivalence considérée.

Les notions présentées dans ce Chapitre 2 constituent les fondations théoriques qui, dans la suite de ce mémoire, permettent de définir certains tests de compatibilité avec une relation d'équivalence précise que nous introduisons par la suite.

Le Chapitre 3 constitue ainsi un chapitre préliminaire aux méthodologies proposées, dans ce mémoire, pour répondre à la problématique de classification considérée.

En effet, nous y précisons la condition d'équivalence $\sim$ qui est la condition d'équivalence associée à la classification considérée dans tout ce document, ainsi que le concept général des tests envisagés.

La complémentarité des concepts développés dans ce Chapitre 2, avec la problématique précisée au Chapitre 3, constitue finalement le cœur de la motivation de nos travaux, présentés dans ce document.

Chapitre 3 Préliminaires

3.1 Equivalence et invariant : une introduction

Pour expliquer la trame de ce chapitre avec des mots simples :

L'objectif principal de cette thèse étant de « mettre dans des cases » les comportements des robots, par l'intermédiaire des fonctions géométriques qui les caractérisent, il est nécessaire de bien définir la propriété « avoir le même comportement ».

On qualifiera d'équivalentes deux fonctions qui correspondent au même comportement.

Ensuite, il est naturellement nécessaire de tester l'équivalence.

Pour cela, on construira, pour chaque fonction considérée, une famille d'objets, des invariants, tels que d'après leurs méthodes de construction, si ceux-ci sont différents pour deux fonctions, alors ces deux fonctions ne peuvent être équivalentes.

Les motivations de ce chapitre sont donc, à la fois de :

1. Définir correctement et formellement les concepts présentés d'équivalence et d'invariant
2. Expliciter les types d'invariants que nous considérons ainsi que leurs constructions
3. Définir les deux processus à mettre en pratique pour la classification proposée

3.1.1. Concepts formels d'Equivalence et d'Invariants

A partir de maintenant, sauf mention contraire, on considèrera que tout espace est topologique et muni, par défaut, de sa topologie usuelle (si une telle topologie est définie).

Tout d'abord, rappelons la définition de deux termes :

Définition 3.1.1.1 (Homéomorphisme)

Application continue inversible et d'application inverse continue.

Définition 3.1.1.2 (Difféomorphisme)

Homéomorphisme h , tel que $\forall H \in \{h, h^{-1}\}$, H est différentiable et sa différentielle, en tout point, est un homéomorphisme (*vis-à-vis des espaces topologiques entre lesquels agit chaque différentielle*).

En utilisant la Définition 3.1.1.1 et la Définition 3.1.1.2, la définition formelle de l'équivalence que nous considérons est explicitée par la Définition 3.1.1.3 des applications équivalentes, ci-dessous.

Définition 3.1.1.3 (Applications équivalentes)

Deux applications $f : X \rightarrow Y$ et $g : X' \rightarrow Y'$ (entre variétés) sont équivalentes, si il existe deux difféomorphismes $h : X \rightarrow X'$ et $k : Y \rightarrow Y'$ tels que le diagramme de la Figure 7 commute. Autrement dit, si il existe deux difféomorphismes $h : X \rightarrow X'$ et $k : Y \rightarrow Y'$ tels que $k \circ f = g \circ h$.

On notera alors $f \sim g$

Si il n'existe pas un tel couple de difféomorphismes, on notera alors $f \not\sim g$

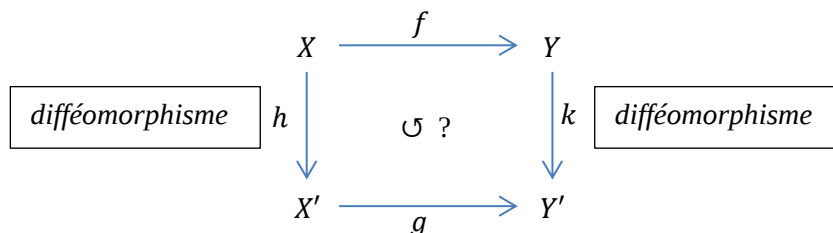


Figure 7 : Lorsque ce diagramme commute f et g sont équivalentes

Remarque 3.1.1.1 (« Equivalence » $\sim$ et relation d'équivalence)

$\sim$ est trivialement une relation d'équivalence (i.e. une relation réflexive, symétrique et transitive).
Par la suite nous utiliserons aussi une « équivalence faible » entre application : *la conjugaison*.

Définition 3.1.1.4 (Applications conjuguées ou « topologiquement équivalentes »)

Deux applications $f : X \rightarrow Y$ et $g : X' \rightarrow Y'$ (entre variétés) sont conjuguées, si il existe deux homéomorphismes $h : X \rightarrow X'$ et $k : Y \rightarrow Y'$ tels que le diagramme de la Figure 8 commute.

On notera alors $f \sim_0 g$

Dans le cas contraire, on notera $f \not\sim_0 g$

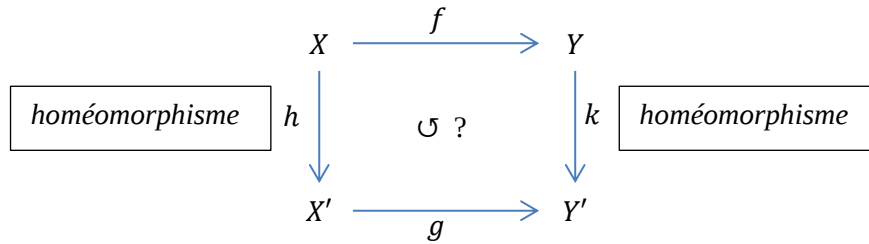


Figure 8 : Lorsque ce diagramme commute f et g sont conjuguées

Remarque 3.1.1.2 (Equivalence et Conjugaison)

On notera que $(f \sim g \Rightarrow f \sim_0 g)$ ce qui induit $(f \not\sim_0 g \Rightarrow f \not\sim g)$

De ce fait, par la suite on se propose de construire un test capable de détecter $\sim_0$, dans certains cas.

Puis, de par cette remarque, on sera alors capable de détecter $\sim$, dans ces mêmes cas.

Afin de construire un test de détection de $\sim_0$ (et par extention de $\sim$) nous construirons des *invariants*.

Définition 3.1.1.5 (Invariant)

Un invariant, pour une relation d'équivalence entre objets, est une construction (algorithmique) qui fournit le même résultat pour deux objets équivalents pour la relation considérée.

Un exemple trivial :

Le nombre de lettres d'un mot est un invariant pour la relation d'équivalence « être des anagrammes ».

Un invariant induit une *condition nécessaire* pour que deux objets vérifient une relation d'équivalence. L'objectif consiste toujours à définir l'invariant le plus précis possible afin de se rapprocher, au plus, de la situation idéale où le partage de l'invariant, constituerait une *condition suffisante* d'équivalence.

La question qui reste en suspens est maintenant :

Quels invariants considérer et construire afin d'obtenir un test de détection « efficace » (i.e. le plus discriminant que nous puissions envisager) de la non équivalence entre deux fonctions ?

3.1.2. Propriété fondamentale de l'équivalence et introduction à la définition d'un invariant pour des fonctions de $\mathcal{D} \subseteq \mathbb{R}$ dans $\mathbb{R}$

Nous proposons, pour bien comprendre les propriétés de la notion d'équivalence présentée, ainsi que la construction d'invariants pour cette équivalence, de parcourir ces propriétés immédiates, en les illustrant par l'application du concept d'équivalence aux fonctions de $\mathcal{D} \subseteq \mathbb{R}$ dans $\mathbb{R}$, qui constituent une famille d'applications relativement simples à caractériser, étant donné le fait que les espaces sources et buts sont unidimensionnels.

Soit $f : \mathcal{D}_f \rightarrow \mathbb{R}$ et $g : \mathcal{D}_g \rightarrow \mathbb{R}$, où $(\mathcal{D}_f, \mathcal{D}_g) \subseteq \mathbb{R}^2$
 $f \sim g$ devient $(\exists ((h : \mathcal{D}_f \rightarrow \mathcal{D}_g), (k : \mathbb{R} \rightarrow \mathbb{R})), \text{ des difféomorphismes tels que } g \circ h = k \circ f)$

On vérifie, ici, que $f \sim g$ implique, naturellement, que $f(\mathcal{D}_f)$ et $g(\mathcal{D}_g)$ doivent être difféomorphes. Plus généralement **la propriété d'équivalence s'étend aux sous-parties des espaces sources et buts**. En d'autres termes :

(Dans l'espace source) : Si $E \subseteq \mathcal{D}_f$ et $f \sim g$ Alors $\exists E' \subseteq \mathcal{D}_g$ tel que :

- E et E' sont difféomorphes
- $f(E)$ et $g(E')$ sont difféomorphes

(Dans l'espace but) : Si $E \subseteq \mathbb{R}$ et $f \sim g$ Alors $\exists E' \subseteq \mathbb{R}$ tel que :

- E et E' sont difféomorphes
- $f^{-1}(E)$ et $g^{-1}(E')$ sont difféomorphes

Cette propriété d'extension aux sous-parties s'appuie sur la continuité des difféomorphismes.

La construction d'invariants devrait donc permettre de mettre en lumière l'absence de cette propriété. Ceci nous amène à concevoir un invariant qui collecterait les propriétés topologiques de certaines sous-parties des espaces sources et buts, qui seraient, en quelque sorte, indissociables de la fonction.

3.2 Détection de lieu caractéristique et construction d'invariant

Comme nous avons pu l'entrevoir, il est nécessaire de bien définir certains lieux caractéristiques qui délimiteraient des zones, elles aussi, caractéristiques du comportement de chaque fonction considérée.

Afin que la compréhension de ce concept de zone caractéristique et que son lien avec la classification de fonctions soit le plus clair possible, nous allons continuer de développer formellement ces concepts, tout en les illustrant simultanément à l'aide du cas particulier des fonctions de $\mathcal{D} \subseteq \mathbb{R}$ dans $\mathbb{R}$.

Nous allons tout d'abord définir des sous-ensembles caractéristiques de l'espace source.

Puis nous démontrerons que ceux-ci sont « topologiquement conservés » par les difféomorphismes.

Ce qui nous amène alors à :

1. Définir des méthodes pour détecter ces lieux caractéristiques.
2. Mettre en place des procédures pour construire un invariant riche à partir des lieux détectés.

Ces deux problématiques définissent le cœur respectif de chacun des deux chapitres suivants.

3.2.1. Lieu singulier caractéristique et Classes de singularités

Les classes de singularités $\mathcal{S}_\nu$ regroupent les points de même niveau et de même type de singularité. Nous reformulons ici leur définition, d'après celle énoncée dans (Golubitsky, et al., 1986) :

Définition 3.2.1.1 (Fonction restreinte à un sous espace)

Soit $f : X \rightarrow Y$, une application entre espaces topologiques et E un sous espace topologique de X .

On note $f|_E$ la fonction f restreinte à E , caractérisée par : $f|_E : E \rightarrow Y$ et $\forall M \in E, f|_E(M) = f(M)$.

Définition 3.2.1.2 (Classes de singularités $\mathcal{S}_v$)

Soit $v \in \mathbb{N}^k$, $k \in \mathbb{N}$ et $f : X \rightarrow Y$, une application entre espaces topologiques.
Chaque classe de singularités $\mathcal{S}_v$ est construite par récurrence selon les éléments de v .
Cette construction s'initialise par $\mathcal{S}_\emptyset(f) = X$

Soit $v = (v_j)_{j \in \llbracket 1; k \rrbracket}$ et en notant $\pi(v) = (v_j)_{j \in \llbracket 1; k-1 \rrbracket}$ on définit $\mathcal{S}_v$ comme étant :

$$\left\{ x \in \mathcal{S}_{\pi(v)} \mid \text{le rang de } f|_{\mathcal{S}_{\pi(v)}} \text{ chute de } v_k \right\}$$

Notons toutefois que cette définition simple fait appel à beaucoup d'implicites :

- Le rang (local) d'une application g , en un point x , est le rang de Dg_x .
- La chute de rang est considérée vis-à-vis du rang maximal, que peut avoir une application entre les variétés considérées.

Nous proposons donc une seconde formulation « duale » plus explicite, pour un même vecteur v .

La définition de $\mathcal{S}_v$ peut donc être aussi formulée de la façon suivante :

$x \in \mathcal{S}_v$ si et seulement si $x \in \mathcal{S}_{\pi(v)}$ et v_k est l'entier m maximum tel que :

$$\exists (u_i)_{i \in \llbracket 1; m \rrbracket} \in (\text{Ker}(Df_x) \cap \mathcal{T}_x \mathcal{S}_{\pi(v)})^m \text{ tel que } \left(\left(\sum_{i \in \llbracket 1; m \rrbracket} \lambda_i u_i \right) = 0 \Rightarrow \forall i \in \llbracket 1; m \rrbracket, \lambda_i = 0 \right)$$

(ou, en français, tel que il existe un m -uplet de vecteurs de $\text{Ker}(Df_x)$, non linéairement dépendants qui sont aussi éléments de l'espace tangent, en x , à $\mathcal{S}_{\pi(v)}$)

Pour la clarté de la lecture, on omet les parenthèses autour des éléments de v dans l'écriture de $\mathcal{S}_v$, lorsque l'on écrit v sous forme du vecteur explicite de tous ses éléments.

Remarque 3.2.1.1 (Propriétés de $\mathcal{S}_v$)

Posons à nouveau $v = (v_j)_{j \in \llbracket 1; k \rrbracket}$ et $f : X \rightarrow Y$ Alors $\mathcal{S}_v \neq \emptyset$ implique :

- Si $\dim(X) = m$ alors $\forall j \in \llbracket 1; k \rrbracket, v_j \leq m$
- $\forall j \in \llbracket 1; k \rrbracket, v_j \leq v_1$
- Si $v_i = 0$ alors $\forall j \geq i, v_j = 0$
- Plus généralement, $(v_j)_{j \in \llbracket 1; k \rrbracket}$ est une suite (tronquée) décroissante (i.e. toutes ses sous-suites ne sont pas strictement croissantes)

La preuve des trois premières propriétés est évidente.

La quatrième et dernière propriété peut paraître moins évidente vis-à-vis des formulations que nous avons choisies pour la définition de $\mathcal{S}_v$. Toutefois, bien que nous n'en aurons pas l'usage, en pratique, dans ce mémoire, cette preuve, énoncée dans (Arnold, et al., 1985), est relativement simple et repose sur l'observation que $\mathcal{S}_v \subseteq \mathcal{S}_{\pi(v)}$, ce qui implique que les noyaux des différentielles des restrictions de f à ces espaces respectifs sont aussi inclus l'un dans l'autre.

En remarquant que $\text{Ker} \left(D \left(f|_{\mathcal{S}_{\pi(v)}} \right)_x \right) = \text{Ker}(Df_x) \cap \mathcal{T}_x \mathcal{S}_{\pi(v)}$, on conclut alors que $\mathcal{S}_v$ est non vide, si et seulement si les éléments du vecteur v , des dimensions successives de ces classes incluses, successivement, les unes dans les autres, sont décroissants (non forcément strictement).

Remarque 3.2.1.2 (Classes de singularités pour $f : \mathfrak{D}_f \rightarrow \mathbb{R}$ où $\mathfrak{D}_f \subseteq \mathbb{R}$)

On rappelle que l'on a initialement $\mathcal{S}_\emptyset(f) = \mathfrak{D}_f$

De plus, Df_x s'identifie à $f'(x)$, la dérivée de f en x et son rang est alors la dimension de $f'(x)\mathbb{R}$.

On a alors $\forall m \in \mathbb{N}$:

$$\mathcal{S}_j(f) = \left\{ x \in \mathfrak{D}_f \mid j = \max_{m \in \mathbb{N}} \left(\exists (u_i)_{i \in \llbracket 1; m \rrbracket} \in \left(\text{Ker}(f'(x)) \right)^m \mid (u_i)_{i \in \llbracket 1; m \rrbracket} \text{ linéairement indépendants} \right) \right\}$$

où $\max_m(\mathcal{P}(m))$ est l'élément m maximal tel que la propriété $\mathcal{P}(m)$ est vraie

On déduit alors que :

- $\forall j \geq 2, \mathcal{S}_j(f) = \emptyset$
- $\mathcal{S}_1(f) = \{x \in \mathfrak{D}_f \mid f'(x) = 0\}$
- $\mathcal{S}_0(f) = \{x \in \mathfrak{D}_f \mid f'(x) \neq 0\}$

Proposition 3.2.1.1 ($\mathcal{S}_v$ est compatible avec l'équivalence)

Soit $f : X \rightarrow Y$ et $g : X' \rightarrow Y'$, deux applications équivalentes.

Alors $\forall k \in \mathbb{N}, \forall v \in \mathbb{N}^k, \mathcal{S}_v(f)$ et $\mathcal{S}_v(g)$ sont difféomorphes.

Preuve 3.2.1.1 (de la Proposition 3.2.1.1)

Soit $f : X \rightarrow Y$ et $g : X' \rightarrow Y'$

Si $f \sim g$ alors il existe deux difféomorphismes $h : X \rightarrow X'$ et $k : Y \rightarrow Y'$ tels que : $k \circ f = g \circ h$

Nous allons prouver la Proposition 3.2.1.1 par récurrence, pour pouvoir nous appuyer sur la construction de $\mathcal{S}_v$, qui a été définie par récurrence.

De plus, on notera que $f \sim g \Rightarrow df_x \sim_0 dg_{h(x)}$, par les homéomorphismes dh_x et $dk_{f(x)}$

Initialisation :

Comme nous l'avons déjà fait remarqué, on a que $\mathcal{S}_\emptyset(f) = X$ et $\mathcal{S}_\emptyset(g) = X'$ sont difféomorphes.

Récurrence :

Supposons que $\mathcal{S}_{\mathfrak{A}(v)}(f)$ est difféomorphe à $\mathcal{S}_{\mathfrak{A}(v)}(g)$, par h , et que $u \in \text{Ker}(Df_x) \cap \mathfrak{T}_x \mathcal{S}_{\mathfrak{A}(v)}$

Par continuité, on a $dh_x(u) \in dh_x(\text{Ker}(Df_x) \cap \mathfrak{T}_x \mathcal{S}_{\mathfrak{A}(v)}(f)) = dh_x(\text{Ker}(Df_x)) \cap dh_x(\mathfrak{T}_x \mathcal{S}_{\mathfrak{A}(v)}(f))$.

Par hypothèse de récurrence, on a aussi $dh_x(\mathfrak{T}_x \mathcal{S}_{\mathfrak{A}(v)}(f)) = \mathfrak{T}_{h(x)} \mathcal{S}_{\mathfrak{A}(v)}(g)$

De plus, $\text{Ker}(Df_x) = \{w \in \mathfrak{T}_x X \mid Df_x(w) = 0\}$, d'où $dh_x(\text{Ker}(Df_x)) = \{dh_x w \in \mathfrak{T}_x X \mid Df_x(w) = 0\}$,

soit encore $dh_x(\text{Ker}(Df_x)) = \{z \in \mathfrak{T}_{h(x)} X \mid Df_x(Dh_x^{-1}(z)) = 0\}$

On utilise maintenant les propriétés du difféomorphisme k , que nous n'avons pas oubliées.

D'où $dh_x(\text{Ker}(Df_x)) = \{z \in \mathfrak{T}_{h(x)} X \mid Dk_{g(h(x))}^{-1}(Dg_{h(x)}(z)) = 0\} = \{z \in \mathfrak{T}_{h(x)} X \mid Dg_{h(x)}(z) = 0\}$

Ce qui nous permet de conclure car, Dh_x étant inversible, elle conserve l'indépendance linéaire.

Proposition 3.2.1.2 (Compatibilité des images et contre images)

L'image de $\mathcal{S}_v$ et les contre images de cette image sont compatibles avec l'équivalence.

Soit $f : X \rightarrow Y$ et $g : X' \rightarrow Y'$, deux applications équivalentes.

Alors $\forall k \in \mathbb{N}, \forall v \in \mathbb{N}^k$:

$f(\mathcal{S}_v(f))$ et $g(\mathcal{S}_v(g))$ sont difféomorphes

$f^{-1}(f(\mathcal{S}_v(f)))$ et $g^{-1}(g(\mathcal{S}_v(g)))$ sont difféomorphes

Preuve 3.2.1.2 (de la Proposition 3.2.1.1)

A partir des résultats fondamentaux de la sous-section 3.1.2 et de la Proposition 3.2.1.1, la démonstration de la Proposition 3.2.1.2 est triviale.

3.2.2. Invariant hybride entre équivalence et conjugaison

Nous avons vérifié que les classes $\mathcal{S}_v(f)$, ainsi que les sous-ensembles qui leur sont associés par f , sont difféomorphes aux classes $\mathcal{S}_v(g)$ et aux sous-ensembles associés par g , si $f \sim g$.

Nous nous intéressons donc à la création d'un invariant qui découle de ces difféomorphismes.

Définition 3.2.2.1 (Invariant riche « $\mathcal{K}_1$ »)

Soit $f : X \rightarrow Y$, $\mathcal{K}_1(f)$ contient l'information sur la topologie et la topologie différentielle de chaque sous-ensemble qui peut être délimité dans X et Y à partir de $\{\mathcal{S}_v(f) \mid v \text{ est un vecteur d'entiers}\}$, par l'intermédiaire d'itérations de f et de f^{-1} (i.e. $\{f(\mathcal{S}_v(f)) \mid v \text{ est un vecteur d'entiers}\}$ et $\{f^{-1}(f(\mathcal{S}_v(f))) \mid v \text{ est un vecteur d'entiers}\}$ ainsi que toute (pré-)image de ces ensembles par f (respectivement f^{-1}) et ainsi de suite) ainsi que sur le lien entre ces sous-ensembles par f .

$\mathcal{K}_1$ est bel et bien un invariant riche pour la relation d'équivalence $\sim$

Toutefois, il est difficile de stocker l'intégralité de l'information différentielle en tout point.

De ce fait, nous proposons de nous affranchir de cette information et de conserver uniquement l'information topologique et de lien par la fonction.

Définition 3.2.2.2 (Invariant simplifié « $\mathcal{K}_0$ » utilisé)

Soit $f : X \rightarrow Y$, $\mathcal{K}_0(f)$ contient l'information sur la topologie de chaque sous-ensemble qui peut être délimité dans X et Y à partir de $\mathcal{S}_v(f)$, par l'intermédiaire d'itérations de f et de f^{-1} , ainsi que sur le lien entre ces sous-ensembles par f .

Bien que $\mathcal{K}_0$ ne conserve pas l'information sur la différentielle en chaque point, il n'est pas réduit à un invariant uniquement basé sur $\sim_0$ car l'appartenance à $\mathcal{S}_v$ est uniquement conservée si les fonctions sont équivalentes.

De ce fait, $\mathcal{K}_0$ est un invariant hybride caractérisant $\sim$, même s'il est déterminé au travers de la vérification de $\sim_0$, pour des sous-ensembles bien choisis, qui ne sont conservés que par la relation $\sim$.

Remarque 3.2.2.1 (Fil conducteur des chapitres suivants)

On notera que les chapitres suivants s'intéressent donc aux 2 aspects nécessaires pour la construction d'un invariant $\mathcal{K}_0$:

1. Caractériser et détecter (algorithmiquement) $\mathcal{S}_v(f)$, $f(\mathcal{S}_v(f))$ et $f^{-1}(f(\mathcal{S}_v(f)))$
2. Définir algorithmiquement un objet précis qui puisse être considéré comme l'invariant $\mathcal{K}_0$.

3.2.3. Cheminement intuitif vers $\mathcal{K}_0$, pour $f : \mathcal{D}_f \rightarrow \mathbb{R}$ où $\mathcal{D}_f \subseteq \mathbb{R}$

3.2.3.1. La base familière du tableau de variation

Considérons les cas simples où $\mathcal{D}_f = [a, b]$, un intervalle fermé de $\mathbb{R}$, ou $\mathcal{D}_f = \mathbb{R}$.

A titre d'illustration nous considérons $f : \mathcal{D}_f \rightarrow \mathbb{R}$ définie par $\forall x \in \mathbb{R}, f(x) = x^2 - x$.

On considèrera, pour cet exemple que $[a, b] = [-2; 2]$.

On peut naturellement construire le tableau simple des variations de $f: \mathfrak{D}_f \ni x \mapsto x^2 - x$.
On présente ainsi, dans le Tableau 1, les variations de f , lorsque $\mathfrak{D}_f = [-2; 2]$ et lorsque $\mathfrak{D}_f = \mathbb{R}$.

x	$-\infty$	-2	$\frac{1}{2}$	2	$+\infty$	
$f': x \mapsto 2x - 1$		$-$	$-$	0	$+$	$+$
$f: x \mapsto x^2 - x$	$+\infty$					$+\infty$
A variation table diagram for the function $f(x) = x^2 - x$. The diagram shows a path starting from $+\infty$ at $x = -\infty$, decreasing to a local minimum at $x = \frac{1}{2}$ (with $y = -\frac{1}{4}$), increasing to a local maximum at $x = 2$ (with $y = 2$), and then decreasing to $+\infty$ at $x = +\infty$. The points -2 and 6 are also marked on the path.						

Tableau 1 : Tableau de variation de la fonction $f: x \mapsto x^2 - x$, pour $\mathfrak{D}_f = [-2; 2]$ et $\mathfrak{D}_f = \mathbb{R}$

En se référant au Tableau 1, on vérifie que $\mathcal{S}_1(f) = \{\frac{1}{2}\}$, d'où $\mathcal{S}_0(f) = \mathfrak{D}_f - \{\frac{1}{2}\}$ et $\forall j > 1, \mathcal{S}_j(f) = \emptyset$.
De même, on a $\forall j > 0, \mathcal{S}_{1,j}(f) = \emptyset$, d'où $\mathcal{S}_1(f) = \mathcal{S}_{1,0}(f)$.

Lorsque $\mathfrak{D}_f = \mathbb{R}$, on a :

- $\mathcal{S}_0(f) =]-\infty; \frac{1}{2}[\cup]\frac{1}{2}; +\infty[$
- $\mathcal{S}_1(f) = \{\frac{1}{2}\}$ et $f(\mathcal{S}_1(f)) = \{-\frac{1}{4}\}$
- $f(]-\infty; \frac{1}{2}[) = f(]\frac{1}{2}; +\infty[) =]-\frac{1}{4}; +\infty[$

Ces propriétés définissent alors entièrement $\mathcal{K}_0(f)$, par définition de cet invariant, lorsque l'on remplace les intervalles considérés par leur topologies.

Si, par convention, chaque lettre désigne un élément de $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$ (tel que les intervalles soient non vides) et que l'on considère les topologies au lieu des ensembles. Alors $\mathcal{K}_0(f)$ s'écrit :

- $\mathcal{S}_0(f) =]a, p[\cup]p, b[$
- $\mathcal{S}_1(f) = \{p\}$ et $f(\mathcal{S}_1(f)) = f(p) = \alpha$
- $f(]a, p[) = f(]p, b[) =]\alpha, \beta[$

Toutefois, lorsque $\mathfrak{D}_f$ est restreint à un intervalle $[a, b]$ tel que $f(a) \neq f(b)$ (par exemple $[-2; 2]$ pour $f: x \mapsto x^2 - x$) la description complète de $\mathcal{K}_0(f)$ requiert de subdiviser plus précisément $\mathcal{S}_0(f)$, afin que $\mathcal{K}_0(f)$ contienne toujours toute l'information du « lien par f » après itérations de f et f^{-1} :

- $\mathcal{S}_0(f) = [a, c[\cup]c, p[\cup]p, b]$
- $\mathcal{S}_1(f) = \{p\}$ et $f(\mathcal{S}_1(f)) = f(p) = \alpha$
- $f([c, p[) = f(]p, b]) =]\alpha, \gamma]$ et $f([a, c[) =]\gamma, \beta]$

On notera que, dans cette description de $\mathcal{K}_0(f)$, il apparait possible de retrouver la nature de chaque élément repère, en s'appuyant sur les liens entre les intervalles où ceux-ci apparaissent *excepté si* $\delta\mathfrak{D}_f \cap (\bigcup_{v \in \mathbb{N}^k, k \in \mathbb{N}} \mathcal{S}_v(f)) \neq \emptyset$ où δE désigne le bord de l'espace E .

Nous souhaiterions donc maintenant disposer d'un objet bien défini qui coderait complètement $\mathcal{K}_0$.
Pour cela nous nous orientons vers les objets combinatoires et les graphes.

3.2.3.2. Visualisation de $\mathcal{K}_0$: progression vers la notion de « portrait »

Coder l'information portée par $\mathcal{K}_0$

Nous proposons, dans cette sous-section, de construire un objet combinatoire simple, le *portrait de f* , portant toutefois toute l'information de l'invariant $\mathcal{K}_0(f)$, lorsque $f : \mathcal{D}_f \rightarrow \mathbb{R}$ où $\mathcal{D}_f \subseteq \mathbb{R}$

Nous allons construire progressivement ce *portrait*, à partir de l'isolation des classes $\mathcal{S}_v(f)$, puis en affinant celles-ci, en considérant les actions successives de f et de f^{-1} sur elles.

De plus, nous proposons de réaliser cette construction, en parallèle de l'affinement du tableau de variation de f , afin qu'il soit possible de construire le portrait d'une fonction quelconque, à partir d'un tableau de variation assez détaillé de celle-ci.

L'esquisse ou la caractérisation de l'action de f

L'esquisse est l'étape de base de tout portrait, ainsi nous désignerons par *l'esquisse de f* , un objet combinatoire contenant l'information sur les classes $\mathcal{S}_v(f)$ ainsi que sur l'action de f sur elles.

Afin de construire l'esquisse, il est uniquement nécessaire de détecter $\mathcal{S}_1(f) = \{x \in \mathcal{D}_f \mid f'(x) = 0\}$, car $\mathcal{S}_0(f) = \mathcal{D}_f - \mathcal{S}_1(f)$. De par la nature unidimensionnelle de $\mathcal{D}_f \subseteq \mathbb{R}$, pour une fonction continue f , les composantes connexes de $\mathcal{S}_0(f)$ sont des intervalles (ouverts, semi-fermés ou fermés) tels que leurs bords sont des éléments de $\delta\mathcal{D}_f \cup \mathcal{S}_1(f)$. On peut donc se restreindre à caractériser l'action de f sur $\delta\mathcal{D}_f \cup \mathcal{S}_1(f)$ pour caractériser son action sur l'intégralité de $\mathcal{D}_f$, tout comme on le représente dans tout tableau de variation simple d'une fonction (dérivable) f .

En reprenant l'exemple de $f : x \mapsto x^2 - x$, on construit alors :

x	$-\infty$	$\frac{1}{2}$	$+\infty$
$f': x \mapsto 2x - 1$	$-$	0	$+$
$f: x \mapsto x^2 - x$	$+\infty$	$-\frac{1}{4}$	$+\infty$

Tableau 2 : Tableau de variation simple de $f : x \mapsto x^2 - x$

La structure topologique induite par le Tableau 2 correspond donc à l'information du « lien par f » connue après s'être restreint à appliquer f aux composantes connexes des $\mathcal{S}_v(f)$ détectés.

L'esquisse de f peut être qualifiée de description du premier ordre de f .

Cette description équivaut donc à décrire la topologie des composantes connexes de $\mathcal{S}_0(f)$ ainsi que la singularité des points « clés » de $\overline{\mathcal{D}_f}$ extérieurs à $\mathcal{S}_0(f)$, où $\bar{E}$ désigne l'adhérence de E .

Toujours du fait de l'unidimensionalité de $\mathcal{D}_f \subseteq \mathbb{R}$ et de $\mathbb{R}$, cela revient à conserver l'information sur :

- Les couples de points clés qui sont bords d'une même composante connexe de $\mathcal{S}_0(f)$
- L'appartenance de chaque point clé à $\mathcal{S}_1(f)$
- L'appartenance de chaque point clé à $\mathcal{D}_f$

L'esquisse de $f : x \mapsto x^2 - x$ se représente donc par le Tableau 3, dans lequel « V » et « F » signifient respectivement « vrai » et « faux » vis-à-vis de l'appartenance et où les symboles ♣, ♦, ♠, ♥, et ◇ représentent des éléments de $\mathbb{R} = \mathbb{R} \cup \{-\infty, +\infty\}$.

$\in \mathcal{D}_f$	F	V	F
$\in \mathcal{S}_1(f)$	F	V	F
f 	♣ $\longleftrightarrow$ intervalle	♦ $\longleftrightarrow$ intervalle	♠ $\longleftrightarrow$ intervalle
	♥ $\longleftrightarrow$ intervalle	◇ $\longleftrightarrow$ intervalle	♥ $\longleftrightarrow$ intervalle

Tableau 3 : Esquisse de $f : x \mapsto x^2 - x$

L'esquisse peut, trivialement, s'affranchir de l'appartenance de chaque point clé à $\mathcal{D}_f$, si il est préétabli que la fonction f considérée est définie ou restreinte sur un espace fermé.

Toutefois, si on veut s'affranchir de caractériser l'appartenance à $\mathcal{S}_1(f)$, il est nécessaire de, clairement, définir que, alternativement :

- $\delta\mathcal{D}_f \in \mathcal{S}_1(f)$
- $\delta\mathcal{D}_f \cap \mathcal{S}_1(f) = \emptyset$

On note que les fonctions f qui, telles $f : x \mapsto x^2 - x$, admettent des limites (éventuellement infinies), sur les bords ouverts de $\mathcal{D}_f$, ont un comportement qui peut être décrit comme un comportement limite de celui d'une famille de fonctions. Chaque fonction g de cette famille est définie comme la restriction de f à $\mathcal{D}_g \subset \mathcal{D}_f$, tel que l'ensemble $\mathcal{D}_g$ est fermé (i.e. $\mathcal{D}_g = \overline{\mathcal{D}_g}$) et tel que la distance entre $\delta\mathcal{D}_g$ et $\delta\mathcal{D}_f$ est choisie arbitrairement petite.

On imposera aussi une certaine égalité des images le long de $\delta\mathcal{D}_g$ afin de respecter les éventuelles égalité des limites (finies ou infinies) de f .

Ainsi, il est possible de décrire f comme la fonction limite lorsque $\mathcal{D}_g$ tend vers $\mathcal{D}_f$ de $g = f|_{\mathcal{D}_g}$ où :

$$\left\{ \begin{array}{l} (a, b) \in (\delta\mathcal{D}_g)^2 \text{ et } a \neq b \\ g(a) = g(b) = z \text{ où } z \in \mathbb{R} = \mathbb{R} \cup \{-\infty, +\infty\} \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{l} (a', b') \in (\delta\overline{\mathcal{D}_f})^2 \text{ et } a' \neq b' \\ f(a') = f(b') = z' \text{ où } z' \in \mathbb{R} = \mathbb{R} \cup \{-\infty, +\infty\} \end{array} \right\}$$

Notre intérêt pour la description du comportement de fonction et pour sa comparaison au comportement de fonctions restreintes provient de notre angle d'étude des fonctions.

En effet, nous nous proposons d'étudier celles-ci par des méthodes numériques, implémentées informatiquement ; ces méthodes étant basées sur le concept de *l'Analyse par Intervalles*.

Or, il est à noter que lors de l'étude numérique d'une fonction, celle-ci doit, dans la plupart des cas, être restreinte à un sous-ensemble fermé (parfois borné) afin que cette étude pratique soit réalisable.

Ainsi avoir conscience de la perturbation du comportement d'une fonction, par restriction et étudier l'impact de celle-ci est primordial, au vu de notre choix d'une étude numérique.

Pour conclure sur la notion d'esquisse et sur l'information de l'action de f sur les classes $\mathcal{S}_v(f)$, on remarquera que l'objet combinatoire qu'est l'esquisse de f ne contient pas, a priori, toute l'information de $\mathcal{K}_0(f)$, car une visualisation de l'invariant $\mathcal{K}_0(f)$ doit aussi contenir l'information associée à une l'action d'une itération de f^{-1} sur les ensembles $f(\mathcal{S}_v(f))$ et $\mathcal{S}_v(f)$.

L'encrage par l'action de f^{-1}

Nous avons déjà pu remarquer, dans la sous-sous-section 3.2.3.1, que, afin de conserver toute l'information du lien par f entre l'espace source et l'espace but, il peut s'avérer nécessaire d'isoler, en complément de $\mathcal{S}_v(f)$, d'autres sous-espaces critiques, dans l'espace source.

Notre objectif est, ici, de montrer qu'il est suffisant d'isoler tous les ensembles $f^{-1}(f(\mathcal{S}_v(f)))$, pour pouvoir extraire toute l'information de l'invariant $\mathcal{K}_0(f)$. (i.e. une seule itération de f puis de f^{-1} sur $\mathcal{S}_v(f)$ décrit l'intégralité des sous-ensembles dont nous savons la topologie conservée par $\sim$)

La suffisance de $f^{-1}(f(\mathcal{S}_v(f)))$ découle simplement de la relation $f(f^{-1}(f(\mathcal{S}_v(f)))) = f(\mathcal{S}_v(f))$

La nécessité de pouvoir isoler ces ensembles pour décrire le lien par f , provient du fait que :

Chaque composante connexe de $f^{-1}(f(\mathcal{S}_v(f)))$ est en bijection avec son image par f .

Décrire le lien par f sur les ensembles dont la topologie est conservée par $\sim$ est synonyme de :

Répertorier la liste la plus fine des ensembles tels que f , restreinte à chacun d'entre eux, est bijective.

Ainsi, afin de décrire $\mathcal{K}_0(f)$, nous proposons de définir un *encrage* de f , comme étant un objet décrivant :

- La partition de $\mathcal{D}_f$ par les composantes connexes de $f^{-1}(f(\mathcal{S}_v(f)))$ (i.e. la topologie de chacun de ces ensembles connexes ainsi que leurs dispositions relatives)
- La partition de l'espace, dans lequel vit $f(\mathcal{D}_f)$, par les composantes connexes de $f(\mathcal{S}_v(f))$
- Le lien par f entre composantes connexes de $f^{-1}(f(\mathcal{S}_v(f)))$ et celles de $f(\mathcal{S}_v(f))$ (i.e. une liste de couples (a, b) tels que $f(a) = b$)

Similairement à la construction de l'esquisse, la construction d'un encrage de $f : \mathcal{D}_f \rightarrow \mathbb{R}$ où $\mathcal{D}_f \subseteq \mathbb{R}$ se résume à décrire la topologie de $\mathcal{D}_f - f^{-1}(f(\mathcal{S}_1(f) \cup \delta\overline{\mathcal{D}_f}))$, par l'intermédiaire de la nature des points critiques qui définissent les bords des intervalles constituant cet ensemble, puis à indexer les images de ces points critiques par f , en prenant à nouveau soin de conserver l'information d'égalité entre les images.

Notons par ailleurs que l'encrage s'identifie à l'esquisse dans le cas où $\forall v, f^{-1}(f(\mathcal{S}_v(f))) = \mathcal{S}_v(f)$ (ce qui est le cas, par exemple, pour $f : x \mapsto x^2 - x$, définie sur $\mathcal{D}_f = \mathbb{R}$).

Illustrons la méthode proposée par la construction progressive de l'encrage de $f : x \mapsto x^3 - x$:

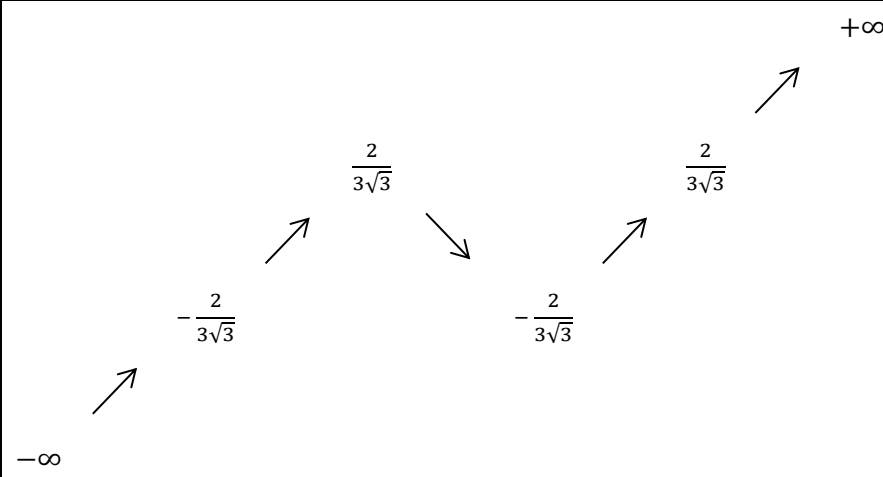
x	$-\infty$	$-\frac{2}{\sqrt{3}}$	$-\frac{1}{\sqrt{3}}$	$\frac{1}{\sqrt{3}}$	$\frac{2}{\sqrt{3}}$	$+\infty$			
$f': x \mapsto 3x^2 - 1$	+	+	+	0	-	0	+	+	+
$f: x \mapsto x^3 - x$									

Tableau 4 : Tableau de variation détaillé de $f : x \mapsto x^3 - x$, définie sur $\mathcal{D}_f = \mathbb{R}$

Dans le Tableau 5 et dans le Tableau 6, « V » et « F » signifient respectivement « vrai » et « faux » vis-à-vis de l'appartenance et les symboles ♣, ♦, ♠, ■, ○, ♥, ◇ et □ représentent des éléments de $\mathbb{R} = \mathbb{R} \cup \{-\infty, +\infty\}$.

$\in \mathcal{D}_f$	F	V	V	F
$\in \mathcal{S}_1(f)$	F	V	V	F
$f \hookrightarrow$	♣ intervalle	♦ intervalle	■ intervalle	♠
	♥ intervalle	◇ intervalle	□ intervalle	♥

Tableau 5 : Esquisse de $f : x \mapsto x^3 - x$

$\in \mathcal{D}_f$	F	V	V	V	V	F
$\in \mathcal{S}_1(f)$	F	F	V	V	F	F
$f \hookrightarrow$	♣ intervalle	○ intervalle	♦ intervalle	■ intervalle	△ intervalle	♠
	♥ intervalle	□ intervalle	◇ intervalle	□ intervalle	◇ intervalle	♥

Tableau 6 : Encrage de $f : x \mapsto x^3 - x$

Concept de portrait

Strictement, le portrait n'est pas un invariant, en général.

Le portrait d'une fonction f est défini comme étant un objet combinatoire, décrivant f .

Les portraits que nous considérerons dans ce mémoire seront plus précisément construits comme des descripteurs de $\mathcal{K}_0(f)$, car l'information conservée dans cet invariant décrit richement f .

En quelques mots, le portrait est un objet constitué d'éléments combinatoires simples, ce qui en fait un candidat privilégié pour une étude informatique des fonctions d'intérêt.

A l'opposé, l'encrage (i.e. une représentation canonique de $\mathcal{K}_0$) se veut être un objet univoque, contenant le maximum d'information de $\mathcal{K}_0$ sous une forme dont la nature importe peu.

Pour une fonction $f : \mathcal{D}_f \rightarrow \mathbb{R}$, telle que l'ensemble $\mathcal{D}_f \subseteq \mathbb{R}$ est une union finie d'intervalles fermés et qui vérifie que $\delta\mathcal{D}_f \cap \mathcal{S}_1(f) = \emptyset$, l'encrage de f s'identifie à un portrait de f , car l'information sur l'appartenance à $\mathcal{D}_f$ est redondante et car l'appartenance à $\mathcal{S}_1(f)$ du point, associé avec $\blacklozenge$, se déduit alors du fait que les points sources, reliés à $\blacklozenge$ ont (ou non) la même image.

Pour une telle fonction f , un portrait (s'appuyant sur $\mathcal{K}_0$) est alors 2 listes de paires de symboles et une liste de couples de symboles, codant l'action de f entre l'espace source $\mathcal{D}_f$ et l'espace image $\mathbb{R}$.

(Les symboles associés $\left\{ (a, b) \in \left(\delta f(\mathcal{D}_f) \right)^2 \mid [a, b] \cap f(\mathcal{D}_f) = \{a, b\} \right\}$ doivent être appariés.)

3.2.3.3. Compléments - Perspectives

Portrait ou encrage ? (i.e. description combinatoire de $\mathcal{K}_0$ ou objet canonique codant $\mathcal{K}_0$?)

Nous verrons, dans la suite de ce mémoire, que construire un portrait ne nécessite finalement que d'être en mesure de définir un objet combinatoire pour chaque topologie que l'on veut coder.

De ce fait, la construction d'un portrait semble plus simple que la définition d'un objet univoque codant l'invariant de $\mathcal{K}_0$.

Toutefois, il existe des pistes possibles de définition générale de l'encrage, cet objet que l'on veut, canoniquement, pouvoir associer à $\mathcal{K}_0$.

La piste plus concluante semblant être celle d'un objet de la théorie des graphes, où chaque ensemble critique définissant $\mathcal{K}_0$ serait relié à ses bords et à son image ou à ses pré-images par f .

Afin de compléter l'information portée par $\mathcal{K}_0$, chaque noeux du graphe serait alors adjoint des caractéristiques telles sa topologie.

L'obstacle majeur à la définition complète de cet objet réside dans la détermination des caractéristiques qui doivent être stockées pour que la représentation de $\mathcal{K}_0$ soit complète.

Toutefois, un encrage « incomplet » où seule la topologie de chaque élément est adjointe définit bien un invariant pour l'équivalence $\sim$

De plus, le fait que chaque portrait décrit bien $\mathcal{K}_0(f)$, lorsque $\mathcal{D}_f = \overline{\mathcal{D}_f}$ et que $\delta\mathcal{D}_f \subset \mathcal{S}_0(f)$, mais que sa construction ne soit pas univoque nous enjoint à considérer le fait que :

La classe de tous les portraits décrivant $\mathcal{K}_0(f)$ définit une représentation théorique de $\mathcal{K}_0(f)$.

Lien entre $\mathcal{K}_0$ et $\mathcal{K}_1$

Cela nécessiterait une étude mathématique relativement poussée mais :

Chaque $\mathcal{S}_v$ est isolé pour définir $\mathcal{K}_0$, et, sous certaines hypothèses, il est possible de garantir la « constance » d'une fonction sur certains espaces (similairement au cas des fonctions étudiées en « analyse complexe »).

Il paraît donc possible que, sous certaines hypothèses, l'information portée par $\mathcal{K}_0(f)$ soit assimilable à l'information portée par $\mathcal{K}_1(f)$ car l'information différentielle serait alors constante et connue sur chaque $\mathcal{S}_v(f)$ « maximal » (i.e. tel que $\forall w, (\exists k \in \mathbb{N} \mid \pi^k(v)) \Rightarrow (\mathcal{S}_w(f) = \mathcal{S}_v(f) \text{ ou } \mathcal{S}_w(f) = \emptyset)$).

Chapitre 4 Caractérisation et Détection des singularités

4.1 Introduction : la Théorie et la Pratique

La détection et l'isolation des singularités est essentielle à la classification des fonctions étant donné l'invariance topologique de la disposition des singularités les unes par rapport aux autres.

A l'issu du Chapitre 3, nous avons pu mettre en évidence un ensemble d'applications $\mathcal{P}: f \mapsto \mathcal{P}(f)$ tel que tout ensemble $\mathcal{P}(f)$ des points vérifiant les propriétés $\mathcal{P}(f)$ est homéomorphe à tout ensemble $\mathcal{P}(g)$ des points vérifiant les propriétés $\mathcal{P}(g)$ lorsque les fonctions f et g considérées vérifient $f \sim g$. On dira alors que l'application $\mathcal{P}$ est compatible avec l'équivalence $\sim$.

L'un des objectifs de ce document est donc de proposer un objet riche, décrivant le comportement de toute fonction d'intérêt f , tout en s'approchant le plus possible d'une représentation de l'invariant $\mathcal{K}_0$. Il est alors primordial de pouvoir détecter, et donc de savoir caractériser, chaque ensemble conservé, topologiquement, par l'équivalence $\sim$.

La méthodologie envisagée pour définir et pour détecter et isoler ces ensembles caractéristiques de f n'est pas dépendante des espaces considérés et se décompose en deux aspect complémentaires :

1. Répertoire et définir par des équations les ensembles caractéristiques pertinents de f pour $\sim$.
2. Mettre en place une méthode pratique de détection de chaque ensemble vérifiant l'un des systèmes d'équations définissant un ensemble caractéristique pertinent de f pour $\sim$.

Nous proposons donc d'étudier, dans chaque section de ce chapitre, une application de la méthodologie générale de caractérisation et de détection à une famille de fonctions $\{f : A \rightarrow \mathcal{P} \mid \dim(A) = \dim(\mathcal{P}) = n\}$, pour tous les entiers $n \in \llbracket 1; 3 \rrbracket$ et ceci dans l'ordre croissant.

La section 4.2 complètera ainsi, par l'exemple la présentation générale du modèle de la méthodologie. La section 4.3, moins élémentaire propose, en complément, un aperçu de l'intérêt pratique de notre problématique de classification, par une application en robotique. Enfin, l'approche de la section 4.4 devient plus théorique et ouvre sur l'étude et la classification de fonctions en dimensions supérieures. L'application en robotique, de la section 4.3, illustre aussi le Chapitre 5, qui est dédié à la suite logique de ce Chapitre 4, que constitue la construction d'un objet pour la classification globale.

4.2 Méthodologie générale et Application en dimension 1

Les deux sous-sections suivantes explicitent en détails et respectivement les deux aspects, de la méthodologie envisagée que sont la caractérisation et la détection des ensembles caractéristiques.

Les différentes étapes de la méthodologie générale seront illustrées au fur et à mesure, dans le cas relativement simple des fonctions entre espaces de dimension 1 (plongeables dans $\mathbb{R}$).

4.2.1. Caractérisation Théorique

Cette première étape de la méthodologie consiste principalement à définir les éléments d'intérêt qui sont pertinents à rechercher pour caractériser la classe de fonctions actuellement considérée.

Au regard de la Définition 3.1.1.3 de l'équivalence $\sim$ entre fonctions, nous ne prétendons pas caractériser chaque sous-ensembles dont la topologie est conservée par les difféomorphismes liés à $\sim$. En effet, l'intégralité E de ces sous-ensembles est décrite par une sous-partie F de celle-ci. Toutefois, les sous-ensembles éléments de F ne sont pas, a priori, décrits par un nombre fini de propriétés et ne sont alors pas, a priori, décrits par un nombre fini de systèmes d'équations.

On notera que cette famille de propriétés s'agrandit avec les dimensions des espaces impliqués.

De ce fait, notre stratégie est d'isoler certains sous-ensembles spécifiques, de telle sorte que ceux-ci décrivent et délimitent les sous-ensembles conservés par $\sim$, pour une famille *générique* de fonctions parmi la classe de fonctions actuellement considérée, comme défini dans la section 2.2.

En ce sens, nous recherchons les sous-ensembles caractéristiques « usuels » des fonctions d'intérêt.

4.2.1.1. Identifier les ensembles $\mathcal{S}_v(f)$ pertinents

Soit une fonction d'intérêt f quelconque. Nous savons, premièrement, que les composantes connexes de chaque ensemble $\mathcal{S}_v(f)$ voient leurs topologies conservées par $\sim$, d'après la Proposition 3.2.1.1.

Il en est de même pour $f(\mathcal{S}_v(f))$ et pour $f^{-1}(f(\mathcal{S}_v(f)))$, d'après la Proposition 3.2.1.2.

La première étape de la méthodologie est donc d'analyser et déterminer les vecteurs v , pour lesquels $\mathcal{S}_v(f)$ est formellement non vide, en s'appuyant sur les propriétés reportées dans la Remarque 3.2.1.1. Parmi ces ensembles $\mathcal{S}_v(f)$ non vides, on peut toutefois vérifier que certains sont vides *génériquement* (i.e. pour une partie au moins *résiduelle* des fonctions d'intérêt). Pour vérifier ceci, on s'appuiera alors sur les concepts et les résultats développés dans la section 2.2.

Nous avons déjà analysé les ensembles $\mathcal{S}_v(f)$ d'une fonction f entre espaces de dimension 1, à l'occasion du § 3.2.3.1 et du § 3.2.3.2. On en déduit alors l'arbre des $\mathcal{S}_v(f)$ représenté à la Figure 9, où chaque flèche relie un ensemble à un de ses sous-ensembles.

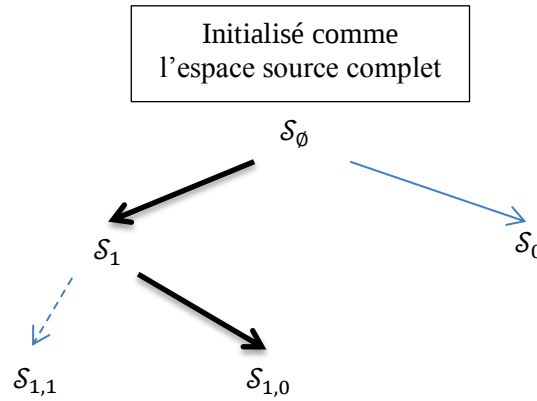


Figure 9 : Arbre des $\mathcal{S}_v$ pour une fonction entre espaces de dimensions 1

L'ensemble $\mathcal{S}_1(f)$ est, dans le cas d'une fonction f , entre espaces de dimension 1, égal à S_f .

On a alors que $\mathcal{S}_1(f) = S_f = \{x | f'(x) = 0\}$. Plus généralement, si f est une fonction, entre espaces de mêmes dimensions, alors on pourra poser $S_f = \{x | \det(Df_x) = 0\}$.

On notera que $\mathcal{S}_{1,0}$ contient, ici, les points de $\mathcal{S}_1$, isolés dans $\mathcal{S}_1$ alors que $\mathcal{S}_{1,1}$ contient les points de $\mathcal{S}_1$ non isolés dans $\mathcal{S}_1$. L'ensemble $\mathcal{S}_{1,1}$ est, en effet, constitué des segments de l'espace source tels que $f' = 0$, sur la continuité du segment. On a alors, localement, $\mathcal{S}_{1,1} = \mathcal{S}_1$ impliquant ainsi $\mathcal{S}_{1,1,1} = \mathcal{S}_{1,1}$ sur l'ensemble du segment. De ce fait, par récurrence, on obtient que :

$$\forall k \in \mathbb{N} - \{0,1\}, \forall v \in \mathbb{R}^k \text{ tel que } (\forall i \in \llbracket 1; k \rrbracket, v_i = 1), \mathcal{S}_v = \mathcal{S}_{1,1}$$

On notera, dès à présent, que les composantes connexes de $\mathcal{S}_{1,1}(f)$ s'identifient aux régions, de l'espace source, sur lesquelles f est constante. On en déduit que $x \in \mathcal{S}_{1,1}(f) \Rightarrow f'(x) = f''(x) = 0$.

Les ensembles $\mathcal{S}_v(f)$ d'intérêt, pour une fonction f entre espaces de dimensions 1, sont donc les classes $S_f = \mathcal{S}_1(f)$ et $\mathcal{S}_{1,1}(f)$ ou $\mathcal{S}_{1,0}(f)$, selon la méthode de distinction entre ces deux classes.

Aussi, les points non isolés de S_f sont, trivialement, les points de $\mathcal{S}_{1,1}(f)$. Notons tout de même que l'étude de f'' permet d'analyser le comportement topologique de f' et donc de caractériser $\mathcal{H}_1(f)$.

Rappelons toutefois que, l'équivalence $\sim$ conserve les topologies des sous-ensembles de $\mathcal{S}_v$. De ce fait, par exemple, l'isolation des points dans $\mathcal{S}_f$ est une propriété invariante par $\sim$ qui est donc complémentaire à la distinction entre $\mathcal{S}_1$ et $\mathcal{S}_{1,1}$, bien que ces deux distinctions se confondent ici.

4.2.1.2. Propriétés d'intérêt liées à l'image par la fonction

La Proposition 3.2.1.2 met en évidence le fait que, l'équivalence $\sim$ respecte les topologies des ensembles $f(\mathcal{S}_v(f))$ et $f^{-1}(f(\mathcal{S}_v(f)))$, pour tout vecteur v tel que $\exists k \in \mathbb{N}$ de sorte que $v \in \mathbb{N}^k$. Afin d'affiner la description de la fonction f étudiée, vis-à-vis de $\sim$, il apparaît donc pertinent d'extraire des caractéristiques, des images et des contres images non singulières des ensembles $\mathcal{S}_v(f)$, qui sont respectées par $\sim$.

En toute généralité, cette extraction de caractéristiques revient à décrire, le plus précisément possible, la topologie de chaque ensemble $f(\mathcal{S}_v(f))$ et $f^{-1}(f(\mathcal{S}_v(f)))$, ceci pour tout vecteur v admissible. Toutefois, tout comme pour décrire les classes de singularités $\mathcal{S}_v(f)$, nous proposons de reconnaître théoriquement des caractéristiques suffisantes à la description des topologies à décrire.

Considérons, l'exemple illustratif actuel des fonctions unidimensionnelles $f : \mathbb{R} \supseteq \mathcal{D}_f \rightarrow \mathbb{R}$.

Nous pouvons remarquer, à la lumière du § 3.2.3.2, que la connaissance de la disposition relative des éléments de $f(\mathcal{S}_f \cup \delta\overline{\mathcal{D}_f})$ et de leurs pré-images vis-à-vis de f (i.e. $\{x \in \overline{\mathcal{D}_f} \mid f(x) \in f(\mathcal{S}_f \cup \delta\overline{\mathcal{D}_f})\}$) est suffisante pour caractériser les ensembles $f(\mathcal{S}_v(f))$ et les ensembles $f^{-1}(f(\mathcal{S}_v(f)))$. Notons que ces informations peuvent être obtenues à l'aide de méthodes numériques ou formelles.

En pratique on se proposera d'extraire l'ensemble $\{(x, y) \in (\mathcal{S}_f \cup \delta\overline{\mathcal{D}_f})^2 \mid f(x) = f(y) \text{ et } x \neq y\}$ afin de construire $f(\mathcal{S}_f \cup \delta\overline{\mathcal{D}_f})$, sans erreurs (éventuellement dues à une imprécision numérique).

Nous noterons, par la suite, que l'ensemble $\{(x, y) \in (\mathcal{S}_f \cup \delta\overline{\mathcal{D}_f})^2 \mid f(x) = f(y) \text{ et } x \neq y\}$ est vide pour un ensemble générique de fonctions. De plus, la caractérisation effective de cet ensemble est dépendante de la méthode de détection pratique choisie.

4.2.2. Détection Pratique

On propose d'utiliser des méthodes basées sur l'Analyse par Intervalles, basées sur les méthodes présentées au § 2.3.4, pour détecter les différents ensembles d'intérêt, pour lesquels des équations ont été mises en évidence théoriquement.

L'intérêt premier de l'Analyse par Intervalles étant d'être capable d'analyser numériquement des situations où l'étude formelle s'avère difficile nous remarquerons dès à présent que :

- Une étude formelle permet d'analyser de façon plus poussée les ensembles définis
- Dans des cas formels simple, une étude formelle est souvent plus pertinente

De ce fait, dans le cas d'une fonction $f : \mathbb{R} \supseteq \mathcal{D}_f \rightarrow \mathbb{R}$, nous proposons une étude théorique couplée à une méthodologie s'appuyant sur l'Analyse par Intervalles.

L'analyse conjointe, d'une fonction $f : \mathbb{R} \supseteq \mathcal{D}_f \rightarrow \mathbb{R}$ générale, par une étude formelle et par une méthodologie basée sur l'Analyse par Intervalles permet de souligner les spécificités de chacune de ces approches.

Nous concluons cette présentation de la méthodologie générale en mettant en évidence la perspective naturelle de construire un portrait de f , à partir de la topologie des ensembles d'intérêt extraits.

En effet, pour la dimension 1, il est assez aisé de définir un portrait, à partir des éléments caractérisés, et qui suit le modèle de l'objet introduit au § 3.2.3.2.

En dimension supérieure, la construction d'un portrait s'avère plus délicate. Toutefois, une méthodologie générale peut toujours être définie, selon les développements proposés au Chapitre 5.

4.2.2.1. Etude formelle

Pour que l'étude formelle de $f : \mathbb{R} \supseteq \mathcal{D}_f \rightarrow \mathbb{R}$ soit possible, il est nécessaire que les racines des fonctions dérivées f' et f'' puissent être suffisamment caractérisées par des éléments formels, afin d'être en mesure d'ordonner ces racines.

La possibilité de caractériser suffisamment les expressions formelles des racines des systèmes, définissant les ensembles d'intérêt recherchés, constitue la motivation principale à considérer une étude formelle, plutôt qu'une autre méthode.

Soit donc une fonction $f : \mathbb{R} \supseteq \mathcal{D}_f \rightarrow \mathbb{R}$ deux fois différentiable.

Soit $E \subseteq \overline{\mathbb{R}}$. Par convention, $\mathbb{I}(E)$ est l'ensemble des éléments connexes de E pour la topologie, induite sur E , par la topologie usuelle sur $\mathbb{R}$ (qui étend la topologie usuelle sur $\mathbb{R}$).

Un algorithme formel caractérisant les ensembles d'intérêt de cette fonction et construisant son portrait serait l'algorithme procédural suivant :

1. Extraire $\mathbb{I}(\delta\overline{\mathcal{D}_f})$ et en déduire $\mathbb{I}(f(\delta\overline{\mathcal{D}_f}))$
2. Déduire $\mathbb{I}(S_f)$ en isolant $\{x \in \mathcal{D}_f \mid f'(x) = 0\}$
3. Déduire $\mathbb{I}(S_{1,1}(f))$ à partir de $\mathbb{I}(S_f)$ en isolant $\{x \in I_f \mid f''(x) = 0\}$
4. Déduire $\mathbb{I}(f(S_f))$ à partir de l'ensemble $\{f(x) \mid x \in I_f \text{ et } f'(x) = 0\}$
 - a. La construction de $\mathbb{I}(f(S_f))$ requiert d'isoler $\{(x, y) \in S_f^2 \mid f(x) = f(y) \text{ et } x \neq y\}$
 - b. Lorsque $\{(x, y) \in S_f^2 \mid f(x) = f(y) \text{ et } x \neq y\} \neq \emptyset$, on conserve une liste L des sous-ensembles singuliers maximaux appariés, car partageant la même image.
5. Déduire $\mathbb{I}(f(S_{1,1}(f)))$, de $\mathbb{I}(f(S_f))$ et de $f(S_{1,1}(f)) = \{f(x) \mid x \in \mathcal{D}_f \text{ et } f''(x) = 0\}$
6. On extrait $\mathbb{I}(f^{-1}(f(S_f) \cup f(\delta\overline{\mathcal{D}_f})))$ en appliquant f^{-1} aux ensembles $f(S_f) \cup f(\delta\overline{\mathcal{D}_f})$

A partir des ensembles isolés, il est donc possible de constituer l'esquisse, l'encrage et le portrait de f (selon les mêmes procédures et raisonnements qu'au § 3.2.3.2).

Cette application de la méthodologie à une fonction f , entre espaces de dimensions 1, montre ainsi que la perspective immédiate de la détection et de la caractérisation des singularités de f consiste à classifier f selon la topologie des espaces délimités par les ensembles caractérisés.

4.2.2.2. Etude à l'aide de l'Analyse par Intervalles

Le plan de l'algorithme de caractérisation des singularités de f est le même que celui considéré pour l'étude formelle de la fonction f .

La principale différence entre les deux études provient des restrictions d'applications de celles-ci :

- L'étude formelle nécessite que des expressions formelles explicites existent pour chaque ensemble à caractériser.
- L'étude à l'aide de l'Analyse par Intervalles ne peut pas, trivialement, confirmer l'égalité de deux valeurs, lorsqu'au moins une est encadrée par un intervalle non ponctuel.

Ainsi, l'Analyse par Intervalles permet d'encadrer chacun des ensembles $\delta\overline{\mathcal{D}_f}$, S_f et $S_{1,1}(f)$, de même que leurs images respectives $f(\delta\overline{\mathcal{D}_f})$, $f(S_f)$ et $f(S_{1,1}(f))$ et la contre image $f^{-1}(f(S_f) \cup f(\delta\overline{\mathcal{D}_f}))$. Toutefois, bien que l'on puisse vérifier, en considérant une précision suffisante, que les ensembles $S_{1,1}(f)$ et $\{(x, y) \in S_f^2 \mid f(x) = f(y) \text{ et } x \neq y\}$ sont vides, il n'existe donc pas de méthode triviale permettant de garantir que $S_{1,1}(f) \neq \emptyset$ ou que $\{(x, y) \in S_f^2 \mid f(x) = f(y) \text{ et } x \neq y\} \neq \emptyset$.

En vérifiant que ces conditions de vacuité sont vérifiées par un ensemble résiduel de fonctions, pour la topologie $\mathcal{T}^\infty(\mathcal{D}_f, \mathbb{R})$, on peut affirmer que réaliser cette étude, à l'aide de l'Analyse par Intervalles, permet de conclure pour une classe générique de fonctions.

On peut donc remplacer, dans la chronologie proposée au § 4.2.2.1, l'étape 3 par une vérification que $\mathcal{S}_{1,1}(f) = \emptyset$ ou, a priori, admettre que $\mathcal{S}_{1,1}(f) = \emptyset$ et omettre cette étape.

De même, l'étude de $\{(x, y) \in S_f^2 \mid f(x) = f(y) \text{ et } x \neq y\}$ peut être envisagée de façon similaire.

Une approche alternative serait, pour chaque recouvrement $\mathfrak{A}$, d'un ensemble d'intérêt, d'identifier les sous-ensembles encadrés par $I_1 \in \mathfrak{A}$ et par $I_2 \in \mathfrak{A}$ lorsque $I_1 \cap I_2 \neq \emptyset$.

Ceci correspond à étudier la fonction $h = \vartheta \circ f \circ \vartheta$ au lieu de f , où ϑ et ϑ sont des transformations sur les espaces source et but de f , traduisant les identifications proposées ci-dessus.

On note, finalement, que, si $\mathcal{S}_{1,1}(f) = \emptyset$, alors il est possible d'appliquer la méthode de Newton par Intervalles pour détecter les points de S_f qui sont ainsi tous isolés et éléments de $\mathcal{S}_{1,0}(f)$.

4.2.3. Conclusion relative aux méthodes en dimension 1

En dimension 1, une étude exhaustive des classes de singularités semble envisageable.

De plus, nous avons pu établir, précédemment, qu'un objet descripteur, relativement à l'équivalence $\sim$ peut être défini à l'aide de la connaissance des topologies des composantes connexes délimitées par les ensembles extraits.

Nous pouvons remarquer que l'approche proposée, basée sur l'Analyse par Intervalles, semble d'intérêt modéré, pour l'étude d'une fonction f , entre espaces de dimension 1.

Toutefois, cette prédominance de l'étude formelle réside principalement dans l'extensivité des méthodes d'études formelles, pour les espaces unidimensionnels.

En effet, nous remarquons que la différence de méthodologie entre l'étude formelle et l'étude basée sur l'Analyse par Intervalles réside dans les restrictions d'applications de chacune des deux études.

Les fonctions qui n'admettent pas d'expressions formelles explicites pour ces ensembles étudiés sont moins triviales que celles ne vérifiant pas les conditions requises par l'étude complète de ces mêmes ensembles à l'aide de l'Analyse par Intervalles.

Ainsi, la méthodologie générale proposée s'applique aux fonctions triviales, de façon élémentaire.

L'étude des classes singulières de fonctions, entre espaces de dimension 1, illustre aussi la perspective de la définition d'un objet général caractérisant les fonctions, relativement à l'équivalence $\sim$.

Cette application introductive de la méthodologie générale permet de préciser les stratégies d'études considérées dans le cas général et l'existence de limites aux méthodologies envisagées.

La considération de la notion de généricité et de stratégies complémentaires s'illustre plus particulièrement pour les fonctions entre espaces de dimensions supérieures, considérées et analysées dans les sections suivantes de ce chapitre.

4.3 Application en dimension 2

4.3.1. Caractérisation Théorique

La caractérisation théorique générale suit le même principe que présenté précédemment bien que la structure des ensembles caractérisés se complexifie avec l'augmentation des dimensions considérées, comme intuitivement prévu.

4.3.1.1. Identifier les ensembles $\mathcal{S}_v(f)$ pertinents

Nous considérons, dans cette section, les fonctions $f : A \rightarrow \mathcal{P}$, telles que $\dim(A) = \dim(\mathcal{P}) = 2$.

La Figure 10 présente l'arbre des inclusions entre les classes de singularités pertinentes pour une certaine sous-famille *générique* de ces fonctions (dont le sens est défini à la section 2.2).

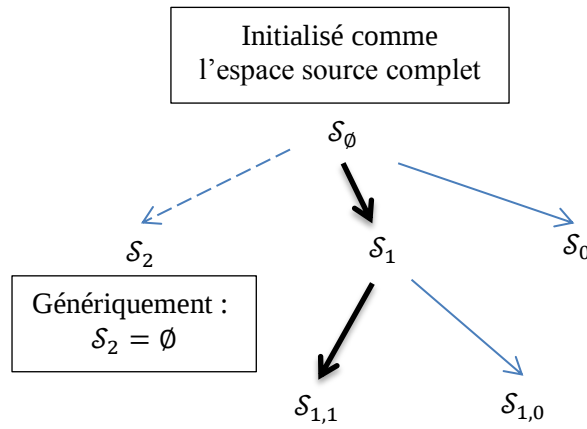


Figure 10 : Arbre des $\mathcal{S}_v$ pour une fonction entre espaces de dimensions 2

La généricité pouvait être omise, dans la détermination des classes de singularités, en dimension 1, du fait du nombre restreint de ces classes. Toutefois, sa pertinence commence à apparaître à partir de l'étude de la dimension 2, puisque $\mathcal{S}_2(f)$ peut, potentiellement, nécessiter l'étude de $\mathcal{S}_{2,2}(f)$ et $\mathcal{S}_{2,1}(f)$ puis celle de leurs sous-classes de singularités successives.

On notera que la généricité peut toujours être omise si l'on se propose de caractériser exhaustivement les fonctions entre espaces de dimension 2.

Les fonctions génériques que nous considérons, dans cette section sont donc $\{g : A \rightarrow \mathcal{P} \mid \mathcal{S}_2(g) = \emptyset\}$.

Il est immédiat, selon la Proposition 2.2.2.1, qu'un ensemble résiduel, pour $\mathcal{T}^\infty(A, \mathcal{P})$, de fonctions $g \in \mathcal{C}^\infty(A, \mathcal{P})$ vérifie $\mathcal{S}_2(g) = \emptyset$. En effet, $\mathcal{S}_2(g) = \emptyset \Leftrightarrow \forall M \in A, Dg(M) \neq 0_{2 \times 2}$ et $Dg(M) = 0_{2 \times 2}$ est un système d'équations de co-dimension 4 (sur le 1-jet d'une fonction sur A , tel que $\dim(A) = 2$). L'ensemble $\{g : A \rightarrow \mathcal{P} \mid \mathcal{S}_2(g) = \emptyset\}$ est donc bien générique selon la Définition 2.2.2.2, si l'on considère des fonctions indéfiniment différentiables.

Notons toutefois que l'étude, proposée ici, s'applique à toute fonction 2 fois différentiable.

Considérons donc une fonction $f : A \rightarrow \mathcal{P}$, vérifiant $\dim(A) = \dim(\mathcal{P}) = 2$ et $\mathcal{S}_2(f) = \emptyset$.

Proposition 4.3.1.1 (Equation définissant $\mathcal{S}_1(f)$)

En s'appuyant sur la vacuité de $\mathcal{S}_2(f)$, il est possible de caractériser simplement $\mathcal{S}_1(f) = \mathcal{S}_f$:

$$\mathcal{S}_1(f) = \{M \in A \mid \det(Df(M)) = 0\}$$

Système d'équations définissant $\mathcal{S}_{1,1}(f)$

Définition 4.3.1.1 (Sous-espace vectoriel engendré par un vecteur)

Soit un espace vectoriel E , considéré vis-à-vis du corps F et soit $\mathcal{E} \subseteq E$.

On notera $\mathcal{V}_E^F(\mathcal{E})$, le sous-espace vectoriel engendré par $\mathcal{E}$, dans le F -espace vectoriel E .

On omet de préciser le corps F ou l'espace vectoriel E , dans le cas où ceux-ci sont sans ambiguïtés.

De même, si $\mathcal{E} = \{w\}$, on écrira $\mathcal{V}_E^F(w)$ plutôt que $\mathcal{V}_E^F(\{w\})$.

Définition 4.3.1.2 (Espace tangent $\mathcal{T}_M E$)

On a noté $\mathcal{T}_M E$, l'espace tangent à l'espace E , au point M .

$\mathcal{T}_M E$ est l'ensemble des vecteurs vitesses possibles, au point M , d'une courbe tracée sur E .

Selon la Définition 3.2.1.2, $\mathcal{S}_{1,1}(f)$ est défini comme étant un sous-ensemble de $\mathcal{S}_1(f)$ contenant les points $M \in A$ vérifiant que $\exists ! E = \mathcal{V}(w)$ tel que $0 \neq w \in \text{Ker}(Df(M))$ et tel que $E \subseteq \mathcal{T}_M \mathcal{S}_1(f)$.

Afin de définir un système caractérisant les points de $\mathcal{S}_{1,1}(f)$, nous nous proposons donc de reformuler la condition ci-avant, sous forme d'une équation, présentée à la Proposition 4.3.1.2.

Définition 4.3.1.3 (Opérateur gradient ∇)

Soit une fonction $h : A \rightarrow E$ différentiable telle que $\dim(E) = 1$.

Le gradient ∇h est la fonction qui, à tout point $M \in A$, associe le vecteur de l'application linéaire dh_M . (Les composantes de $\nabla h(M)$ sont celles de $Dh(M)$, dans la même base de l'espace tangent à A en M .)

D'après la Proposition 4.3.1.1, si f est générique, l'ensemble $\mathcal{S}_1(f)$ est défini par une unique équation.

L'espace tangent $\mathcal{T}_M \mathcal{S}_1(f)$, lorsque $M \in \mathcal{S}_1(f)$, est alors orthogonal à $n(M) = \left(\nabla(\det(Df)) \right)(M)$ si :

- L'espace A est orthonormé
- $n(M) \neq 0$

On supposera donc, par la suite, que l'espace A est orthonormé et admet un produit scalaire « $\bullet$ », induit par la norme considérée sur A .

Remarque 4.3.1.1 (Hypothèse facultative sur les fonctions considérées)

Afin de simplifier le raisonnement actuel, il serait possible de considérer un sous-ensemble, toujours générique, des fonctions g telles que $\mathcal{S}_2(g) = \emptyset$, en imposant que les fonctions g considérées vérifient $\{ M \in A \mid \det(Dg(M)) = 0 \} \cap \{ M \in A \mid \left(\nabla(\det(Dg)) \right)(M) = (0,0) \} = \emptyset$.

En effet, cette condition se traduit en trois équations indépendantes sur le 2-jet des fonctions.

Cette hypothèse permet d'affirmer que l'aire des lieux singuliers $\mathcal{S}_g$ est nulle, puisque vérifier les propriétés $\det(Dg(M)) = 0$ et $\left(\nabla(\det(Dg)) \right)(M) = (0,0)$ équivaut à vérifier la propriété :

« Il existe un voisinage V de M tel que $V \subseteq \mathcal{S}_g$ »

(Si $\mathcal{S}_2(g) = \emptyset$ alors ce voisinage V vérifie $\forall k \in \mathbb{N}, \forall v \in \mathbb{N}^k, (\forall l \in \llbracket 1; k \rrbracket, v_l = 1) \Leftrightarrow V \subseteq \mathcal{S}_v(g)$.)

Toutefois, cette hypothèse n'est pas nécessaire puisque l'application pratique de la méthode nécessite de gérer les fonctions telles que cette hypothèse n'est pas vérifiée.

Ainsi, soit un vecteur $w(M) \neq 0$, tel que $w(M) \bullet n(M) = 0$, alors on a $\mathcal{T}_M \mathcal{S}_1(f) = \mathcal{V}(w(M))$.

D'où, lorsque $Df(M) \neq 0$ et lorsque $n(M) \neq 0$:

$M \in \mathcal{S}_1(f) \Leftrightarrow Df(M) \times w(M) = 0$, pour un vecteur $w(M)$ tel que $w(M) \bullet n(M) = 0$ et $w(M) \neq 0$

Alternativement, les vecteurs lignes de $Df(M)$ définissent une base de l'orthogonal de $\text{Ker}(Df(M))$.

En s'appuyant sur cette remarque on peut également obtenir la caractérisation proposée de $\mathcal{S}_{1,1}(f)$.

Quel que soit le raisonnement suivi, on obtient ainsi la Proposition 4.3.1.2 :

Proposition 4.3.1.2 (Equations définissant $\mathcal{S}_{1,1}(f)$)

Si $M \notin \mathcal{S}_2(f)$ et si $n(M) = \left(\nabla(\det(Df)) \right)(M) \neq 0$.

Pour tout vecteur $w(M)$, tel que $w(M) \neq 0$ et $w(M) \cdot n(M) = 0$, on a :

$$M \in \mathcal{S}_{1,1}(f) \Leftrightarrow \begin{cases} Df_1(M) \cdot w(M) = 0 \\ Df_2(M) \cdot w(M) = 0 \end{cases} \text{ où } Df(M) = \begin{pmatrix} Df_1(M) \\ Df_2(M) \end{pmatrix}$$

4.3.1.2. Propriétés d'intérêt liées à l'image par la fonction

Similairement à la caractérisation des points singuliers partageant la même image, proposée lors de l'étude d'une fonction entre espaces de dimensions 1, au § 4.2.1.2, nous proposons de caractériser l'ensemble $\{(M_1, M_2) \in (S_f)^2 \mid f(M_1) = f(M_2) \text{ et } M_1 \neq M_2\}$.

On obtient trivialement la Proposition 4.3.1.3 :

Proposition 4.3.1.3 (Equations définissant un couple de points de $\mathcal{S}_f$ de même image)

$$\left((M_1, M_2) \in (S_f)^2 \text{ et } f(M_1) = f(M_2) \right) \Leftrightarrow \begin{cases} f(M_1) = f(M_2) \\ \det(Df(M_1)) = 0 \\ \det(Df(M_2)) = 0 \end{cases}$$

Conceptuellement, il serait pertinent de vérifier le niveau de singularité des points de même image. Toutefois, en nous appuyant sur le concept de généricité, on peut affirmer qu'il existe un ensemble résiduel de fonctions telles que $\{(M_1, M_2) \in (S_f)^2 \mid f(M_1) = f(M_2) \text{ et } M_1 \neq M_2\} \subseteq (\mathcal{S}_{1,0}(f))^2$.

En effet, ajouter la condition que l'un des deux éléments appartient à $\mathcal{S}_{1,1}(f)$ ajoute une équation au système considéré, dont la co-dimension dépasse alors la dimension de l'espace d'étude, A^2 , dont la dimension topologique est égale à $(\dim(A))^2 = 2^2 = 4$.

Dans le cadre d'une étude formelle exhaustive, il reste cependant pertinent de vérifier effectivement les niveaux de singularité des points partageant la même image, si cela est possible.

4.3.1.3. Considération pratique : Caractérisation d'un robot 3R

Fonction géométrique et concepts de singularité

Précisons à nouveau que *les points cusps et les points nodes dans l'espace de travail* sont des positions singulières de l'effecteur du robot, vérifiant certaines propriétés additionnelles :

- Un cusp admet trois solutions géométriques inverses confondues.
- Un node admet deux paires distinctes de solutions géométriques inverses confondues.

Toutefois, au lieu de caractériser ces points spécifiques de l'espace de travail, nous nous proposons de caractériser leurs solutions géométriques inverses.

Les points d'intérêt que nous considérons donc sont *les points cusps et nodes de l'espace articulaire*, que nous définissons comme étant *les ensembles des solutions géométriques inverses singulières, respectifs, des points cusps et nodes dans l'espace de travail*.

Fonction géométrique simplifiée décrivant un manipulateur 3R

Nous avons déjà introduit le concept du manipulateur robotique 3R, en tant qu'exemple de robot pouvant être cuspidal, au § 0.

Afin de formaliser la fonction géométrique considérée pour caractériser un tel manipulateur, nous nous référons à la Figure 11, qui est une reproduction de la Figure 3, reportée ici pour simplifier la lecture.

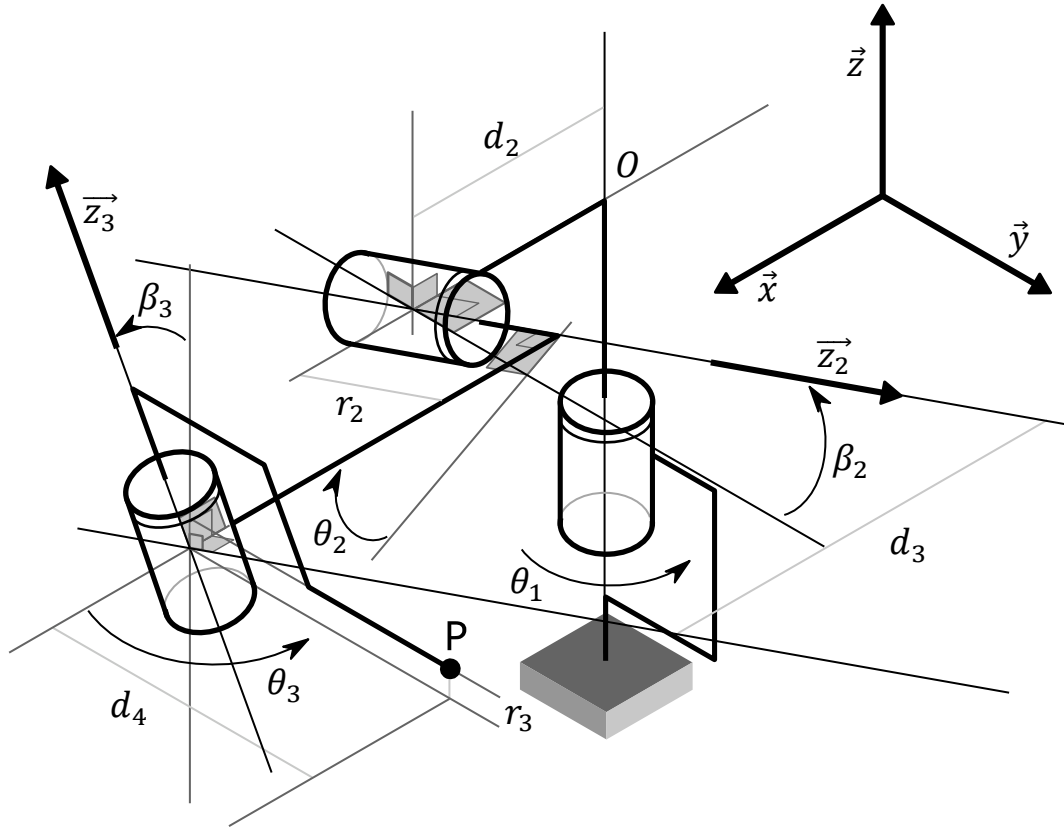


Figure 11 : Schéma cinématique d'un manipulateur 3R

La fonction géométrique, d'un tel manipulateur 3R, $g : (\theta_1, \theta_2, \theta_3) \mapsto (x, y, z, u_x, u_y, u_z)$, qui opère de $\mathbb{R}^3$ vers $\mathbb{R}^6$, peut être décrite par une fonction géométrique simplifiée $f : (\theta_2, \theta_3) \mapsto (\rho, z)$.

C'est cette fonction f , opérant du tore vers $\mathbb{R}^2$, que nous considérons pour illustrer la méthodologie de détection d'ensembles d'intérêt pour les fonctions entre espaces de dimensions 2.

On considère uniquement le manipulateur 3R en tant que robot positionneur. De ce fait, on omet les composantes d'orientation u_x, u_y et u_z de la fonction $g : (\theta_1, \theta_2, \theta_3) \mapsto (x, y, z, u_x, u_y, u_z)$.

Par la suite, la fonction $G : (\theta_1, \theta_2, \theta_3) \mapsto (x, y, z)$ obtenue en omettant les composantes d'orientation s'avère admettre une invariance le long de son premier axe d'orientation, caractérisé par l'angle θ_1 .

En effet, si la rotation mesurée par l'angle θ_1 n'est pas limitée et si, pour tout couple (θ, Θ) de mesures d'angles pour cette rotation, on vérifie la relation :

$$\begin{array}{ccc} (\theta, \theta_2, \theta_3) & & (\Theta, \theta_2, \theta_3) \\ \text{est une configuration} & \Leftrightarrow & \text{est une configuration} \\ \text{possible du robot} & & \text{possible du robot} \end{array}$$

alors la fonction $G_{\theta_1} : (\theta_2, \theta_3) \mapsto (x_{\theta_1}(\theta_2, \theta_3), y_{\theta_1}(\theta_2, \theta_3), z_{\theta_1}(\theta_2, \theta_3)) = G(\theta_1, \theta_2, \theta_3)$ admet, pour tout paramètre θ_1 , un même domaine de définition.

Ceci implique que DG et $\det(DG)$ admettent un domaine de définition indépendant du paramètre θ_1 .

Complémentairement, l'expression formelle de la fonction $\det(DG)$ est indépendante du paramètre θ_1 . Ainsi, soit Q , l'espace articulaire accessible du robot 3R considéré S_G est invariant selon θ_1 lorsque :

$$\exists E \text{ tel que } \forall \theta_1 \in]-\pi, \pi], \quad E = \{(\theta_2, \theta_3) | (\theta_1, \theta_2, \theta_3) \in S_G \cap Q\}$$

Remarque 4.3.1.2 (Interprétations dans l'espace opérationnel)

Lorsqu'un ensemble E de couples d'angles vérifie $(\forall \theta_1 \in]-\pi, \pi], \forall (\theta_2, \theta_3) \in E, (\theta_1, \theta_2, \theta_3) \in Q)$ alors l'ensemble $G(E)$ peut être décrit par la révolution, autour de l'axe de la rotation d'angle θ_1 , d'une section de $G(E)$ par un plan contenant cet axe de rotation.

En particulier, cette remarque s'applique à l'ensemble S_G , lorsqu'il est invariant selon le paramètre θ_1 .

Remarque 4.3.1.3 (Expression formelle de G)

Nous ne reporterons pas, dans le corps de ce mémoire, les expressions formelles des fonctions g et G , qui se déduisent du schéma cinématique d'un manipulateur 3R, reproduit tout au long de ce manuscrit. Toutefois, les expressions formelles utilisées par les algorithmes, implémentés au cours de cette thèse, sont toutes reportées en annexe au § A.1 et disponibles avec le code source de ces algorithmes.

On suppose donc que l'ensemble $S_G \cap Q$ est invariant vis-à-vis de l'angle θ_1 .

Afin de décrire $S_G \cap Q$, il est donc suffisant de décrire une section génératrice de cet ensemble.

Pour ce faire, on considère $G_{\theta_1} : (\theta_2, \theta_3) \mapsto (x_{\theta_1}, y_{\theta_1}, z_{\theta_1}) = G(\theta_1, \theta_2, \theta_3)$ et $\rho_{\theta_1} = \sqrt{x_{\theta_1}^2 + y_{\theta_1}^2}$.

On note, en se reportant à la Figure 11, que $\rho_{\theta_1}(\theta_2, \theta_3)$ et $z_{\theta_1}(\theta_2, \theta_3)$ sont indépendants de θ_1 . Complémentairement, l'application $(\theta_2, \theta_3) \mapsto (\rho_{\theta_1}, 0, z_{\theta_1})$ décrit aussi un générateur de l'espace de travail de la fonction G et, comme G_{θ_1} admet un lieu singulier générant S_G par ajout d'une composante associée à l'angle θ_1 .

De ce fait, il suffit de considérer un angle θ_1 fixé et d'étudier $f_{\theta_1} : (\theta_2, \theta_3) \mapsto (\sqrt{x_{\theta_1}^2 + y_{\theta_1}^2}, z_{\theta_1})$ pour caractériser la fonction G , initialement prise en compte pour caractériser le robot positionneur 3R.

Par la suite, on considère $\theta_1 = 0$ et on pose $f = (f_\rho, f_z) = f_0 : (\theta_2, \theta_3) \mapsto (\sqrt{x_0^2 + y_0^2}, z_0)$.

Explicitement on a donc :

$$f(\theta_2, \theta_3) = (f_\rho(\theta_2, \theta_3), f_z(\theta_2, \theta_3))$$

où :

$$\begin{aligned} f_\rho(\theta_2, \theta_3) &= \sqrt{(\cos \theta_2 (d_4 \cos \theta_3 + d_3) + d_2 - r_3 \sin \theta_2)^2 + (d_4 \sin \theta_3 + r_2)^2} \\ f_z(\theta_2, \theta_3) &= \sin \theta_2 (d_4 \cos \theta_3 + d_3) + r_3 \cos \theta_2 \end{aligned}$$

De plus, on notera que f est périodique, de période 2π , selon θ_2 et θ_3 . De ce fait, il est suffisant d'étudier f en tant que fonction du tore dans $\mathbb{R}^2$, en paramétrant le tore par un couple d'intervalles, tout deux de taille 2π .

Remarque 4.3.1.4 (Identification pratique)

Pour la simplicité de l'implémentation on identifiera, en pratique, à la fonction f , la fonction (f_{ρ^2}, f_z)

où $f_{\rho^2}(\theta_2, \theta_3) = (f_\rho(\theta_2, \theta_3))^2 = (\cos \theta_2 (d_4 \cos \theta_3 + d_3) + d_2 - r_3 \sin \theta_2)^2 + (d_4 \sin \theta_3 + r_2)^2$, afin de s'affranchir de la racine carrée.

De même, on considèrera f en tant que fonction 2π périodique, de $\mathbb{R}^2$ dans $\mathbb{R}^2$, plutôt qu'en tant que fonction du tore vers $\mathbb{R}^2$, afin de considérer la caractérisation simple, proposée au § 2.3.2.1, des intervalles de l'espace $\mathbb{R}^2$.

Systèmes d'équations caractérisant les cusps et les nodes articulaires

Le point principal à souligner est que :

- Les cusps articulaires de f sont les points de $\mathcal{S}_{1,1}(f)$ isolés.
- Les nodes articulaires sont les paires d'éléments tels que les couples que celles-ci induisent sont des éléments isolés de l'ensemble $\{(M_1, M_2) \in (S_f)^2 \mid f(M_1) = f(M_2) \text{ et } M_1 \neq M_2\}$

Ainsi, on peut déduire, de la Proposition 4.3.1.2 et de la Proposition 4.3.1.3, des équations explicites caractérisant les cusps et les nodes articulaires.

Application aux cusps

Un point cusp articulaire M étant un point de $\mathcal{S}_{1,1}(f)$, d'après la Proposition 4.3.1.2, on a donc :

Si $Df(M) \neq 0$ et si $n(M) = (\nabla(\det(Df)))(M) \neq 0$. Soit $w(M) \neq 0$, tel que $w(M) \cdot n(M) = 0$ alors :

$$\begin{cases} Df_1(M) \cdot w(M) = 0 \\ Df_2(M) \cdot w(M) = 0 \end{cases} \quad \text{où } Df(M) = \begin{pmatrix} Df_1(M) \\ Df_2(M) \end{pmatrix}$$

Or, les vecteurs $Df_1(M)$ et $Df_2(M)$ s'identifient, respectivement, à $D(f_\rho)(M)$ et à $D(f_z)(M)$, si on considère $f = (f_\rho, f_z)$. (Si on pose $f = (f_{\rho^2}, f_z)$ alors $Df_1(M)$ s'identifie à $D(f_{\rho^2})(M)$.)

De plus, dans $\mathbb{R}^2$, si $(0,0) \neq (u_1, u_2)$ alors on a $(0,0) \neq (-u_2, u_1)$ et $(-u_2, u_1) \cdot (u_1, u_2) = (0,0)$.

Si on considère la fonction d'intérêt $f = (f_\rho, f_z)$, on obtient donc que :

Si $Df(M) \neq 0$ et si $(\nabla(\det(Df)))(M) = \left(\frac{\partial \det(Df)}{\partial \theta_2}(M), \frac{\partial \det(Df)}{\partial \theta_3}(M) \right) \neq 0_{\mathbb{R}^2}$, alors :

$$\begin{array}{lcl} M \text{ est un} & & \\ \text{point cusp} & \Leftrightarrow & \\ \text{articulaire} & & M \text{ est une} \\ & & \text{racine} \\ & & \text{isolée de} \end{array} \begin{cases} \frac{\partial f_\rho}{\partial \theta_2}(M) \left(-\frac{\partial \det(Df)}{\partial \theta_3}(M) \right) + \frac{\partial f_\rho}{\partial \theta_3}(M) \left(\frac{\partial \det(Df)}{\partial \theta_2}(M) \right) = 0 \\ \frac{\partial f_z}{\partial \theta_2}(M) \left(-\frac{\partial \det(Df)}{\partial \theta_3}(M) \right) + \frac{\partial f_z}{\partial \theta_3}(M) \left(\frac{\partial \det(Df)}{\partial \theta_2}(M) \right) = 0 \end{cases} \quad (e)$$

Application aux nodes

Un node étant décrit par un élément isolé dans $\{(M_1, M_2) \in (S_f)^2 \mid f(M_1) = f(M_2) \text{ et } M_1 \neq M_2\}$, d'après la Proposition 4.3.1.3, un node articulaire vérifie donc :

$$\begin{cases} f(M_1) = f(M_2) \\ \det(Df(M_1)) = 0 \\ \det(Df(M_2)) = 0 \end{cases}$$

En considérant $f = (f_\rho, f_z)$, on obtient donc que :

$$\begin{array}{lcl} \{M_1, M_2\} \text{ est} & & \\ \text{un node} & \Leftrightarrow & \{M_1, M_2\} \text{ est} \\ \text{articulaire} & & \text{une racine} \\ & & \text{isolée de} \end{array} \begin{cases} f_\rho(M_1) = f_\rho(M_2) \\ f_z(M_1) = f_z(M_2) \\ \det(Df(M_1)) = 0 \\ \det(Df(M_2)) = 0 \end{cases} \quad (f)$$

Mouvements internes

Un mouvement interne est défini par un continuum $\mathcal{I}$ de points articulaires admettant la même image.

Nous notons, premièrement, que $\forall M \in \mathcal{I}, \forall u \in \mathbb{T}_M \mathcal{I} \supseteq \mathbb{T}_M \mathcal{S}_f, Df(M) \times u = 0$.

De ce fait, si $\mathcal{I} \subseteq \mathcal{S}_1(f)$ alors la propriété précédente correspond à la caractérisation d'un point de l'ensemble $\mathcal{S}_{1,1}(f)$. Dans ce cas, $\mathcal{I}$ est donc un sous-ensemble continu de $\mathcal{S}_{1,1}(f)$.

Ceci implique que tout point M de $\mathcal{I}$ vérifie le système d'équations caractérisant un cusp articulaire.

On remarque que $\forall (M_1, M_2) \in \mathcal{I}^2, \begin{cases} f(M_1) = f(M_2) \\ \det(Df(M_1)) = 0 \\ \det(Df(M_2)) = 0 \end{cases}$

Cette remarque implique que toute paire de points $\{M_1, M_2\}$ telle que $M_1 \in \mathcal{I}$ et $M_2 \in \mathcal{I}$, vérifie le système d'équations caractérisant un node articulaire.

On conclut donc que tout point d'une courbe associée à un mouvement interne, vérifie le système d'équations caractérisant les cusps articulaires (de même toute paire de points de cette courbe vérifie le système d'équation caractérisant les nodes articulaire) tout en n'étant ni un cusp ni un node, puisque la propriété d'être une racine isolée n'est pas vérifiée.

Pour un robot positionneur 3R, un mouvement interne se produit lorsque le point effecteur P se situe dans l'axe de l'une des liaisons pivot (en pratique, l'axe de la rotation d'angle θ_2).

4.3.2. Détection Pratique (et performances des algorithmes)

4.3.2.1. Algorithmes d'encadrement proposés

Principe général des algorithmes

Selon les équations (e) et (f), les cusps et les nodes articulaires sont des solutions isolées de systèmes carrés. Nous nous proposons donc de construire des algorithmes basés sur la méthode de Newton par Intervalles, présentée au § 2.3.4.2, afin de détecter et d'encadrer ces points d'intérêt.

Toutefois, les deux algorithmes proposés sont implémentés de façon à gérer les situations, pour lesquelles les équations (e) et (f) ne peuvent pas être appliquées ou ne peuvent pas être traitées directement par la méthode de Newton par Intervalles. De même, les implémentations proposées prennent en compte certaines propriétés qui permettent d'optimiser l'efficacité d'exécution des algorithmes.

En pratique, les algorithmes proposés retournent chacun deux listes d'intervalles $\mathcal{R}$ et $\mathcal{U}$.

Ces deux listes vérifient que $\mathcal{R} \cup \mathcal{U}$ est un recouvrement de l'ensemble des points d'intérêt et que chaque intervalle de $\mathcal{R}$ contient un unique point d'intérêt.

Encadrement des cusps

Premièrement, nous notons que tout point cusp articulaire M est un point singulier (i.e. $M \in S_f$).

On en conclut que tout point cusp vérifie $\det(Df(M)) = 0$.

L'algorithme encadrant les cusps articulaires commence donc par tester si l'intervalle étudié I vérifie $0 \in \det(Df(I))$. L'intervalle I est alors traité uniquement si le test est positif.

L'équivalence de l'équation (e) n'est valide que lorsque les points M considérés vérifient, à la fois, $Df(M) \neq 0_{2 \times 2}$ et $[\nabla(\det(Df))](M) \neq 0$.

Or, si $Df(M) = 0_{2 \times 2}$ alors $M \notin S_1(f)$, impliquant que M ne peut être un cusp articulaire.

De plus, en se reportant à la Remarque 4.3.1.1, la propriété $(\nabla(\det(Df)))(M) = 0$ implique que :

$$\ll M \in S_f \gg \Leftrightarrow \ll \text{Il existe un voisinage } V \text{ de } M \text{ tel que } V \subseteq S_g \gg$$

Toutefois, si on considère des fonctions d'inclusion, il est nécessaire de préciser que :

$$0 \in [\nabla(\det(Df))](I) \not\Rightarrow \exists M \in I \text{ tel que } (\nabla(\det(Df)))(M) = 0$$

De ce fait, l'algorithme proposé teste donc si $0_{2 \times 2} \in Df(I)$ ou si $0 \in [\nabla(\det(Df))](I)$, pour chaque intervalle étudié I . Ensuite, dans le cas où l'un des deux tests d'appartenances est positif, l'algorithme omet le test élémentaire de Newton par Intervalles, qui n'est alors pas pertinent pour conclure, puis subdivise directement I en nouveaux intervalles à étudier.

L'algorithme encadrant les cusps s'identifie donc à une version modifiée de la méthode de Newton par Intervalles, proposée à la Figure 6, où la fonction d'intérêt est celle du système de l'équation (e).

Les modifications consistent en :

- Les listes d'intervalles retournées sont :
 - La liste $\mathcal{R}$ qui reste inchangée
 - La liste $\mathcal{U}$ contenant les intervalles dans $\mathcal{U}$ qui sont absents de $\mathcal{R}$
- L'ajout de tests, dans l'ordre, dans la boucle principale (où I est l'intervalle étudié) :
 - « $0 \in \det(Df(I))$? » : Si « Non » la boucle principale est omise
 - « $0_{2 \times 2} \in Df(I)$? » : Si « Oui » le test élémentaire de la méthode est omis
 - « $0 \in [\nabla(\det(Df))](I)$? » : Si « Oui » le test élémentaire de la méthode est omis

Encadrement des nodes

Le système de l'équation (f) est carré, avec 4 variables articulaires pour 4 équations indépendantes, lorsque $(M_1, M_2) \in \mathfrak{N}_2(\mathbb{R}^2)$ (i.e. lorsque $M_1 \neq M_2$).

On en déduit qu'il est possible d'appliquer la méthode de Newton par Intervalles, associée au système définissant les nodes, pour tout intervalle d'étude $I = I_a \times I_b$, tel que $(I_a, I_b) \in (\mathbb{IR}^2)^2$ et tel que $I_a \cap I_b = \emptyset$, afin de détecter les couples de points associés à un node articulaire, dans (I_a, I_b) .

La clé de voute de la recherche garantie des nodes à l'aide de cette méthode de Newton par Intervalles consiste donc à vérifier l'absence de nodes articulaires, dans (I_a, I_b) , lorsque $I_a \cap I_b \neq \emptyset$.

On remarque toutefois que, en dehors de $S_f \times S_f$, un intervalle d'étude I ne peut pas contenir de solution au système d'équations définissant un node.

En nous appuyant sur les propositions relatives à l'injectivité des fonctions de $\mathbb{R}^2$ dans $\mathbb{R}^2$, énoncées dans (Delanoue, et al., 2014), nous sommes en mesure de définir deux conditions qui impliquent l'injectivité de $f|_{S_f \cap B}$ où $B \in \mathbb{IR}^2$.

En effet, soit $B \in \mathbb{R}^2$, alors $f|_{S_f \cap B}$ est injective si l'une des deux propriétés suivantes est vraie :

- $0 \notin Df_1(B) \cdot w(B)$ ou $0 \notin Df_2(B) \cdot w(B)$ où :
 - $\forall M \in \mathbb{R}^2, w(M) \cdot n(M) = 0$, pour $n(M) = \left(\nabla(\det(Df)) \right)(M)$
- $\exists B_0 \in \mathbb{R}^2$, tel que $0 \notin \alpha(B_0) \cdot \mathfrak{x}(B)$ où :
 - $\exists M_0 \in B_0 \cap B$ tel que M_0 est un point cusp articulaire
 - $\forall M \in \mathbb{R}^2, \alpha(M) \cdot (Df(M) \times n(M)) = 0$, pour $n(M) = \left(\nabla(\det(Df)) \right)(M)$
 - $\forall M \in \mathbb{R}^2, \mathfrak{x}(M) = \frac{\partial^3 f}{\partial w(M)^3}(M)$, pour $w(M)$, tel que $w(M) \cdot n(M) = 0$

Remarque 4.3.2.1 (Nécessité des deux tests d'injectivité)

La première propriété « $(0 \notin Df_1(B) \cdot w(B) \text{ ou } 0 \notin Df_2(B) \cdot w(B)) \Rightarrow f|_{S_f \cap B}$ est injective » est insuffisante pour conclure totalement sur l'injectivité de $f|_{S_f}$ de par l'équation (e).

Ainsi, puisque $\begin{cases} Df_1(M) \cdot w(M) = 0 \\ Df_2(M) \cdot w(M) = 0 \end{cases}$ pour tout point cusp articulaire M , il est pertinent de compléter cette première propriété d'injectivité par une seconde, valide au voisinage des points cusps.

Toutefois, la complexité formelle d'un test d'injectivité associé à la seconde propriété, nous incite à favoriser l'appel à la première propriété, avant de tester la présence d'un cusp dans B , pour être en mesure d'appliquer la seconde propriété.

Soit le processus qui propose, pour un intervalle d'étude $I = I_a \times I_b$ tel que $(I_a, I_b) \in (\mathbb{R}^2)^2$:

- Si $I_a \cap I_b = \emptyset$ alors :
 1. On applique le test élémentaire de Newton par Intervalles à I
 - Si le test conclut à une racine unique, I est ajouté aux intervalles d'intérêt
 - Si le test conclut à l'absence de solutions, I est retiré des intervalles d'étude
 2. Si le test ne permet pas de conclure sur l'existence d'une racine sur I
 - Si la taille limite est atteinte, I est ajouté aux intervalles indéterminés
 - Sinon, I est subdivisé en plusieurs sous-intervalles à traiter
- Si $I_a \cap I_b \neq \emptyset$ alors :
 1. On teste l'injectivité de $f|_{S_f \cap (I_a \cup I_b)}$
 - Si $\langle I_a \cup I_b \rangle$ ne contient pas de cusps, on teste si :
 - $0 \notin Df_1(B) \cdot w(B)$
 - $0 \notin Df_2(B) \cdot w(B)$
 - Si $\langle I_a \cup I_b \rangle$ contient un cusp $M_0 \in B_0$, on teste si $0 \notin \alpha(B_0) \cdot \mathfrak{x}(B)$
 2. Si $f|_{S_f \cap (I_a \cup I_b)}$ est injective alors I est retiré des intervalles d'étude
 3. Si le test ne permet pas de conclure sur l'injectivité de $f|_{S_f \cap (I_a \cup I_b)}$
 - Si la taille limite est atteinte, I est ajouté aux intervalles indéterminés
 - Sinon, I est subdivisé en plusieurs sous-intervalles à traiter

Ce processus trivial permet de détecter les nodes articulaires mais présente le défaut d'être très consommateur de temps, du fait des tests d'injectivité répétés pour chaque intervalle d'étude tel que les deux composantes dont il est le produit ne sont pas étrangères.

Nous proposons donc de séparer la recherche des nodes en deux sous-étapes successives :

1. Construire un recouvrement R , par des intervalles, de S_f tel que (idéalement) :

$$\forall (I_a, I_b) \in (R)^2, (I_a \cap I_b \neq \emptyset) \Rightarrow (f|_{S_f \cap (I_a \cup I_b)} \text{ est injective})$$
2. Appliquer la méthode de Newton par Intervalles, à chaque intervalle $I = I_a \times I_b$ tel que $(I_a, I_b) \in (R)^2$ et $I_a \cap I_b = \emptyset$ (en omettant $I_b \times I_a$ si $I_a \times I_b$ est déjà traité).

On notera que, lorsqu'il est constructible, R est un recouvrement garanti du lieu singulier S_f (restreint à l'espace décrit par les intervalles initiaux d'étude).

Méthode de construction du recouvrement R

Le recouvrement R est, en pratique, construit en deux phases :

- Construction d'un recouvrement R_1 , tel que $\forall I \in R_1, f|_{S_f \cap I}$ est injective, par :
 - Subdivision de chaque intervalles d'étude I jusqu'à obtenir, alternativement :
 - $0 \notin [\det(Df)](I)$
 - $f|_{S_f \cap I}$ est injective d'après l'un des deux tests appliqué à I
 - $\omega(I)$ est inférieure à la taille limite ε choisie
 - On pose $R_1 = \{I \mid f|_{S_f \cap I} \text{ est injective}\}$ et $R_1' = \{I \mid \omega(I) < \varepsilon\}$
- Construction d'un recouvrement R_2 , tel que

$$\forall (I_a, I_b) \in (R)^2, (I_a \cap I_b \neq \emptyset) \Rightarrow (f|_{S_f \cap (I_a \cup I_b)} \text{ est injective})$$
 réalisé par :
 - Définition d'une liste d'intervalles $R_0 = R_1$
 - Subdivision de chaque intervalle I dans R_0 jusqu'à ce que, alternativement :
 - $0 \notin [\det(Df)](I)$
 - $\omega(I)$ est inférieure à la taille limite ε choisie
 - Pour tout intervalle $I_0 \in R_0$ tel que $I \cap I_0 \neq \emptyset$, $f|_{S_f \cap (I \cup I_0)}$ est injective d'après l'un des deux tests d'injectivité appliqué à $\langle I \cup I_0 \rangle$.
 - On pose
 - $R_2 = \{IR_0 \mid \forall I_0 \in R_0, f|_{S_f \cap (I \cup I_0)} \text{ est injective, si } I \cap I_0 \neq \emptyset\}$
 - $R_2' = \{I \in R_0 \mid \omega(I) < \varepsilon\}$
- On définit alors $R = R_2 \cup R_2' \cup R_1'$

Lorsque le recouvrement R de S_f ne peut être construit de telle sorte que :

$$\forall (I_a, I_b) \in (R)^2, (I_a \cap I_b \neq \emptyset) \Rightarrow (f|_{S_f \cap (I_a \cup I_b)} \text{ est injective})$$

(i.e. $R_2' \cup R_1' \neq \emptyset$ ce qui implique que $R \neq R_2$)

On considère alors uniquement le recouvrement R_2 à la place du recouvrement complet R , de S_f , afin d'appliquer la méthode de Newton par Intervalles associée à la recherche de nodes.

(On déduit alors un recouvrement de l'ensemble des nodes articulaires en ajoutant, aux intervalles encadrant les nodes détectés, les couples d'intervalles (I_a, I_b) tels que $I_a \in (R_2' \cup R_1')$ et $I_b \in R$.)

Situation particulière des mouvements internes

Lorsque le robot associé à un fonction f admet un mouvement interne associé à la courbe $\mathcal{I}$, dans l'espace articulaire, d'après la définition d'un mouvement interne au § 4.3.1.3 on vérifie que :

- L'algorithme de détection des cusps articulaires retourne une liste d'intervalles $\bar{U}$ encadrant $\mathcal{I}$.
- La liste d'intervalles R_1' retournée par l'algorithme de détection des nodes articulaires encadre la courbe $\mathcal{I}$ associée au mouvement interne.

Ainsi, sur tout intervalle non étranger à une courbe $\mathcal{I}$ associée à un mouvement interne, chacun des deux algorithmes de détection n'est pas en mesure de conclure.

4.3.2.2. Implémenter et exécuter les algorithmes d'encadrement

On se propose d'implémenter, en C++ (version 11), les algorithmes définis au § 4.3.2.1 et de les appliquer aux fonctions géométriques simplifiées f , décrivant les manipulateurs 3R considérés, formés par des solides reliés par trois liaisons pivots d'axes mutuellement orthogonaux.

Pour cela, il est nécessaire de calculer, au préalable et en s'appuyant sur l'expression formelle de f , toutes les expressions formelles des fonctions nécessaires aux algorithmes proposés.

Les algorithmes s'appuyant sur l'Analyse par Intervalles, ceux-ci évaluent les expressions nécessaires sur chaque intervalle étudié, substituant aux fonctions naturelles des fonctions d'inclusion de celles-ci. Afin de manipuler des intervalles et d'effectuer des opérations sur ceux-ci, les algorithmes font appel à la librairie de calcul sur les intervalles « Filib++ » (Filib++ version 3.02) disponible, de même que sa documentation, à l'adresse <http://www2.math.uni-wuppertal.de/org/WRST/software/filib.html>.

On notera, dès à présent, que les algorithmes d'encadrement, proposés au § 4.3.2.1, sont applicables quels que soient les paramètres géométriques de la fonction d'intérêt considérée. Toutefois, le comportement de la fonction variant selon ses paramètres géométriques, les performances des algorithmes varient, elles aussi, selon ces mêmes paramètres.

Ainsi, l'application des algorithmes à des manipulateurs 3R admettant $\beta_2 \neq 0$ ou $\beta_3 \neq 0$ est possible, après calcul de la fonction géométrique générale d'un manipulateur 3R.

Toutefois, la complexité formelle des expressions nécessaires aux algorithmes augmente dans ce cas. Or, avec cette augmentation de complexité formelle, les temps d'exécution des algorithmes peuvent augmenter significativement. De plus, les précisions nécessaires à l'encadrement des points d'intérêt peuvent être significativement plus fines, dans le cas d'une telle augmentation de complexité formelle.

Les algorithmes implémentés admettent toute liste initiale d'intervalles d'études encadrant des couples d'angles $(\theta_2, \theta_3) \in \mathbb{R}^2$. Ceci permet notamment de considérer la périodicité de la fonction d'intérêt.

Complémentairement, les paramètres de la fonction d'intérêt f peuvent être définis comme étant les éléments de tout intervalle, représentable par l'ordinateur.

De plus, les algorithmes proposés peuvent être combinés à une procédure encadrant l'espace articulaire utilisable par le robot, comme proposée au § 4.3.2.3.

Une telle procédure permet de prendre en compte certaines butées articulaires et, à l'aide d'un modèle volumétrique simple, d'encadrer les régions articulaires n'induisant pas de collision.

Cet encadrement, par des intervalles, de l'espace articulaire accessible peut, notamment, être choisi comme liste initiale d'intervalles d'étude.

La Table 2 présente les résultats de l'exécution des algorithmes appliqués aux fonctions géométriques simplifiées de robots 3R orthogonal positionneurs, dont les paramètres constitutifs d_2, d_3, d_4, r_2 et r_3 sont inclus dans les intervalles proposés à la Table 1, des caractéristiques de ces manipulateurs.

Les résultats de la Table 2 sont proposés pour une liste initiale d'intervalles d'études restreinte à l'intervalle $[-3.1415 ; 3.1415]^2$, proche de l'intervalle $[-\pi, \pi]$ qui est suffisant à l'étude de tous les couples d'angles (θ_2, θ_3) , étant donnée la 2π périodicité des fonctions d'intérêt considérées.

Désignation	Paramètres géométriques					Propriétés du manipulateur		
	d_2	d_3	d_4	r_2	r_3	mouvements internes?	Cusps	Nodes
a	1	2	1.5	1	0	Non	4	0
b	1	2	1.5	1	0.5	Non	4	0
c	[1,1.001]	[2,2.001]	[1.5,1.501]	[1,1.001]	0	ND	4	ND
d	1	$\langle 0.7 \rangle_{\mathbb{IR}}$	$\langle 0.3 \rangle_{\mathbb{IR}}$	$\langle 0.2 \rangle_{\mathbb{IR}}$	0	Non	0	0
e	1	1.5	$\langle 0.7 \rangle_{\mathbb{IR}}$	0.5	0	Non	4	2
f	1	0.5	$\langle 1.3 \rangle_{\mathbb{IR}}$	$\langle 0.2 \rangle_{\mathbb{IR}}$	0	Oui	0	2

Table 1 : Exemples de manipulateurs robotiques étudiés (ND : Non Déterminé)

cas	Algorithme des cusps				Algorithme des nodes				
	précision	cusps	indéterminés	temps	précision	nodes	indéterminés	temps	
								sans Ibex	avec Ibex
a	10^{-4}	4	Non	32 s	2.5×10^{-10}	0	Non	10 h	23 min
b	10^{-4}	4	Non	46 s	2.5×10^{-10}	0	Non	18 h	45 min
c	10^{-4}	4	Non	35 s	2.5×10^{-10}	NA	Oui	NA	Mémoire dépassée
d	10^{-4}	0	Non	12 s	2.5×10^{-10}	0	Non	52 s	16 s
e	10^{-4}	4	Non	52 s	2.5×10^{-10}	2	Non	35 h	5h et 42 min
f	10^{-2}	0	Oui	12 min	10^{-2}	2	Oui	42 s	42 s
f	10^{-3}	0	Oui	90 min	10^{-3}	2	Oui	41 s	41 s

Table 2 : Performances des algorithmes pour les manipulateurs robotiques de la Table 1 (NA : l'algorithme n'a pas abouti et on ne peut donc pas conclure)

Dans la Table 1, $\langle p \rangle_{\mathbb{IR}}$ désigne $\{p\}_{\mathbb{IR}}$ tel que introduit à la Définition 2.3.1.5, où $\mathbb{IR}$ est l'ensemble d'intervalles construit au § 2.3.2.1. Ainsi $\langle p \rangle_{\mathbb{IR}}$ représente l'un des plus petits intervalles, contenant la valeur p et dont les bornes sont représentables par l'ordinateur (i.e les bornes de $\langle p \rangle_{\mathbb{IR}}$ sont des éléments de $\mathbb{IR}$, où l'index considéré est celui du codage standard du type informatique « double »). Les temps d'exécution présentés dans la Table 2 sont obtenus pour des exécutions réalisées sur un ordinateur utilisant un système d'exploitation Debian version 7.8, 64 bits et un CPU Intel® Core™ i7 (2.90 GHz).

Performances de l'algorithme encadrant les cusps

Manipulateurs n'induisant pas d'indéterminés (cas « a », « b », « d » et « e » de la Table 1.) :

L'algorithme a été appliqué à chaque exemple d'ensemble de paramètres géométriques constitutifs proposés dans (Baili, et al., 2004).

On en déduit que, lorsque le manipulateur n'admet pas de mouvement interne, et si la précision choisie est modérément fine, alors l'algorithme nécessite peu de temps d'exécution pour obtenir des encadrements garantis des cusps, tout en ne retournant aucun intervalle indéterminé.

Manipulateurs induisant des indéterminés (cas « f » de la Table 1) :

Lorsqu'il est appliqué à un robot admettant des mouvements internes, l'algorithme est en mesure de détecter les cusps, en dehors des mouvements internes, en un temps d'exécution comparable à ceux observés pour l'application aux manipulateurs sans mouvement interne.

Toutefois, le temps d'exécution effectif de l'algorithme est bien supérieur à ce temps de détection. Ceci s'explique par le fait que l'algorithme est implémenté de telle sorte qu'il subdivise, en amont, les intervalles étudiés jusqu'à ce que leurs tailles soient majorées par la précision choisie.

L'algorithme n'étant pas en mesure de conclure sur la nature des points, le long des mouvements internes, les intervalles à leur voisinage sont donc subdivisés jusqu'à la précision choisie, avant d'être qualifiés d'intervalles indéterminés. De ce fait, le temps d'exécution est dépendant de la précision.

Performances de l'algorithme encadrant les nodes

Si chaque intervalle étudié, ne contient ni cusp ni mouvement interne (par exemple dans le cas « d » de la Table 1) l'algorithme encadrant les nodes conclut après un temps d'exécution proche de celui nécessaire à la détection d'un cusp, par l'algorithme consacré, sur un intervalle qui n'admet pas d'intersection avec une courbe associée à un mouvement interne.

Toutefois, lorsqu'un intervalle contient un cusp (ce qui est possible pour les cas « a », « b » et « e » de la Table 1) le temps d'exécution de l'algorithme augmente très significativement, puisque, au voisinage des cusps, l'injectivité de la fonction f , restreinte à S_f , ne peut être vérifiée que sur des intervalles de petites tailles.

Similairement, soit I un intervalle étudié tel que $\mathbb{R}^4 \supseteq I = I_a \times I_b$ et tel que $(I_a, I_b) \in (\mathbb{IR}^2)^2$.

On remarque que, si $\langle I_a \cup I_b \rangle$ est proche d'un point cusp, alors la méthode de Newton par Intervalles n'est en mesure de conclure que lorsque les tailles $w(I_a)$ et $w(I_b)$ sont relativement petites.

Amélioration des performances à l'aide de méthodes de contraction

Comme cela a déjà été mis en lumière, l'inconvénient principal de l'algorithme proposé est la vérification relativement lente de l'absence de node près des cusps.

Afin de palier à cet inconvénient, nous avons décidé de faire appel à la librairie de méthodes de contraction d'intervalles nommée *Ibex* (ibex version 2.1.10) disponible gratuitement, avec sa documentation à l'adresse <http://www.ibex-lib.org/>.

Un *contracteur* est un opérateur sur les intervalles, associé à un ensemble, qui réduit l'intervalle étudié à un intervalle de plus petite taille, sans retirer d'élément de l'ensemble associé à l'intervalle initial.

Une *méthode de contraction*, en Analyse par Intervalles, est une méthode d'encadrement d'ensemble, s'appuyant sur des contracteurs, associés à l'ensemble d'intérêt, et sur des subdivisions éventuelles.

L'intérêt principal d'une telle méthode réside dans le fait que la réduction d'un intervalle, à l'aide de contracteurs, est beaucoup moins consommatrice de temps que l'application successive de tests et de subdivisions, uniquement, jusqu'à la précision limite choisie.

L'une des procédures de contraction, proposées par la librairie *Ibex*, est incluse dans l'algorithme en tant que test additionnel avant l'application d'une itération de la méthode de Newton par Intervalles pour les nodes, sur chaque intervalle d'étude (i.e. un intervalle de dimension 4 associé à un couple d'intervalles de dimensions 2, disjoints).

La procédure utilisée s'appuie sur un contracteur, basé sur une méthode de Newton par Intervalles, et qui admet comme paramètre le système, définissant un couple de points associés à un node.

Le contracteur de la procédure *Ibex* réduisant plutôt efficacement les intervalles étudiés, nous utilisons cette procédure afin de vérifier, rapidement, l'absence de node dans chaque intervalle étudié. En effet, si la procédure retourne un intervalle vide, alors il n'y a pas de node dans l'intervalle étudié initial. Ainsi, cet intervalle d'étude peut être immédiatement omis pour la recherche et la détection de nodes. Ceci permet donc d'éviter des subdivisions et des tests élémentaires de Newton par Intervalles qui s'avèreraient n'aboutir à aucune détection, réduisant ainsi le temps d'exécution global de l'algorithme.

En conséquence de l'inclusion du test utilisant la librairie Ibex, dans l'étape de recherche des nodes, les performances de l'algorithme, vis-à-vis de la rapidité de la vérification d'absence de nodes, ont été grandement améliorées. Par exemple, le temps nécessaire à l'exécution de l'algorithme de recherche des nodes, pour un manipulateur 3R admettant des nodes et des cusps, est *réduit jusqu'à moins d'un cinquième de sa valeur initiale* (dans le cas « e » de la Table 1).

Application avec des intervalles (non ponctuels) de paramètres géométriques constitutifs

Nos algorithmes ont été implémentés de façon à gérer des intervalles de paramètres géométriques. Pour considérer tout intervalle de paramètres non ponctuel (comme dans le cas « b » de la Table 1) il suffit donc de définir un intervalle de paramètres géométriques qui n'est pas restreint à un point.

Si l'un des deux algorithmes de détection trouve un intervalle solution, alors *pour tout choix de paramètres géométriques parmi l'intervalle de paramètres* il existe une unique solution (cusp ou node, respectivement, selon l'algorithme) dans l'intervalle solution trouvé.

Il n'y a pas de point d'intérêt, dans tout intervalle qui n'admet pas d'intersection avec, au moins, un intervalle solution ou un intervalle indéterminé, *pour tout choix de paramètres géométriques parmi l'intervalle choisi de paramètres*.

Finalement, *pour tout choix de paramètres géométriques parmi l'intervalle choisi de paramètres*, il ne peut exister des points d'intérêt que dans les intervalles solutions et les intervalles indéterminés.

Ainsi, lorsqu'un manipulateur admet des mouvements internes, les algorithmes retournent, tout de même, des encadrements des points d'intérêt. De plus, les algorithmes fournissent des encadrements d'un sous-ensemble des points d'intérêt, de telle sorte que, chaque intervalle de ces recouvrements contient un unique point d'intérêt, parmi le type de point d'intérêt associé au recouvrement considéré.

4.3.2.3. Détection de collisions à l'aide de l'Analyse par Intervalles

Une procédure complémentaire a été ajoutée à nos algorithmes, permettant à l'utilisateur d'avoir accès à un encadrement de l'ensemble des paramètres articulaires induisant des collisions ainsi qu'à un encadrement de l'ensemble des paramètres articulaires n'induisant aucune collision.

Modèle considéré

Les solides qui peuvent entrer en collision (que ce soient des éléments de la chaîne cinématique du manipulateur robotique ou des obstacles dans l'environnement) sont modélisés à l'aide d'objets oblongs $\mathcal{O}(\mathcal{Y}, r)$ paramétrés par un segment de trame $\mathcal{Y}$ et par un rayon de diffusion r .

$\mathcal{O}(\mathcal{Y}, r)$ est l'ensemble de tous les points, distants du segment $\mathcal{Y}$ d'au plus le rayon de diffusion r .

Ce modèle volumique, du robot et de son environnement, présente l'intérêt de se déduire et d'être caractérisé, trivialement, à partir des segments de trame du schéma cinématique du robot et de la donnée d'un rayon de diffusion, pour chaque solide du robot, décrivant le volume de ce solide associé.

Avec ce modèle, deux objets entrent en collision si et seulement si la distance, entre leurs segments de trames respectifs, est égale ou inférieure à la somme des deux rayons de diffusion.

Soient :

- $\mathcal{B}(M, r)$ la boule de centre M et de rayon r .
- $\mathcal{C}(Y, r)$ le cylindre d'axe Y et de rayon r

Comme présenté à la Figure 12, le volume $\mathcal{O}(\mathcal{Y}, r)$ peut être interprété, alternativement, comme :

- $\mathcal{O}(\mathcal{Y}, r) = \{\mathcal{B}(M, r) \mid M \in \mathcal{Y}\}$
- $\mathcal{O}(\mathcal{Y}, r) = \mathcal{C}(\mathcal{Y}, r) \cup \mathcal{B}(M_1, r) \cup \mathcal{B}(M_2, r)$ où M_1 et M_2 sont les deux extrémités du segment $\mathcal{Y}$

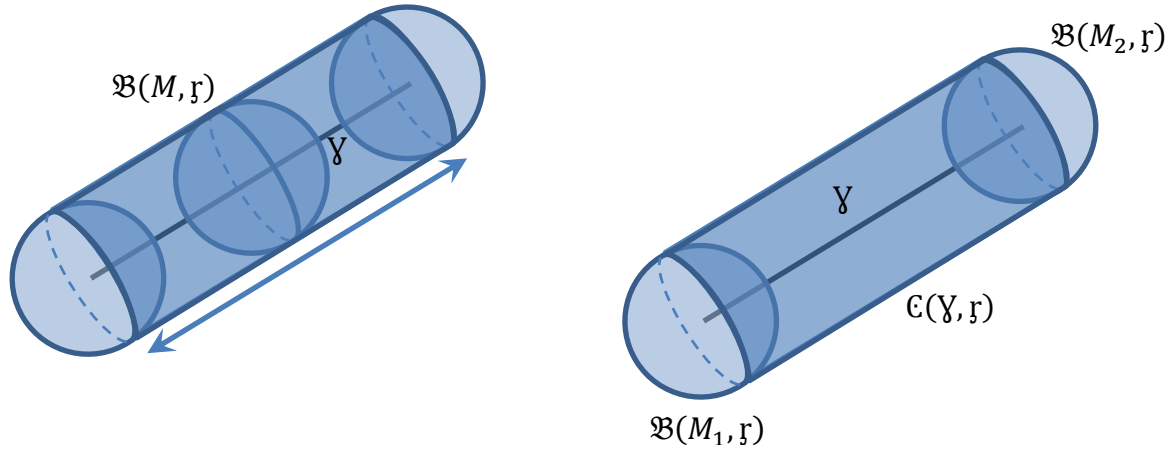


Figure 12 : Interprétations géométriques du modèle volumique de solide oblong $\mathcal{O}(Y, r)$

Procédure implémentée

La procédure implémentée est basée sur l'algorithme d'inversion ensembliste SIVIA (tel que présenté à la Figure 5) et consiste en son application aux distances entre chaque paire de segments de trame $\{Y_1, Y_2\}$ considérée. Puisque l'objectif est donc de calculer le minimum de la distance entre un point d'un des deux segments et un point de l'autre segment, les deux segments sont subdivisés jusqu'à une taille limite et la distance entre chaque paire de sous-segments est comparée à la somme des rayons des deux solides oblong associés aux segments considérés.

Puisque les distances varient avec les paramètres articulaires, le processus précédent est appliqué à des sous-intervalles de la liste initiale d'intervalles d'étude, des paramètres articulaires.

En résumé, la procédure proposée consiste à appliquer successivement, pour chaque paire de solides qui peuvent entrer en collision, définie préalablement par l'utilisateur, un double algorithme SIVIA.

Utilisation conjointe avec la détection de cusp et de node

L'intérêt de la procédure de détection de collision, est de contrôler, pour le manipulateur étudié, la faisabilité de chemins choisis dans l'espace articulaire. L'étape de préparation de la procédure d'encadrement des nodes fournit, quant à elle, un encadrement du lieu singulier. De ce fait, en combinant ces deux connaissances, il est possible de vérifier que deux postures peuvent être reliées par un chemin non singulier, dont la faisabilité est garantie.

L'utilisation conjointe de cette procédure supplémentaire avec l'algorithme principal de détection permet aussi, assez naturellement, de savoir si le manipulateur étudié peut atteindre les points d'intérêt choisis. Dans le cas que nous étudions, l'inaccessibilité des points cusps et des paires de points nodes, peut ne pas modifier le comportement du manipulateur considéré. Toutefois, par exemple, une large zone de collision, autour d'un cusp, peut impliquer que, en pratique, le manipulateur *n'est pas cuspidal*, si aucun chemin articulaire ne peut relier deux Solutions Géométriques Inverses, sans admettre, soit une intersection avec le lieu singulier, soit induire une collision.

4.3.3. Conclusion relative aux méthodes en dimension 2

La méthode proposée a pour intérêt principal d'être applicable à la recherche de tout type de point, d'intérêt pour l'analyse du comportement d'un manipulateur ou d'un système quelconque.

De plus, la méthodologie présentée permet, lorsque les points recherchés sont des racines isolées d'un système d'équations carré, de garantir l'existence et l'unicité de certaines de ces solutions isolées.

Cette méthodologie, s'appuyant sur l'Analyse par Intervalles, constitue donc une façon possible de décrire le lieu singulier d'un manipulateur robotique et permet une détection garantie de certaines configurations singulières isolées et d'intérêt.

Il est à noter que la plupart du temps d'exécution de l'algorithme total est consommé par le traitement des intervalles sur lesquels la procédure de Newton par Intervalles échoue à conclure. Pour améliorer les performances de l'algorithme proposé, l'utilisation de méthodes alternatives à la bisection des intervalles et de tests localisés semble nécessaire. Ces procédures alternatives sont encore à l'étude.

Tout comme de nombreux algorithmes basés sur l'Analyse par intervalles, celui que nous proposons peut être chronophage lorsque celui-ci traite des fonctions géométriques d'expressions compliquées ou des intervalles de dimensions élevées. Ceci est particulièrement le cas pour l'algorithme encadrant les nodes, du fait de la dimension doublée des intervalles d'étude, quoi que cet impact est atténué par une subdivision préparatoire dans l'espace articulaire.

Toutefois, formellement, l'algorithme « conclut en temps fini ». En d'autres termes, si l'algorithme est exécuté pendant le temps qui lui est nécessaire, avec une précision suffisante, celui-ci est en mesure de trouver, sans erreurs, soit :

- des encadrements pour les points recherchés,
- des encadrements pour les éléments d'un sous-ensemble des points recherchés ainsi qu'un recouvrement de l'ensemble des points recherchés.

En utilisation conjointe avec la procédure pour la collision, l'algorithme se propose de permettre un accès efficace et garanti à des éléments d'information pertinents vis-à-vis des propriétés intrinsèques du système étudié (i.e. les propriétés géométriques du manipulateur, pour l'exemple considéré).

L'algorithme serait en mesure de fournir des informations supplémentaires, pertinentes vis-à-vis des intérêts de l'utilisateur, au moyen de procédures additionnelles. On citera, par exemple, l'encadrement des positions singulières dans l'espace de travail, ainsi que l'encadrement des solutions géométriques inverses non singulières de ces positions singulières.

De ce fait nous vous proposons le code source de l'algorithme intégral à l'adresse http://perso-laris.univ-angers.fr/~benoitr/contenu/thom_2d_online.zip.

4.4 Application en dimension 3

4.4.1. Etude théorique

Nous considérons, dans cette section, les fonctions $f : A \rightarrow \mathbb{P}$, telles que $\dim(A) = \dim(\mathbb{P}) = 3$. Toutefois, l'étude théorique de cette partie se limite à l'étude des classes de singularités pertinentes. L'omission de l'étude de l'image est motivée par la complexité théorique, a priori, de cette étude. En contrepartie, l'étude théorique de l'espace source ouvre vers des résultats et des perspectives qui s'appliquent et restent pertinents en dimensions quelconques et supérieures à 3. On notera par ailleurs que les raisonnements étant similaires à ceux effectués pour la dimension 2, nous proposons de les reformuler vis-à-vis du § 4.3.1.1, afin de faciliter leur assimilation.

4.4.1.1. Classes de singularités et généricité considérées

Similairement à la Figure 9 et à la Figure 10, la Figure 13 introduit l'arbre des classes de singularités $\mathcal{S}_v$ considérées pertinentes. Ces ensembles s'avèrent être les uniques classes de singularités $\mathcal{S}_v$, non vides, pour une famille générique de fonctions que nous considérons dans cette section.

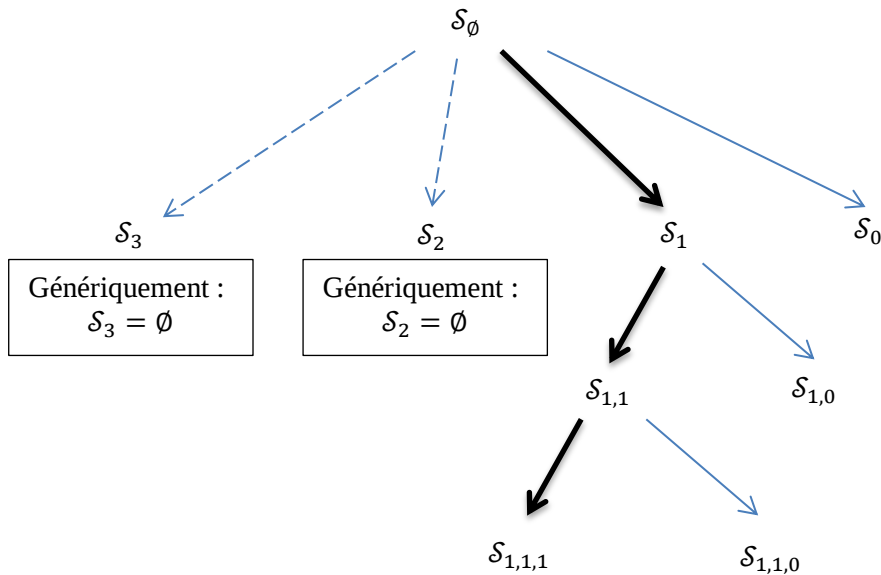


Figure 13 : Arbre des $\mathcal{S}_v$ pour une fonction entre espaces de dimensions 3

Proposition 4.4.1.1 (Les classes $\mathcal{S}_3(f)$ et $\mathcal{S}_2(f)$ sont génériquement vides)

Il existe un ensemble de fonctions E , résiduel pour $\mathcal{T}^\infty(A, \mathbb{P})$, tel que $\forall f \in E, \mathcal{S}_3(f) = \mathcal{S}_2(f) = \emptyset$.

Afin de prouver la Proposition 4.4.1.1, il est possible de s'appuyer sur la Proposition 4.4.1.2, ci-après. Toutefois cette proposition a seulement valeur de lemme, pour ce mémoire, lu sans ses références. En effet, cette affirmation est reformulée, à partir de la proposition 5.3 de (Golubitsky, et al., 1986) et est proposée sans démonstration.

Proposition 4.4.1.2 (Codimension de l'ensemble des applications linéaires de corang fixé)

Soit $L^r(X, Y) = \{h : X \rightarrow Y \mid h \text{ est linéaire et corang}(h) = r\}$

Soient $\dim(X) = m$, $\dim(Y) = n$ et $q = \min(m, n)$.

Alors $\text{codim}(L^r(X, Y)) = (m - q + r)(n - q + r)$

La Proposition 4.4.1.2 permet de distinguer automatiquement, pour chaque ensemble $\{f \mid f: X \rightarrow Y\}$, certaines classes de singularités $\mathcal{S}_r$ qui sont vides pour un sous-ensemble générique de $\{f \mid f: X \rightarrow Y\}$. En effet, en s'appuyant sur la Proposition 2.2.2.1, on obtient que, si $\text{codim}(L^r(X, Y)) > \dim(X)$, alors il existe un sous-ensemble résiduel E , de $\mathcal{T}^\infty(X, Y)$, tel que $\forall f \in E, \mathcal{S}_r(f) = \emptyset$.

On notera toutefois que l'utilité de la Proposition 4.4.1.2 s'illustre principalement lorsque l'on considère des fonctions entre espaces de dimensions élevées, car la caractérisation explicite des classes $\mathcal{S}_r$ devient plus difficile avec l'augmentation des dimensions. De plus il devient plus délicat de vérifier l'indépendance d'un ensemble d'équations lorsque leur nombre augmente ou que les espaces impliqués admettent une topologie plus riche.

Preuve 4.4.1.1 (de la Proposition 4.4.1.1)

En utilisant la Proposition 2.2.2.1, la Proposition 4.4.1.1 peut se démontrer en vérifiant que l'on a :

$$\text{codim}(R_3) > 3 \text{ et } \text{codim}(R_2) > 3$$

où R_3 est le système définissant la classe $\mathcal{S}_3$ et où le système R_2 définit la classe $\mathcal{S}_2$

Il est ainsi possible de démontrer la Proposition 4.4.1.1 de deux manières différentes :

- En déterminant la codimension de R_2 et de R_3 à l'aide de la Proposition 4.4.1.2
- En déterminant la codimension de R_2 et de R_3 à l'aide de leurs expressions explicites

Preuve à l'aide de la Proposition 4.4.1.2 :

Rappelons que $\forall k \in \llbracket 2; 3 \rrbracket$, R_k est le système décrivant une matrice Jacobienne de co-rang k .

La matrice Jacobienne Df , d'une fonction $f: A \rightarrow \mathcal{P}$ entre espace de dimension 3, est assimilable, en tout point $M \in A$, à une application linéaire entre espaces de dimension 3.

On en conclut donc que $\forall k \in \llbracket 2; 3 \rrbracket$, le système R_k décrit $L^k(A, \mathcal{P})$.

En appliquant la Proposition 4.4.1.2, on déduit donc que l'on a :

- $\text{codim}(R_3) = \text{codim}(L^3(A, \mathcal{P})) = 3^2 = 9 > 3$
- $\text{codim}(R_2) = \text{codim}(L^2(A, \mathcal{P})) = 2^2 = 4 > 3$

Ainsi on vérifie donc que l'on a $\text{codim}(R_3) > 3$ et $\text{codim}(R_2) > 3$.

Nécessité de présenter une seconde preuve :

La Proposition 4.4.1.2 ayant seulement valeur de lemme et n'ayant pas été prouvée dans ce mémoire, toute preuve s'appuyant sur cette proposition est incomplète, à la lecture de ce mémoire seul.

Preuve à l'aide d'expressions explicites :

Nous proposons donc de mettre en évidence des systèmes explicites de 4 équations ou plus, fonctionnellement indépendantes, qui caractérisent chacun des systèmes R_3 et R_2 . Comme annoncé, à la Proposition 2.2.2.1 cela implique que $\{f \mid \mathcal{S}_3(f) = \mathcal{S}_2(f) = \emptyset\}$ est résiduel pour $\mathcal{T}^\infty(A, \mathcal{P})$.

Nous rappelons que $M \in \mathcal{S}_3(f) \Leftrightarrow Df(M) = 0_{3 \times 3}$.

Ainsi, le système R_3 définissant la classe $\mathcal{S}_3$, est équivalent au système formé par les 9 équations, qui caractérisent la propriété $Df = 0_{3 \times 3}$, considérée en tant que relation sur le jet des fonctions $f: A \rightarrow \mathcal{P}$.

Ces équations étant formellement indépendantes, R_3 vérifie ainsi que $\text{codim}(R_3) = 9 > 3$.

La démonstration du fait que $\text{codim}(R_2) = 4$ est plus délicate, si l'on omet la Proposition 4.4.1.2.

Toutefois, le plan de la démonstration proposée est simple une fois celle-ci exposée.

Afin de simplifier la présentation de cette preuve, par la suite, nous désignerons par $\det_{i,j}(B)$, le déterminant de la matrice, obtenue en retirant la ligne i et la colonne j de la matrice B .

En utilisant la notation précédemment introduite, nous pouvons alors écrire :

$$M \in \mathcal{S}_2(f) \Leftrightarrow (Df(M) \neq 0_{3 \times 3} \text{ et } \forall (i,j) \in \llbracket 1; 3 \rrbracket^2, \det_{i,j}(Df(M)) = 0)$$

La codimension de R_2 est ainsi le nombre d'équations sur f , fonctionnellement indépendantes parmi l'ensemble $\{\det_{i,j}(Df) = 0 \mid (i,j) \in \llbracket 1; 3 \rrbracket^2\}$, lorsque $\forall M \in A, Df(M) \neq 0_{3 \times 3}$.

On a donc que : $\forall M \in \mathcal{S}_2(f), \exists (\alpha, \beta) \in \llbracket 1; 3 \rrbracket^2$ tel que $(Df(M))_{\alpha, \beta} \neq 0$

L'ensemble d'équations $\{\det_{i,j}(Df(M)) = 0 \mid (i,j) \in (\llbracket 1; 3 \rrbracket - \{\alpha\}) \times (\llbracket 1; 3 \rrbracket - \{\beta\})\}$ implique alors que toutes les lignes de $Df(M)$ sont proportionnelles, ce qui équivaut à vérifier :

$$\forall (i,j) \in \llbracket 1; 3 \rrbracket^2, \det_{i,j}(Df(M)) = 0$$

Or les équations de l'ensemble $\{\det_{i,j}(Df) = 0 \mid (i,j) \in (\llbracket 1; 3 \rrbracket - \{\alpha\}) \times (\llbracket 1; 3 \rrbracket - \{\beta\})\}$ sont fonctionnellement indépendantes.

On en conclut donc que :

$\forall M \in \mathcal{S}_2(f)$, le système engendré par $\{\det_{i,j}(Df(M)) = 0 \mid (i,j) \in \llbracket 1; 3 \rrbracket^2\}$ est de co-dimension 4.

Puisque le système R_2 est décrit par $\forall (i,j) \in \llbracket 1; 3 \rrbracket^2, \det_{i,j}(Df) = 0$ et par $\forall M \in A, Df(M) \neq 0_{3 \times 3}$, on en conclut donc que $\text{codim}(R_2) = 4 > 3$.

Nous avons ainsi explicitement montré que $\text{codim}(R_3) > 3$ puis que $\text{codim}(R_2) > 3$.

En nous appuyant sur la Proposition 2.2.2.1 nous concluons donc que les ensembles de fonctions $\{f \mid \mathcal{S}_3(f) = \emptyset\}$ et $\{f \mid \mathcal{S}_2(f) = \emptyset\}$ sont résiduels pour la topologie $\mathcal{T}^\infty(A, \mathcal{P})$.

Par conservation de la résidualité par l'intersection on en déduit ainsi que $\{f \mid \mathcal{S}_3(f) = \mathcal{S}_2(f) = \emptyset\}$ est résiduel pour $\mathcal{T}^\infty(A, \mathcal{P})$.

Remarque 4.4.1.1 (Non omission des fonctions f telles que $\mathcal{S}_3(f) \neq \emptyset$ ou $\mathcal{S}_2(f) \neq \emptyset$)

D'après la Proposition 4.4.1.1, il est valide de considérer une méthodologie orientée vers les fonctions de l'ensemble $\{f : A \rightarrow \mathcal{P} \mid \mathcal{S}_3(f) = \mathcal{S}_2(f) = \emptyset\}$ puisque celui-ci peut être qualifié de « générique ».

Toutefois, il paraît pertinent de ne pas imposer cette condition sur les fonctions, traitables par les algorithmes implémentés par la suite. En effet, vérifier qu'une fonction f admet $\mathcal{S}_3(f) = \mathcal{S}_2(f) = \emptyset$, requiert, en toute rigueur, d'effectuer une étude formelle préalable des singularités de f .

La nécessité d'une telle étude irait à l'encontre de l'intérêt d'une implémentation, telle que nous les proposons, utilisant l'Analyse par Intervalles, pour entre autres, être applicable à toute fonction d'expression formelle difficile à analyser.

De ce fait, bien que, par la suite, les équations mises en évidences seront valides, sous l'hypothèse que la fonction f étudiée vérifie $\mathcal{S}_3(f) = \mathcal{S}_2(f) = \emptyset$, les implémentations pratiques des algorithmes d'encadrement de $\mathcal{S}_1(f)$, de $\mathcal{S}_{1,1}(f)$ et de $\mathcal{S}_{1,1,1}(f)$ prendra en compte la possibilité que les équations proposées ne sont pas vérifiées.

Ainsi, toute fonction sera traitable par les algorithmes implémentés, bien que les fonctions génériques considérées soient traitées, en pratique, de façon plus fine.

4.4.1.2. Equations définissant $\mathcal{S}_1(f)$, $\mathcal{S}_{1,1}(f)$ et $\mathcal{S}_{1,1,1}(f)$

Afin d'isoler chaque ensemble d'intérêt $\mathcal{S}_v(f)$, un système d'équations le définissant est requis. Nous construirons ces systèmes d'équations récursivement, en admettant que l'application considérée $f : A \rightarrow \mathcal{P}$ est générique, selon la généricité considérée dans cette section.

Equation définissant $\mathcal{S}_1(f)$

Nous rappelons qu'une application générique f vérifie $\mathcal{S}_1(f) = \mathcal{S}_f$.

De ce fait, puisque $M \in \mathcal{S}_f \Leftrightarrow \det(Df(M)) = 0$, nous déduisons la Proposition 4.4.1.3.

Proposition 4.4.1.3 (Equation définissant génériquement $\mathcal{S}_1$)

Si $f : A \rightarrow \mathcal{P}$, telle que $\dim(A) = \dim(\mathcal{P}) = 3$, est une fonction générique alors $M \in \mathcal{S}_1(f)$ ssi :

$$\det(Df(M)) = 0 \quad (g)$$

Système d'équations définissant $\mathcal{S}_{1,1}(f)$

$\mathcal{S}_{1,1}(f)$ est défini comme étant un sous-ensemble de $\mathcal{S}_1(f)$ contenant les points $M = (x, y, z)$ vérifiant que l'espace tangent $\mathcal{T}_M \mathcal{S}_{1,1}(f)$ contient un unique ensemble $\mathcal{V}(w)$ tel que $0 \neq w \in \text{Ker}(Df(M))$.

L'ensemble $\mathcal{S}_1(f)$ étant défini par une unique équation (g), selon la Proposition 4.4.1.3, celui-ci est localement une surface, au point $M \in \mathcal{S}_1(f)$ ssi $n(M) = \nabla(\det(Df(M)))$ n'est pas le vecteur nul. Pour chacun de ces points $M \in \mathcal{S}_1(f)$, vérifiant $n(M) \neq 0$, $n(M)$ définit alors un vecteur normal à la courbe $\mathcal{S}_1(f)$, au point M .

Un vecteur $w(M)$ est alors dans l'espace $\mathcal{T}_M \mathcal{S}_{1,1}(f)$ ssi $w(M) \cdot n(M) = 0$, où l'opérateur $\cdot$ désigne un produit scalaire induit par une norme sur A (que l'on suppose donc orthonormé).

Notre objectif principal est donc de définir un vecteur $w(M)$ tel que $\text{Ker}(Df(M)) = \mathcal{V}(w(M))$.

Désignons par $Df_i(M)$, où $i \in \llbracket 1; 3 \rrbracket$, les 3 composantes vectorielles définies par :

$$\forall (i, j) \in \llbracket 1; 3 \rrbracket^2, \quad (Df_i(M))_j = (Df(M))_{i,j}$$

A l'aide de cette notion il est possible d'écrire :

$$(w_0 \in \text{Ker}(Df(M)) = \mathcal{V}(w)) \Leftrightarrow (\forall i \in \llbracket 1; 3 \rrbracket, \quad Df_i(M) \cdot w_0 = 0)$$

Puisque l'espace A est de dimension 3 et puisque nous avons supposé que A est un espace orthonormé, alors l'espace A est normé et admet une base de 3 vecteurs orthogonaux deux à deux, de norme unité.

De ce fait, il est possible de définir un produit vectoriel $\wedge$ sur l'espace A .

Par suite, nous sommes donc en mesure de définir :

$$w_k : A \ni M \mapsto Df_i(M) \wedge Df_j(M) \text{ où } k \neq i < j \neq k$$

En nous appuyant sur le fait que, sur $\mathcal{S}_1(f)$, $Df(M)$ est de rang 2, nous concluons que :

$$\forall M \in \mathcal{S}_1(f), \quad \forall k \in \llbracket 1; 3 \rrbracket, \quad Df_k(M) \cdot w_k(M) = 0$$

En effet, on a alternativement, lorsque $w_k(M) = Df_i(M) \wedge Df_j(M)$:

- $Df_i(M)$ et $Df_j(M)$ sont proportionnels, ce qui implique que $w_k(M) = (0,0,0)$
- $Df_i(M)$ et $Df_j(M)$ sont linéairement indépendants, ce qui implique que le vecteur $Df_k(M)$ est linéairement dépendant de $Df_i(M)$ et de $Df_j(M)$. En conséquence, le vecteur $Df_k(M)$ est orthogonal à $w_k(M) = Df_i(M) \wedge Df_j(M)$.

Le point important, que l'on souhaite souligner ici, est le fait que $\forall k \in \llbracket 1; 3 \rrbracket, w_k(M) \in \text{Ker}(Df(M))$. De ce fait, le vecteur $w(M)$ recherché peut être défini comme étant n'importe lequel des vecteurs $w_k(M)$, tel que $w_k(M) \neq (0; 0; 0)$. En utilisant à nouveau le fait que, sur $\mathcal{S}_1(f)$, $Df(M)$ est de rang 2, nous pouvons affirmer qu'il existe au moins une paire de lignes de $Df(M)$ tel que ces deux lignes soient linéairement indépendantes.

En reformulant cette dernière affirmation on conclut que :

$$\forall M \in \mathcal{S}_1(f), \quad \exists k \in \llbracket 1; 3 \rrbracket, \quad (0; 0; 0) \neq w_k(M) = Df_i(M) \wedge Df_j(M) \text{ où } k \neq i < j \neq k$$

On posera alors $w(M) = w_k(M)$.

Nous appuyant sur les raisonnements effectués jusqu'à présent, nous énonçons la Proposition 4.4.1.4 :

Proposition 4.4.1.4 (Système générique définissant $\mathcal{S}_{1,1}(f)$)

Si $f : A \rightarrow \mathcal{P}$, telle que $\dim(A) = \dim(\mathcal{P}) = 3$, est une fonction générique alors $M \in \mathcal{S}_{1,1}(f)$ ssi :

$$\begin{cases} \det(Df(M)) = 0 \\ w(M) \cdot \nabla(\det(Df(M))) = 0 \end{cases} \quad (\text{h})$$

où $w(M)$ vérifie :

$$0 \neq w(M) = Df_i(M) \wedge Df_j(M) \text{ où } Df(M) = \begin{pmatrix} Df_1(M) \\ Df_2(M) \\ Df_3(M) \end{pmatrix} \quad (\text{i})$$

Système d'équations définissant $\mathcal{S}_{1,1,1}(f)$

Un point $M \in \mathcal{S}_{1,1}(f)$ est un élément de $\mathcal{S}_{1,1,1}(f)$ si et seulement si l'espace tangent à $\mathcal{S}_{1,1}(f)$ en M contient un vecteur non nul de $\text{Ker}(df_M)$.

Si un ensemble E peut-être décrit comme une intersection $E = E_1 \cap E_2$ où E_1 et E_2 sont des ensembles alors son espace tangent en M , $\mathcal{T}_M E$, est caractérisé comme étant l'ensemble des vecteurs orthogonaux à tous les vecteurs normaux, en M , à l'ensemble E_1 ou à l'ensemble E_2 .

En appliquant cette propriété à $M \in \mathcal{S}_{1,1}(f)$ on obtient que $\mathcal{T}_M \mathcal{S}_{1,1}(f)$ est l'ensemble des vecteurs simultanément orthogonaux à $n(M) = \nabla(\det(Df(M)))$ et à $n_2(M) = \nabla(w(M) \cdot n(M))$.

Lorsque $n(M)$ et $n_2(M)$ sont linéairement indépendants, i.e. lorsque $t(M) = n(M) \wedge n_2(M) \neq 0$, alors le vecteur $t(M)$ vérifie $\mathcal{T}_M \mathcal{S}_{1,1}(f) = \mathcal{V}(t(M))$.

Remarque 4.4.1.2 (Cas de la proportionnalité)

Si $n(M)$ et $n_2(M)$ sont proportionnels, i.e. $t(M) = 0$, alors $\mathcal{T}_M \mathcal{S}_{1,1}(f) = \mathcal{T}_M \mathcal{S}_1(f)$.

Ceci implique que $M \in \mathcal{T}_M \mathcal{S}_{1,1,1}(f)$ et que $\mathcal{T}_M \mathcal{S}_{1,1,1}(f)$ est localement confondu avec $\mathcal{T}_M \mathcal{S}_{1,1}(f)$.

On souhaite définir un système d'équations qui est équivalent à $M \in \mathcal{T}_M \mathcal{S}_{1,1,1}(f)$ lorsque $t(M) \neq 0$.

Le premier candidat évident pour un tel système serait $Df(M) \times t(M) = 0$.

Il est à noter que ce système d'équations doit être considéré sans lui adjoindre les équations du système (h). En effet, $Df(M) \times t(M) = 0$ implique tout d'abord que $M \in \mathcal{S}_f$ qui est assimilable à la propriété $M \in \mathcal{S}_1(f)$, puisque f est considérée générique. De plus, par définition, $t(M)$ est orthogonal à $n(M) = \nabla(\det(Df(M)))$, impliquant que $M \in \mathcal{S}_{1,1}(f)$.

Toutefois, les équations du système $Df(M) \times t(M) = 0$ peuvent s'avérer admettre des expressions formelles compliquées. Puisque notre objectif est d'utiliser des méthodes numériques pour détecter les racines des systèmes définissant les points d'intérêt recherchés, nous nous proposons de considérer un autre système d'équations présentant des expressions, a priori, formellement moins compliquées. De ce fait, nous nous proposons de définir un système d'équations s'appuyant sur le système (h), de telle sorte que la méthodologie construite fasse appel à moins d'expressions formelles.

Nous rappelons que, fondamentalement, un point $M \in \mathcal{S}_{1,1,1}(f)$ vérifie $t(M) \in \text{Ker}(Df(M))$ en complément du système (h). Par ailleurs, nous avons, précédemment, mis en lumière un générateur de $\text{Ker}(Df(M))$ puisque, lorsque $M \in \mathcal{S}_{1,1}(f)$, $\text{Ker}(Df(M)) = \mathcal{V}(w(M))$. Ainsi vérifier $M \in \mathcal{S}_{1,1,1}(f)$ est équivalent à vérifier $(M \in \mathcal{S}_{1,1}(f) \text{ et } t(M) \text{ est proportionnel à } w(M))$.

De plus, $t(M)$ est proportionnel à $w(M)$ ssi $w(M)$ est orthogonal à la fois à $n(M)$ et à $n_2(M)$. Puisque $M \in \mathcal{S}_{1,1}(f)$, nous avons $w(M) \cdot n(M) = 0$. Par suite, nous pouvons donc conclure que vérifier $M \in \mathcal{S}_{1,1,1}(f)$ est équivalent à simultanément vérifier le système (h) et $w(M) \cdot n_2(M) = 0$.

Nous concluons en énonçant la Proposition 4.4.1.5 :

Proposition 4.4.1.5 (Système générique définissant $\mathcal{S}_{1,1,1}(f)$)

Si $f : A \rightarrow \mathcal{P}$, telle que $\dim(A) = \dim(\mathcal{P}) = 3$, est une fonction générique alors $M \in \mathcal{S}_{1,1}(f)$ ssi :

$$\begin{cases} \det(Df(M)) = 0 \\ w(M) \cdot \nabla(\det(Df(M))) = 0 \\ w(M) \cdot \nabla(w(M) \cdot \nabla(\det(Df(M)))) = 0 \end{cases} \quad (j)$$

où $w(M)$ est défini par l'équation (i).

4.4.1.3. Etude formelle d'une famille de fonctions paramétrées

Afin de préparer la détection pratique de la sous-section 4.4.2, nous introduisons une famille $\mathcal{F}$ de fonctions paramétrées par $(b, m, p, q) \in \mathbb{R}^4$ et telle que :

$$\forall f \in \mathcal{F}, \quad f : \mathbb{R}^3 \rightarrow \mathbb{R}^3 \text{ et } \exists (b, m, p, q) \in \mathbb{R}^4 \text{ tel que } f = f_{b,m,p,q}$$

où $\forall (b, m, p, q) \in \mathbb{R}^4$, $f_{b,m,p,q}$ est définie par l'équation (k).

$$\forall (x, y, z) \in \mathbb{R}^3, \quad f_{b,m,p,q}(x, y, z) = (x, \quad y^4 + my^2x + py^2z + qyx + byz, \quad z) \quad (k)$$

Nous allons donc formellement résoudre la recherche de $\mathcal{S}_1(f)$, de $\mathcal{S}_{1,1}(f)$ et de $\mathcal{S}_{1,1,1}(f)$ pour $f \in \mathcal{F}$. De ce fait, nous aurons accès à des résultats théoriques que nous pourrions comparer aux détections numériques retournées par les algorithmes d'Analyse par Intervalles que nous allons définir.

Par simplicité d'écriture, nous omettrons parfois les parenthèses du vecteur (x, y, z) .

Notons premièrement que l'on a l'implication successive $(k) \Rightarrow (l) \Rightarrow (m)$.

$$Df(x, y, z) = \begin{pmatrix} 1 & 0 & 0 \\ ny^2 + qy & 4y^3 + 2myx + 2pyz + qx + bz & py^2 + by \\ 0 & 0 & 1 \end{pmatrix} \quad (l)$$

$$\det(Df(x, y, z)) = 4y^3 + 2myx + 2pyz + qx + bz \quad (m)$$

De ce fait, à partir de la Proposition 4.4.1.3 et de l'équation (m) , nous obtenons que le point (x, y, z) est un élément de $S_f = S_1(f)$ ssi (x, y, z) vérifie l'équation (n) ci-dessous.

$$4y^3 + 2myz + 2pyz + qx + bz = 0 \quad (n)$$

Pour définir $S_{1,1}(f)$ il nous est nécessaire de calculer l'expression du gradient $\nabla(\det(Df))$, donnée dans l'équation (o) et obtenue à partir de l'équation (m) .

$$[\nabla(\det(Df))](x, y, z) = (2my + q, \quad 12y^2 + 2mx + 2pz, \quad 2py + b) \quad (o)$$

Pour tout vecteur $(x, y, z) \in \mathbb{R}^3$, le vecteur $w_2(x, y, z)$, dont l'expression a été définie au § 4.4.1.2, vérifie $w_2(x, y, z) = (0; -1; 0)$. Nous pouvons ainsi poser $\forall (x, y, z) \in \mathbb{R}^3$, $w(x, y, z) = (0; -1; 0)$. De ce fait, nous pouvons affirmer que $(x, y, z) \in S_{1,1}(f)$ ssi (x, y, z) vérifie le système (p) qui suit.

$$\begin{cases} 4y^3 + 2myz + 2pyz + qx + bz = 0 \\ -12y^2 - 2mx - 2pz = 0 \end{cases} \Leftrightarrow \begin{cases} 6y^2 + mx + pz = 0 \\ 8y^3 = qx + bz \end{cases} \quad (p)$$

Posons $n(x, y, z) = [\nabla(\det(Df))](x, y, z)$ et $n_2(x, y, z) = \nabla(w(x, y, z) \cdot n(x, y, z))$. Nous pouvons alors déduire, de l'équation (o) , l'expression vérifiée par $n_2(x, y, z)$, donnée par l'équation (q) .

$$n_2(x, y, z) = \nabla(-12y^2 - 2mx - 2pz) = (-2m, \quad -24y, \quad -2p) \quad (q)$$

On constate que, si $(pq \neq mb)$ alors $n(x, y, z) \wedge n_2(x, y, z) \neq (0; 0; 0)$.

En effet, affirmer que $n(x, y, z) \wedge n_2(x, y, z) = (0; 0; 0)$ est équivalent à affirmer :

$$(2my + q, \quad 12y^2 + 2mx + 2pz, \quad 2py + b) \wedge (-2m, \quad -24y, \quad -2p) = (0; 0; 0)$$

Cette dernière équation se réécrit :

$$(24py^2 - 4pmx + 24by - 4p^2z, \quad 2mb - 2pq, \quad 4pmz - 24qy + 4m^2x - 24my^2) = (0; 0; 0)$$

On déduit donc finalement que :

$$n(x, y, z) \wedge n_2(x, y, z) = (0, 0, 0) \Leftrightarrow \begin{cases} pq = mb \\ 6py^2 + 6by = pmx + p^2z \\ 6my^2 + 6qy = pmz + n^2x \end{cases}$$

Posons que $(pq \neq mb)$. La Proposition 4.4.1.5 nous conduit à conclure $(x, y, z) \in S_{1,1,1}(f) \Leftrightarrow (r)$.

$$\begin{cases} 4y^3 + 2myx + 2pyz + qx + bz = 0 \\ -12y^2 - 2mx - 2pz = 0 \\ w(M) \cdot n_2(M) = 0 \end{cases} \Leftrightarrow \begin{cases} 6y^2 + mx + pz = 0 \\ 8y^3 = qx + bz \\ 24y = 0 \end{cases} \quad (r)$$

Résoudre (r) en posant ($pq \neq mb$), fournit la succession de systèmes (s).

$$\begin{cases} y = 0 \\ mx + pz = 0 \\ qx + bz = 0 \end{cases} \Leftrightarrow \left(\begin{cases} y = 0 \\ qmx + pqz = 0 \\ qmx + bmz = 0 \end{cases} \text{ ou } \begin{cases} y = 0 \\ z = 0 \\ x = 0 \end{cases} \right) \Leftrightarrow \begin{cases} y = 0 \\ z = 0 \\ x = 0 \end{cases} \quad (s)$$

Afin de visualiser géométriquement les classes de singularités, précédemment définies formellement, ainsi que l'action de f sur ces classes de singularités, la Figure 14 présente, pour un vecteur choisi de paramètres :

- Deux vues en projection des classes $\mathcal{S}_v(f)$, pour chaque vecteur v de $\{(1), (1,1), (1,1,1)\}$
- Deux vues en projection de $f(\mathcal{S}_v(f))$, pour chaque vecteur v de $\{(1), (1,1), (1,1,1)\}$

Ces vues laissent apparaître :

- Par une couleur identique les points projetés, en relation (i.e. projections d'un même point ou de deux points dont l'un est l'image de l'autre).
- Par une colorisation en nuances de rouge, de vert et de bleu, la position des points dans l'espace source. En effet, la densité de rouge, de vert et de bleu de la couleur d'un point augmente, respectivement, avec la valeur des coordonnées en $\vec{x}$, en $\vec{y}$ et en $\vec{z}$.
- Par une superposition de couleurs, mise en évidence par effet de transparence, chaque zone où plusieurs points sont projetés, de telle sorte que la couleur utilisée correspond à la somme des couleurs associées aux points projetés. La transparence est appliquée dans l'ordre de leurs coordonnées suivant l'axe de projection.

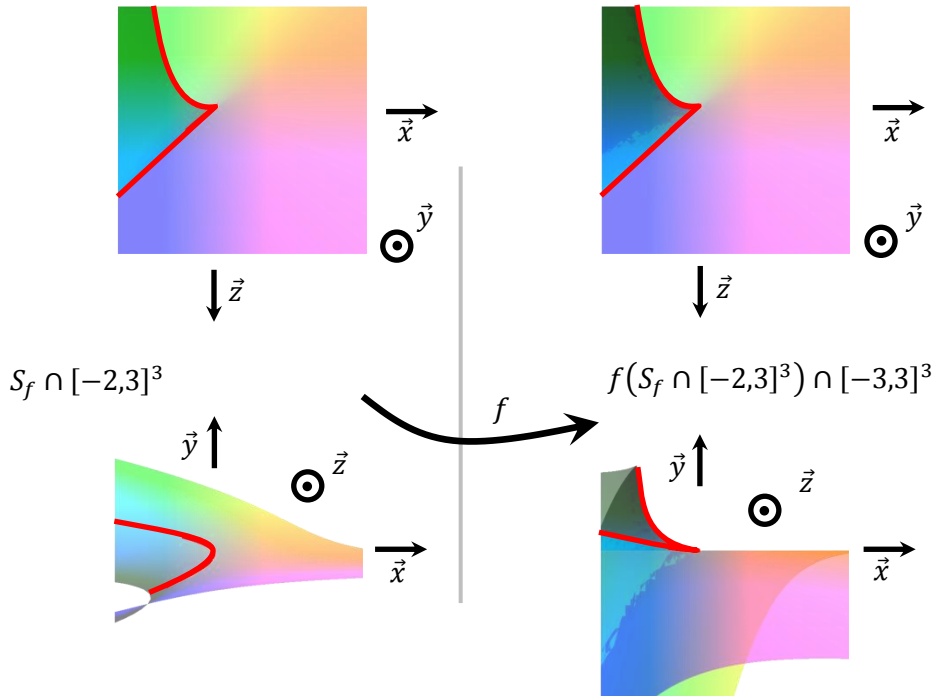


Figure 14 : Vues en projection de $\mathcal{S}_f$ et de $f(\mathcal{S}_f)$ et (en rouge) de $\mathcal{S}_{1,1}(f)$ et de $f(\mathcal{S}_{1,1}(f))$ où :
 $f = f_{b,m,p,q}$ avec $b = m = 2$ et $p = q = 1$

4.4.2. Détection Pratique

4.4.2.1. Algorithmes implémentés

Tout comme pour les algorithmes proposés au § 4.2.2 et au § 4.3.2, nous proposons d'implémenter des algorithmes de détection et d'encadrement garantis, s'appuyant sur l'Analyse par Intervalles.

Comme en dimension inférieure, en nous référant à la section 2.3, relative à l'Analyse par Intervalles, chaque algorithme proposé est paramétré par :

- une précision
- une liste d'intervalles d'étude
- une fonction formelle d'intérêt, convertie, trivialement, en une fonction d'inclusion naturelle

Il est ainsi nécessaire, en préliminaire à l'exécution des algorithmes, de calculer formellement les expressions de:

- $\det(Df)$
- $\forall k \in \llbracket 1; 3 \rrbracket :$

- $w_k : \mathbb{R}^3 \ni M \mapsto Df_i(M) \wedge Df_j(M)$ où $Df(M) = \begin{pmatrix} Df_1(M) \\ Df_2(M) \\ Df_3(M) \end{pmatrix}$ et où $k \neq i < j \neq k$
- $g_k = w_k \cdot \nabla(\det(Df))$
- $h_k = w_k \cdot \nabla(g_k) = w_k \cdot \nabla(w_k \cdot \nabla(\det(Df)))$
- Du_k où $u_k = (\det(Df), g_k, h_k)$

En pratique, les trois algorithmes proposés sont implémentés de manière à charger en mémoire, au début de leur exécution, les expressions calculées formellement, précédemment listées, à partir de fichiers textes externes.

Ce choix permet de modifier les expressions d'intérêt indépendamment du fichier exécutable.

Il n'est donc pas nécessaire de recompiler le programme exécutable pour l'appliquer à des expressions et à des paramètres différents, qui peuvent être ainsi édités sans expertise vis-à-vis du langage C++.

Il est à noter que les algorithmes implémentés sont, intrinsèquement, des fusions d'algorithmes IVIA et d'algorithmes de Newton par Intervalles d'expressions d'intérêt choisies pertinemment.

En effet, chacun des trois algorithmes proposés contient une boucle principale, qui analyse chaque élément d'une liste d'intervalles, tant que cette liste n'est pas vide. De même, chaque algorithme incorpore, selon une organisation différente, des tests élémentaires successifs, associés à des méthodes de Newton par Intervalles et à des procédures IVIA, paramétrées par des fonctions d'intérêt en entrée.

Les algorithmes implémentés *n'exigent pas que la fonction d'intérêt soit générique* et, de ce fait, peuvent *traiter toute fonction* de $\mathbb{R}^3$ vers $\mathbb{R}^3$, tant que celle-ci est suffisamment différentiable sur chaque intervalle d'étude, afin de garantir que les expressions requises par chaque algorithme existent.

Les 3 algorithmes retournant des encadrements garantis de $S_f \supseteq \mathcal{S}_1(f)$, de $\mathcal{S}_{1,1}(f)$ et de $\mathcal{S}_{1,1,1}(f)$ sont respectivement présentés, sous forme de logigrammes, à la Figure 15, à la Figure 16 et à la Figure 17.

Ces 3 algorithmes sont conçus de façon à être exécutés dans l'ordre de l'inclusion entre les ensembles qu'ils encadrent respectivement. En effet, puisque l'on vérifie que $S_f \supseteq \mathcal{S}_1(f) \supseteq \mathcal{S}_{1,1}(f) \supseteq \mathcal{S}_{1,1,1}(f)$, appliquer l'algorithme encadrant $\mathcal{S}_{1,1}(f)$ au recouvrement obtenu de S_f , puis appliquer l'algorithme, détectant $\mathcal{S}_{1,1,1}(f)$, à cet encadrement déduit de $\mathcal{S}_{1,1}(f)$ permet de réduire, dans ces deux cas, le volume du recouvrement initial à considérer et donc, a priori, d'optimiser les performances des algorithmes recouvrant $\mathcal{S}_{1,1}(f)$ et $\mathcal{S}_{1,1,1}(f)$.

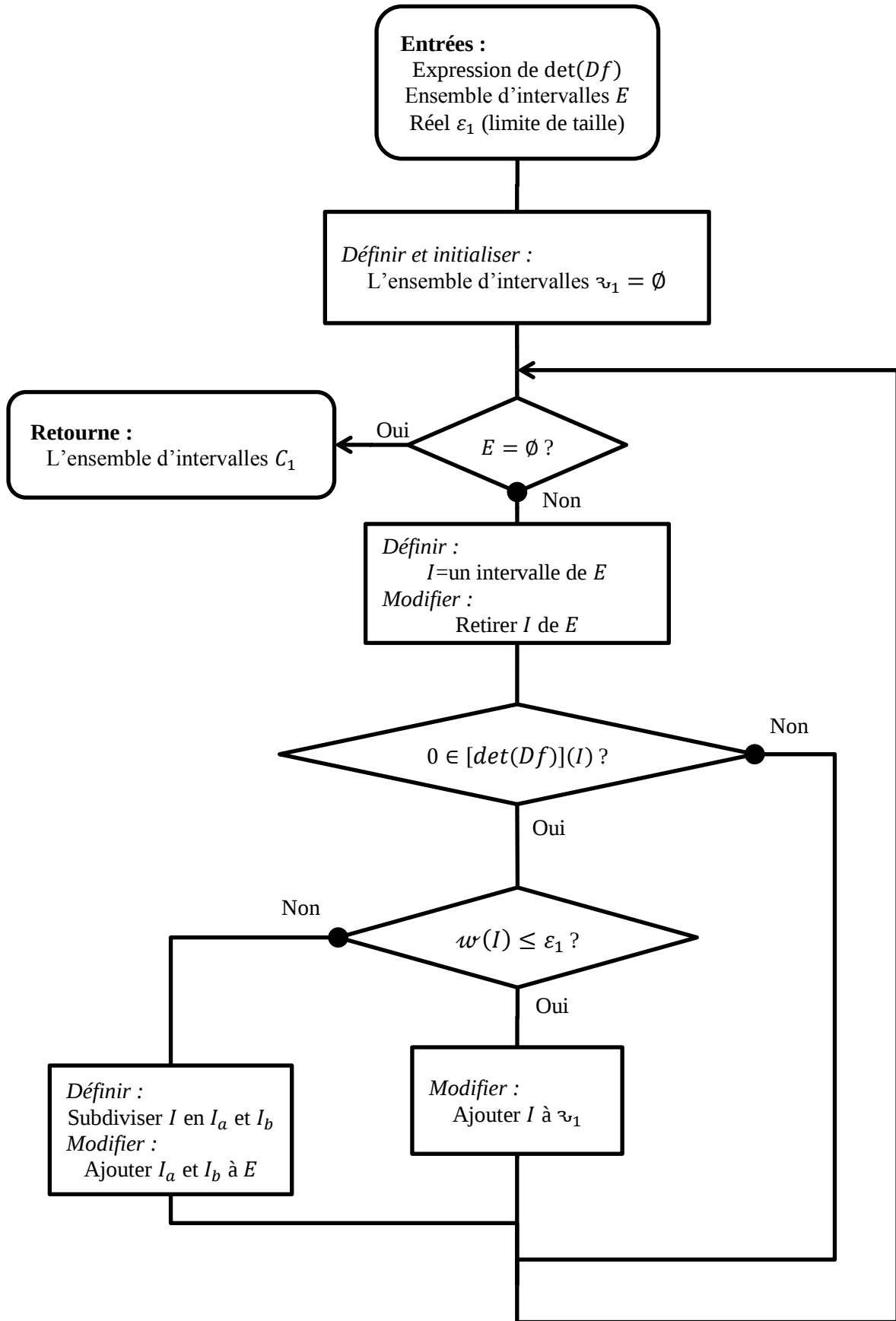


Figure 15 : Algorithme encadrant S_f (génériquement $S_f = \mathcal{S}_1(f)$)

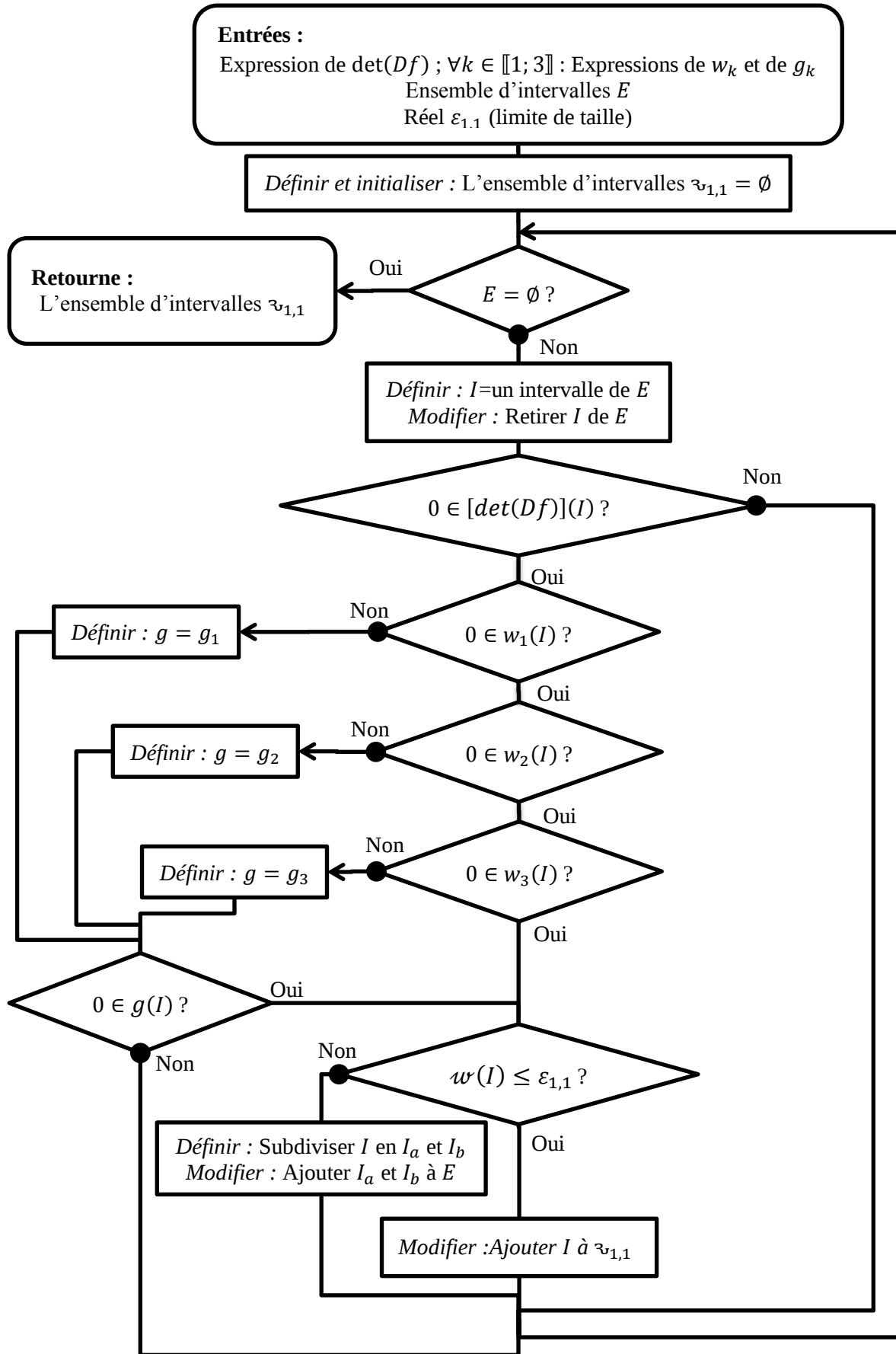


Figure 16 : Algorithme encadrant $\mathcal{S}_{1,1}(f)$

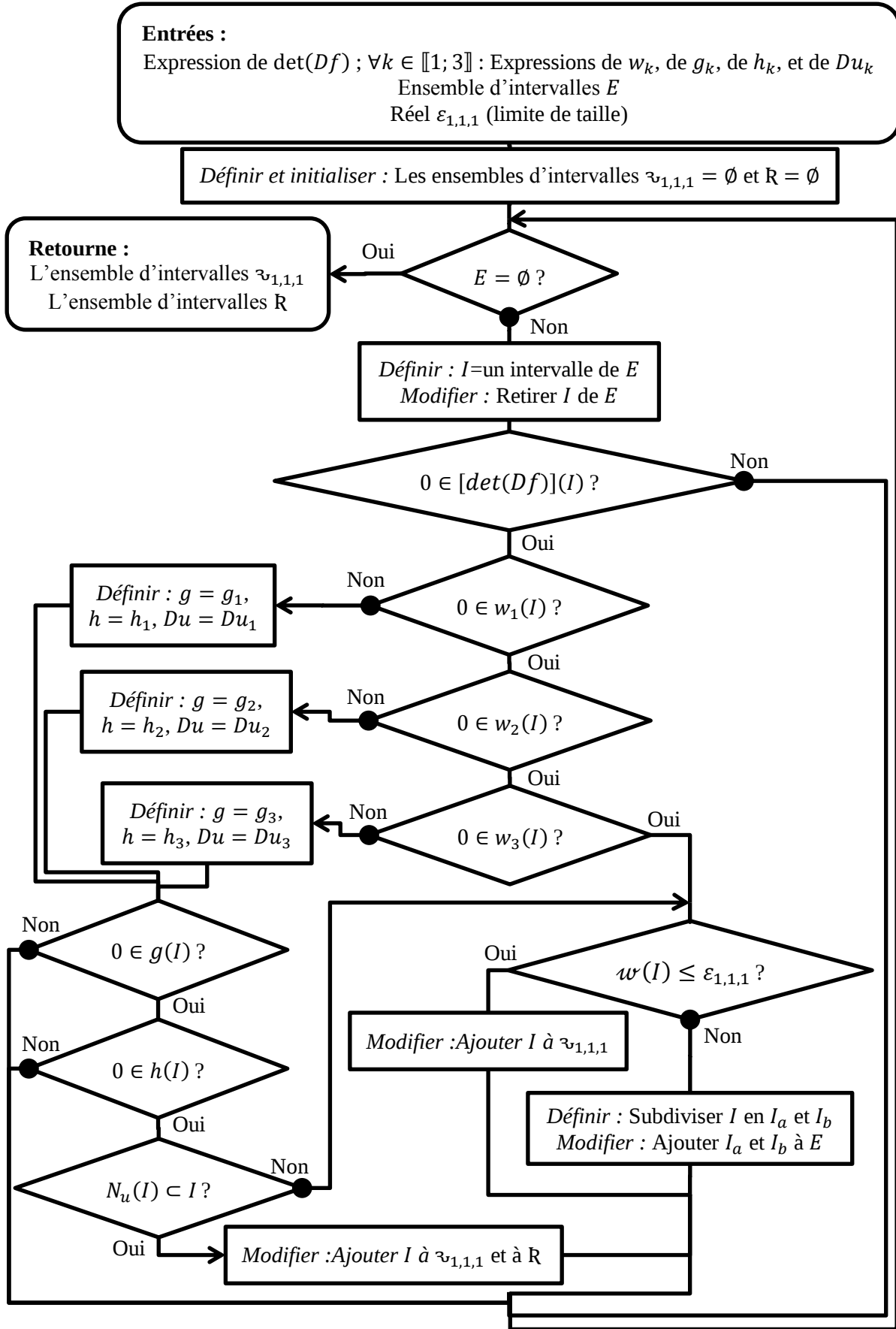


Figure 17 : Algorithme encadrant $\mathcal{S}_{1,1,1}(f)$

En comparant la Figure 5 et la Figure 15, il est évident que l'algorithme encadrant le lieu singulier S_f est un algorithme IVIA, de fonction d'intérêt $\det(Df)$.

De ce fait, cet algorithme requiert seulement que f soit différentiable sur chaque intervalle étudié.

De plus, l'algorithme encadre, trivialement, toujours $\mathcal{S}_1(f)$, puisque $S_f = \mathcal{S}_1(f) \cup \mathcal{S}_2(f) \cup \mathcal{S}_3(f)$.

Rappelons qu'une fonction f est générique lorsque $\mathcal{S}_2(f) = \mathcal{S}_3(f) = \emptyset$. On conclut alors que :

Si f est générique, l'encadrement retourné est un encadrement de l'ensemble $\mathcal{S}_1(f)$ seul.

L'algorithme de la Figure 16, qui encadre génériquement $\mathcal{S}_{1,1}(f)$ est conçu dans l'objectif d'encadrer, à la fois, les points de $\mathcal{S}_{1,1}(f)$, pour lesquels la Proposition 4.4.1.4 est applicable et les points pour lesquels cette dernière proposition ne peut être appliquée, car l'Analyse par Intervalles échoue à conclure que $\exists k \in \llbracket 1; 3 \rrbracket$ tel que $w_k \neq 0$. Ceci implique que l'algorithme de la Figure 16 retourne un recouvrement de $\mathcal{S}_{1,1}(f) \cup \mathcal{S}_2(f) \cup \mathcal{S}_3(f)$. Tout comme l'algorithme précédent, ceci implique que, génériquement, seul $\mathcal{S}_{1,1}(f)$ est encadré. Pour que l'algorithme puisse s'exécuter, il est nécessaire que f soit deux fois différentiable, sur chaque intervalle étudié, puisque $\nabla(\det(Df))$ est seulement défini si les dérivées secondes de f sont elles-mêmes définies.

Complémentairement aux algorithmes précédemment introduits, l'algorithme présenté à la Figure 17 est en mesure de *prouver l'existence et l'unicité* de points isolés de $\mathcal{S}_{1,1,1}(f)$, en plus de fournir un *encadrement* de l'ensemble $\mathcal{S}_{1,1,1}(f)$. En effet, pour toute fonction f , le test d'inclusion lié à l'opérateur de Newton par Intervalles permet de détecter certains points isolés de $\mathcal{S}_{1,1,1}(f)$.

Cet algorithme retourne donc une liste R , d'intervalles encadrant chacun un point M tel que :

- $M \in \mathcal{S}_{1,1,1}(f)$
- $\exists k \in \llbracket 1; 3 \rrbracket$ tel que $u_k(M) = 0$ et $w_k(M) \neq 0$

Nous nous appuyons ainsi sur la Proposition 4.4.1.5 afin d'affirmer que, si on vérifie que $w_k(M) \neq 0$ alors il est équivalent d'affirmer « $u_k(M) = 0$ » ou « $M \in \mathcal{S}_{1,1,1}(f)$ ».

Sur tout intervalle I tel que où $0 \notin w_k(I)$, il est trivial que $\forall M \in I, w_k(M) \neq 0$.

Sur un tel intervalle I , toute racine isolée de u_k , détectée par la méthode de Newton par Intervalles est donc un élément de l'ensemble $\mathcal{S}_{1,1,1}(f)$, selon la Proposition 4.4.1.5.

La vérification préliminaire que $0 \notin w_k(I)$ (ou, au minimum que $\forall M \in I, w_k(M) \neq 0$) est donc primordiale pour justifier que les racines isolées détectées sont bien des éléments de $\mathcal{S}_{1,1,1}(f)$.

Afin de n'omettre aucun élément de $\mathcal{S}_{1,1,1}(f)$ et afin de pouvoir traiter toute fonction, l'algorithme proposé pour détecter $\mathcal{S}_{1,1,1}(f)$ retourne aussi un recouvrement $\mathfrak{S}_{1,1,1}$ de cet ensemble.

Le recouvrement $\mathfrak{S}_{1,1,1}$ contient les intervalles I sur lesquels l'Analyse par Intervalles, étant donnée la taille limite $\varepsilon_{1,1,1}$, n'est pas en mesure de mettre en évidence que « $\exists k \in \llbracket 1; 3 \rrbracket$ tel que $0 \notin w_k(I)$ ».

De même, ce recouvrement comporte les intervalles I pour lesquels la méthode de Newton par Intervalles ne peut prouver l'existence et l'unicité d'un point $M \in I$ tel que $M \in \mathcal{S}_{1,1,1}(f)$. Cette incapacité peut être liée à une précision trop grossière ou au fait que $\mathcal{S}_{1,1,1}(f)$ n'est pas, localement, sur l'intervalle I considéré, restreint à des points isolés.

Chaque intervalle $I \in \mathfrak{S}_{1,1,1}$ peut ainsi contenir :

- Des points isolés de $\mathcal{S}_{1,1,1}(f)$ non détectables à la précision $\varepsilon_{1,1,1}$
- Des sous-ensembles non ponctuels de $\mathcal{S}_{1,1,1}(f)$
- Des sous-ensembles de $\mathcal{S}_3(f) \cup \mathcal{S}_2(f)$

Similairement aux deux autres algorithmes, l'algorithme encadrant $\mathcal{S}_{1,1,1}(f)$ retourne donc aussi, parmi le recouvrement $\mathfrak{S}_{1,1,1}$, les intervalles sur lesquels la méthodologie de la Proposition 4.4.1.5 ne peut pas être appliquée, incluant les intervalles encadrant l'ensemble $\mathcal{S}_3(f) \cup \mathcal{S}_2(f)$.

Ainsi, la liste d'intervalles $\mathfrak{I}_{1,1,1}$ encadre, en général, $\mathcal{S}_3(f) \cup \mathcal{S}_2(f) \cup \mathcal{S}_{1,1,1}(f)$ et encadre génériquement l'ensemble $\mathcal{S}_{1,1,1}(f)$ seul.

Complémentairement, la seconde liste d'intervalles $\mathbf{R}$ retournée est une sous-liste de $\mathfrak{I}_{1,1,1}$, telle que :

$$\forall I \in \mathbf{R}, \quad \exists ! M \in I, M \in \mathcal{S}_{1,1,1}(f)$$

Afin que $\mathcal{S}_{1,1,1}(f)$ soit défini sur tout un intervalle I , il est nécessaire que f soit trois fois différentiable sur cet intervalle I considéré, car la caractérisation proposée de $\mathcal{S}_{1,1,1}(f)$ requiert que l'expression de la fonction $h = w \cdot \nabla(g) = w \cdot \nabla[w \cdot \nabla(\det(Df))]$ soit définie. Toutefois, l'opérateur de Newton par Intervalles utilisé, N_u , est construit à partir de Du , la matrice Jacobienne de u , où $u = (\det(Df), g, h)$. Ceci impose une dérivation supplémentaire de la fonction d'intérêt f et donc 4 dérivations de celle-ci. La fonction f doit donc être 4 fois différentiable sur chaque intervalle étudié par l'algorithme.

On conclut donc que, si on pose $f: U \subseteq A \rightarrow \mathbb{P}$, une fonction qui est, au moins 4 fois différentiable, sur chaque intervalle $I \in E$, où E est la liste des intervalles d'étude et où $\{x \in A \mid \exists I \in E, x \in I\} \subseteq U$, alors f peut être traitée par les 3 algorithmes proposés à la Figure 15, à la Figure 16 et à la Figure 17, pour obtenir, respectivement, un encadrement de $\mathcal{S}_1(f) \cup \mathcal{S}_2(f) \cup \mathcal{S}_3(f)$, un encadrement de $\mathcal{S}_{1,1}(f) \cup \mathcal{S}_2(f) \cup \mathcal{S}_3(f)$ et un couple, formé par un encadrement de $\mathcal{S}_{1,1,1}(f) \cup \mathcal{S}_2(f) \cup \mathcal{S}_3(f)$ et par une liste $\mathbf{R}$ d'intervalles telle que $\forall I \in \mathbf{R}, \exists ! M \in I, M \in \mathcal{S}_{1,1,1}(f)$.

4.4.2.2. Application et performances des 3 algorithmes implémentés

Afin d'étudier et de vérifier l'applicabilité des 3 algorithmes proposés, ainsi que leurs performances, nous nous intéressons à leur application aux fonctions de la famille paramétrée introduite au § 4.4.1.3. Nous rappelons que ces fonctions $f \in \mathcal{F}$ sont définies par $\exists (b, m, p, q) \in \mathbb{R}^4$ tel que $f = f_{b,m,p,q}$ où :

$$f_{b,m,p,q}(x, y, z) = (x, \quad y^4 + my^2x + py^2z + qyx + byz, \quad z)$$

Afin de garantir que $\mathcal{S}_{1,1,1}(f_{b,m,p,q}) = \{(0; 0; 0)\}$, en accord avec l'étude réalisée au § 4.4.1.3, nous nous restreignons à considérer des vecteurs de paramètres $(b, m, p, q) \in \mathbb{R}^4$ tels que $(bm \neq pq)$.

Nous proposons d'étudier les performances des 3 algorithmes proposés, au regard de l'influence de deux facteurs principaux que sont les précisions choisies, pour chaque algorithme, et l'imprécision sur les paramètres constitutifs b, m, p et q (i.e. la taille des intervalles encadrant leurs valeurs).

La méthodologie proposée compare les performances obtenues à celles d'une exécution de référence dont les caractéristiques sont :

- La fonction de référence est $f_0 = f_{2,2,1,1}$ (i.e. on pose $b = m = 2$ et $p = q = 1$).
- Les 3 algorithmes sont appliqués successivement, dans l'ordre de l'inclusion à la fonction f_0 , (i.e. l'encadrement de $\mathcal{S}_{f_0}$, puis l'encadrement de $\mathcal{S}_{1,1}(f_0)$ suivis par la détection de $\mathcal{S}_{1,1,1}(f_0)$).
- Les précisions sont $\varepsilon_1 = 0.025$, $\varepsilon_{1,1} = 0.01$ et $\varepsilon_{1,1,1} = 0.003$
- L'ensemble d'intervalles initial d'étude est $E = \{[-2; 3]^2\}$
- La fonction de taille $\mathcal{W}$ considérée est définie par :

$$\forall (\alpha, \beta, \rho, \mu) \in \mathbb{R}^4, \mathcal{W}([\alpha, \beta] \times [\rho, \mu]) = |[\alpha, \beta] \times [\rho, \mu]|_\infty = \max(\min((\beta - \alpha), (\mu - \rho)); 0)$$
- La subdivision de chaque intervalle I en deux intervalles I_a et I_b vérifie :
 - $I_a \cup I_b = I$
 - $I_a \neq \emptyset$ et $I_b \neq \emptyset$
 - $\mathcal{W}(I_a \cap I_b) = 0$

Afin de visualiser les recouvrements obtenus à partir de l'exécution de référence, nous proposons un rendu de ceux-ci à la Figure 18 (selon des vues en projection d'une scène composée de pavés représentant les intervalles de ces recouvrements, réalisée à l'aide de la méthodologie POVRAY).

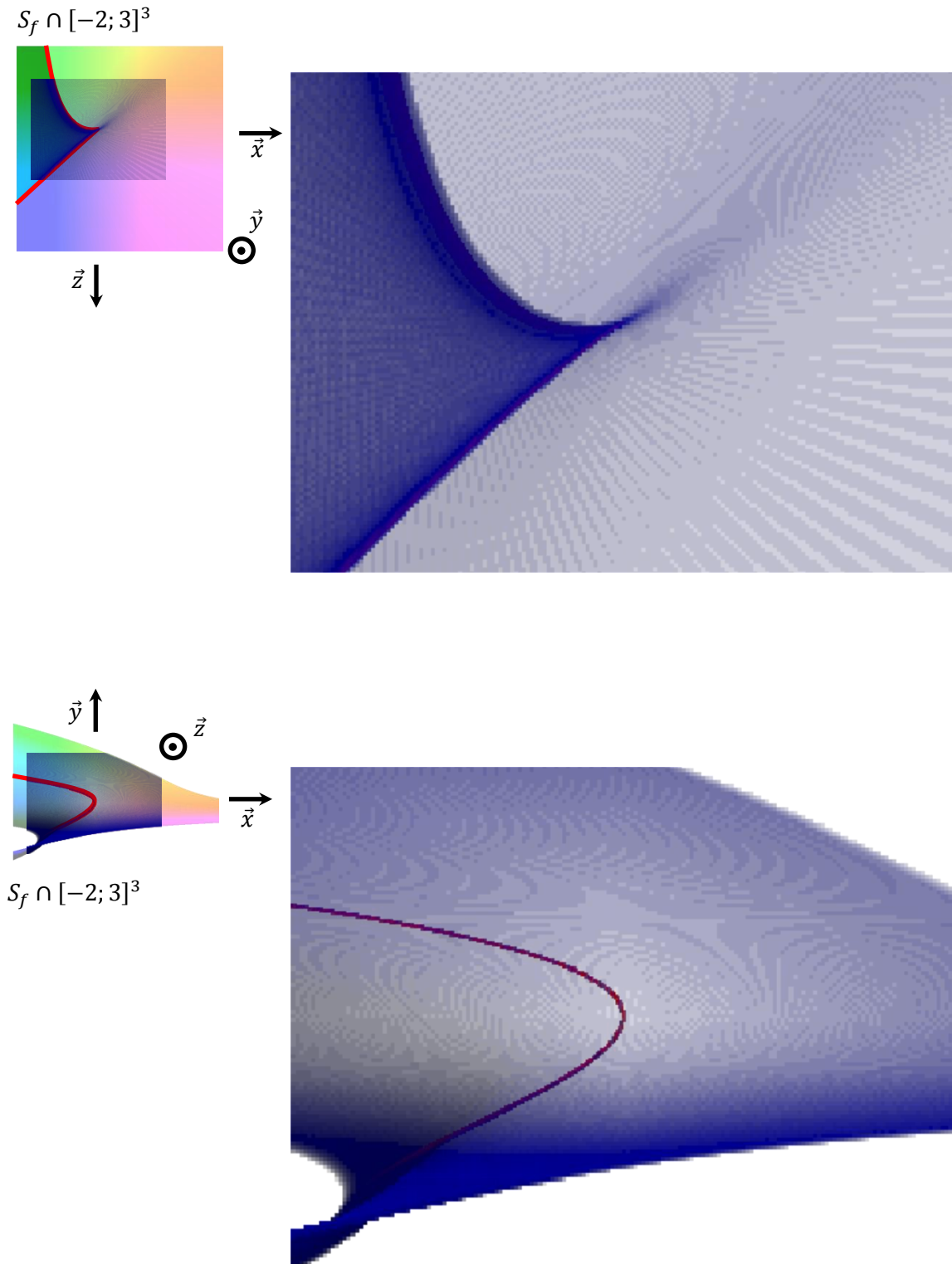


Figure 18 : Vues en projection, de S_{f_0} et $S_{1,1}(f_0)$ (en rouge), avec un rendu, par la méthode POVRAY, des recouvrements de S_{f_0} (pavés en bleu et en rouge) et de $S_{1,1}(f_0)$ (pavés en rouge) fournis par l'exécution de référence.

Nous étudions, premièrement, l'influence des précisions ε_1 , $\varepsilon_{1,1}$ et $\varepsilon_{1,1,1}$ choisies respectivement pour l'exécution des algorithmes encadrant $\mathcal{S}_{f_0}$, $\mathcal{S}_{1,1}(f_0)$ et $\mathcal{S}_{1,1,1}(f_0)$. Les performances considérées pour cette étude sont le temps d'exécution total des 3 algorithmes et le nombre d'intervalles encadrant chaque classe singulière. Ces performances sont reportées dans la Table 3, pour 6 exécutions, incluant l'exécution de référence 0.

Numéro d'exécution		0	1	2	3	4	5
Précision respective pour chaque algorithme	$\mathcal{S}_1$	0.025	0.025	0.25	0.25	0.725	1.25
	$\mathcal{S}_{1,1}$	0.01	0.005	0.05	0.125	0.5	0.625
	$\mathcal{S}_{1,1,1}$	0.003	0.001	0.01	0.05	0.2	0.15
Temps d'exécution total		147.745 s	166.287 s	4.277 s	3.60 s	0.306 s	0.305 s
nombre d'intervalles englobants :	$\mathcal{S}_1$	239915	239915	3700	3700	205	205
	$\mathcal{S}_{1,1}$	5028	10000	1255	646	160	160
	$\mathcal{S}_{1,1,1}$	1	1	1	1	1	1

Table 3 : Evolution des performances en fonction des précisions choisies

Nous proposons d'étudier le temps d'exécution total des trois algorithmes car, étant donné que les algorithmes encadrant $\mathcal{S}_{1,1}(f_0)$ et $\mathcal{S}_{1,1,1}(f_0)$ admettent pour listes d'étude initiales, respectivement, l'encadrement obtenu pour $\mathcal{S}_{f_0}$ et celui obtenu pour $\mathcal{S}_{1,1}(f_0)$, leurs temps d'exécutions dépendent des paramètres choisis pour l'exécution de l'algorithme encadrant $\mathcal{S}_{f_0}$.

Les temps d'exécution reportés dans la Table 3 sont donnés pour un ordinateur utilisant un système d'exploitation Debian, version 7.8, 64 bits et un CPU Intel® Core™ i7 (2.90 GHz).

Le nombre d'intervalles englobants, pour chaque recouvrement, est à considérer conjointement à la précision choisie comme paramètre de l'algorithme, afin de caractériser la « proximité » de l'encadrement vis-à-vis de l'ensemble encadré par chaque algorithme.

Il est à noter que les performances observées paraissent admettre des paliers. Ce que semblent indiquer les exécutions 4 et 5, qui présentent des performances similaires pour des précisions différentes. L'évolution, selon des paliers, des performances est formellement justifiée par le rôle de la précision dans les algorithmes implémentés, où celle-ci sert de seuil limite à la subdivision d'intervalles. Ainsi, l'impact de la précision choisie sur les performances ne peut changer que lorsque la précision atteint une taille possible d'un intervalle obtenu par subdivisions successives d'un intervalle d'étude initial. Dans le cas de l'application considérée, la fonction de taille ω est induite par la norme infinie sur $\mathbb{R}^3$ et les subdivisions des intervalles étant réalisées selon les points milieux des composantes de ceux-ci, si $\forall k \in \mathbb{N}$, $5/2^k \in \mathbb{IR}$ (i.e. ces valeurs sont représentables dans l'index de nombres réels considéré) alors les performances peuvent évoluer lorsque la précision atteint une valeur égale à $\omega([-2; 3]^3)/2^k = 5/2^k$ où $k \in \mathbb{N}$.

Globalement, dans la Table 3, les temps d'exécution sont modérés, même pour les exécutions admettant les précisions les plus fines. Il est, par ailleurs, pertinent de garder à l'esprit, qu'en pratique, les temps d'exécution et les performances en général, sont aussi dépendants de la complexité de la fonction d'intérêt considérée ainsi que de la liste initiale d'intervalles étudiés. Tout ceci implique que les algorithmes proposés devraient être applicables à toute fonction admettant à la fois une expression formelle de complexité modérée, étudiée sur un volume d'intervalles lui aussi modéré.

Puisque, plutôt que de considérer des fonctions scalaires, les algorithmes s'appliquent à des *fonctions d'inclusions* qui agissent sur des intervalles, *les algorithmes proposés peuvent s'appliquer à toute fonction d'intérêt, même si ses paramètres constitutifs sont de valeurs inconnues, tant que ceux-ci restent constants et sont inclus dans des intervalles connus, induits par les spécifications du robot.*

Dans ce cas, les paramètres constitutifs, de la fonction d'inclusion considérée, sont ces intervalles induits, au lieu d'être des intervalles minimaux contenant les valeurs de paramètres ponctuels.

La capacité à gérer des paramètres légèrement inconnus mais contrôlés est une motivation majeure, conjointement, pour des futures applications à des manipulateurs robotiques et, plus généralement, pour l'analyse des systèmes, puisque leurs paramètres constitutifs ne sont pas fixés mais sont bornés par les spécifications de construction.

Nous nous intéressons donc à l'étude de l'impact de l'imprécision, relative aux paramètres constitutifs, sur les performances des algorithmes. Afin de ce faire, nous proposons, pour chaque imprécision considérée δ , d'exécuter les algorithmes en considérant que chaque paramètre constitutif est encadré par un intervalle de contrôle $[v, v + \delta]$, où v est la valeur nominale de référence de ce paramètre.

Les performances de ces exécutions sont rapportées dans la Table 4.

Numéro d'exécution		0	1	2	3
Imprécision δ		0	0.1	0.3	0.5
temps d'exécution total		157.745 s	358.101 s	777.204 s	1314.771 s
nombre d'intervalles recouvrant :	$\mathcal{S}_1$	239915	544124	1134372	1703263
	$\mathcal{S}_{1,1}$	5028	49578	272006	664045
	$\mathcal{S}_{1,1,1}$	1	1	1	1 ($\mathcal{S}_{1,1,1}$ encadré mais non détecté)

Table 4 : Evolution des performances en fonction de l'imprécision sur les paramètres constitutifs.

A partir des résultats fournis dans la Table 4, nous sommes en mesure de conclure que les performances des algorithmes restent bonnes pour des imprécisions modérées et que leur capacité à encadrer les ensembles d'intérêt est conservée, tant que la mémoire disponible est suffisante pour stocker la quantité, croissante avec l'imprécision, d'intervalles dans les recouvrements retournés.

Il faut toutefois noter que, à partir d'une certaine imprécision, la procédure de Newton par Intervalles peut ne plus être en mesure de conclure sur l'existence et l'unicité de points dans $\mathcal{S}_{1,1,1}(f)$.

Enfin, comme pour l'étude précédente, de l'impact des précisions considérées, les performances des algorithmes sont dépendantes de la complexité formelle de la fonction d'intérêt. Ceci induit que l'application des algorithmes, à des fonctions robotiques formellement compliquées, reste une perspective encore à l'étude, car des optimisations de code et des simplifications formelles peuvent s'avérer nécessaires afin d'étudier efficacement les manipulateurs robotiques qui leur sont associées.

4.4.3. Conclusions et perspectives de l'étude en dimension 3

En tout premier lieu, nous souhaitons souligner que, au cours de cette étude, nous avons proposé des systèmes d'équations caractérisant $\mathcal{S}_{1,1}(f)$ et $\mathcal{S}_{1,1,1}(f)$, pour toute fonction f telle que :

- $f : A \rightarrow \mathbb{P}$ où $\dim(A) = \dim(\mathbb{P}) = 3$
- $\mathcal{S}_3(f) = \mathcal{S}_2(f) = \emptyset$

En combinant cette contribution au concept de généricité (i.e. il existe un ensemble de fonctions, résiduel pour $\mathcal{T}^\infty(X, Y)$, tel que $\mathcal{S}_3(f) = \mathcal{S}_2(f) = \emptyset$) implique que, *pour presque toutes les fonctions (lisses) entre espaces de dimension 3, toutes leurs classes de singularité peuvent être isolées*. En effet, pour une fonction f parmi ces fonctions génériques, les seules classes singulières non vides vérifient :

$$S_f = \mathcal{S}_1(f) = \mathcal{S}_{1,0}(f) \cup \mathcal{S}_{1,1}(f) = \mathcal{S}_{1,0}(f) \cup \mathcal{S}_{1,1,0}(f) \cup \mathcal{S}_{1,1,1}(f)$$

où S_f vérifie l'équation $\det(Df) = 0$

La mise en évidence de systèmes caractérisant les classes de singularités n'est que la première contribution de l'étude proposée dans cette section. En effet, une implémentation *d'algorithmes basés sur l'Analyse par Intervalles*, s'appuyant sur ces systèmes mis en évidence, est aussi considérée pour détecter ces classes de singularités d'intérêt. De plus, même si les classes considérées suffisent, génériquement, à définir toutes les classes de singularités non vides, les algorithmes proposés sont en mesure de traiter toute fonction d'intérêt, puisqu'ils sont implémentés pour tester la généricité de la fonction traitée, sur chaque intervalle, et pour agir en conséquence lorsque le test échoue.

Les propriétés des algorithmes implémentés garantissent que ceux-ci encadrent toujours les ensembles d'intérêt recherchés, si l'exécution des algorithmes n'est pas prématurément interrompue. En effet, les temps d'exécution des algorithmes ne peuvent être infinis, ce qui implique que, si la mémoire disponible est suffisante pour stocker les encadrements obtenus, les algorithmes finiront par conclure.

La *méthode de Newton par Intervalles* est utilisée afin de détecter certains des points isolés de $\mathcal{S}_{1,1,1}$. Toutefois, on notera que si celle-ci ne peut pas conclure sur l'existence ou l'unicité d'un point de $\mathcal{S}_{1,1,1}$ dans tout intervalle de taille inférieure à la précision, alors l'algorithme considère que cet intervalle « encadre seulement » l'ensemble $\mathcal{S}_{1,1,1}$.

Les causes de cette impossibilité de conclure peuvent être :

- Le pessimisme, intrinsèque à l'Analyse par intervalles
- La non généricité de la fonction d'intérêt

Ainsi, une étude locale plus fine (éventuellement formelle) de la fonction peut permettre de conclure sur la nature de l'encadrement obtenu, sur tout intervalle indéterminé pour cette méthode de Newton.

Le second intérêt de l'Analyse par Intervalles demeure son utilisation de *fonctions d'inclusion* qui se substituent intrinsèquement aux fonctions scalaires formelles. Cette substitution permet toujours de remplacer chaque paramètre ponctuel de la fonction formelle par un intervalle de paramètres, prenant en compte l'imprécision qui existe sur chaque paramètre constitutif de la fonction d'intérêt formelle. Cette faculté de prendre en compte des imprécisions s'illustre naturellement par l'application des algorithmes à l'étude de fonctions géométriques de manipulateur robotiques, dont les paramètres constitutifs sont toujours bornés mais n'admettent pas de valeurs exactement connues.

La perspective immédiate des développements détaillés dans cette section demeure la même que pour les études en dimensions inférieures et consiste en l'optimisation des méthodes et des algorithmes. En effet, cette optimisation permettrait d'analyser plus efficacement les fonctions géométriques entre espaces de dimensions 3, dont les singularités ne sont pas encore complètement catégorisées.

Afin d'illustrer notre propos, nous présentons une famille de manipulateurs, admettant une fonction géométrique, entre espaces de dimension 3 :

Les *manipulateurs parallèles plan 3-RPR*, dont le schéma de principe est proposé à la Figure 19.

En effet, un manipulateur de cette famille peut déplacer sa plateforme mobile, dans le plan, selon trois degrés de liberté (deux translations et une rotation) par l'intermédiaire de ses trois jambes extensibles.

Pour un tel manipulateur, la fonction géométrique d'intérêt principal est $g: (x, y, z) \mapsto (\rho_1, \rho_2, \rho_3)$.

Pour une introduction plus complète à cette classe de manipulateur, le lecteur pourra, par exemple, se référer à (Moroz, et al., 2010).

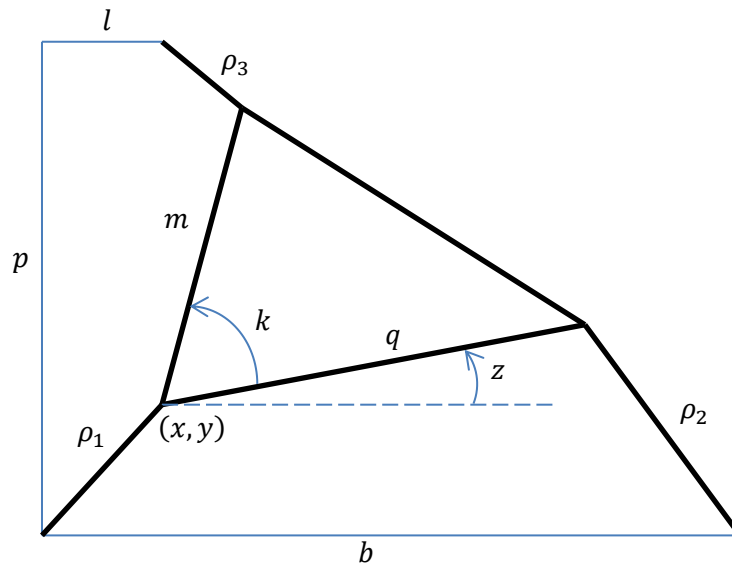


Figure 19 : Schéma de principe et paramètres géométriques d'un manipulateur parallèle 3-RPR

Comme affirmé précédemment, les applications à la robotique sont encore en cours de développement. Toutefois, les concepts des algorithmes proposés sont totalement fonctionnels et leurs premières implémentations (en C++), sont disponibles en ligne, sur la page <http://perso-laris.univ-angers.fr/~benoitr/contenu/singdim3online.zip> et peuvent être appliquées à toute fonction présentant une complexité formelle modérée.

4.5 Conclusions relatives à ce chapitre et à ses perspectives

4.5.1. Méthodologie générale

La méthodologie générale, proposée pour caractériser les ensembles d'intérêt d'une fonction f , consiste à :

- Décrire l'ensemble S_f (caractérisé par $\det(Df) = 0$ si $f : A \rightarrow \mathcal{P}$ lorsque $\dim(A) = \dim(\mathcal{P})$ et par « $\det(\mathcal{M}) = 0$, pour toute sous-matrice de Df de rang maximal » dans le cas général).
- Caractériser des classes de singularités $\mathcal{S}_v(f)$ qui vérifient, lorsque $f : A \rightarrow \mathcal{P}$, la propriété « pour toute fonction g , d'une famille de fonction, résiduelle pour $T^\infty(A, \mathcal{P})$, $\mathcal{S}_v(g) \neq \emptyset$ ».
- Caractériser des sous-ensembles V , des classes de singularités $\mathcal{S}_v(f)$ décrites, tels que l'image de V est restreinte à un point de $\mathcal{P}$.
- Caractériser les sous-ensembles V' qui partagent l'image de l'un des ensembles caractérisés précédemment (i.e. l'image d'une classe de singularités $\mathcal{S}_v(f)$ ou d'un sous-ensemble V).
- A partir des sous-ensembles extraits, il est possible de déduire des composantes connexes appartenant aux mêmes sous-ensembles isolés.

Les topologies des composantes connexes ultimement isolées étant invariantes pour l'équivalence $\sim$, cette information ouvre vers la constitution d'un objet invariant ou descripteur de f , vis-à-vis de $\sim$.

L'intérêt de la généricité augmente avec la dimension des espaces source et image de la fonction f . Toutefois, il reste pertinent de ne pas omettre une classe de singularités $\mathcal{S}_v(f)$ que l'on sait non vide, même si celle-ci est génériquement vide, puisque sa non vacuité permet notamment de distinguer la fonction f étudiée, de toutes les fonctions génériques entre les espaces source et but de f .

4.5.2. Utilisation de l'Analyse par Intervalles

Nous avons pu observer que l'Analyse par Intervalles ne se substitue pas à l'étude formelle.

La complémentarité de l'Analyse par Intervalles réside principalement dans son affranchissement de toute nécessité d'expressions formelles des solutions des équations caractéristiques d'intérêt.

Toutefois on remarque que l'Analyse par Intervalles ne permet pas de conclure dans certaines situations particulières, rendant cette méthode complémentaire à la notion de généricité.

En effet, il est trivial que $(M_a \in I \text{ et } M_b \in I) \not\Rightarrow M_a = M_b$.

Or, dans certains cas, conclure quant à l'égalité de points encadrés à l'aide de l'Analyse par Intervalles nécessiterait d'être en mesure de conclure que $M_a = M_b$, sachant seulement que $(M_a \in I \text{ et } M_b \in I)$.

Nous avons pu aussi remarquer que, lorsque l'on considère la méthode de Newton par Intervalles, pour détecter des ensembles de points isolés partageant une image, il est nécessaire de définir des tests d'injectivité locaux, afin de pouvoir conclure sur tout produit d'intervalles, non tous étrangers.

Plus généralement, les procédures basées sur les algorithmes élémentaires SIVIA ou IVIA permettent toujours d'obtenir un recouvrement de l'ensemble recherché.

L'Analyse par Intervalles permet donc d'isoler des ensembles étrangers aux ensembles d'intérêt.

Complémentairement, lorsque une méthode de Newton par Intervalles est applicable, il est possible d'extraire une liste d'intervalles, contenant chacun un unique point d'intérêt isolé.

Si la topologie locale d'un ensemble d'intérêt est connue, en considérant une précision suffisante, il est ensuite possible d'extraire, à partir du recouvrement obtenu, la topologie globale de l'ensemble d'intérêt étudié.

L'Analyse par Intervalles constitue ainsi une alternative pertinente, à l'étude théorique d'une fonction, de complexité formelle considérée élevée, pour un objectif de classification.

La capacité pratique à caractériser, des fonctions de toutes complexités formelles, renforce la pertinence d'une méthodologie générale retournant, à partir des topologies des composantes connexes isolées, un objet invariant ou descripteur de f , vis-à-vis de $\sim$.

Chapitre 5 Classification globale

La conception de nouveaux manipulateurs robotiques implique de pouvoir analyser et interpréter les comportements cinématiques de ceux-ci et *donc d'analyser leurs fonctions géométriques respectives*. De ce fait, notre objectif reste la caractérisation d'une fonction (géométrique) $f : A \rightarrow \mathcal{P}$, aux changements de repères près, sur l'espace (articulaire) A ou sur l'espace (opérationnel) $\mathcal{P}$ ou bien sur l'espace A et sur l'espace $\mathcal{P}$.

Nous avons déjà présenté, dans les chapitres précédents, et plus précisément au Chapitre 4, des méthodes de détection des lieux singuliers et de certains points singuliers spécifiques. Toutefois, comme nous l'avons précisé, la détection de classes de singularités va de pair avec la classification de la fonction par la topologie de ses diverses classes de lieux singuliers. De plus, il est à noter que la topologie des lieux singuliers n'offre pas une description complète de la fonction et doit être considérée conjointement avec l'action de la fonction elle-même. En effet, en particulier, deux fonctions (géométriques) peuvent partager une même topologie singulière tout en étant non équivalentes.

Ce chapitre propose donc de s'intéresser à une description combinatoire d'une fonction donnée. Les fonctions considérées agissent entre deux espaces de dimension 2, toutefois la méthodologie s'étend à toute dimension et la description combinatoire proposée ouvre la voie vers un invariant combinatoire, non encore défini clairement mais qui serait le descripteur de la classe des objets combinatoires possibles pour caractériser le comportement des fonctions (géométriques).

L'objet combinatoire proposé est une application $c_f : c_A \rightarrow c_{\mathcal{P}}$ entre complexes simpliciaux abstraits. La réalisation géométrique de c_f est une application d'une triangulation c_A de l'espace source (articulaire) A , vers une triangulation $c_{\mathcal{P}}$ d'un espace contenant l'image de A (l'espace opérationnel $\mathcal{P}$). Cette réalisation géométrique de c_f a la propriété d'être topologiquement conjuguée à f .

La constitution de cette méthodologie étant motivée par son applicabilité à la robotique, nous illustrons cette méthodologie en l'appliquant à différentes classes de comportements d'une famille de manipulateurs robotiques : *Les robots 3R (orthogonaux)*.

Il est à noter que, lorsque le robot associé à f admet des mouvements internes, f n'est pas strictement descriptible, à l'aide d'une application combinatoire telle que proposée initialement, et dont la réalisation serait topologiquement conjuguée à f .

Toutefois, f peut être décrite par une extension de la notion de triangulation à une nouvelle notion que nous appellerons *quadrilatation*, utilisant *une description topologique d'un éclatement (mathématique)* dans un espace de dimension 2, ainsi qu'un équivalent combinatoire de celui-ci.

5.1 Contribution vis-à-vis de la robotique

Du point de vue robotique, la contribution de ce chapitre se veut utile à la conception et à l'étude de manipulateurs robotiques novateurs. Précisons à nouveau que l'objectif de l'étude de manipulateurs novateurs est double, puisqu'il est de comprendre leurs comportements, principalement par l'intermédiaire des propriétés de leurs fonctions géométriques respectives, mais aussi de classer ces comportements qui dépendent des paramètres constitutionnels et cinématiques de ces manipulateurs.

Des comportements sont considérés équivalents lorsque les fonctions géométriques associées sont les mêmes à des changements de repères près, sur les espaces articulaires et les espaces de travail.

Cette équivalence est exactement l'équivalence $\sim$ définie par la Définition 3.1.1.3.

Le comportement des manipulateurs et de leurs fonctions géométriques peut être étudié par l'intermédiaire de diverses méthodes formelles telles que celles présentées dans (Wenger, 2007), (Pai, et al., 1992) et (Smith, et al., 1993) ou par l'intermédiaire de méthodes numériques comme détaillé dans (Merlet, 2011), (Bohigas, et al., 2012), (Bohigas, et al., 2013) et dans (Benoit, et al., 2015) qui présente la méthode développée à la section 4.3.

Ces méthodes permettent d'obtenir des informations sur le comportement d'un manipulateur, principalement sous forme d'information relative à ses singularités.

Chaque type d'information extrait de la fonction géométrique peut être conservé sous forme d'une construction, qui va d'un objet topologique à une quantité de points d'intérêt.

La contribution de ce chapitre est complémentaire aux approches existantes, incluant l'approche que nous avons développée au Chapitre 4. Celle-ci propose une approche qualitative qui s'appuie sur la notion d'*abstraction* (détaillée par la suite) qui constitue son innovation majeure.

Cette *abstraction* donne accès à une construction qui contient assez d'information pour décrire complètement le comportement de la fonction. De plus, étant donné que l'*abstraction* obtenue est un objet combinatoire, celle-ci peut être aisément traitée par ordinateur.

En quelques mots, l'abstraction c_f est une application de c_A vers c_p , qui sont deux complexes simpliciaux abstraits, tels que définis dans (Mauder, 1996), qui décrivent respectivement A et $\mathcal{P}$.

Si l'on reste colloquial, c_A , c_p et, finalement, c_f sont construits de telle sorte qu'ils décrivent la façon dont l'espace source A est « envoyé » dans $\mathcal{P}$ par $f : A \rightarrow \mathcal{P}$.

De l'abstraction proposée, l'utilisateur est en mesure d'extraire l'information topologique originale, de la fonction, relative à l'espace articulaire et à l'espace de travail. Parmi cette information extractible, on citera, dans l'espace de travail, la disposition relative des points cusps et des points nodes, le long de l'image du lieu singulier, nommé le *contour apparent*, dont la topologie peut elle aussi être extraite. Relativement à l'espace articulaire, la topologie des solutions géométriques inverses du contour apparent peut être extraite, incluant la disposition topologique relative des solutions géométriques inverses des points cusp et des points node. De plus, il est possible de séparer le lieu singulier et les *surfaces caractéristiques* (Wenger, 2004). Enfin, la topologie des aspects ainsi que les *régions et zones d'intérêt*, délimitées par les ensemble précédemment délimités, peuvent être extraites.

Seule une certaine partie de l'information, sur les singularités de la fonction, s'avère nécessaire, pour construire algorithmiquement l'*abstraction* que nous allons définir.

De plus, ces informations nécessaires peuvent être obtenues à l'aide de plusieurs méthodes qui sont présentées elles aussi dans les références précédemment citées (Merlet, 2011), (Bohigas, et al., 2012), (Bohigas, et al., 2013) et (Benoit, et al., 2015).

5.2 Plan du chapitre

La méthode algorithmique pour construire l'abstraction proposée constitue la contribution principale de ce chapitre.

Dans ce chapitre, nous nous proposons d'abord de définir rigoureusement les constructions et objets précédemment cités en introduction.

Ce chapitre est organisé selon les étapes suivantes :

La Section 5.4 définit la notion de *comportement* d'un robot et comment le caractériser. La notion d'*abstraction* y est aussi formellement définie. Puis, à partir de la définition de l'abstraction, les propriétés de ses réalisations géométriques sont soulignées.

Une méthode pour construire cet objet est donnée, pour une fonction $f : A \rightarrow \mathcal{P}$, entre deux espaces de dimension 2, en Section 5.5, à partir de la connaissance de la *trace singulière* (*constituée par le lieu singulier, son image et la pré-image de cette dernière*).

Plus précisément, à partir de la trace singulière notre processus algorithmique génère des cellules topologiques qui sont abstraites par des *complexes simpliciaux abstraits*.

Ces complexes simpliciaux abstraits sont construits selon des templates, explicités au § 5.5.3, qui sont paramétrés par la topologie de la cellule ainsi que par ses bords. Un tel paramétrage permet de prendre en compte le comportement global de la fonction. De ce fait, l'association de l'ensemble de ces constructions locales engendre naturellement un complexe simplicial global et cohérent.

A titre d'illustration de la méthode de construction proposée, celle-ci est mise en pratique pour trois exemples de comportements de manipulateurs 3R orthogonaux.

Les manipulateurs 3R orthogonaux illustrent idéalement la méthodologie proposée, du fait que ceux-ci peuvent être décrits par une fonction entre espaces de dimension 2, tel que le détaillent (Wenger, 2007), (Burdick, 1995) et (Wenger, 1998).

Certains comportements de manipulateurs 3R présentent des *mouvements internes*.

Dans ces cas-là, la fonction géométrique associée ne peut pas être décrite par cette première construction présentée, basée sur des *simplexes*.

(Un *simplexe* basé sur l'ensemble fini Ω est l'ensemble des n -uplets formés d'éléments distincts de Ω . Il est le composant élémentaire des complexes et généralise la notion de triangle.)

Toutefois, des mouvements internes peuvent apparaître pour une quantité, élevée et significative, de comportements et il est donc nécessaire de pouvoir décrire ces mouvements internes, afin de pouvoir décrire, de ce fait, un vaste panorama de comportements.

En définissant un objet combinatoire composé de simplexes et de « quadrilatères », nous sommes en mesure d'étendre la construction précédente afin de prendre en considération, à la Section 5.8, les cas, auparavant omis, de manipulateurs 3R, qui présentent des mouvements internes.

5.3 Résumé en images de la construction de l'abstraction

Nous proposons un aperçu de la construction de cet objet combinatoire à la Figure 20.

En quelques mots, la procédure proposée construit une représentation combinatoire d'une fonction f , par l'intermédiaire d'une application c_f entre *complexes simpliciaux abstraits*.

Plus précisément, c_f représente f , au sens où il existe des *homéomorphismes* φ et ψ tels que $f = \psi^{-1} \circ |c_f| \circ \varphi$ où $|c_f|$ est une réalisation géométrique de c_f .

f et $|c_f|$ sont ainsi *conjuguées* (on lira aussi parfois « topologiquement équivalentes ») i.e. $f \sim_0 |c_f|$.

Le formalisme encadrant c_f et $|c_f|$ et présenté à la Figure 20 sera développé dans la sous-section 5.4.2.

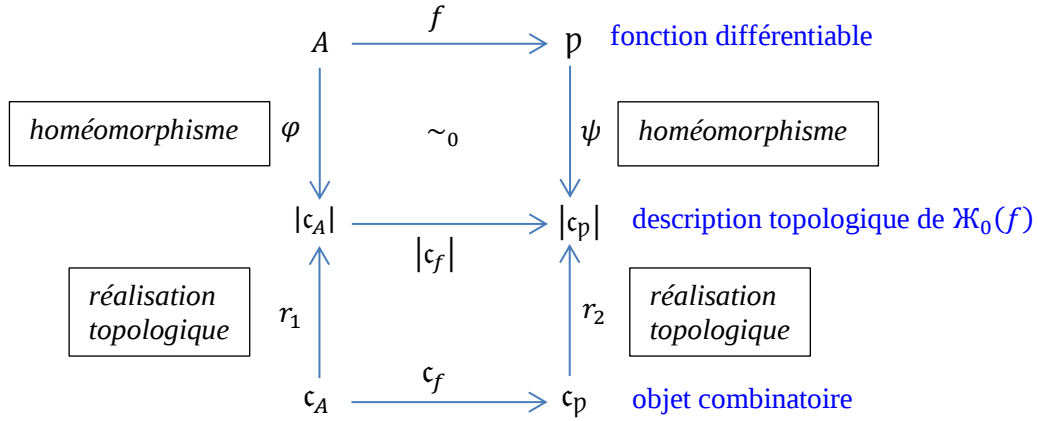


Figure 20 : Diagramme commutatif d'objets caractérisant f

Afin de présenter la logique et les propriétés de la méthode d'abstraction proposée, nous illustrons cette procédure par l'intermédiaire de son application à un exemple.

La fonction f choisie comme exemple présente la particularité de pouvoir être assimilée à la fonction géométrique d'un manipulateur 3R orthogonal, de par l'invariance de celui-ci selon son premier axe.

Avec les conventions du schéma cinématique de la Figure 21, les paramètres du robot décrit par f sont :

$$d_2 = 1, \quad d_3 = 2, \quad d_4 = 1.5, \quad r_2 = 1, \quad r_3 = 0.5$$

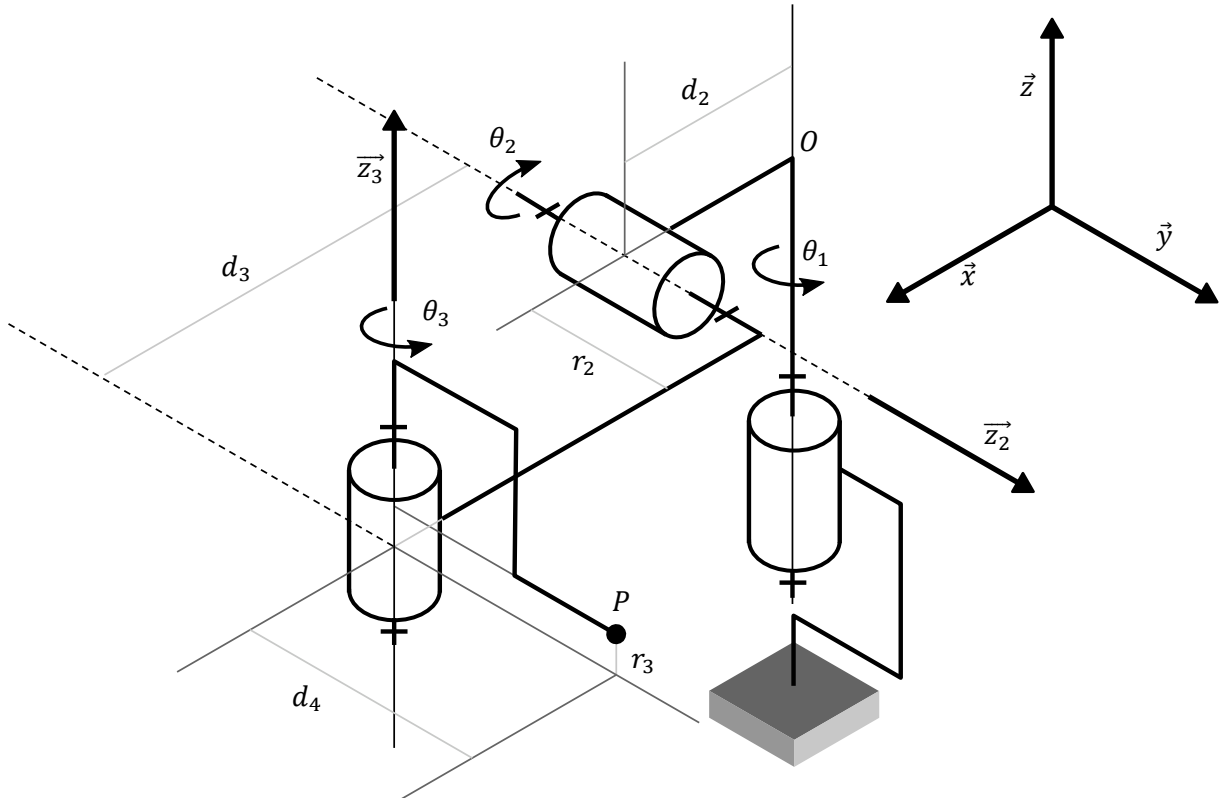
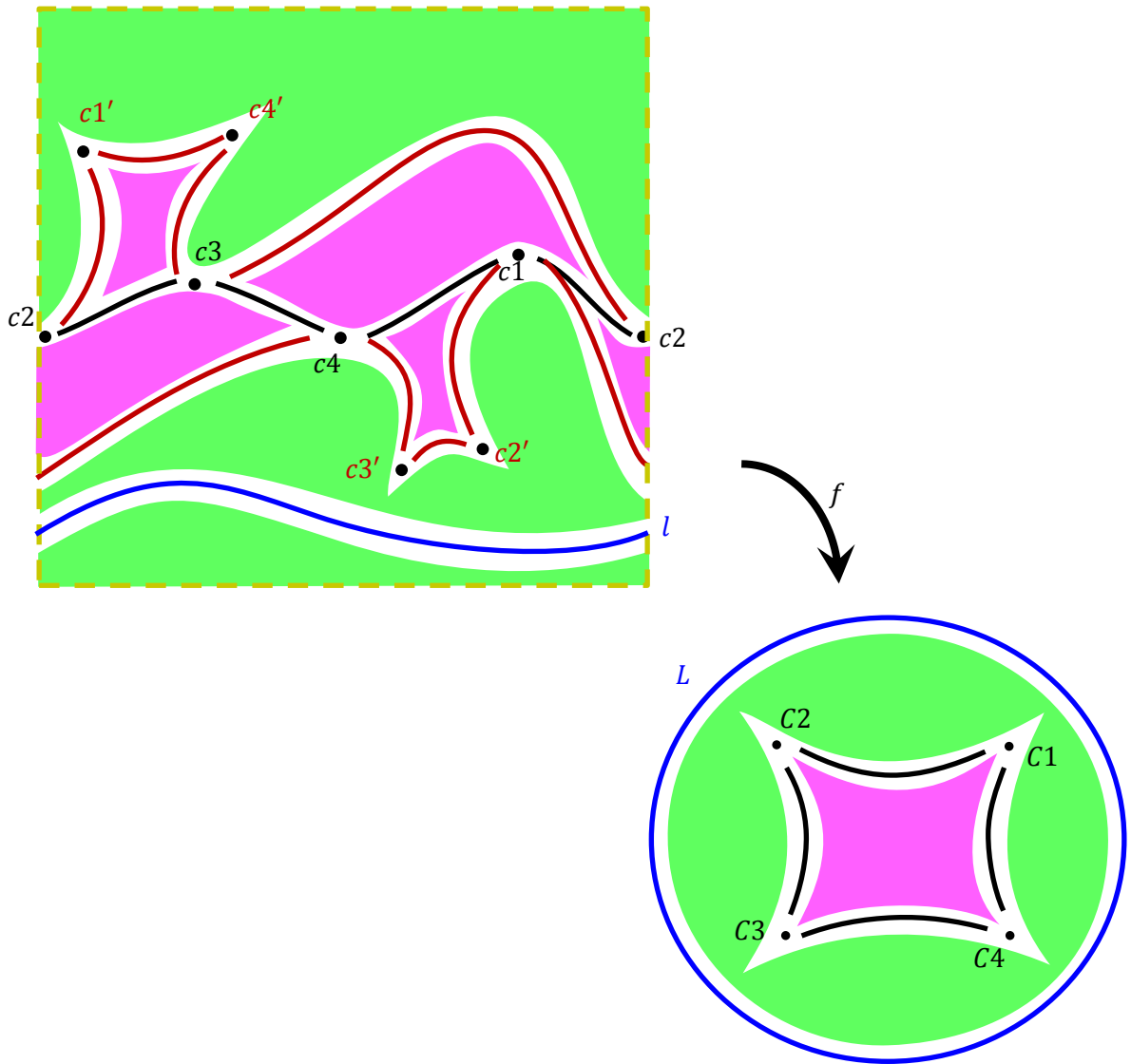


Figure 21 : Schéma cinématique d'un manipulateur 3R orthogonal.

La première étape consiste à isoler le *lieu singulier* S_f , son image par f : le *contour apparent* $f(S_f)$, ainsi que les solutions géométriques inverses de $f(S_f)$ qui se séparent alors entre le lieu singulier et les *courbes caractéristiques* (i.e. $f^{-1}(f(S_f)) - S_f$), qui sont des courbes régulières, définies dans (Wenger, 2004).

A l'aide des ensembles isolés, il est possible de définir une partition, de l'espace articulaire mais aussi de l'espace de travail et ceci, tout en faisant en sorte que, pour tout élément E de la partition de l'espace articulaire, $f(E)$ est un élément de la partition de l'espace de travail.

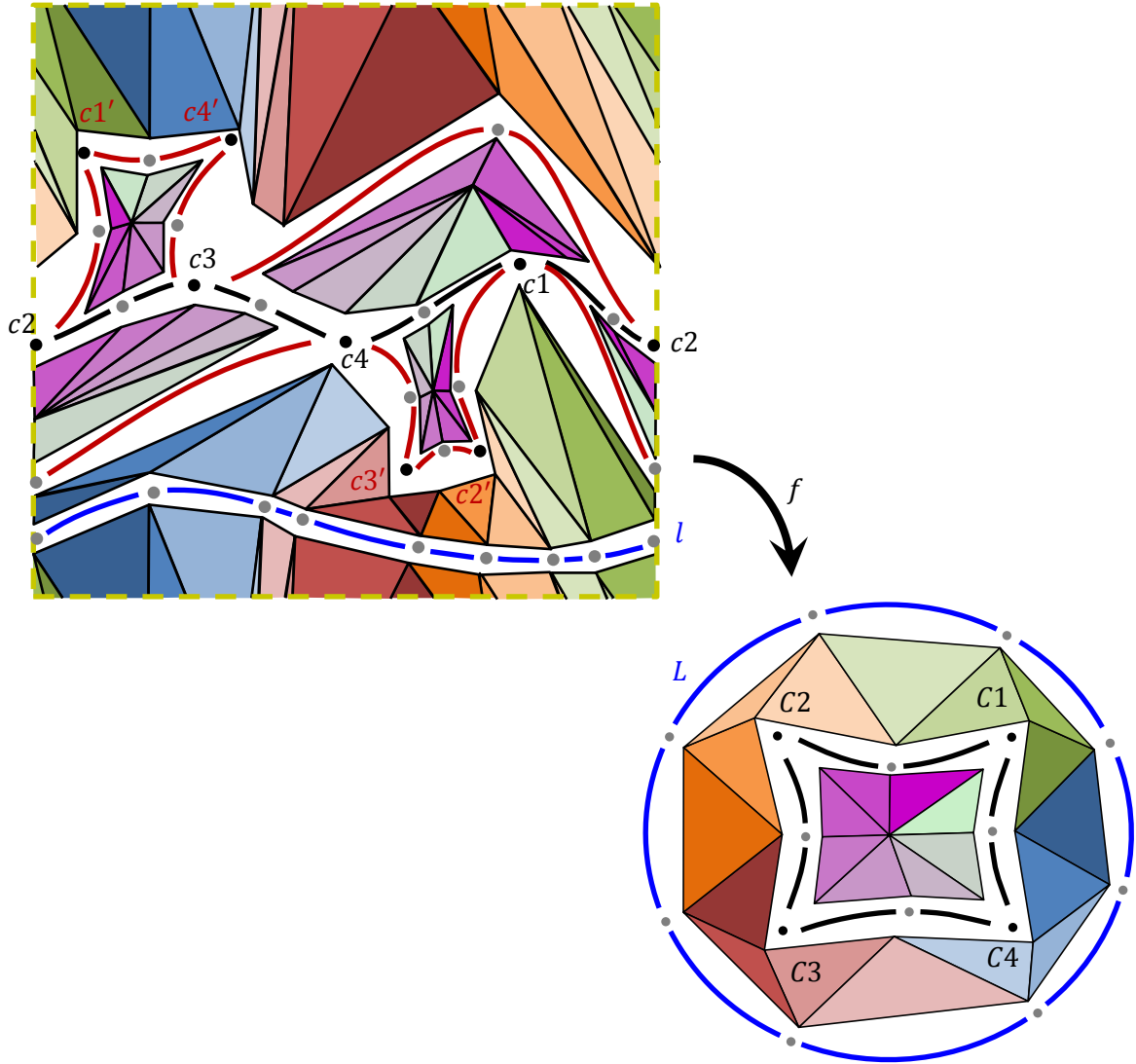
Ce couple de partitions est illustré à la Figure 22 où les points cusps $C1$, $C2$, $C3$ et $C4$ sont repérés et où leurs solutions géométriques inverses $c1$, $c2$, $c3$, $c4$ (singulières) et $c1'$, $c2'$, $c3'$, $c4'$ (non singulières) sont mises en avant.



$$\{c1, c1'\} \xrightarrow{f} C1 ; \{c2, c2'\} \xrightarrow{f} C2 ; \{c3, c3'\} \xrightarrow{f} C3 ; \{c4, c4'\} \xrightarrow{f} C4 ; l \xrightarrow{f} L$$

Figure 22 : Partitions des espaces articulaire et opérationnel selon la fonction f , décrivant le manipulateur représenté à la Figure 21

A partir de ces constructions, il est possible de créer une *triangulation* telle que définie dans (Maunder, 1996) représentée par la Figure 23 et telle que chaque triangle dans l'espace articulaire soit converti en un triangle de l'espace de travail, par l'action de f .



$$\{c1, c1'\} \xrightarrow{f} C1 ; \{c2, c2'\} \xrightarrow{f} C2 ; \{c3, c3'\} \xrightarrow{f} C3 ; \{c4, c4'\} \xrightarrow{f} C4 ; l \xrightarrow{f} L$$

Figure 23 : Triangulation des partitions de la Figure 22

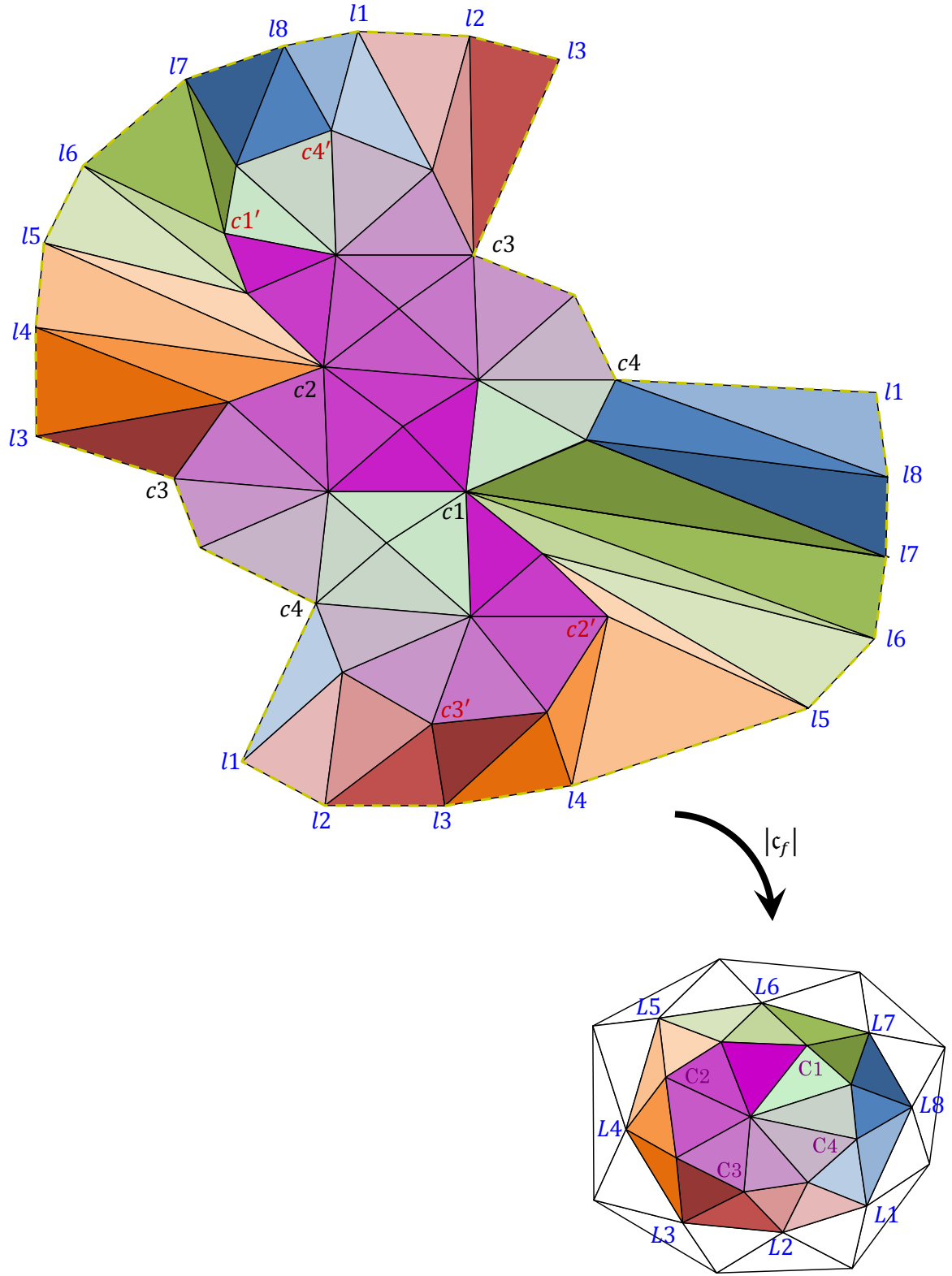
L'étape finale du processus est de conserver chaque triangulation sous forme d'un objet topologique connu sous le nom de *complexe simplicial abstrait* et de relier ces deux objets par une application : l'abstraction proposée c_f .

L'abstraction c_f admet une représentation géométrique, du fait que les complexes simpliciaux abstraits peuvent être interprétés géométriquement par des triangulations, ce que représente la Figure 24.

Cette application géométrique $|c_f|$ entre les deux triangulations est *conjuguée* à la fonction f .

Il est à noter que l'espace articulaire du manipulateur 3R est un tore et que, de ce fait, les côtés opposés de l'espace articulaire, tel que représenté à gauche de la Figure 22, de la Figure 23 et de la Figure 24, doivent être identifiés.

Cette remarque reste valide pour tous les espaces articulaires, dans la suite de ce chapitre.



$$\{c1, c1'\} \xrightarrow{|c_f|} C1 ; \{c2, c2'\} \xrightarrow{|c_f|} C2 ; \{c3, c3'\} \xrightarrow{|c_f|} C3 ; \{c4, c4'\} \xrightarrow{|c_f|} C4 ; \forall k \in \llbracket 1; 8 \rrbracket, lk \xrightarrow{|c_f|} Lk$$

Figure 24 : Représentation géométrique de l'abstraction du manipulateur exemple de la Figure 21

5.4 Bases théoriques

La construction proposée est une *abstraction* d'une fonction qui décrit son *comportement*, par l'intermédiaire d'une application combinatoire.

Afin de bien comprendre comment ce concept s'inclut dans notre logique de classification des fonctions et comment cette logique bénéficie au domaine robotique, nous commencerons par rappeler et par étendre certaines bases théoriques sur la caractérisation et la classification de fonctions.

Ainsi, nous clarifierons la notion de *comportement* et sa dualité vis-à-vis de l'équivalence $\sim$.

Puis, après avoir rappelé les concepts liés à l'invariant $\mathcal{K}_0$, nous introduirons les définitions formelles de l'*abstraction* et des notions théoriques qui lui sont connexes.

5.4.1. Caractérisation et Classification (remise en situation)

Nous classifions toujours les fonctions selon leur appartenance à une même classe d'équivalence, pour la relation $\sim$. En termes de robotique, comme nous l'avons déjà brièvement fait remarquer, cela correspond à ce que les fonctions géométriques des manipulateurs soit égales à des changements de repères près, dans les espaces *articulaires* et *opérationnels* (ce qui implique un changement de repère, dans l'espace de travail).

L'abstraction que nous proposons de construire est, en réalité et tout simplement, une représentation, pour les fonctions f entre variétés de dimension 2, de l'invariant $\mathcal{K}_0(f)$.

Nous avons déjà prouvé que, en particulier, la topologie du lieu singulier S_f , d'une quelconque fonction f différentiable, est un invariant pour $\sim$, de même que la topologie de $f(S_f)$.

Nous avons aussi pu vérifier que ces topologies invariantes ne contiennent pas toute l'information que l'on peut extraire de $\mathcal{K}_0(f)$ et que ces invariants ont donc certaines limites.

Toutefois, il faut noter que le *contour apparent* (i.e. $f(S_f)$) permet de vérifier l'existence de *nodes*, ce que ne permet pas l'observation du lieu singulier S_f seul.

En robotique, l'information sur les *nodes* et la connaissance de l'espace de travail sont primordiales pour maîtriser la pertinence d'un manipulateur. Ainsi, dans le cadre de la définition d'un objet unique caractérisant un manipulateur robotique, par l'intermédiaire de sa fonction géométrique f le contour apparent semble constituer une meilleure base que le lieu singulier S_f .

De plus, $f(S_f)$ peut être calculé numériquement, notamment de façon garantie, comme mis en avant dans (Delanoue, et al., 2014) et (Benoit, et al., 2015).

Dans un objectif de prise en compte de la partie articulaire de f , en plus de sa partie opérationnelle, décrite par $f(S_f)$, un invariant plus riche est présenté dans (Wenger, 2007) et consiste en un objet, $f_{\#}(S_f)$, constitué de $f(S_f)$ et d'entiers, un pour chaque composante connexe de $\mathcal{P}$, délimitée par $f(S_f)$, et tels que chacun correspond au nombre de préimages pour f de cette composante connexe.

Malheureusement, il est à noter que $f_{\#}(S_f)$ n'est pas aussi riche que $\mathcal{K}_0(f)$.

Ainsi, comme le montre la Figure 25, on peut avoir $g_{1\#}(S_{g_1}) = g_{2\#}(S_{g_2})$ et $g_1 \not\sim g_2$.

g_1 et g_2 ne sont ici pas équivalentes du fait d'une différence de topologies pour les préimages (notamment non singulières) du contour apparent. L'information sur ces topologies n'est pas totalement contenue dans $g_{1\#}(S_{g_1})$ et $g_{2\#}(S_{g_2})$. De ce fait, nous nous orientons de nouveau vers une méthodologie de construction d'un objet codant $\mathcal{K}_0(f)$ par l'intermédiaire des topologies des composantes connexes des espaces $f^{-1}(\mathcal{S}_v(f))$ et du codage de l'action de f sur ces espaces.

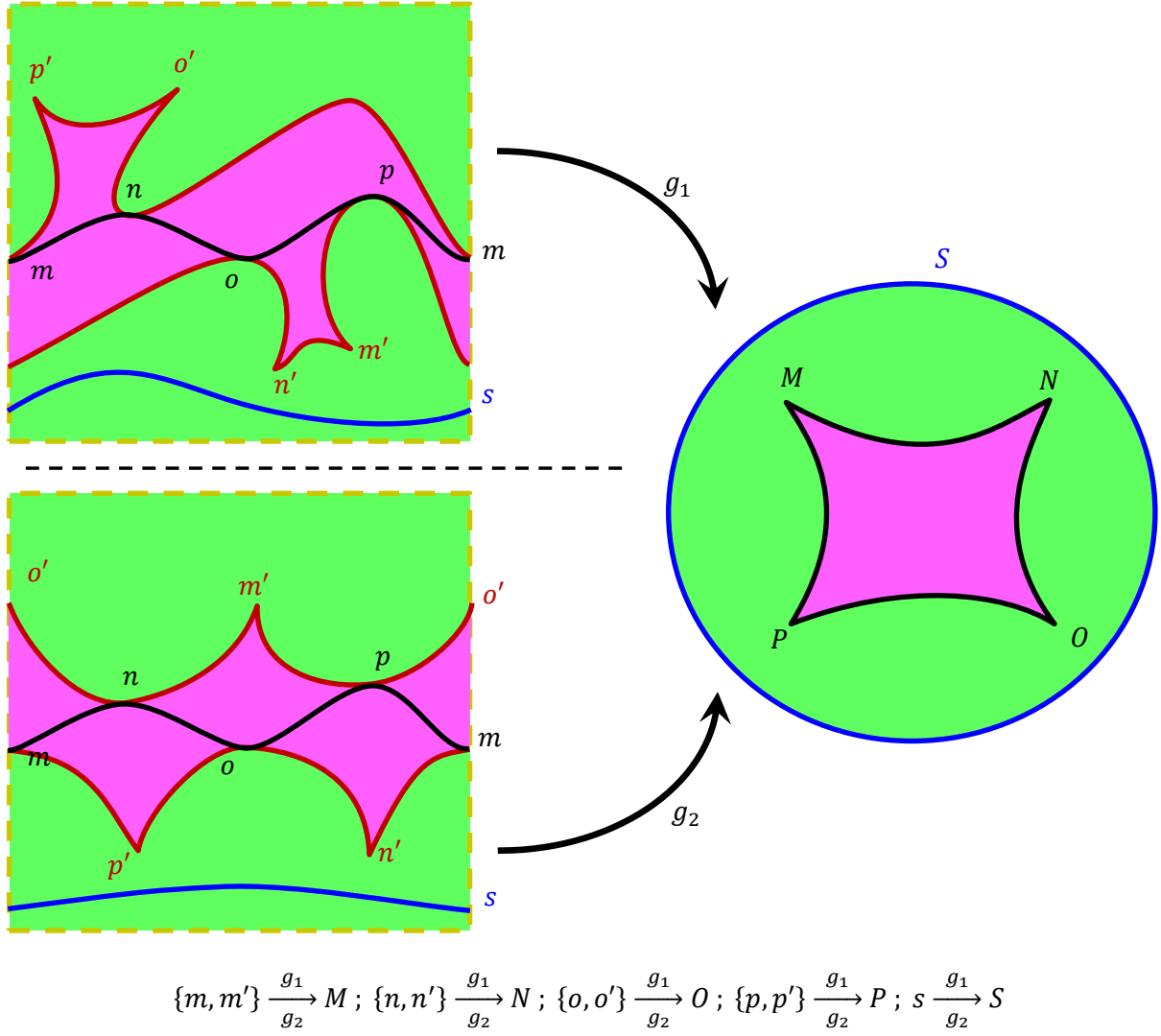


Figure 25 : Deux fonctions géométriques g_1 et g_2 distinctes et de contours apparents homéomorphes. g_1 est la fonction géométrique d'un robot série 3R et g_2 est la fonction géométrique d'un robot parallèle 3-RPR. g_1 et g_2 diffèrent par les pré-images de leurs contours apparents.¹

Les limitations du contour apparent, en tant qu'invariant topologique, nous motivent à considérer une approche similaire à celle de la construction de l'encrage et du portrait dans le § 3.2.3.2.

Toutefois, certaines simplifications que nous pouvions faire en dimension 1 ne sont pas valides pour les espaces de dimension 2.

Ainsi il est nécessaire de définir plus généralement et précisément les ensembles critiques.

De même, il est nécessaire de considérer les diverses topologies possibles des composantes connexes, délimitées par les ensembles critiques qui admettent eux aussi une plus grande variété de topologies.

Pour définir les ensembles critiques, en général, nous proposons de définir des *traces spécifiques* d'ordres croissants, caractérisant progressivement et conjointement la singularité des sous-ensembles du lieu singulier, du contour apparent mais aussi des *courbes caractéristiques*, non singulières mais dont la pertinence est mise en avant dans (Wenger, 2004).

¹ Nous remercions chaleureusement Michel Coste, qui nous a aimablement présenté les exemples de la Figure 25.

Définition 5.4.1.1 (Trace singulière)

La trace singulière $\mathcal{Z}(f)$ de la fonction géométrique f est définie comme étant le couple :

$$\mathcal{Z}(f) = \left(f^{-1} \left(f(S_f) \right), f(S_f) \right)$$

En premier aperçu, les traces spécifiques $(\mathcal{Z}_n(f))_{n \in \mathbb{N}}$ sont une suite de sous-ensembles de $\mathcal{Z}(f)$, décroissante pour l'inclusion et la régularité.

Définition 5.4.1.2 (Traces spécifiques)

Similairement aux classes de singularités $\mathcal{S}_\nu(f)$, les traces spécifiques $\mathcal{Z}_n(f)$ caractérisent la régularité et sont construites par récurrence.

Initialisation :

Soit une fonction géométrique $f : A \rightarrow \mathcal{P}$. Sa trace spécifique d'ordre 0 est sa trace singulière :

$$\mathcal{Z}_0(f) = \mathcal{Z}(f)$$

Récurrence :

$$\text{Posons } \mathcal{Z}_n(f) = \left(\mathcal{Z}_{n_a}(f), \mathcal{Z}_{n_p}(f) \right)$$

Alors $\mathcal{Z}_{n+1}(f) = \left(\mathcal{Z}_{n+1_a}(f), \mathcal{Z}_{n+1_p}(f) \right)$ est défini par :

$\mathcal{Z}_{n+1_a}(f)$ et $\mathcal{Z}_{n+1_p}(f)$ sont les plus petits ensembles tels que :

- $\mathcal{Z}_{n+1_a}(f) \supseteq \{z \in \mathcal{Z}_{n_a}(f) \mid \mathcal{Z}_{n_a}(f) \text{ n'est pas localement une variété en } z\}$
- $\mathcal{Z}_{n+1_p}(f) \supseteq \{z \in \mathcal{Z}_{n_p}(f) \mid \mathcal{Z}_{n_p}(f) \text{ n'est pas localement une variété en } z\}$
- $f^{-1}(\mathcal{Z}_{n+1_p}(f)) = \mathcal{Z}_{n+1_a}(f)$

Convention :

Pour simplifier les définitions et écritures, par la suite, nous définissons $\mathcal{Z}_{-1}(f) = (A, \mathcal{P})$

Dans la Figure 25, par exemple, on a $\forall k \in \llbracket 1; 2 \rrbracket$:

- $\mathcal{Z}_{1_p}(g_k) = \{M, N, O, P\}$ et $\mathcal{Z}_{1_a}(g_k) = \{m, n, o, p, m', n', o', p'\}$
- $\forall i > 1, \mathcal{Z}_{i_p}(g_k) = \emptyset$ et $\mathcal{Z}_{i_a}(g_k) = \emptyset$

Notre objectif est maintenant de définir une stratification, telle que définie dans (Mather, 1976) du couple $(A, \mathcal{P})$, selon la fonction géométrique $f : A \rightarrow \mathcal{P}$, à partir de $(\mathcal{Z}_n(f))_{n \in \mathbb{N} \cup \{-1\}}$

Définition 5.4.1.3 (Partition et Stratification d'un espace)

Une partition $\mathfrak{p}(E)$, de l'ensemble E , est un ensemble de sous-ensembles de E , tel que :

$$\forall M \in E, \exists ! \mathcal{E} \in \mathfrak{p}(E) \text{ tel que } M \in \mathcal{E}$$

Une stratification de l'espace topologique E est une partition $\mathfrak{p}(E)$ telle que, pour chacun des éléments $\mathcal{E} \in \mathfrak{p}(E)$, la dimension topologique m de $\mathcal{E}$ est constante en tout point de ce sous ensemble de E .

Chaque élément d'une stratification est appelé une strate (de dimension topologique m).

Définition 5.4.1.4 (Stratification tracée et ensembles de spécificités)

Soit $f : A \rightarrow \mathcal{P}$, une fonction géométrique.

Définissons tout d'abord, ses *ensembles de spécificités* d'ordre $k \in \mathbb{N}$, notés $\mathcal{E}_k(f)$ et formant la suite

$$\left(\mathcal{E}_k(f) = \left(\mathcal{E}_{k_a}(f), \mathcal{E}_{k_p}(f) \right) \right)_{k \in \mathbb{N}}, \text{ en posant :}$$

$$\forall k \in \mathbb{N}, \quad \forall z \in \{a, p\}, \quad \mathcal{E}_{k_z}(f) = \mathcal{Z}_{k-1_z}(f) - \mathcal{Z}_{k_z}(f)$$

On définit alors la *stratification tracée* $\mathfrak{s}(f) = (\mathfrak{s}_a(f), \mathfrak{s}_p(f))$ de f par :

$$\forall z \in \{a, p\}, \quad \mathfrak{s}_z(f) = \bigcup_{k \in \mathbb{N}} \{\text{composantes connexes de } \mathcal{E}_{k_z}(f)\}$$

Remarque 5.4.1.1 (Propriétés des stratifications tracées et des ensembles de spécificités)

Les ensembles de composantes connexes $\mathfrak{s}_a(f)$ et $\mathfrak{s}_p(f)$ définissent bien des stratifications d'espaces (de l'espace articulaire de f et de l'espace opérationnel de f , respectivement).

$\forall z \in \{a, p\}$, chaque strate $\mathfrak{E}_{k_z}(f)$ contient les points de spécificité k (ou « critiques d'ordre k »).

(i.e. il faut itérer $k + 1$ fois la recherche de points critiques (parmi les points critiques déjà isolés) pour que ces points apparaissent réguliers.)

Remarque 5.4.1.2 (Stratification tracée d'une fonction entre espaces de dimension 2)

Pour une fonction $f : A \rightarrow \mathcal{P}$, entre espaces de dimension 2 :

1. $\forall z \in \{a, p\}$, les composantes connexes de $\mathfrak{E}_{0_z}(f)$ sont usuellement des strates de dimension 2
2. $\forall (z, Z) \in \{(a, A), (p, \mathcal{P})\}$, $\delta\mathfrak{E}_{0_z}(f) \cap Z = \mathfrak{E}_{1_z}$ (où δE est le bord de E)
3. $\forall n \in \mathbb{N}, \forall (z, Z) \in \{(a, A), (p, \mathcal{P})\}$, toute composante connexe de $\mathfrak{E}_{n_z}(f)$, $\delta\mathfrak{E}_{n_z}(f) \cap Z$, se décompose usuellement en une liste de composantes connexes de $\{\mathfrak{E}_{k_z} \mid k \in \mathbb{N} \text{ et } k > n\}$

La définition et la construction, que nous proposons, de l'abstraction/du portrait d'une telle fonction f découle de l'observation de ces 3 remarques et de notre tentative d'en interpoler un objet descriptif.

Remarque 5.4.1.3 (Construire un invariant topologique ou un objet combinatoire ?)

A partir de l'information contenue dans $\mathfrak{s}(f)$ et de l'action de f , il est possible de s'orienter soit vers :

- Une représentation « canonique » de $\mathcal{K}_0(f)$, par un objet de la théorie des graphes, où chaque nœud est annoté d'une topologie et de son appartenance à l'un des deux espaces et où chaque nœud est connecté aux nœuds associés à ses bords et à son image.
- Une représentation *combinatoire* de l'information contenue dans $\mathfrak{s}(f)$ et de l'action de f , non univoque, mais de structure simple, interprétable et qui peut être analysée par ordinateur.

La suite de ce chapitre s'oriente vers la représentation combinatoire.

Toutefois la question suivante demeure :

Quelle est la différence d'information entre la représentation combinatoire de f et $\mathcal{K}_0(f)$?

Notre objectif étant de construire, algorithmiquement, à partir de $\mathfrak{s}(f)$, un objet combinatoire c_f , constitué par des associations, en n -upplets, de labels, nous proposons de définir un objet hybridant des associations de labels et des stratifications d'espace : la stratification tracée labélisée.

Définition 5.4.1.5 (Fonction de labélisation et Stratification tracée labélisée)

La labélisation $\mathcal{L}(E)$ d'un ensemble E se définit par :

$$\mathcal{L}(E) = \{(z, l_z) \mid z \in E\} \text{ où } \forall z \in E, l_z \text{ est un label}$$

La labélisation d'un ensemble par $\mathcal{L}$ s'assimile à son indexation par l'ensemble des labels utilisés.

Soit $\mathfrak{s}(f) = (\mathfrak{s}_a(f), \mathfrak{s}_p(f))$, la stratification tracée de $f : A \rightarrow \mathcal{P}$

La stratification tracée labélisée est alors définie comme $\mathcal{L}(\mathfrak{s}(f)) = (\mathcal{L}(\mathfrak{s}_a(f)), \mathcal{L}(\mathfrak{s}_p(f)))$.

Remarque 5.4.1.4 (Motivation et intérêt du couplage labélisation et stratification)

On notera d'abord que les objets topologiques définis dans cette sous-section préparent la description de chaque fonction d'intérêt f et, plus précisément, sa description par l'intermédiaire de $\mathcal{K}_0(f)$.

La définition d'un objet tel que $\mathcal{L}(\mathfrak{s}(f))$ est motivée par les développements suivants de ce chapitre.

En effet, notre stratégie consiste à convertir les éléments topologiques de $\mathfrak{s}(f)$ en *simplexes abstraits*, des *objets combinatoires* construits à partir d'ensembles de symboles, selon des templates choisis, de façon à ce que ces objets combinatoires décrivent complètement f , tout en pouvant être facilement traités par ordinateur. La construction $\mathcal{L}(\mathfrak{s}(f))$ et la fonction $\mathcal{L}$ permettent alors de formaliser et de rendre triviale cette conversion, que nous définirons exhaustivement, par la suite.

5.4.2. Triangulations et complexes simpliciaux abstraits

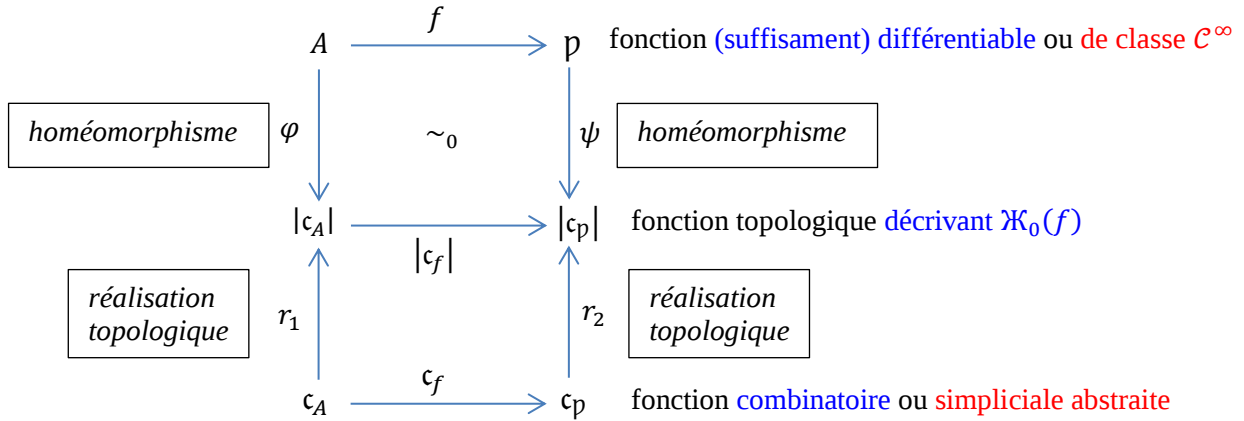


Figure 26 : Diagramme commutatif détaillé d'objets caractérisant f

Figure 26 détaille à la fois le modèle de construction choisi, dans son cas général et le premier cas particulier de construction considéré, formellement défini dans cette sous-section.

Ainsi, pour la Figure 26 :

- Les qualificatifs en **bleu** sont les qualificatifs pour le modèle général de construction proposé.
- Les qualificatifs en **rouge** sont ceux que nous considérons pour cette première construction

La définition que nous proposons d'un objet combinatoire est assez intuitive.

Définition 5.4.2.1 (Objet combinatoire et Application combinatoire)

Un objet est combinatoire s'il peut être écrit sous forme d'un ensemble dénombrable d'éléments, où chacun de ces éléments est lui-même un ensemble d'éléments plus basiques.

Soit l'objet combinatoire $\{E \ni v_j \mid j \in J\}$ où J est un ensemble d'index.

Alors pour tout j de J , v_j code une combinaison valide d'éléments de l'ensemble E .

Une application combinatoire est une application entre objets combinatoires.

De plus, en considérant l'action de l'application combinatoire comme un ensemble de deux ensembles (l'un étant dans l'espace source et l'autre dans l'espace but) on vérifie que :

Une application combinatoire est bien un objet combinatoire (admettant une certaine structure).

c_f est l'abstraction proposée dans ce chapitre pour décrire le comportement d'une fonction f .

Cette application combinatoire sera construite de telle sorte que sa réalisation géométrique $|c_f|$ soit conjuguée à f (i.e. $f \sim_0 |c_f|$).

Les définitions formelles d'un *simplexe abstrait*, d'un *complexe simplicial abstrait*, d'une triangulation et des applications entre ces éléments se retrouvent dans (Maunder, 1996).

Nous proposons, ci-après, un résumé des définitions et propriétés essentielles de ces concepts.

Définition 5.4.2.2 (simplexe et complexe simplicial abstraits)

Un *simplexe abstrait* s est engendré par un n -upplet de symboles $z = \{s_k \mid k \in \llbracket 1; n \rrbracket\}$, et est défini comme l'ensemble des parties de z , ou, écrit mathématiquement, $s = \mathcal{P}(z)$.

On écrira par la suite, pour plus de simplicité, $s = \langle z \rangle$.

Un *complexe simplicial abstrait* c est une union de simplexes abstraits.

Définition 5.4.2.3 (simplexe et complexe simplicial géométriques)

Un *simplexe géométrique* $r(s)$ est un objet représentant géométriquement le simplexe abstrait s :

- A chaque symbole de $z = \{s_k \mid k \in \llbracket 1; n \rrbracket\}$, tel que $s = \langle z \rangle$ est associé un point M_k tel que : $\mathcal{V}(\{M_k \mid k \in \llbracket 1; n \rrbracket\})$ est un espace de dimension $n - 1$
- Les éléments descripteurs de $r(s)$, de dimension j sont exactement les enveloppes convexes des $(j + 1)$ -upplets de points parmi $\{M_k \mid k \in \llbracket 1; n \rrbracket\}$.
(i.e. les sommets de $r(s)$ sont les points $M_k \mid k \in \llbracket 1; n \rrbracket$, ses arrêtes sont les segments $[M_k M_i] \mid k \in \llbracket 1; n \rrbracket, i \in \llbracket k + 1; n \rrbracket$, ses faces sont les triangles dont les sommets sont les triplets de points $\{M_k, M_i, M_l\} \mid k \in \llbracket 1; n \rrbracket, i \in \llbracket k + 1; n \rrbracket, l \in \llbracket i + 1; n \rrbracket$ et ainsi de suite...)

Un *complexe simplicial géométrique* est une union des simplexes géométriques $\bigcup_{m \in J \subset \mathbb{N}} r(s_m)$.

Ce complexe réalise géométriquement le complexe simplicial abstrait défini par $\bigcup_{m \in J \subset \mathbb{N}} s_m$

Remarque 5.4.2.1 (réalisations d'un complexe simplicial abstrait homéomorphes)

Il est à noter que toutes les réalisations d'un complexe simplicial abstrait sont homéomorphes.

Ainsi, un tel homéomorphisme $\phi : r_1(c) \rightarrow r_2(c)$, où $c = \bigcup_{i \in I \subset \mathbb{N}} z_i$ (union d'ensembles de symboles), conserve aussi les réalisations des ensembles de symboles z_i , au sens où $\forall i \in I, \phi(r_1(z_i)) = r_2(z_i)$ et où $\forall i \in I, \phi : r_1(z_i) \rightarrow r_2(z_i)$ est un homéomorphisme.

Cette propriété est démontrée dans (Maunder, 1996) et implique que l'on peut définir une application « réalisation topologique » ($c \mapsto |c|$) qui, à un complexe c , associe la classe des réalisations de c .

De plus, l'image $|c|$ d'un complexe c par l'application ($c \mapsto |c|$) peut être représentée par n'importe quelle réalisation géométrique de c , sans perte d'information sur ce complexe simplicial.

Enfin, il est même possible de définir une « topologie » sur c , en l'assimilant à la topologie de $|c|$.

Définition 5.4.2.4 (Application entre complexes simpliciaux)

Notons $\mathcal{Y}(c)$ l'ensemble des symboles apparaissant dans au moins un élément de c .

Dans tout ce chapitre, une application $g : c_1 \rightarrow c_2$ entre complexes simpliciaux abstraits sera toujours implicitement définie à partir d'une application $\mathcal{Y}(g) : \mathcal{Y}(c_1) \rightarrow \mathcal{Y}(c_2)$, de la façon suivante :

$$\forall z = \{s_k \mid k \in \llbracket 1; n \rrbracket\} \in c_1, \quad g(z) = \{[\mathcal{Y}(g)](s_k) \mid k \in \llbracket 1; n \rrbracket\}$$

(g doit vérifier aussi, par définition : $\forall z \in c_1, g(z) \in c_2$.)

En choisissant de se restreindre à de telles fonctions entre complexes, nous nous assurons que toute fonction $g : c_1 \rightarrow c_2$ respecte l'inclusion (i.e. $\forall (z', z) \in c_1^2, z' \subseteq z \Rightarrow g(z') \subseteq g(z)$).

Enfin, on définit une réalisation de $g : c_1 \rightarrow c_2$ comme étant une fonction continue $|g| : |c_1| \rightarrow |c_2|$ telle que : $\forall z \in c_1, |g|(|z|) = |g(z)|$.

Définition 5.4.2.5 (Abstraction et Triangulation)

Soit un espace E .

Soit un complexe simplicial c_E , dont la réalisation géométrique $|c_E|$ est homéomorphe à l'espace E .

On qualifiera alors :

- c_E d'abstraction de E
- $|c_E|$ de triangulation de E

Par extension :

Soit $f : E \rightarrow F$ et c_E et c_F , des abstractions respectives de E et de F définies comme précédemment.

Soit une application $c_f : c_E \rightarrow c_F$ telle que une réalisation $|c_f|$ vérifie $|c_f| \sim_0 f$.

On qualifiera alors :

- c_f d'abstraction de f
- $|c_f|$ de triangulation de f

Remarque 5.4.2.2 (Bilan sur le concept de description proposé)

Nous proposons donc que la fonction combinatoire c_f de la Figure 26 soit une abstraction de f .

On notera que construire l'abstraction c_f requiert d'abstraire E et F selon l'action de f entre eux.

Ceci justifie le développement d'un algorithme capable de générer une telle abstraction c_f .

5.5 Construction d'une abstraction de f

La construction de l'abstraction $c_f : c_A \rightarrow c_P$, telle que définie par la Figure 26, peut être fractionnée en plusieurs étapes selon le processus évolutif décrit par la Figure 27.

Ces étapes sont détaillées, dans les sous-sections suivantes, en particulier l'étape 2 afin d'explicitier la nature algorithmique de la construction de l'abstraction.

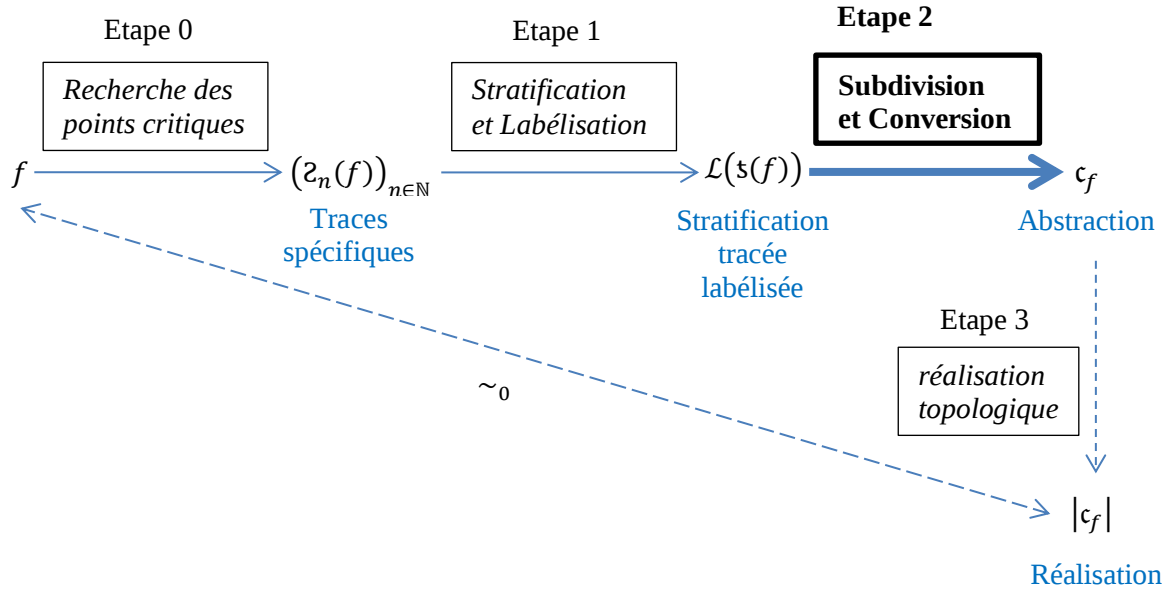


Figure 27 : Plan du processus de construction d'une abstraction c_f de f

On notera qu'il est nécessaire d'avoir accès à la définition de la fonction f , tout au long du processus, puisqu'il est nécessaire de conserver le lien entre les parties articulaires et opérationnelles des constructions intermédiaires $(\mathcal{Z}_n(f))_{n \in \mathbb{N}}$ et $\mathcal{L}(\mathfrak{s}(f))$.

5.5.1. Etape 0 : Isoler les points et ensembles critiques

Cette étape peut être effectuée par l'intermédiaire de diverses techniques, le choix de celle-ci est libre quoi qu'il dépende des contraintes imposées et des technologies accessibles à l'utilisateur pour l'étude du comportement de la fonction géométrique d'intérêt.

5.5.1.1. Etape 0a : Isoler la trace singulière $\mathcal{Z}(f) = \mathcal{Z}_0(f)$

Afin d'initialiser la construction de $(\mathcal{Z}_n(f))_{n \in \mathbb{N}}$, il est nécessaire d'isoler la trace singulière $\mathcal{Z}(f)$.

Pour ce faire, le lieu singulier S_f doit être déterminé, à l'aide de la technique choisie par l'utilisateur. Puis, le contour apparent $f(S_f)$ doit être calculé à partir de l'ensemble S_f déterminé. Enfin, il est nécessaire de compléter la partie articulaire de $\mathcal{Z}(f)$ par la détermination des courbes caractéristiques, en calculant $f^{-1}(f(S_f))$, c'est-à-dire les solutions géométriques inverses du contour apparent.

5.5.1.2. Etape 0b : Isoler les traces spécifiques $(\mathcal{Z}_n(f))_{n > 0}$

Une fois la trace singulière isolée, il est possible d'isoler, successivement, les éléments de $(\mathcal{Z}_n(f))_{n > 0}$ de façon à analyser le plus précisément possible l'action de f sur des sous-ensembles de $\mathcal{Z}(f)$.

En s'appuyant sur la Définition 5.4.1.2, par récurrence, de $(\mathcal{Z}_n(f))_{n \in \mathbb{N}} = \left(\left(\mathcal{Z}_{n_a}(f), \mathcal{Z}_{n_p}(f) \right) \right)_{n \in \mathbb{N}}$,

on déduit que pour tout entier n , pour construire $\mathcal{Z}_{n+1}(f) = \left(\mathcal{Z}_{n+1_a}(f), \mathcal{Z}_{n+1_p}(f) \right)$ il suffit d'isoler, pour chaque $z \in \{a, p\}$, l'ensemble E_z des points M de $\mathcal{Z}_{n_z}(f)$ tels que $\mathcal{Z}_{n_z}(f)$ n'est pas, localement, une variété au point M .

$\mathcal{Z}_{n+1}(f)$ est alors obtenu en posant, respectivement :

- Dans l'espace articulaire : $\mathcal{Z}_{n+1_a}(f) = E_a \cup \left(f^{-1}(E_p) \right)$
- Dans l'espace opérationnel : $\mathcal{Z}_{n+1_p}(f) = E_p \cup \left(f(E_a) \right)$

Dans le cas d'une fonction entre espaces de dimensions 2, les ensembles isolés pour construire $\mathcal{Z}_1(f)$ seront usuellement des nodes (i.e. des croisements sur $f(S_f)$) et des cusps (le long de $f(S_f)$).

On notera tout de même que seul un nombre fini d'ensembles $\mathcal{Z}_n(f)$ est à construire, en pratique.

En effet, lorsque $(\mathcal{Z}_n(f))_{n \in \mathbb{N}}$ devient stationnaire, il n'est plus nécessaire d'itérer la recherche de sous-ensembles critiques d'ordres supérieurs car cela revient à répéter indéfiniment la même itération.

Remarque 5.5.1.1 (Fonction d'expression formelle « compliquée »)

Lorsque l'on considère une fonction dont l'expression formelle induit une recherche difficile, par des méthodes formelles, des ensembles critiques considérés dans cette sous-section, l'utilisation de méthodes numériques peut être envisagée, notamment celles basées sur l'Analyse par Intervalles.

L'existence de telles fonctions motive le développement de méthodes basées sur cette approche.

On citera notamment les développements, réalisés par les contributeurs à cette thèse, tels que (Benoit, et al., 2015) et (Delanoue, et al., 2014).

Complémentairement, (Imbach, et al., 2015) propose un autre point de vue sur l'utilisation de ces méthodes numériques, pour un objectif connexe de caractérisation de courbes.

5.5.2. Etape 1 : Construire la stratification labellisée tracée

Une fois la suite $(\mathcal{Z}_n(f))_{n \in \mathbb{N}}$ connue, la construction de la stratification tracée $\mathfrak{s}(f)$ est naturelle, d'après sa définition (Définition 5.4.1.4). De même, la labélisation d'un ensemble est triviale.

On rappelle tout de même que construire $\mathfrak{s}(f)$ revient à considérer, pour tout entier n , l'ensemble des composantes connexes (dans l'espace articulaire et dans l'espace opérationnel) de $\mathcal{Z}_n(f) - \mathcal{Z}_{n-1}(f)$.

De plus, ici encore, on notera que si $(\mathcal{Z}_n(f))_{n \in \mathbb{N}}$ est stationnaire à partir d'un certain indice, alors cette décomposition en composantes connexes n'est à réaliser qu'un nombre fini de fois.

Pour une fonction f , entre espaces de dimensions 2, $\mathcal{L}(\mathfrak{s}(f))$ est usuellement constituée dans l'ordre décroissant des dimensions, de disques, de cylindres, de cercles, d'arcs et de points (tous labélisés).

5.5.3. Etape 2 : Créer l'abstraction

Pour créer une abstraction de f , notre stratégie est, ici, de créer un complexe simplicial abstrait précis, pour l'espace articulaire et pour l'espace opérationnel, afin de définir une application combinatoire c_f les liant et ayant des propriétés contrôlées qui font d'elle un descripteur riche de f .

Pour ce faire, comme nous l'avons déjà annoncé, nous nous basons sur $\mathcal{L}(\mathfrak{s}(f))$, et créons, pour chaque élément de cette stratification labélisée, un complexe simplicial abstrait représentant ce dernier, mais aussi son interaction avec ses bords.

Ces divers complexes sont alors recollés de façon à définir deux complexes globaux, qui, liés par l'action induite de f fournissent l'abstraction voulue. Afin que ce recollement global soit possible et cohérent, il est nécessaire de construire chaque complexe simplicial abstrait avec soin.

La suite de cette sous-section contiendra donc, la trame explicite d'un algorithme d'abstraction à partir de $\mathcal{L}(\mathfrak{s}(f))$, afin de donner forme au concept théorique proposé d'algorithme d'abstraction.

Par souci de complétude, nous proposons des méthodes modèles pour générer un complexe simplicial abstrait représentant chaque topologie usuelle dans un espace de dimension 2.

Nous proposons donc, en particulier, un modèle pour représenter avec un objet combinatoire un point, un arc, un cercle, un disque, un cylindre et un disque troué n fois. En utilisant ces modèles conjointement à la trame d'algorithme proposé, le lecteur devrait être en mesure de construire l'abstraction de toute fonction f ne présentant que ce type de topologie dans $\mathfrak{s}(f)$, et telle que $f : E \rightarrow f(E)$ est bijective, pour tout élément E de $\mathfrak{s}_a(f)$.

5.5.3.1. Algorithme d'abstraction

L'algorithme d'abstraction requérant de définir un template pour abstraire chaque topologie qui apparaît dans les stratifications $\mathfrak{s}_z(f)$ où $z \in \{a, p\}$, toujours extractibles de $\mathcal{L}(\mathfrak{s}(f))$, il apparaît alors nécessaire que $\forall z \in \{a, p\}, \forall E \in \mathfrak{s}_z(f)$, E vérifie les conditions imposées, sur sa topologie et la topologie de ses bords, par au moins l'un des templates prédéfinis.

Toutefois, il est possible de décrire toute partition $\mathfrak{p}_0(Z)$ de Z par une sous-partition $\mathfrak{p}_1(Z)$, i.e. $(\forall F \in \mathfrak{p}_1(Z), \exists E \in \mathfrak{p}_0(Z) \mid F \subseteq E)$ et $(\forall E \in \mathfrak{p}_0(Z), E = \bigcup_{i \in I} F_i \text{ où } \forall i \in I, F_i \in \mathfrak{p}_1(Z))$.

Les stratifications sont, par définition, des partitions dont les éléments ont la particularité d'avoir une dimension topologique constante en chacun de leurs points.

On propose donc d'appliquer l'algorithme d'abstraction à une *stratification tracée adaptée labélisée* $\mathcal{L}(\mathfrak{t}(f))$ où $\mathfrak{t}(f)$ est une sous-stratification de $\mathfrak{s}(f)$ conservant la compatibilité avec la fonction f .

En posant $\mathfrak{t}(f) = (\mathfrak{t}_a(f), \mathfrak{t}_p(f))$, ceci implique que la construction $\mathfrak{t}(f)$ doit alors vérifier:

1. $\forall z \in \{a, p\}, \forall E \in \mathfrak{t}_z(f), \exists F \in \mathfrak{s}_z(f) \text{ tel que } E \subseteq F$
2. $\forall E \in \mathfrak{t}_a(f), f(E) \in \mathfrak{t}_p(f)$
3. $\forall F \in \mathfrak{t}_p(f), f^{-1}(F) = \bigcup_{i \in I} E_i \text{ où } \forall i \in I, E_i \in \mathfrak{t}_a(f)$

La partie majeure de la préparation de $\mathcal{L}(\mathfrak{t}(f))$ pour que l'algorithme lui soit trivialement applicable réside donc dans la subdivision de $\mathfrak{s}(f)$, de telle sorte que les conditions imposées par les templates de représentations de topologies et d'interaction avec leur bords soit satisfaites.

Il est alors suffisant de re-subdiviser les topologies articulaires et opérationnelles obtenues, de telle sorte que les propriétés (2) et (3) précédentes soient vérifiées.

Le principe de l'algorithme repose sur :

- Construire les complexes décrivant chaque élément de $\mathfrak{t}(f)$
- Construire la restriction de c_f aux labels isolés/aux singletons

Il est à noter que c_p est belle et bien l'abstraction de l'espace opérationnel $\mathcal{P}$ et non celle de l'espace de travail accessible $W = f(A) \subseteq \mathcal{P}$, image de l'espace articulaire A . De ce fait, c_p distingue les bords d'un trou dans l'espace de travail W d'un bord extérieur de ce même espace de travail $W \subseteq \mathcal{P}$.

5.5.3.2. Abstraction d'une topologie admettant un bord hors de $\mathfrak{s}(f)$

Notre concept d'abstraction d'un élément topologique labélisé requiert que son bord se décompose en éléments appartenant à la stratification labélisée. Or, certaines topologies de la stratification $\mathfrak{s}(f)$ peuvent vérifier qu'une part du bord δE , de l'élément E associé, ne soit pas un élément de $\mathfrak{s}(f)$.

(i.e. cette part de bord n'est pas incluse dans l'espace A ou $\mathcal{P}$, où s'inclut l'élément topologique.)

Dans ce cas, l'abstraction consistera alors en celle d'un élément E' formé par l'intersection de E avec un disque de rayon suffisant pour que toute intersection de E avec un disque de plus grand rayon possède la même topologie que E' . (On utilisera alors le template approprié à E')

Cette procédure sera, par exemple, usuellement nécessaire afin d'abstraire l'extérieur de l'espace de travail accessible d'un robot considéré, qui peut contenir, par exemple lorsque l'on pose $\mathcal{P} = \mathbb{R}^2$, un espace, topologiquement identifiable à $\mathbb{R}^2$, privé d'un disque.

On notera alors que $\mathfrak{t}_a(f)$ et $\mathfrak{t}_p(f)$ sont les stratifications des espaces A et $\mathcal{P}$, *restreints au minimum* afin que leur abstraction soit possible, tout en permettant à $\mathfrak{t}(f)$ de vérifier les trois propriétés nécessaires.

5.5.3.3. Abstraction d'un cercle

Un cercle peut seulement être abstrait en tant que deux arcs ou plus, connectés en boucle fermée par un même nombre de points. Cela implique, pour construire l'abstraction et donc $\mathfrak{t}(f)$, à partir de $\mathfrak{s}(f)$, de séparer chaque cercle de $\mathfrak{s}(f)$ en un ensemble d'arcs et de points de connection.

5.5.3.4. Abstraction d'un disque

Nous proposons d'abstraire un disque topologique à l'aide du complexe simplicial abstrait, généré par l'ensemble des triplets qui regroupent le label marquant le disque topologique, un label marquant un arc de son bord et un label marquant l'un des deux points de bord de l'arc choisi.

Pour faciliter la visualisation de cette construction, la Figure 28 présente un exemple de réalisation géométrique pour l'abstraction d'un disque topologique.

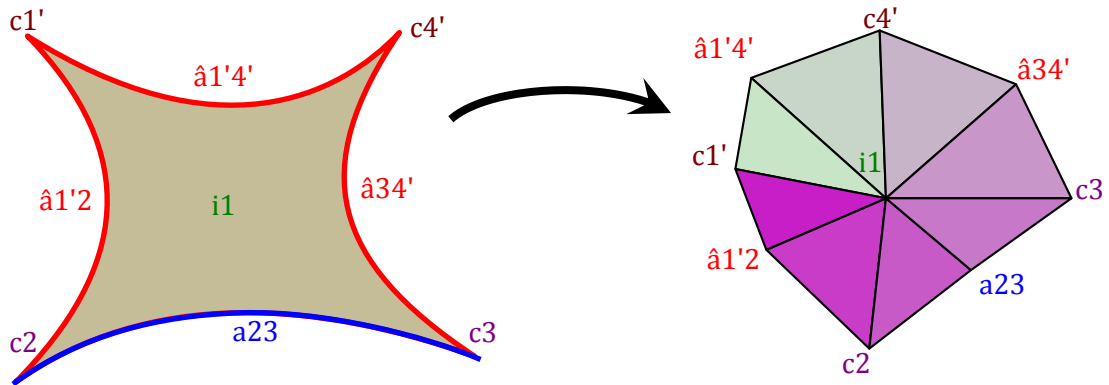


Figure 28 : (Réalisation de) l'abstraction d'un disque topologique $i1$ de bords labélisés

5.5.3.5. Abstraction d'un cylindre (i.e. un disque troué une fois)

Pour le dire avec des mots simples, nous proposons d'abstraire les cylindres topologiques par des « accordéons fermés ».

Cette abstraction requiert que les deux bords séparés de l'élément soit constitués d'une même quantité d'éléments labélisés. Si nécessaire, certains arcs de chacun des bords du cylindre topologique, dans $\mathfrak{s}(f)$, devront être subdivisés en plusieurs arcs connectés par de nouveaux points, jusqu'à ce que les deux bords possèdent autant d'éléments.

Lors de la construction de $\mathfrak{t}(f)$, si une subdivision des bords a été nécessaire, afin de rester compatible avec f , chaque arc, image, pré-image ou ayant la même image qu'un arc subdivisé, doit être subdivisé de la même manière (les arcs et points issus des subdivisions devront être labélisés à nouveau).

De plus, on notera que les deux bords du cylindre topologique doivent être *orientés*, de façon à conserver l'orientation relative de ceux-ci, que respecte tout homéomorphisme sur un cylindre.

Nous choisissons de représenter l'orientation de chaque bord par une liste, liant, dans un seul sens, tous les labels de ce bord considéré.

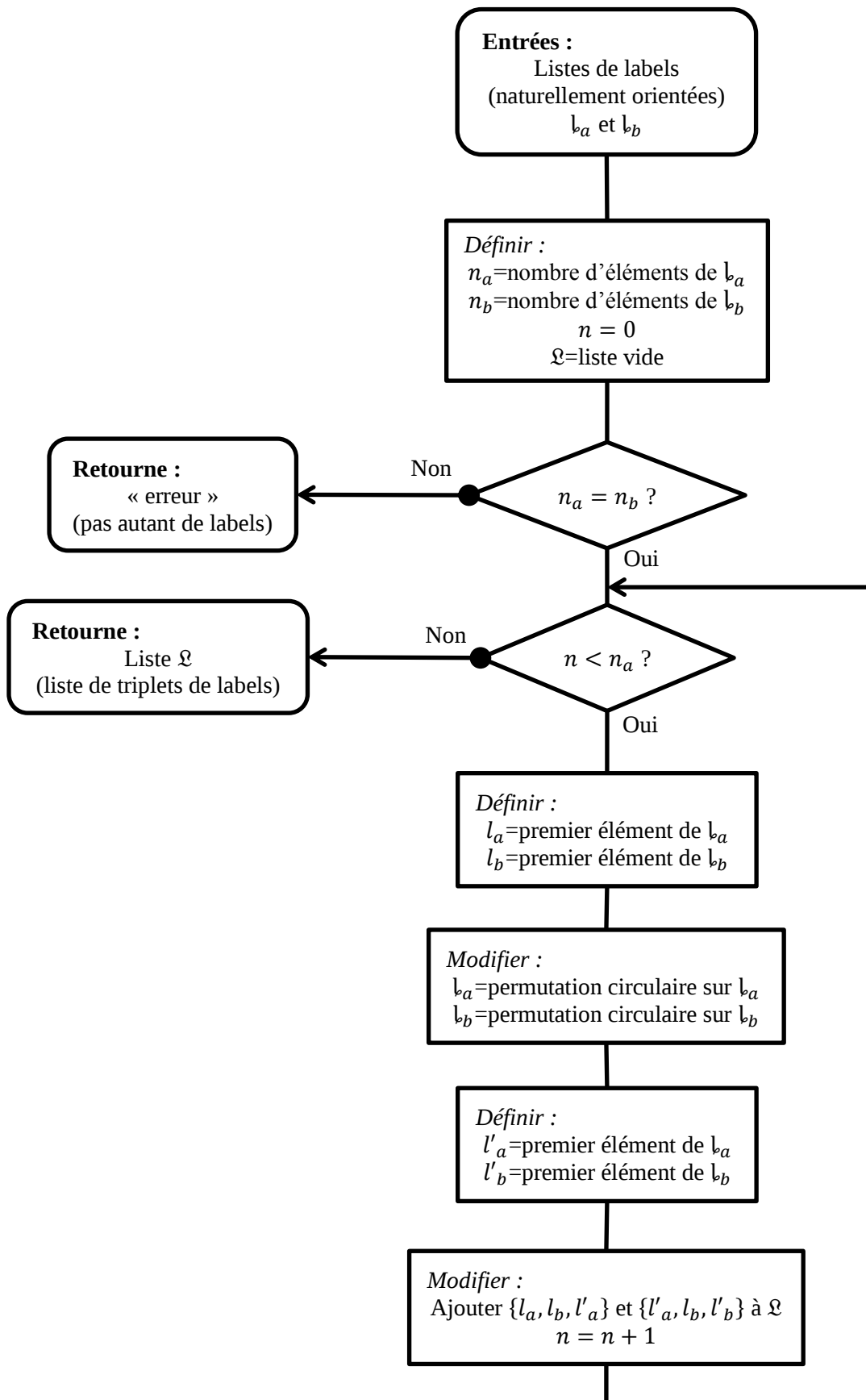


Figure 29 : Méthode algorithmique de construction des triplets de labels pour abstraire un cylindre

Une fois que les deux bords, dans $\mathcal{L}(\mathfrak{t}(f))$, contiennent le même nombre d'éléments labélisés, la méthode algorithmique de la Figure 29 est utilisée pour construire l'ensemble des triplets de points sur la base desquels sera construit le complexe simplicial abstrait proposé, décrivant le cylindre étudié.

Par exemple, la Figure 30 montre que, pour abstraire l'élément E, le bord L est séparé en quatre arcs et quatre points, tout comme est subdivisé l'autre bord de E. On obtient alors l'ensemble des simplexes abstraits, qui définissent une abstraction de E, en appliquant l'algorithme de la Figure 29 au bord L subdivisé et au second bord de E (tous deux munis d'une orientation).

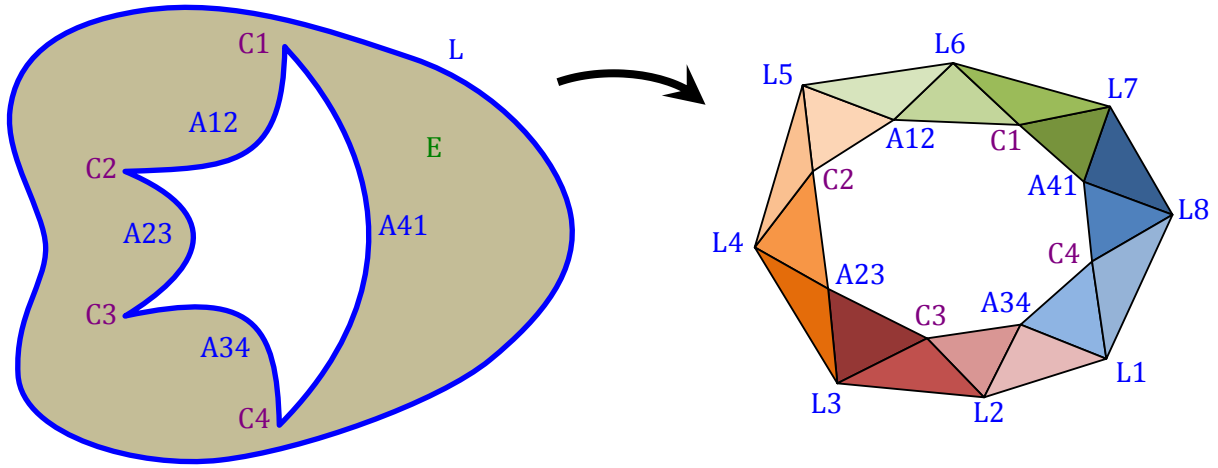


Figure 30 : (Réalisation de) l'abstraction d'un cylindre topologique E de bords labélisés

Remarque 5.5.3.1 (Eviter l'association d'une même désignation à plusieurs espaces)

Soient deux cylindres topologiques $\mathfrak{D}_1$ et $\mathfrak{D}_2$ dont les deux composantes connexes de bords sont, respectivement, $(\mathfrak{A}_{1,1}$ et $\mathfrak{A}_{1,2})$ et $(\mathfrak{A}_{2,1}$ et $\mathfrak{A}_{2,2})$.

Si $\mathfrak{A}_{1,1} \cap \mathfrak{A}_{2,1} \neq \emptyset$ et $\mathfrak{A}_{1,2} \cap \mathfrak{A}_{2,2} \neq \emptyset$, alors, en appliquant la méthode décrite précédemment, il est possible que plusieurs triangles topologiques soient décrits par les mêmes triplets de labels.

Or, deux simplexes abstraits se différencient uniquement par les labels du triplet qui définissent chacun d'eux. Ainsi le complexe simplicial abstrait, qui serait construit à partir de ces triplets décrivant plusieurs éléments, ne décrirait pas les espaces étudiés selon les conventions attendues.

La solution que nous proposons, pour résoudre cette difficulté, est de définir deux cercles topologiques $\mathfrak{A}_{1,3}$ et $\mathfrak{A}_{2,3}$ qui subdivisent chacun des deux cylindres $\mathfrak{D}_1$ et $\mathfrak{D}_2$ en deux nouveaux cylindres, que nous notons, respectivement, $(\mathfrak{D}_{1,1}$ et $\mathfrak{D}_{1,2})$ et $(\mathfrak{D}_{2,1}$ et $\mathfrak{D}_{2,2})$. Ces cylindres sont définis de telle sorte que

$$\forall (i, j) \in \llbracket 1; 2 \rrbracket^2, \text{ les deux bords de } \mathfrak{D}_{i,j} \text{ sont } (\mathfrak{A}_{i,j} \text{ et } \mathfrak{A}_{i,3})$$

Si $\mathfrak{D}_1$ et $\mathfrak{D}_2$ appartiennent à l'espace articulaire et ont la même image par f , on choisira $\mathfrak{A}_{1,3}$ et $\mathfrak{A}_{2,3}$ de telle sorte que $f(\mathfrak{A}_{1,3}) = f(\mathfrak{A}_{2,3})$

De plus, on notera, à nouveau que, pour que $\mathfrak{t}(f)$ reste compatible avec f , chaque ensemble image, pré-image ou ayant la même image qu'un ensemble subdivisé, doit être subdivisé de la même manière. Cette subdivision additionnelle est nécessaire pour les comportements étudiés au § 5.6.2.2 (exemple 1) et au § 5.6.2.4 (exemple 3)

Enfin, pour chaque paire, de cylindres parmi $\{\mathfrak{D}_{1,1}, \mathfrak{D}_{1,2}, \mathfrak{D}_{2,1}, \mathfrak{D}_{2,2}\}$, se trouvant dans la situation décrite au début de cette Remarque 5.5.3.1, on applique de nouveau la procédure proposée.

5.5.3.6. Abstraction d'un disque troué n fois

La stratégie consiste ici à subdiviser l'élément topologique en n éléments dont la topologie commune est celle d'un cylindre. Ensuite, il devient possible d'abstraire chacun de ces nouveaux éléments en faisant appel à la méthode appropriée pour abstraire un cylindre.

Une méthode choisie pour réaliser effectivement la subdivision peut être d'utiliser un (ou des) point(s) et des arcs supplémentaires, similairement à l'approche choisie pour subdiviser un cylindre.

On propose de subdiviser le disque E par l'intermédiaire d'arcs supplémentaires concourants.

Pour cela, on pose $\delta_0 E$ le bord extérieur de E (i.e. la composante connexe de δE qui n'est pas le bord de l'un des n trous) et $\forall i \in \llbracket 1; n \rrbracket$, on pose $\delta_i E$ le bord d'un des trous, de sorte que $\bigcup_{i \in \llbracket 0; n \rrbracket} \delta_i E = \delta E$.

On subdivise alors E en n cylindres E_i , où $\forall i \in \llbracket 1; n \rrbracket$ les bords de E_i sont $(\delta_i E \text{ et } \lambda_i)$ où λ_i vérifie :

- λ_i est un cercle topologique
- $\lambda_i = \#_i \cup M \cup \gamma_{i-1} \cup \gamma_{[i|n]}$ où :
 - $\#_i = \lambda_i \cap \delta_0 E$ est un arc topologique tel que $\forall N \in \delta_0 E, \exists k \in \llbracket 1; n \rrbracket$ tel que $N \in \#_k$
 - $[i|n]$ est le reste de la division euclidienne de i par n
 - $\forall j \in \llbracket 0; n-1 \rrbracket, \gamma_j$ est un arc topologique
 - $M \in \bigcap_{j \in \llbracket 0; n-1 \rrbracket} \delta \gamma_j$ est un point concourant des arcs γ_j

On notera à nouveau que, pour conserver la compatibilité avec f , chaque ensemble image, pré-image ou ayant la même image qu'un ensemble subdivisé, doit être subdivisé de la même manière, dans $\mathfrak{t}(f)$.

5.5.3.7. Abstraction de c_f (à partir de sa restriction aux singletons)

Posons $(\mathcal{L}(\mathfrak{t}_a(f)), \mathcal{L}(\mathfrak{t}_p(f))) = \mathcal{L}(\mathfrak{t}(f))$. On définit alors :

- $\ell = \ell_{\mathcal{L}(\mathfrak{t}(f))} : \forall z \in \{a, p\}, \mathcal{L}(\mathfrak{t}_z(f)) \ni (E, l) \mapsto l$
- $\gamma = \gamma_{\mathcal{L}(\mathfrak{t}(f))} : \ell(\mathcal{L}(\mathfrak{t}_a(f)) \cup \mathcal{L}(\mathfrak{t}_p(f))) \ni l \mapsto E \mid (E, l) \in \mathcal{L}(\mathfrak{t}_a(f)) \cup \mathcal{L}(\mathfrak{t}_p(f))$

Après avoir défini c_a et c_p à l'aide de complexes combinant, respectivement, les éléments de $\ell(\mathcal{L}(\mathfrak{t}_a(f)))$ et de $\ell(\mathcal{L}(\mathfrak{t}_p(f)))$, $c_f : c_a \rightarrow c_p$ est alors engendrée et totalement décrite par :

$$\forall (l_a, l_p) \in \ell(\mathcal{L}(\mathfrak{t}_a(f))) \times \ell(\mathcal{L}(\mathfrak{t}_p(f))), (c_f(l_a) = l_p) \Leftrightarrow f(\gamma(l_a)) = \gamma(l_p)$$

En français :

c_f associe à chaque label marquant un élément de $\mathfrak{t}_a(f)$, le label marquant son image par f .

Remarque 5.5.3.2 (Abstraction combinatoire et représentation générale de $\mathcal{K}_0$)

Toujours vis-à-vis de la comparaison abstraction/représentation canonique de $\mathcal{K}_0$, nous noterons que la topologie des éléments de $\mathfrak{t}(f)$ et leurs propriétés d'être bord d'autres éléments sont à la base de l'abstraction construite, ce qui incite à considérer un objet intermédiaire qui serait défini à l'aide de ces informations (on remarquera que l'orientation relative des bords des cylindres doit être connue).

C'est cette remarque qui inspire à définir une représentation « incomplète » de $\mathcal{K}_0$ par un objet de la théorie des graphes où les nœuds sont des éléments topologiques, annotés de leur topologie respective (dans l'espace A ou $\mathcal{P}$) et où les connexions signifient que l'élément de dimension inférieure fait partie du bord de l'élément de dimension supérieure, si ceux-ci sont dans le même espace, et que l'élément, dans $\mathcal{P}$, est image de celui dans A , si les deux éléments ne sont pas dans le même espace.

L'aspect « incomplet » provient de notre remarque sur l'orientation relative des bords des cylindres, qui n'est pas conservée dans l'objet proposé. Toutefois, bien que ce type d'information ne semble pas admettre (pour nous) un codage évident dans un graphe (à part, peut être en utilisant des connexions orientées), l'objet proposé est un invariant, et, sous certaines conditions, il est possible que ces informations supplémentaires soit redondantes avec l'information contenue dans l'objet proposé.

5.5.4. Etape 3 (optionnelle) : Réalisation

La réalisation de l'abstraction est optionnelle puisque le comportement de la fonction f abstraite est déjà totalement décrit par c_f . L'intérêt de la réalisation $|c_f|$ réside donc en sa lisibilité différente de celle de c_f . En effet, $|c_f|$ étant une application entre simplexes (géométrique), sa lisibilité peut être meilleure que celle de la fonction combinatoire c_f (cette lisibilité dépend du lecteur).

De par la nature graphique des réalisations d'abstraction, nous illustrerons l'abstraction, de chaque exemple considéré, dans le corps de ce chapitre, par l'intermédiaire d'une réalisation géométrique de cette abstraction.

Toutefois, par souci de complétude, les abstractions (qui peuvent être considérées comme absconses) sont toutes reportées dans la sous-section annexe A.2.2.

5.5.5. Application à une fonction géométrique d'un robot 3-RPR

On se propose d'appliquer la méthode d'abstraction proposée aux fonctions géométriques g_1 et g_2 , illustrées par la Figure 25, associées, respectivement, à un robot 3R et d'un robot 3-RPR.

Les traces singulières des fonctions g_1 et g_2 , sont définies par les courbes tracées sur la Figure 25, dans l'espace opérationnel et l'espace articulaire associé à chaque fonction.

Nous avons déjà remarqué au §5.4.1 que $\forall k \in \llbracket 1; 2 \rrbracket, \forall i > 1, \mathcal{Z}_{i_p}(g_k) = \emptyset$ et $\mathcal{Z}_{i_a}(g_k) = \emptyset$.

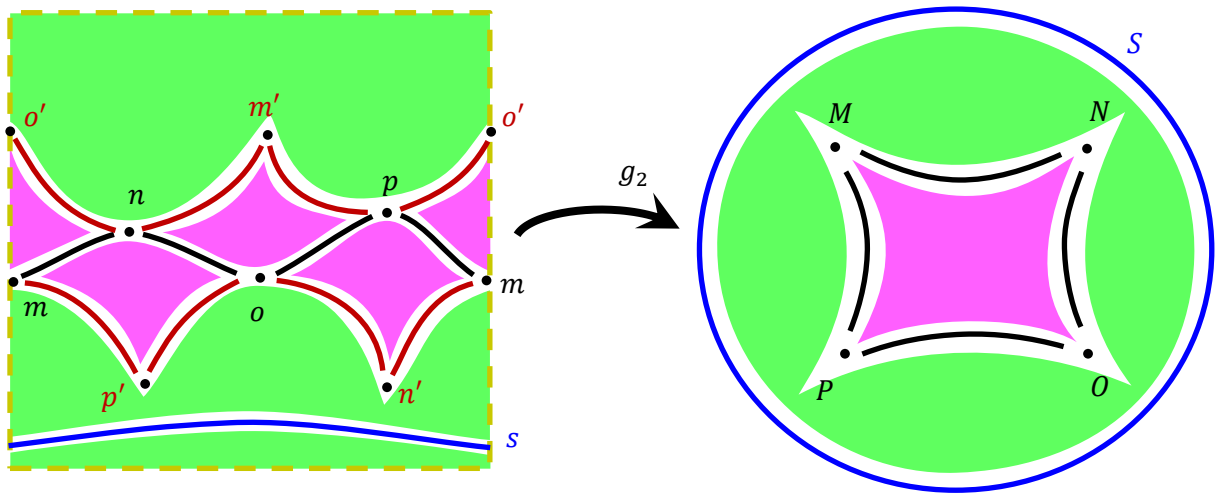
D'où $\forall k \in \llbracket 1; 2 \rrbracket, \forall i > 1, \mathcal{Z}_i(g_k) = (\emptyset, \emptyset)$.

Nous avons aussi noté que $\forall k \in \llbracket 1; 2 \rrbracket, \mathcal{Z}_{1_p}(g_k) = \{M, N, O, P\}$ et $\mathcal{Z}_{1_a}(g_k) = \{m, n, o, p, m', n', o', p'\}$

On en déduit alors les stratifications tracées $\mathfrak{s}(g_k)$, pour $k \in \llbracket 1; 2 \rrbracket$ en remarquant que, même si $\mathfrak{s}_p(g_1) = \mathfrak{s}_p(g_2)$, on a déjà $\mathfrak{s}(g_1) \neq \mathfrak{s}(g_2)$ car $\mathfrak{s}_a(g_1) \neq \mathfrak{s}_a(g_2)$

Notons qu'il serait redondant de construire la stratification tracée $\mathfrak{s}(g_1)$ et l'abstraction c_{g_1} , puisque la fonction g_1 est équivalente à la fonction géométrique f_2 (i.e. $g_1 \sim f_2$) du manipulateur 3R, étudié au § 5.6.2.3. De ce fait $\mathfrak{s}(g_1)$ est égale à $\mathfrak{s}(f_2)$, à des homéomorphismes près sur les éléments. De même, l'abstraction proposée pour f_2 est une abstraction décrivant g_1 . (On peut poser $c_{g_1} = c_{f_2}$)

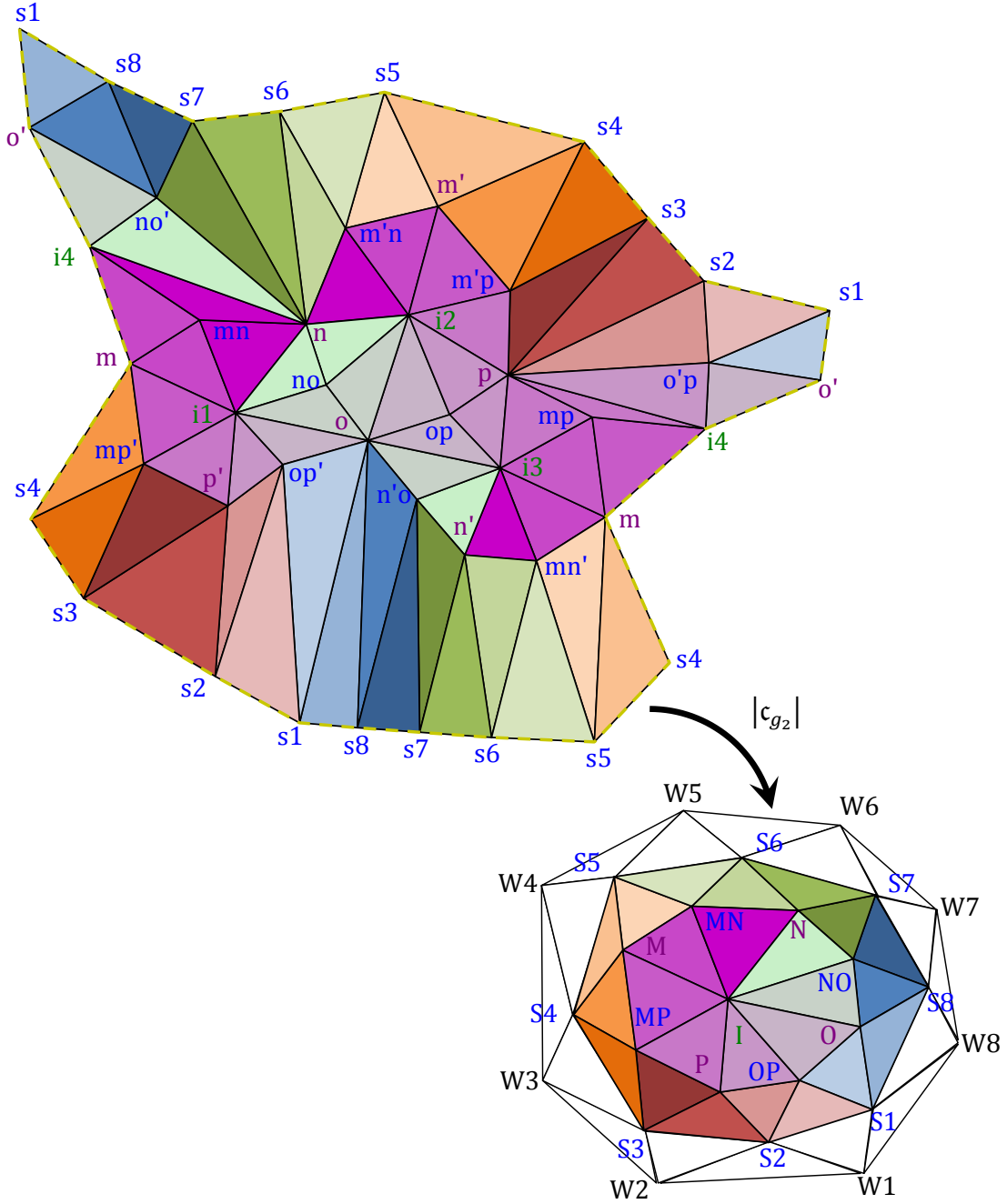
$\mathfrak{s}(g_2)$ est représentée à la Figure 31 par l'isolation de chacun de ses éléments.



$$\{m, m'\} \xrightarrow{g_2} M ; \{n, n'\} \xrightarrow{g_2} N ; \{o, o'\} \xrightarrow{g_2} O ; \{p, p'\} \xrightarrow{g_2} P ; s \xrightarrow{g_2} S$$

Figure 31 : Stratification tracée $\mathfrak{s}(g_2)$ de la fonction g_2 de la Figure 25

A partir de $\mathfrak{s}(g_2)$, nous sommes en mesure de construire l'abstraction, dont la réalisation est présentée à la Figure 32, à l'aide de la procédure détaillée dans cette section.



Pour toutes lettres i et j , $c_{g_2}(i) = c_{g_2}(i') = I$ et $c_{g_2}(ij) = c_{g_2}(i'j) = c_{g_2}(ij') = IJ$
 Pour toute lettre i et pour tout nombre k , $c_{g_2}(ik) = Ik$

Figure 32 : Triangulation $|c_{g_2}|$, réalisation géométrique de l'abstraction de g_2 , issue de la Figure 25

5.6 Application à la robotique

5.6.1. Notations, modèles et conventions

5.6.1.1. Notations utilisées pour écrire l'abstraction d'une fonction

Nous représenterons une abstraction $c_f : c_A \rightarrow c_P$ d'une fonction $f : A \rightarrow P$, par l'intermédiaire des abstractions c_A et c_P (des espaces A et P , respectivement) et de l'application induite par c_f qui associe les singletons de c_A aux singletons de c_P .

(Ces éléments de représentation sont univoques et définissent entièrement $c_f : c_A \rightarrow c_P$)

En réinterprétant (Delanoue, et al., 2014), nous choisissons d'utiliser les notations suivantes :

- $\langle \xi_1, \dots, \xi_n \rangle$ est le simplexe engendré par le n -uplet de symboles $\{\xi_1, \dots, \xi_n\}$
- $c_1 \cup c_2$ est le complexe simplicial abstrait généré par les complexes simpliciaux c_1 et c_2

5.6.1.2. Modèle géométrique d'un manipulateur 3R orthogonal (Rappels)

Nous avons déjà présenté dans le § 4.3.1.3, dans le cas général, l'étude et la caractérisation des manipulateurs robotiques 3R, par des fonctions géométriques du tore dans le plan $\mathbb{R}^2$.

De plus, nous avons déjà introduit, avec la Figure 21, le schéma cinématique d'un robot 3R orthogonal.

Toutefois, pour simplifier la lecture, nous allons reporter ici un résumé de l'approche considérée.

La Figure 33 reproduit le schéma cinématique d'un robot 3R orthogonal, ici dans sa position d'origine.

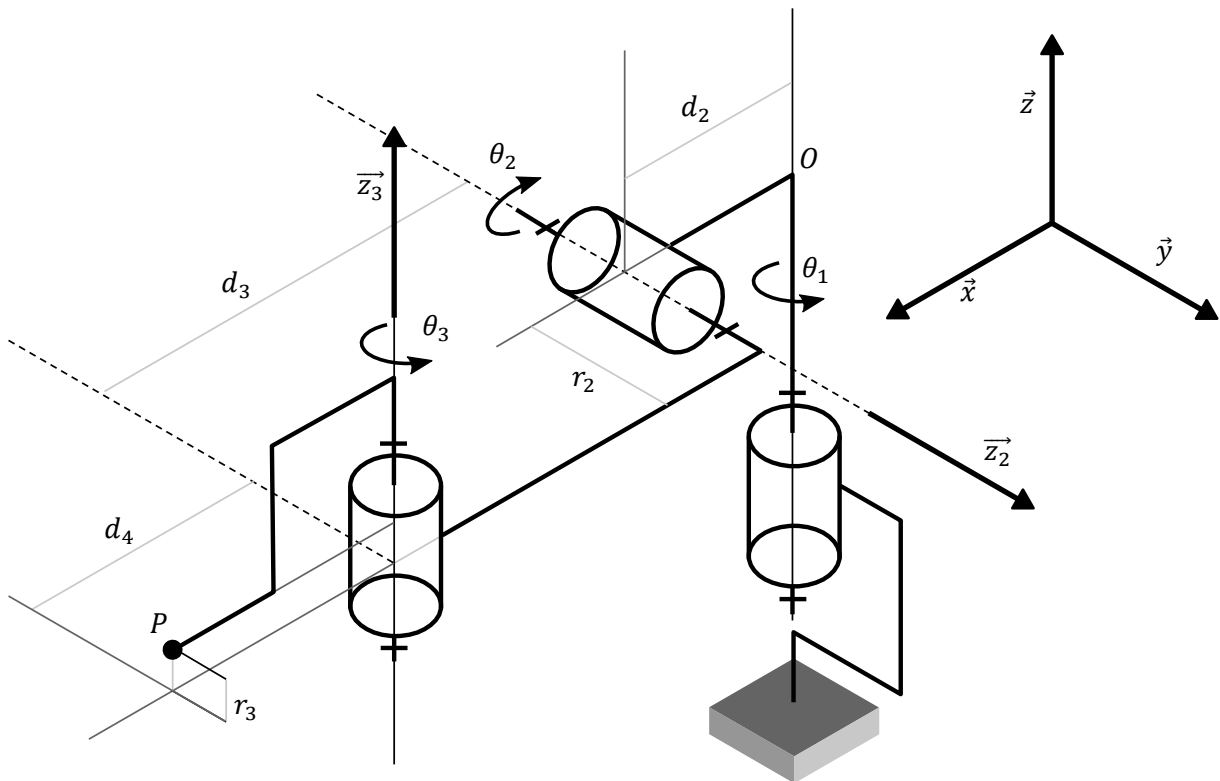


Figure 33 : Schéma cinématique d'un manipulateur 3R orthogonal dans sa position d'origine.

Ces manipulateurs 3R sont spécifiques au sens où les axes de leurs liaisons pivot sont mutuellement orthogonaux, d'où leur qualification de manipulateur orthogonal comme détaillé dans (Wenger, 2007).

La caractérisation de ces robots par une fonction f du tore dans le plan $\mathbb{R}^2$, s'appuie sur le fait que, lorsque $\forall \theta_1, (\theta_2, \theta_3)$ peut parcourir le même sous-ensemble du tore, le lieu singulier S_G de la fonction géométrique totale $G : (\theta_1, \theta_2, \theta_3) \mapsto (x, y, z)$ est invariant selon θ_1 . De plus, en posant, $\forall \theta_1, G_{\theta_1} : (\theta_2, \theta_3) \mapsto (x_{\theta_1}, y_{\theta_1}, z_{\theta_1})$, on peut écrire $\exists(\rho, z), \forall \theta_1, (z = z_{\theta_1} \text{ et } \rho = \sqrt{x_{\theta_1}^2 + y_{\theta_1}^2})$ et donc décrire le robot par $f : (\theta_2, \theta_3) \mapsto (\sqrt{x_0^2 + y_0^2}, z_0)$ (ou par $f : (\theta_2, \theta_3) \mapsto (x_0^2 + y_0^2, z_0)$ qui est équivalente, si l'on souhaite s'affranchir de la racine carrée).

5.6.1.3. Illustration des applications (réalisant les abstractions)

En complément des abstractions, nous proposons une réalisation géométrique pour chacune de celles-ci, afin de faciliter la compréhension de la façon dont une abstraction décrit le comportement d'un manipulateur robotique. Ainsi pour une lecture simplifiée des figures contenant les réalisations, certaines conventions (que l'on espère intuitives) sont utilisées. Leurs descriptions sont détaillées dans la sous-section annexe A.2.1.

Nous proposons, ici une liste condensée de ces conventions :

- Une même couleur est utilisée pour les triangles ayant la même image et pour leur image.
- On identifie les points des bords marqués des mêmes labels (et les arcs les reliant)
- Les points labélisés et leurs images vérifient :
 - Les labels pour l'espace articulaire utilisent des *lettres minuscules*.
 - Les labels pour l'espace opérationnel utilisent des *lettres majuscules*.
 - Un point est une pré-image d'un autre ssi son label contient le label de ce dernier, après retrait de tout symbole apostrophe et guillemet et mise en majuscule des lettres.

5.6.2. Exemples de comportements abstraits

5.6.2.1. Plan de présentation

Les comportements présentés constituent un sous-ensemble de ceux isolés dans (Wenger, 2007). Chaque abstraction présentée est, par ailleurs, réalisée à partir d'un exemple de manipulateur dont les paramètres ont été choisis proches de ceux sélectionnés dans (Wenger, 2007), pour illustrer le même comportement. On notera principalement que ces trois premiers comportements abstraits vérifient que chaque point de l'espace opérationnel admet un nombre fini de pré-images.

Les exemples de comportements seront présentés, en accord avec la procédure d'abstraction, définie à la section 5.5, de la façon suivante :

- Nombre de cusps et de nodes et paramètres géométriques du manipulateur exemple.
- Stratification labellisée tracée (Etape 1 de l'abstraction).
- La triangulation réalisant l'abstraction (Etape 3 de l'abstraction).

Les abstractions elles-mêmes ne seront pas reportées dans ce chapitre, du fait de leur nature absconse, exception faite de celle associée au comportement 1, qui reste simple tout en présentant l'intégralité du concept d'abstraction.

Afin d'être exhaustif, nous avons tout même choisi de reporter l'abstraction associée à chaque comportement étudié, mais seulement dans la sous-section annexe A.2.2, qui leur est dédiée.

Pour chaque comportement robotique étudié, les points des espaces articulaire et opérationnel sont ceux de la fonction $f_i : (\theta_2, \theta_3) \mapsto (\sqrt{x_0^2 + y_0^2}, z_0)$, associée comme défini au § 5.6.1.2, à la fonction géométrique G_i du robot 3R possédant les paramètres géométriques choisis.

S_{f_i} est généré en parcourant le domaine de définition de θ_2 et en résolvant, pour chaque θ_2 considéré, l'équation induite $\det(Df) = 0$, en la variable θ_3 et où, rappelons-le, Df est la matrice Jacobienne du robot et de f , tel que détaillé dans (Wenger, 2004)

Remarque 5.6.2.1

Cette section étant une application directe de la méthode d'abstraction, nous nous abstenons de détailler ses différentes étapes, tant qu'aucun *choix explicite* n'est pas requis.

L'abstraction construite peut toujours être déduite des figures en utilisant les conventions introduites pour décrire le lien entre l'espace articulaire et opérationnel et en extrayant, de chaque figure, la liste des triplets de labels qui définissent un triangle réalisant géométriquement un simplexe abstrait.

5.6.2.2. Comportement 1: « Tore vu du dessus »

Le comportement 1 est associé à une fonction géométrique f_1 présentant 0 cusp et 0 node. Pour cet exemple, les paramètres géométriques du robot 3R orthogonal considéré sont :

$$d_2 = 1, \quad d_3 = 0.7, \quad d_4 = 0.3, \quad r_2 = 0.3, \quad r_3 = 0$$

Ce robot admet un *void* V (i.e. un espace vide bordé par l'espace de travail) dans l'espace opérationnel et chaque point de l'espace labellisé I possède deux pré-images dans l'espace articulaire, par f_1^{-1} . La courbe intérieure de $f_1(S_{f_1})$ borde un void. Celle-ci ne génère donc aucune courbe caractéristique, dans l'espace articulaire, par l'action de f_1^{-1} .

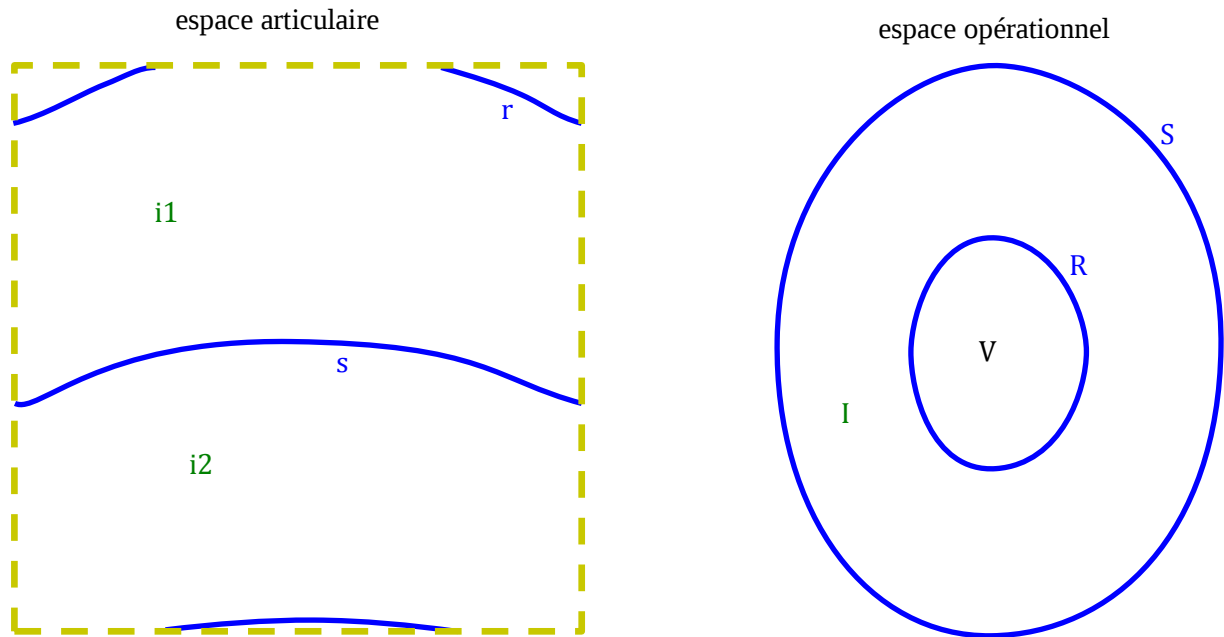


Figure 34 : Stratification tracée labélisée $\mathcal{L}(s(f_1))$ de l'exemple pour le comportement 1

De la stratification tracée labélisée $\mathcal{L}(s(f_1))$, représentée à la Figure 34, on extrait l'abstraction c_{f_1} , de la fonction f_1 , décrivant le manipulateur orthogonal ayant les paramètres géométriques choisis.

En utilisant les labels de la Figure 34 pour désigner les éléments topologiques, on notera que l'espace articulaire, qui est un tore, se sépare selon deux cylindres topologiques $i1$ et $i2$, de bords circulaires communs labélisés r et s .

De plus, d'après les conventions, la Figure 34 implique $f(i1) = f(i2)$, $f(r) = R$ et $f(s) = S$.

Cet exemple de comportement présente, par ailleurs le cas particulier présenté § 5.5.3.3 (nécessité de séparer un cercle complet labélisé). De ce fait, les éléments s , r , S et R seront, chacun, subdivisés en 2 arcs connectés par 2 points, de façon à rester compatible avec f , pour construire $\mathfrak{t}(f_1)$.

La Remarque 5.5.3.1 (plusieurs cylindres partageant deux sous-ensembles de leurs bords communs) étant valide pour les cylindres i_1 et i_2 , ceux-ci doivent être subdivisés, de façon appropriée, par l'intermédiaire de deux cercles, choisis pour avoir la même image par f .

Ces cercles additionnels devront alors être subdivisés en 2 arcs connectés par 2 points, de telle sorte que les nouveaux cylindres formés puissent être abstraits selon le template proposé pour les cylindres.

On rappellera alors que, pour finaliser la construction de $\mathfrak{t}(f_1)$, l'image commune de ces nouveaux éléments subdivisés doit remplacer les images des éléments initiaux, dans la nouvelle stratification, avant que celle-ci soit finalement re-labélisée.

Enfin, l'application de la procédure à la stratification labélisée $\mathcal{L}(\mathfrak{t}(f_1))$ issue des subdivisions induit, naturellement, une abstraction c_{f_1} de la fonction f_1 , dont une réalisation est présentée à la Figure 35.

Abstraction du comportement 1 (c_{f_1}) :

- source :

$$\begin{aligned} &\langle s_2, s_3, l_1h \rangle \cup \langle s_1, s_2, l_4h \rangle \cup \langle s_1, s_4, l_3h \rangle \cup \langle s_3, s_4, l_2h \rangle \cup \langle s_2, s_3, l_1b \rangle \cup \langle s_1, s_2, l_4b \rangle \cup \langle s_1, s_4, l_3b \rangle \\ &\cup \langle s_3, s_4, l_2b \rangle \cup \langle r_2, r_3, l_4h \rangle \cup \langle r_1, r_2, l_3h \rangle \cup \langle r_1, r_4, l_2h \rangle \cup \langle r_3, r_4, l_1h \rangle \cup \langle r_2, r_3, l_4b \rangle \cup \langle r_1, r_2, l_3b \rangle \\ &\cup \langle r_1, r_4, l_2b \rangle \cup \langle r_3, r_4, l_1b \rangle \cup \langle l_2h, l_3h, s_4 \rangle \cup \langle l_1h, l_2h, s_3 \rangle \cup \langle l_1h, l_4h, s_2 \rangle \cup \langle l_3h, l_4h, s_1 \rangle \\ &\cup \langle l_2b, l_3b, s_4 \rangle \cup \langle l_1b, l_2b, s_3 \rangle \cup \langle l_1b, l_4b, s_2 \rangle \cup \langle l_3b, l_4b, s_1 \rangle \cup \langle l_2h, l_3h, r_1 \rangle \cup \langle l_1h, l_2h, r_4 \rangle \\ &\cup \langle l_1h, l_4h, r_3 \rangle \cup \langle l_3h, l_4h, r_2 \rangle \cup \langle l_2b, l_3b, r_1 \rangle \cup \langle l_1b, l_2b, r_4 \rangle \cup \langle l_1b, l_4b, r_3 \rangle \cup \langle l_3b, l_4b, r_2 \rangle \end{aligned}$$

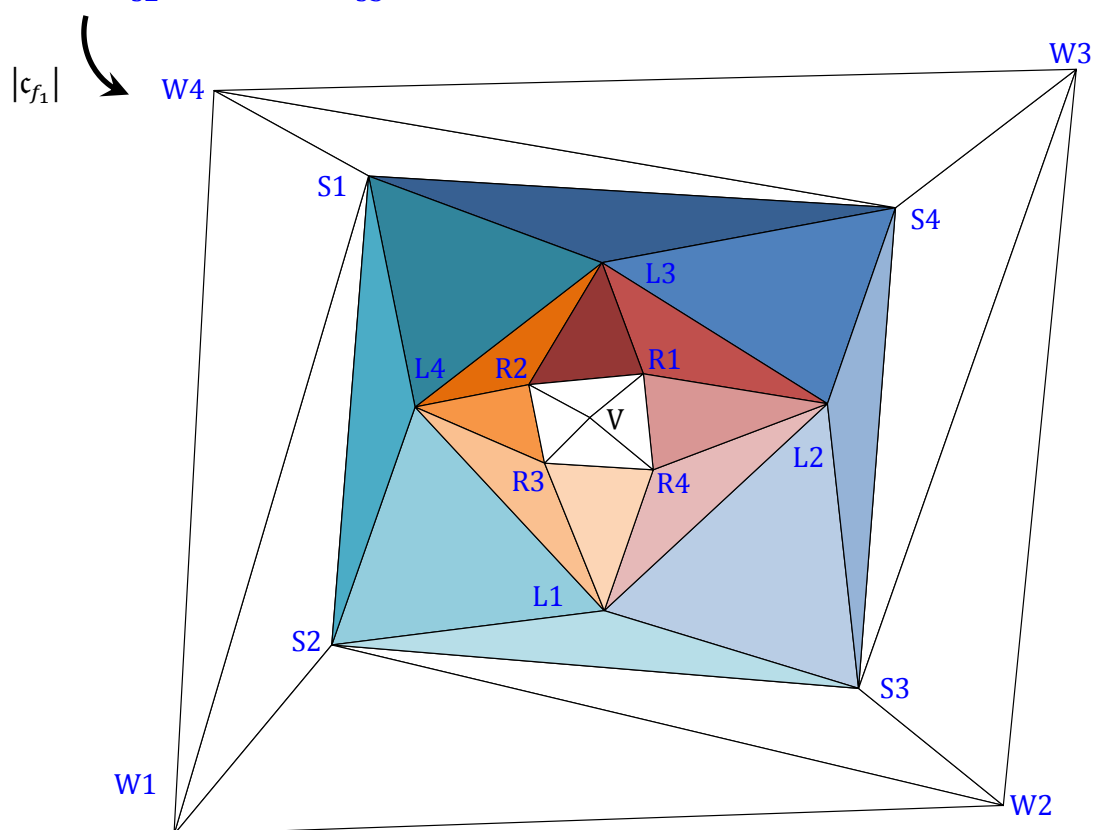
- image :

$$\begin{aligned} &\langle S_2, S_3, L_1 \rangle \cup \langle S_1, S_2, L_4 \rangle \cup \langle S_1, S_4, L_3 \rangle \cup \langle S_3, S_4, L_2 \rangle \cup \langle S_2, S_3, W_2 \rangle \cup \langle S_1, S_2, W_1 \rangle \\ &\cup \langle S_1, S_4, W_4 \rangle \cup \langle S_3, S_4, W_3 \rangle \cup \langle R_2, R_3, L_4 \rangle \cup \langle R_1, R_2, L_3 \rangle \cup \langle R_1, R_4, L_2 \rangle \cup \langle R_3, R_4, V \rangle \\ &\cup \langle R_2, R_3, V \rangle \cup \langle R_1, R_2, V \rangle \cup \langle R_1, R_4, V \rangle \cup \langle R_3, R_4, V \rangle \cup \langle L_2, L_3, S_4 \rangle \cup \langle L_1, L_2, S_3 \rangle \\ &\cup \langle L_1, L_4, S_2 \rangle \cup \langle L_3, L_4, S_1 \rangle \cup \langle L_2, L_3, R_1 \rangle \cup \langle L_1, L_2, R_4 \rangle \cup \langle L_1, L_4, R_3 \rangle \cup \langle L_3, L_4, R_2 \rangle \\ &\cup \langle W_2, W_3, S_3 \rangle \cup \langle W_1, W_2, S_2 \rangle \cup \langle W_1, W_4, S_1 \rangle \cup \langle W_3, W_4, S_4 \rangle \end{aligned}$$

- lien :

$$\forall k \in \llbracket 1; 3 \rrbracket, c_{f_1}(rk) = Rk, c_{f_1}(sk) = Sk, c_{f_1}(lkh) = c_{f_1}(lkb) = Lk$$

Notons que la triangulation, réalisant géométriquement l'abstraction de l'espace articulaire, est homéomorphe au tore et que la Figure 35 montre seulement un patron de ce tore, dont les bords opposés doivent être identifiés le long des points ayant les mêmes labels ainsi que le long des arcs reliant ces mêmes points, afin de construire complètement la triangulation de l'espace articulaire.



109

5.6.2.3. Comportement 2 : « Cusps seuls »

Le comportement 2 est associé à une fonction géométrique f_2 présentant 4 cusps et 0 node.

Pour cet exemple, les paramètres géométriques du robot 3R orthogonal considéré sont :

$$d_2 = 1, \quad d_3 = 2, \quad d_4 = 1.5, \quad r_2 = 1, \quad r_3 = 0.5$$

La stratification labélisée $\mathcal{L}(\mathfrak{s}(f_2))$ de la fonction f_2 est proposée et représentée par la Figure 36.

Pour ce robot, chaque point de I admet quatre pré-images dans l'espace articulaire.

L'application de f^{-1} à la courbe singulière intérieure (dans l'espace opérationnel) génère des courbes caractéristiques dans l'espace articulaire (représentées en rouge dans la Figure 36).

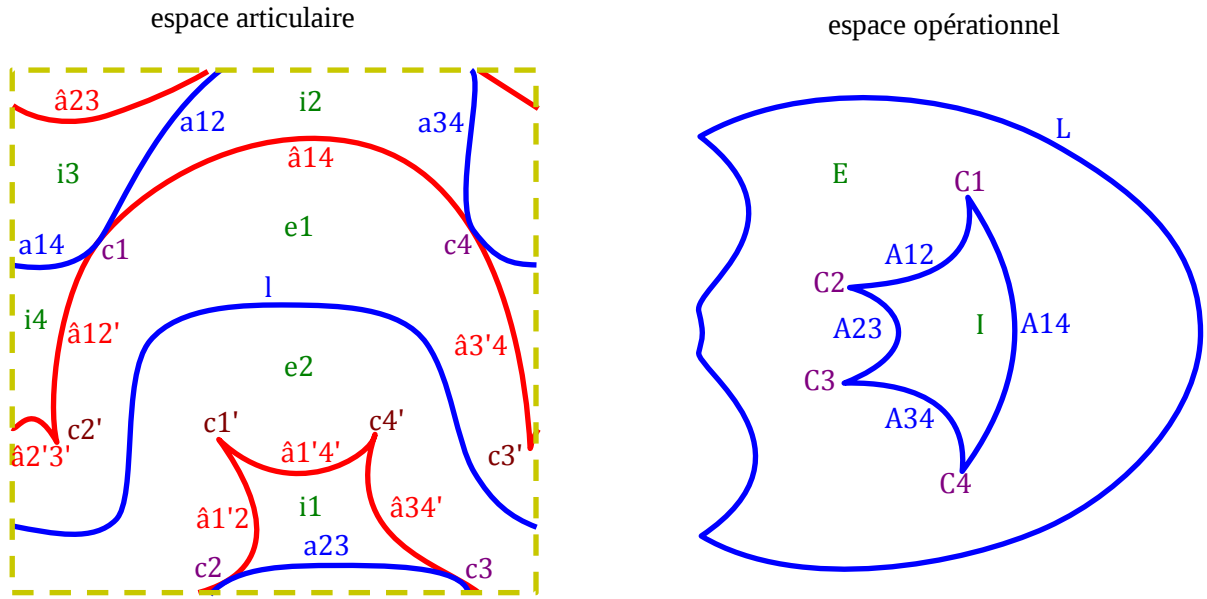


Figure 36 : Stratification tracée labélisée $\mathcal{L}(\mathfrak{s}(f_2))$ de l'exemple pour le comportement 2

Aucun choix explicite n'est réalisé dans le cas de cet exemple. En effet, la seule modification requise est la subdivision de I (et de son image L) en autant d'éléments que les autres bords de $e1$ et de $e2$.

Toutefois, tous les points de I (et de L) se révèlent être de même spécificité, impliquant que le choix des éléments subdivisés est totalement transparent vis-à-vis de la description induite de f_2 .

De la stratification construite $\mathfrak{t}(f_2)$, on extrait alors l'abstraction c_{f_2} , qui possède la propriété d'être l'unique abstraction constructible selon la méthode proposée, au choix des labels près, et dont une réalisation $|c_{f_2}|$ est donnée par la Figure 37.

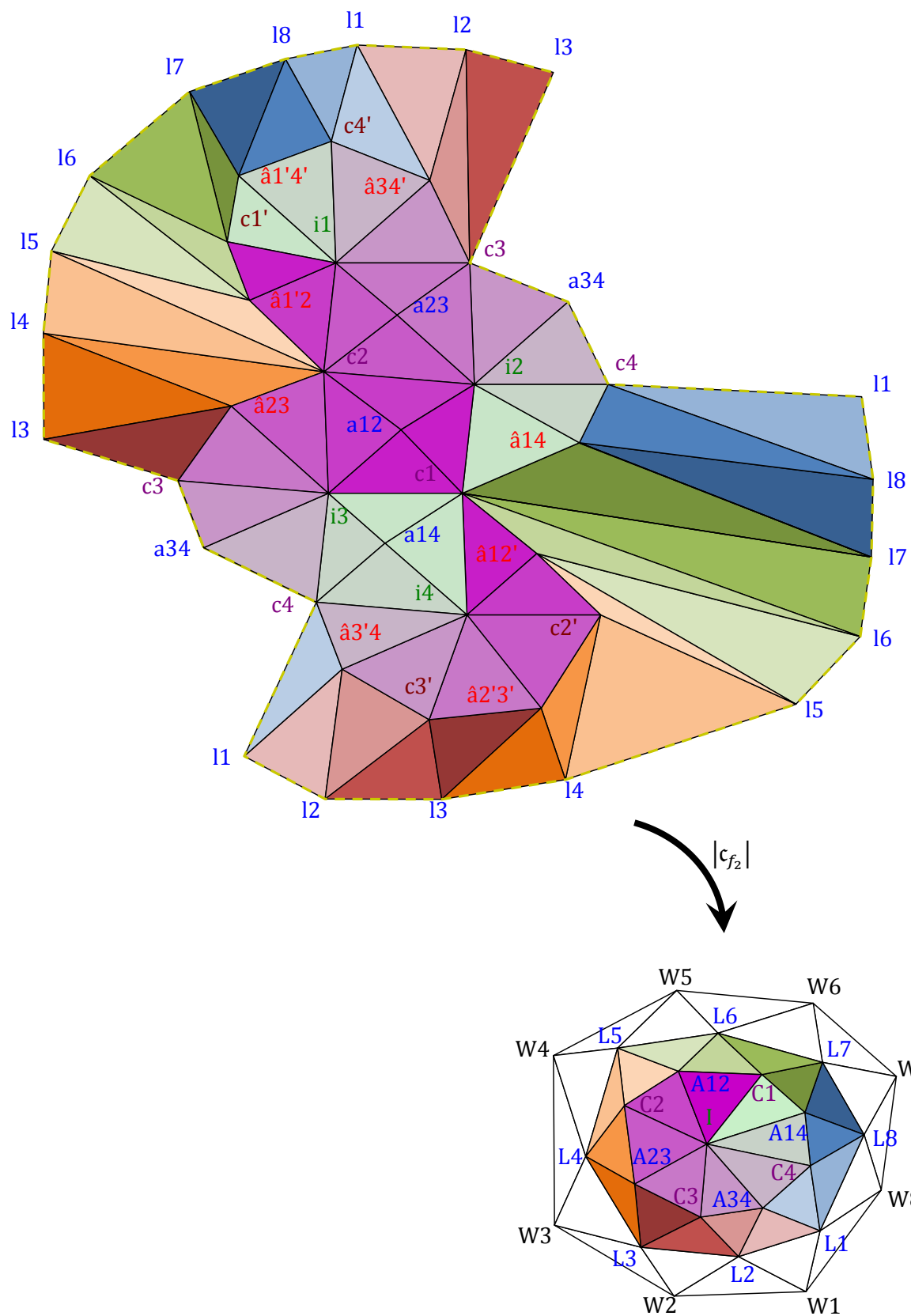


Figure 37 : Triangulation du comportement 2

5.6.2.4. Comportement 3 : « Cusps et nodes »

Le comportement 3 est associé à une fonction géométrique f_3 présentant 4 cusps et 2 nodes.

Pour cet exemple, les paramètres géométriques du robot 3R orthogonal considéré sont :

$$d_2 = 1, \quad d_3 = 1.5, \quad d_4 = 0.7, \quad r_2 = 0.5, \quad r_3 = 0.5$$

La stratification labélisée $\mathcal{L}(\mathfrak{s}(f_3))$ de la fonction f_3 est présentée séparément selon sa partie articulaire $\mathcal{L}(\mathfrak{s}_a(f_3))$ à la Figure 38 et sa partie opérationnelle $\mathcal{L}(\mathfrak{s}_p(f_3))$ à la Figure 39.

L'espace de travail W de ce robot borde la région V , qui est un void pour W (i.e. un sous ensemble, étranger à W , mais inclus dans l'enveloppe convexe de W). Ce même espace de travail W contient deux régions admettant chacune quatre pré-images (régions I2 et I3) en plus d'une région I1 admettant deux pré-images.

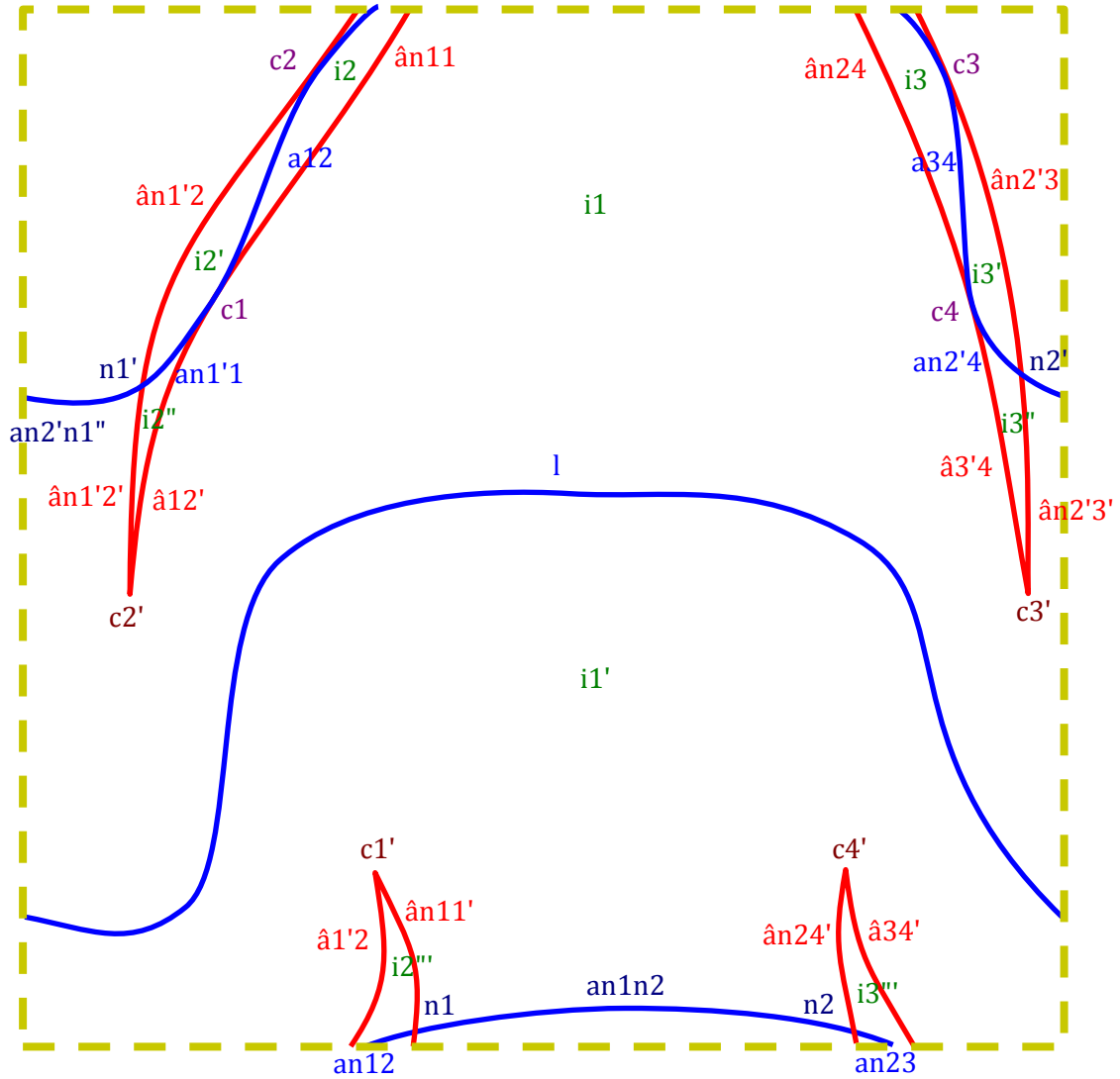


Figure 38 : Partie articulaire $\mathcal{L}(\mathfrak{s}_a(f_3))$ de $\mathcal{L}(\mathfrak{s}(f_3))$ illustrant le comportement 3

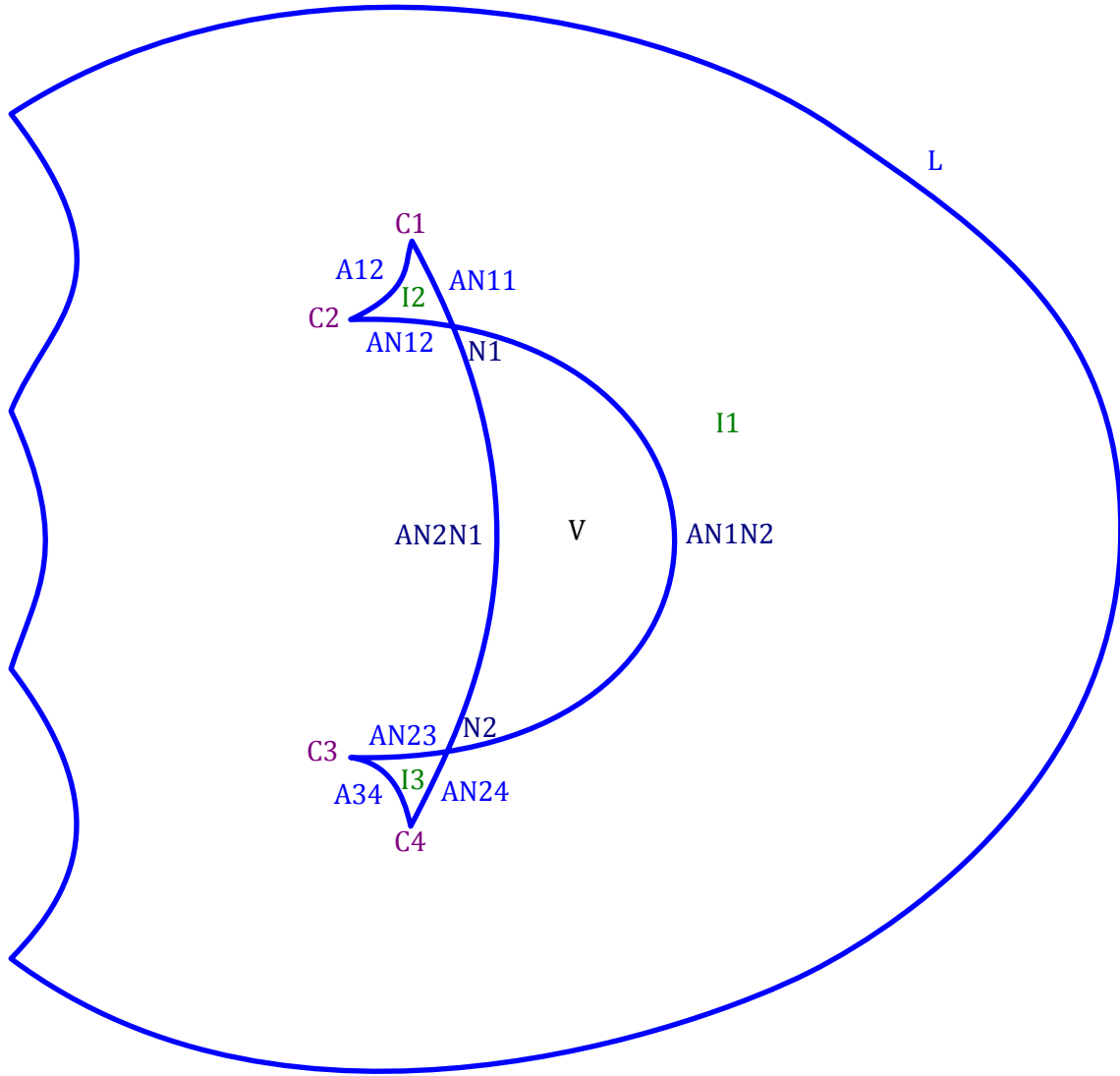


Figure 39 : Partie opérationnelle $\mathcal{L}(\mathfrak{s}_p(f_3))$ de $\mathcal{L}(\mathfrak{s}(f_3))$ illustrant le comportement 3

Comme le montre la Figure 38, le cercle l doit être subdivisé, à la fois pour décrire, selon le template proposé au § 5.5.3.3, sa topologie et pour présenter un même nombre d'éléments que les autres bords des cylindres $i1$ et $i1'$.

De plus, on note que $i1$ et $i1'$, en plus de partager l en tant que bord commun, admettent tout deux $\{an1n2, an2'n1'\}$ comme sous-ensemble de leur second bord.

Cela implique donc la nécessité de subdiviser ces deux cylindres, afin de garantir l'unicité de chaque simplexe abstrait associé à un même n -uplet de labels.

En complétant la stratification afin de conserver la compatibilité avec f_3 (en subdivisant $I1$ et L dans l'espace opérationnel de façon appropriée) et en re-labélisant l'abstraction obtenue $\mathfrak{s}(f_3)$, la procédure proposée retourne l'abstraction recherchée $\mathfrak{c}_{f_3}$, réalisée géométriquement par $|\mathfrak{c}_{f_3}|$ dont une représentation est donnée à la Figure 40.

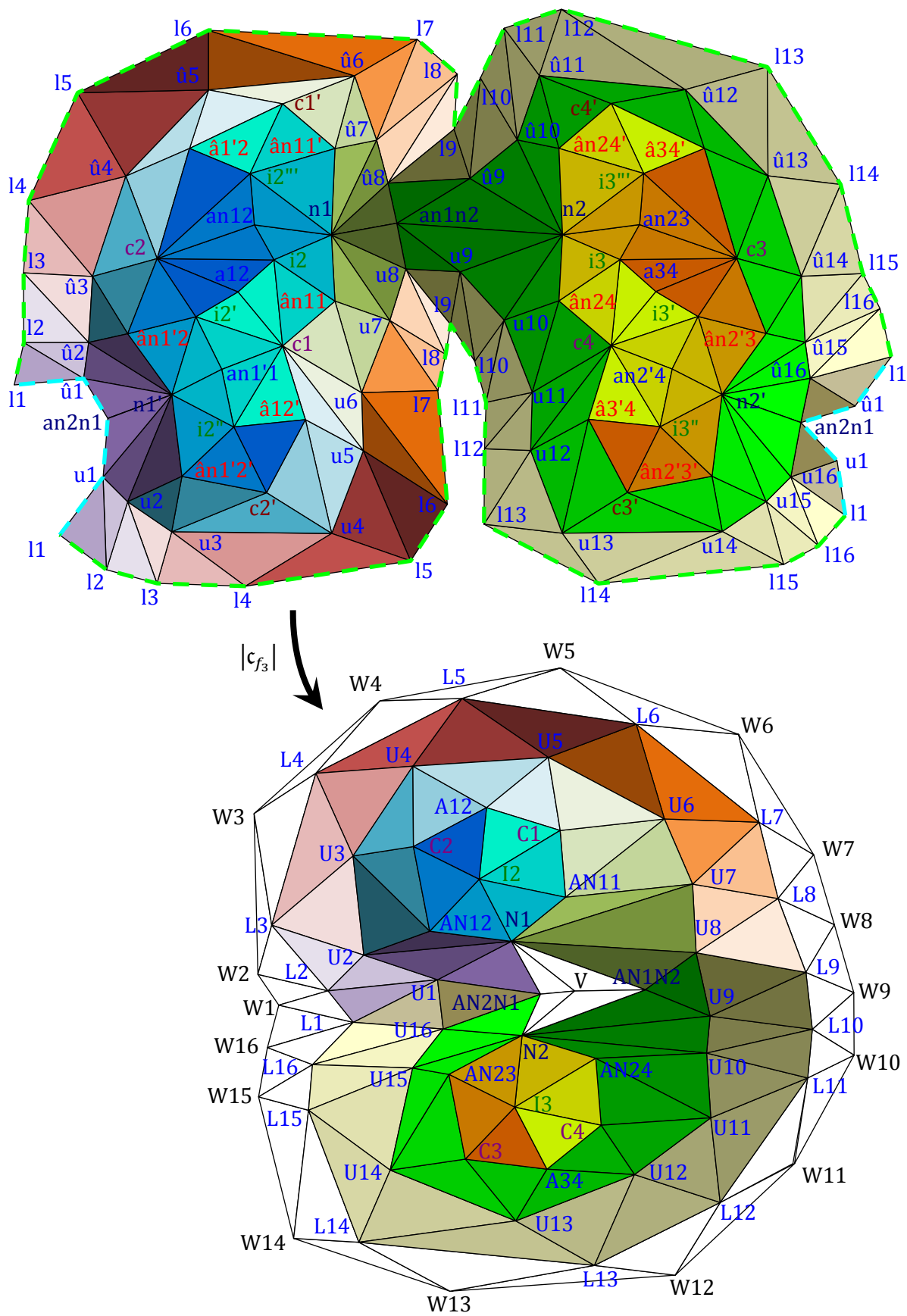


Figure 40 : Triangulation du Comportement 3

5.7 Mouvements internes et abstraction

Lorsqu'un robot présente un mouvement interne, sa fonction géométrique transforme un sous-ensemble de l'espace articulaire en un sous-ensemble de l'espace opérationnel de dimension inférieure (tel que la transformation d'un ensemble de dimension 1 en un point, par exemple). Par conséquence, il est alors impossible de décrire cette fonction géométrique par une application entre triangulation qui lui soit conjuguée (et qui réalise géométriquement une abstraction basées sur des simplexes abstraits).

Pour les robots 3R que nous considérons, la présence de mouvement interne s'observe le long des sous-ensembles de l'espace articulaire qui induisent que l'effecteur du robot (point P de la Figure 33) est dans l'axe de la seconde rotation (d'angle θ_2). Ce type de situation s'observe sur un grand nombre de robots 3R, notamment lorsque $r_3 = 0$ et $d_4 > d_3$, comme détaillé dans (Wenger, 2007).

Par soucis de complétude nous souhaitons proposer un moyen de décrire à l'aide d'objets combinatoires ces mouvements internes. La description envisagée se base sur une définition proposée d'un *éclatement* mathématique, un concept qui peut être comparé à un mouvement interne.

La description combinatoire avancée pourra alors décrire le comportement d'une fonction présentant certains mouvements internes et plus particulièrement ceux présentés par les robots 3R considérés. Toutefois, la réalisation géométrique de cet objet combinatoire n'est conjuguée à la fonction qu' :

- *en dehors des sous-espaces associés aux mouvements internes.*
- *après identification des mouvements internes aux sous-espaces en lesquels ils se transforment.*

5.7.1. Concepts de Mouvement interne et d'Eclatement

Un mouvement interne s'observe lorsque le point terminal P se situe sur un axe de rotation du robot car, dans ce cas la rotation autour de cet axe n'affecte pas la position de P .

Le mouvement interne est alors associé à un ensemble continu, dans l'espace articulaire, non restreint à un point (un cercle dans ce cas précis) dont l'image par la fonction géométrique f est un point.

La notion d'éclatement est souvent utilisée en théorie des singularités. On se référera, par exemple à (Hu, 2011) et à (Herrmann, et al., 1988).

L'une de ses utilités réside dans son utilisation en tant que moyen de « transformer » un point singulier en une courbe (projective) « moins singulière ». L'objectif de cette transformation est de fournir une façon de décrire une singularité au travers de ses multiplicités et de sa *forme locale*.

La définition formelle *algébrique* d'un éclatement peut être trouvée dans l'ouvrage de référence (Hartshorne, 1977).

Toutefois, la notion d'éclatement peut aussi être considérée dans notre contexte *topologique*.

Une définition topologique d'un éclatement est d'ailleurs énoncée dans (Arnold, 1988) et est compatible avec sa définition *algébrique*.

La définition que nous proposons d'un éclatement *topologique*, d'un point sur une surface vers un cercle topologique, est formulée similairement à la définition énoncée dans (Hartshorne, 1977), de façon à mettre l'accent sur sa similarité avec un éclatement *algébrique*.

Définition 5.7.1.1 (Eclatement topologique)

Soit une surface W et un point $P \in W$.

Un éclatement de P vers un cercle topologique $L \in A$, « tracé sur la surface A » est défini par une *fonction d'éclatement* continue $\vartheta : A \rightarrow W$ telle que $\vartheta|_{A-L} : A - L \rightarrow W - P$ est bijective.

L'éclatement est alors égal à $\vartheta|_{A-L}^{-1}$, en dehors de P qui est, lui, transformé en L , par l'éclatement.

Entre tout couple d'espace topologique, un *éclatement* peut être interprété comme une application entre ces deux topologies (et compatible avec l'intersection et le passage au complémentaire).
La particularité d'un éclatement, parmi les applications entre topologies, est que l'image, par lui, d'au moins un élément topologique, est un élément topologique de dimension *topologique* supérieure.

Dans un espace de dimension 2, l'éclatement $\mathfrak{E}$ d'un point O vers un cercle L (respectivement ou localement une ligne) transforme tout triangle, admettant O comme sommet, en un quadrilatère dont une arête est étendue à L tout entier, comme illustré à la Figure 41.

La fonction d'éclatement $\mathfrak{g}$ vérifie $\mathfrak{g} \circ \mathfrak{E} = Id$

De plus, $\mathfrak{g}$ transforme tout quadrilatère dont une arête est incluse dans L en un triangle admettant O comme l'un de ses sommets, ce que la Figure 41 met également en évidence.

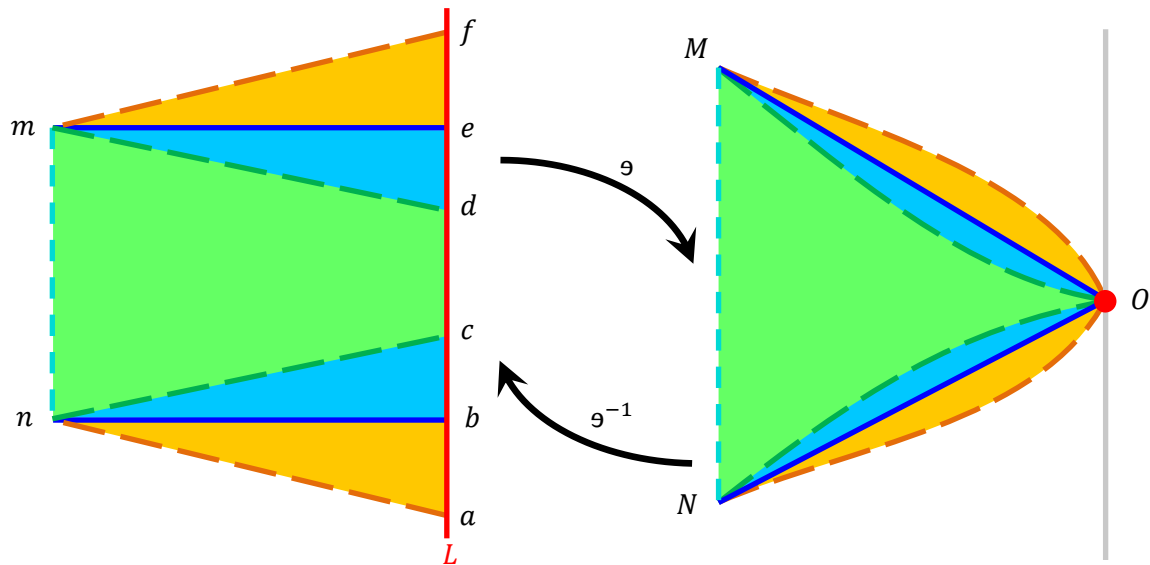


Figure 41 : Eclatement $\mathfrak{E}$ et sa fonction d'éclatement $\mathfrak{g}$, localement équivalente à $(x, y) \mapsto (x, xy)$

5.7.2. Lien entre mouvements internes et éclatement

Un mouvement interne est caractérisé par un sous-ensemble continu de l'espace articulaire, ayant la même image par l'intermédiaire de la fonction f considérée.

L'idée directrice est donc que la fonction f peut être assimilée à une *fonction d'éclatement*, au voisinage du sous-ensemble articulaire associé à chaque mouvement interne.

En se basant sur cette considération et sur le comportement d'un éclatement d'un point vers un cercle, on conclut que, pour chaque mouvement interne de la fonction f , associé à une courbe, f peut être, localement et topologiquement, décrite comme transformant un quadrilatère en un triangle, par identification de l'une des arêtes du premier en un sommet du deuxième, tout en transformant, bijectivement, le reste du quadrilatère en le triangle privé de ce dernier sommet.

Cette description d'un éclatement est illustrée à la Figure 42 ci-après.

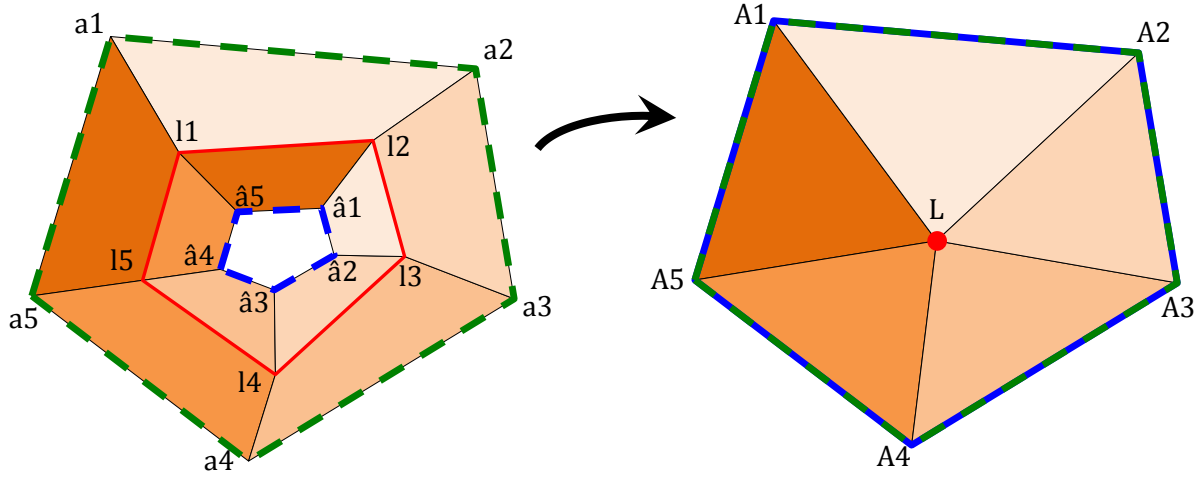


Figure 42 : Description combinatoire locale d'une fonction avec mouvement interne

5.7.3. Quatrilation

Définition 5.7.3.1 (Quatrilation abstraite)

Une quatrilation abstraite $q = c \cup \{z_k \mid k \in I \subset \mathbb{N}\}$ est un ensemble de n -upplets de symboles formé par la réunion d'un complexe simplicial c avec un ensemble d'ensembles de symboles $\{z_k \mid k \in I \subset \mathbb{N}\}$.

On qualifiera de *simplexe généralisé dans* q , tout $s \subseteq q$, tel que $(\exists z \in q \text{ tel que } \forall z' \in s, z' \subseteq z)$.

On dira alors que s est engendré par z dans q et on notera $s = \langle\langle z \rangle\rangle_q$

Dans ce chapitre nous utiliserons plus particulièrement des quatrilations abstraites telles que :

$\forall z \in (q - c)$, il existe $\{\xi_1, \xi_2, \xi_3, \xi_4\}$, un quadruplet de symboles tel que :

$$\langle\langle z \rangle\rangle_q = \{\xi_1, \xi_2, \xi_3, \xi_4\} \cup \langle\langle \xi_1, \xi_2 \rangle\rangle \cup \langle\langle \xi_2, \xi_3 \rangle\rangle \cup \langle\langle \xi_3, \xi_4 \rangle\rangle \cup \langle\langle \xi_4, \xi_1 \rangle\rangle$$

Nous appelons un *carré* de q , élément $\langle\langle z \rangle\rangle_q$ de cette forme et on notera :

$$\langle\langle \xi_1, \xi_2, \xi_3, \xi_4 \rangle\rangle = \{\xi_1, \xi_2, \xi_3, \xi_4\} \cup \langle\langle \xi_1, \xi_2 \rangle\rangle \cup \langle\langle \xi_2, \xi_3 \rangle\rangle \cup \langle\langle \xi_3, \xi_4 \rangle\rangle \cup \langle\langle \xi_4, \xi_1 \rangle\rangle$$

Définition 5.7.3.2 (Quatrilation géométrique)

Une quatrilation géométrique $|q|$ est un objet représentant géométriquement la quatrilation q .

Similairement à la Définition 5.4.2.3, on définit une quatrilation géométrique $|q|$ comme un ensemble minimal d'éléments topologiques incluant une réalisation pour chaque *simplexe généralisé dans* q .

Afin que la réalisation géométrique $|q|$ de la quatrilation q soit définie il faut qu'une réalisation $|s|$ soit définie pour chaque *simplexe généralisé* $s = \langle\langle z \rangle\rangle_q$.

On pourra alors construire $|q|$, dans un espace topologique de dimension suffisante, si toutes les conditions imposées sur les objets constituant chaque réalisation $|s|$ sont compatibles, dans cet espace.

La construction d'une réalisation $|s|$ se fait par récurrence selon les simplexes généralisés $s' \subseteq s$, inclus en cascade les uns dans les autres, à partir de leurs réalisations $|s'| \subseteq |s|$.

Récurrance :

Si $\mathcal{s} = \langle \mathcal{z} \rangle_{\mathcal{q}} = \mathcal{z} \cup (\bigcup_{i \in I} \mathcal{s}_i)$ tel que $|\mathcal{s}_i|$ est défini et est de dimension topologique m_i

Alors $|\mathcal{s}| = |\langle \mathcal{z} \rangle_{\mathcal{q}}|$ vérifie la réunion, sur $i \in I$, des conditions vérifiées par la réalisation $|\mathcal{s}_i|$ et est définie comme $|\mathcal{z}| \cup (\bigcup_{i \in I} |\mathcal{s}_i|)$ où $|\mathcal{z}|$ est l'enveloppe convexe des points réalisant les symboles de $\mathcal{z}$, telle que la dimension topologique de $|\mathcal{z}|$ est $1 + \max_{i \in I} m_i$.

Initialisation :

La réalisation d'un simplexe $\mathcal{s} = \{\{\mathfrak{s}_j\}\}$ est un point M_j (réalisant le symbole $\mathfrak{s}_j$).

Remarque 5.7.3.1 (Analyse de la construction d'une quadrilatation géométrique)

Lorsque la quadrilatation abstraite est un complexe simplicial, la construction proposée est équivalente à celle développée dans la Définition 5.4.2.3. En effet, dans ce cas, la méthode par récurrence proposée aboutit au même résultat que la méthode spécifique aux simplexes abstraits.

Une quadrilatation est donc bien une extension de la notion de complexe simplicial.

Ainsi, tout comme pour une application entre complexes simpliciaux abstraits, dans tout ce chapitre, une application $g : \mathcal{q}_1 \rightarrow \mathcal{q}_2$ sera engendrée par une application $\mathcal{Y}(g) : \mathcal{Y}(\mathcal{q}_1) \rightarrow \mathcal{Y}(\mathcal{q}_2)$ telle que :

$$\forall \mathcal{z} = \{\mathfrak{s}_k \mid k \in \llbracket 1; n \rrbracket\} \in \mathcal{q}_1, \quad g(\mathcal{z}) = \{(\mathcal{Y}(g))(\mathfrak{s}_k) \mid k \in \llbracket 1; n \rrbracket\}$$

De même, une réalisation de $g : \mathcal{q}_1 \rightarrow \mathcal{q}_2$ est une fonction continue $|g| : |\mathcal{q}_1| \rightarrow |\mathcal{q}_2|$ telle que :

$$\forall \mathcal{z} \in \mathcal{q}_1, \quad |g|(|\mathcal{z}|) = |g(\mathcal{z})|$$

Enfin, nous qualifierons de « quadrilatère » la réalisation géométrique d'un carré d'une quadrilatation.

Comme suggéré précédemment, nous proposons de décrire, localement et topologiquement, une *fonction d'éclatement* $\mathfrak{g}$ par une fonction transformant un *quadrilatère* en un *triangle*.

Cette fonction induit une *bijection* en dehors du segment projeté en un point et est équivalente, sur le quadrilatère, à la fonction $(x, y) \mapsto (x, xy)$.

Ainsi, lorsque le comportement considéré, décrit par la fonction f , présente un mouvement interne, projetant localement, par $f : A \rightarrow \mathcal{P}$, un segment en un point, l'espace source A sera alors décrit par une quadrilatation, incluant des carrés, au lieu d'être décrit à l'aide d'un complexe simplicial. On notera que cette quadrilatation $\mathcal{q}_A$ est alors construite en subdivisant les courbes, associées aux mouvements internes, de telle sorte que chacun de ces segments définisse une arête pour deux *carrés* qui seront transformés, par une abstraction de f , en *triangles abstraits* de $\mathcal{c}_{\mathcal{P}}$ (i.e. des simplexes engendrés par des triplets de symboles) par la projection du segment, associé au mouvement interne, en un point.

En utilisant cette méthodologie, toute restriction à $|\mathcal{q}_A|$ d'une réalisation $|\mathcal{c}_f|$ de l'abstraction étendue aux quadrilatations $\mathcal{c}_f : \mathcal{q}_A \rightarrow \mathcal{c}_{\mathcal{P}}$ est conjuguée à f , restreinte à $A' \subset A$, pour tout espace A' ne contenant pas les courbes associées aux mouvements internes de f .

Pour que la réalisation $|\mathcal{c}_f|$ de l'abstraction étendue $\mathcal{c}_f : \mathcal{q}_A \rightarrow \mathcal{c}_{\mathcal{P}}$ de f soit globalement conjuguée à f , il est nécessaire de pouvoir *simultanément* décrire l'action de f , de chaque côté de chaque courbe associée à un mouvement interne, à l'aide d'un nombre fini de carrés.

Toutefois, la vérification qu'un mouvement interne vérifie la propriété d'être descriptible sur un voisinage ouvert à l'aide d'un nombre fini de carrés peut s'avérer difficile.

Nous proposons finalement de décrire f par une quadrilatation $\mathcal{c}_f : \mathcal{q}_A \rightarrow \mathcal{c}_{\mathcal{P}}$, un objet combinatoire dont la réalisation est conjuguée à f , en dehors des courbes associées aux mouvements internes de f .

5.7.4. Construction de la quadrilatation d'une fonction

Les étapes de cette construction sont les mêmes que pour construire l'abstraction d'une fonction.

La seule différence réside dans la nécessité de définir un template pour chaque topologie admettant un bord, associé à un mouvement interne de la fonction f d'intérêt, à l'aide de carrés.

5.7.5. Quatrilation c_f décomposée à l'aide d'une abstraction c_{f_9}

Soit $\vartheta_f : A \rightarrow \vartheta_f(A)$, une fonction d'éclatement, transformant A en identifiant, à un point $O_{i \in I}$, dans $\vartheta_f(A)$, chaque courbe $J_{i \in I}$ associée à un mouvement interne de f , et telle que :

$$\vartheta_f|_{A - \{J_{i \in I} | i \in I\}} : A - \{J_{i \in I} | i \in I\} \rightarrow \vartheta_f(A) - \{O_{i \in I} | i \in I\} \text{ est un homéomorphisme}$$

Soit $f_9 : \vartheta_f(A) \rightarrow \mathcal{P}$ définie par $f = f_9 \circ \vartheta_f$.

Alors on peut écrire $c_f = c_{f_9} \circ c_{\vartheta_f}$ où $c_{f_9} : \vartheta_f(A) \rightarrow \mathcal{C}_P$ abstrait f_9 et où $c_{\vartheta_f} : q_A \rightarrow c_{\vartheta_f(A)}$ abstrait ϑ_f .

Dans ce cas on pourra observer que :

- $|c_{f_9}| \sim_0 f_9$
- $|c_{\vartheta_f}| \sim_0 \vartheta_{f_9}$ si et seulement si q_A décrit A tout en étant compatible avec l'action de f au voisinage des courbes associées aux mouvements internes de f

L'abstraction c_{f_9} code ainsi l'action de f sur un espace $\vartheta_f(A)$ projeté de A où l'action de f , liée à ses mouvement internes est reportée sur l'espace $\vartheta_f(A)$.

La fonction f_9 décrit « la fonction f , épurée de ses mouvements internes ».

Remarque 5.7.5.1 (Construction d'une quatrilation c_f à partir d'une abstraction c_{f_9})

Supposons construite l'abstraction $c_{f_9} : \vartheta_f(A) \rightarrow \mathcal{C}_P$.

Une abstraction $c_f : q_A \rightarrow \mathcal{C}_P$ de $f : A \rightarrow \mathcal{P}$ est alors un objet issu de la procédure suivante :

- On pose initialement $q = c_{\vartheta_f(A)}$.
- $\forall \xi_P \in \mathcal{Y}(c_{\vartheta_f(A)})$ tel que ξ_P abstrait un point $P \in \vartheta_f(A)$ vérifiant que $\vartheta^{-1}(P)$ est un cercle :
 $\forall \mathcal{Z} \in q$ tel que $\mathcal{Z} = \{\xi_P, \xi_1, \xi_2\}$, on substitue le carré $\left(c_{\vartheta_f}^{-1}(\mathcal{Z})\right)_q = \langle \xi_1, \xi_2, \xi_3, \xi_4 \rangle$ à $\mathcal{Z} = \langle \mathcal{Z} \rangle$, dans la quatrilation q , de telle sorte que :
 - Si $\#\mathcal{Z} = \#\mathcal{Z}' = 3$, $\xi_P \in \mathcal{Z} \cap \mathcal{Z}'$ et $\#(\mathcal{Z} \cap \mathcal{Z}') = 2$ Alors $\#(c_{\vartheta_f}^{-1}(\mathcal{Z}) \cap c_{\vartheta_f}^{-1}(\mathcal{Z}')) = 2$
 - q abstrait l'espace A au voisinage de $\vartheta^{-1}(P)$
- On pose $q_A = q$ et $c_f = c_{f_9} \circ c_{\vartheta_f}$.

5.8 Manipulateurs 3R avec mouvements internes

5.8.1. Comportement 4 : « Mouvements internes isolés »

Le comportement 4 est associé à une fonction géométrique f_4 présentant 0 cusp et 0 node et admettant 2 mouvements internes chacun associé à un cercle, isolé dans S_{f_4} (i.e. à une distance non nulle de tout autre point de S_{f_4}).

Pour cet exemple, les paramètres géométriques du robot 3R orthogonal considéré sont :

$$d_2 = 1, \quad d_3 = 0.5, \quad d_4 = 0.8, \quad r_2 = 0.2, \quad r_3 = 0$$

La stratification labélisée $\mathcal{L}(\mathfrak{s}(f_4))$ de la fonction f_4 est représentée par la Figure 43.

En utilisant les désignations de la Figure 43, on note que chaque point de la région I de l'espace opérationnel admet quatre pré-images et que chaque point de E admet deux pré-images, selon f_4 . Toutefois, les points P1 et P2 admettent, eux, une infinité de pré-images, étant donné que le robot décrit par f_4 , admet des mouvements internes au niveau des positions associées à ces points.

De ce fait, la quatrilation géométrique $|c_{f_4}|$, de f_4 , représentée par la Figure 44 n'est pas une triangulation et utilise des *quadrilatères*, réalisant géométriquement des *carrés*, tels que introduits par la Définition 5.7.3.1. Ces *carrés* permettent de décrire l'espace articulaire A , en accord avec l'action de f_4 qui transforme les cercles, associés aux mouvements internes du robot considéré, en les points P1 et P2.

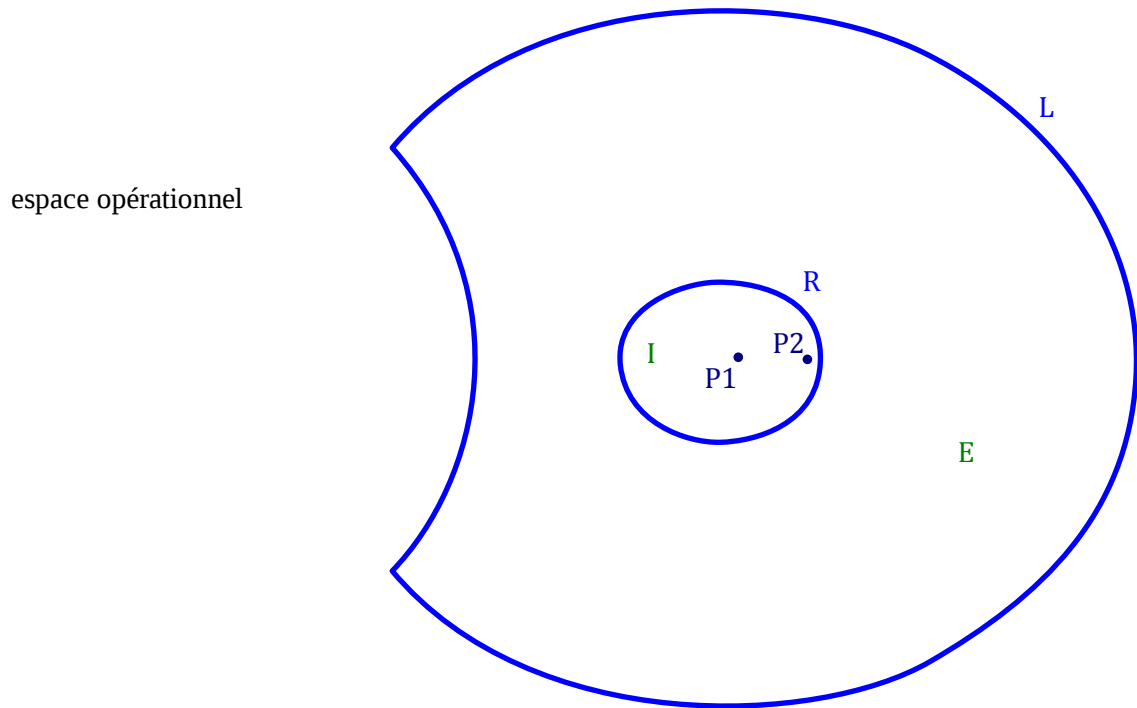
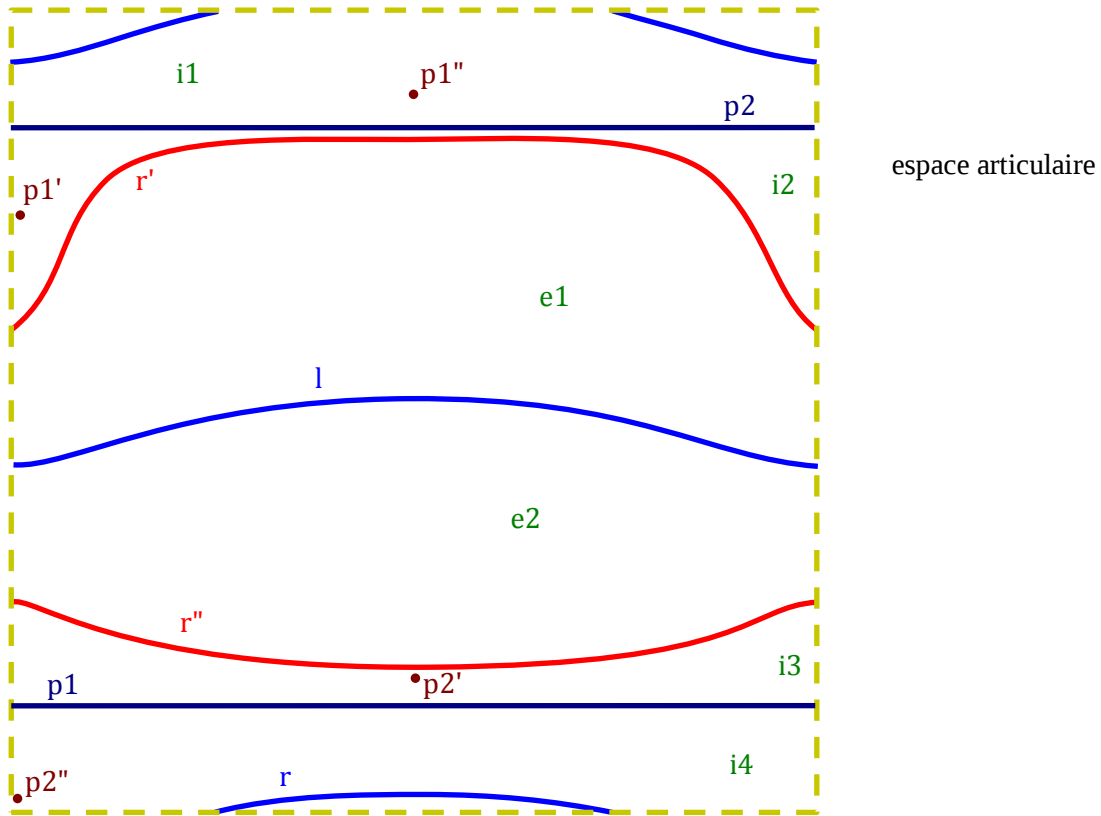


Figure 43 : Stratification tracée labélisée $\mathcal{L}(\mathfrak{s}(f_4))$ de l'exemple pour le comportement 4

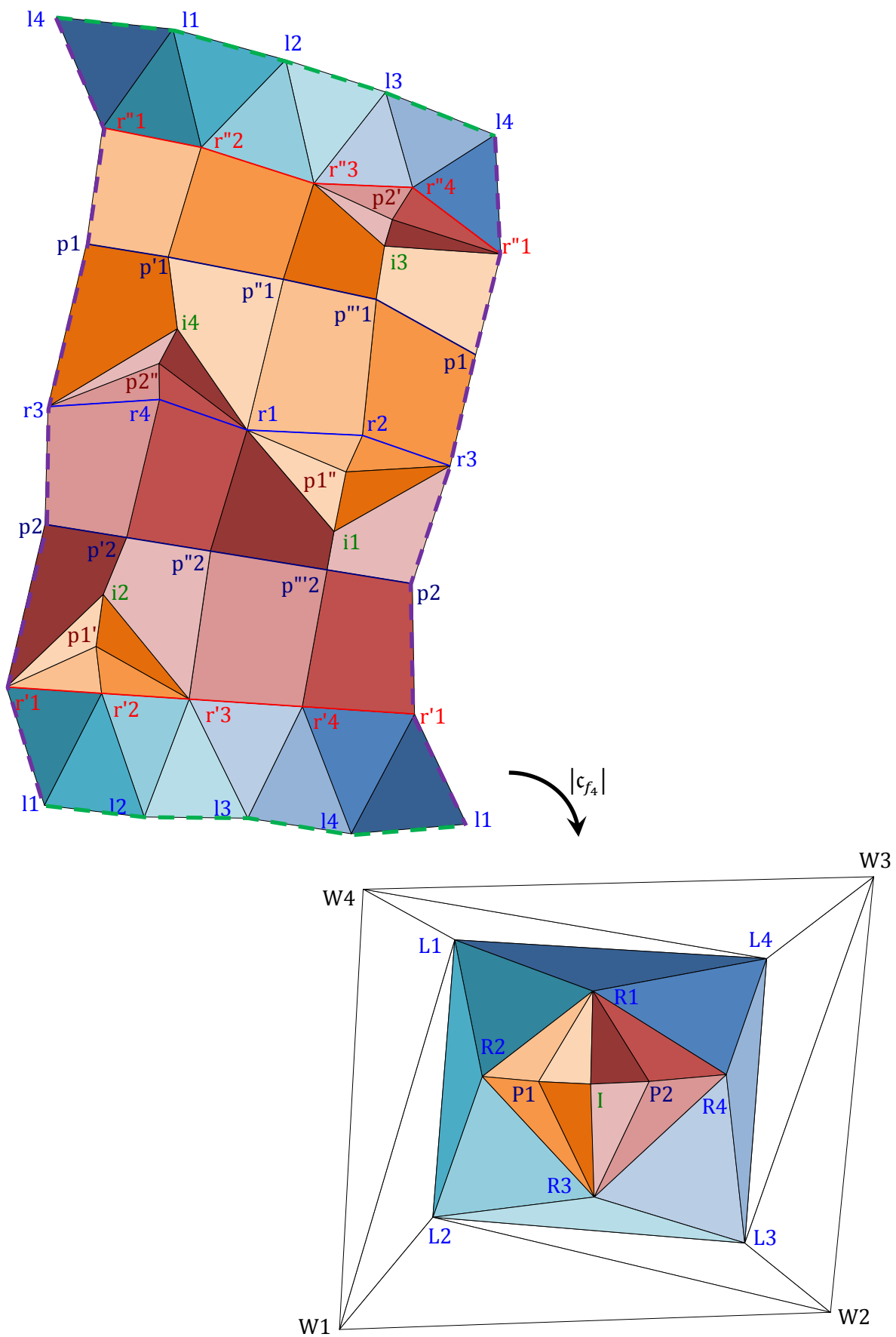


Figure 44 : Quadrilatation du comportement 4

Remarque 5.8.1.1 (Permutations circulaires le long des mouvements internes isolés)

Nous avons détaillé, dans la Remarque 5.7.5.1, une procédure de construction d'une quatrilation c_f de $f : A \rightarrow \mathcal{P}$ en tant que « pré-image » d'une abstraction $c_{f_9} : c_{9(A)} \rightarrow c_{\mathcal{P}}$, où $f = f_9 \circ \vartheta_f$.

Lorsqu'un cercle, associé à un mouvement interne, n'est pas un élément de la stratification $\mathfrak{s}_a(f)$, alors la procédure, de construction de la quatrilation c_f , impose des contraintes, sur la correspondance entre les symboles utilisés pour représenter les éléments topologiques, dont les bords intersectent le cercle associé à ce mouvement interne (du fait que certains points du cercle ont un « rôle particulier »).

Dans le cas où le cercle associé au mouvement interne est isolé dans S_f et est donc une strate de $\mathfrak{s}_a(f)$, les contraintes précédentes ne s'imposent pas. Désignons par $\mathfrak{l}$ la liste des labels associés à un élément de la subdivision, dans $\mathfrak{t}_a(f)$, du cercle associé au mouvement interne, constituée de telle sorte que deux éléments voisins de cette liste sont associés à deux éléments topologiques dont l'un est inclus dans le bord de l'autre. Alors, toute permutation circulaire, sur les éléments de $\mathfrak{l}$, appliquée aux symboles des simplexes généralisés représentant les objets, d'un côté du cercle, induit une autre quatrilation de la fonction f .

5.8.2. Comportement 5 : « Mouvements internes non isolés »

Le comportement 5 est associé à une fonction géométrique f_5 présentant 2 cusps et 2 nodes et admettant 1 mouvement interne associé à un cercle, isolé dans S_{f_4} , mais aussi 1 mouvement interne associé à un cercle, non isolé dans S_{f_4} (i.e. tout voisinage ouvert du cercle contient un point de S_{f_4} , privé du cercle considéré).

Pour cet exemple, les paramètres géométriques du robot 3R orthogonal considéré sont :

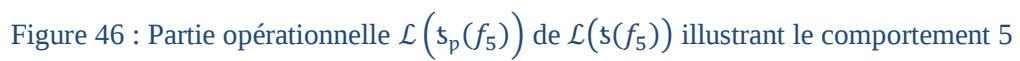
$$d_2 = 1, \quad d_3 = 3, \quad d_4 = 4, \quad r_2 = 2, \quad r_3 = 0$$

La stratification labélisée $\mathcal{L}(\mathfrak{s}(f_5))$ de la fonction f_5 est présentée séparément selon sa partie articulaire $\mathcal{L}(\mathfrak{s}_a(f_5))$ à la Figure 45 et sa partie opérationnelle $\mathcal{L}(\mathfrak{s}_p(f_5))$ à la Figure 46.

En utilisant les désignations de la Figure 45 et de la Figure 46, on vérifie que chaque point de la région I2 et de la région I3 de l'espace opérationnel admet quatre pré-images, selon f_5 , alors que chaque point de la région I1 admet deux pré-images, toujours selon f_5 . Tout comme pour l'exemple illustrant le comportement 4, les points P1 et P2 admettent, eux, une infinité de pré-images, le robot décrit par f_5 , admettant des mouvements internes au niveau des positions associées à ces points.

Tout comme pour la fonction f_4 étudiée à la sous-section 5.8.1, construire une quatrilation c_{f_5} de f_5 nécessite donc d'abstraire certains éléments de l'espace articulaire à l'aide de carrés.

On notera que ce comportement présente la spécificité que le point P2 est confondu avec un node, impliquant qu'aucune permutation circulaire, telle que définie à la Remarque 5.8.1.1 ne peut être appliquée, sur les symboles utilisés pour abstraire $f_5^{-1}(P2)$ dans la quatrilation de l'espace articulaire, utilisée pour définir c_{f_5} , dont une réalisation est présentée à la Figure 47.



Remarque 5.8.2.1 (Cylindre admettant un point appartenant à chacun de ses bords)

La Figure 45 et la Figure 46 montrent que les éléments topologiques e , e' et E peuvent être décrits soit en tant que « disques », admettant chacun deux points confondus sur son bord, soit en tant que cylindres, admettant chacun un point qui appartient simultanément à ses deux bords.

Le template que nous avons proposé pour un disque n'est pas valide pour ce type d'élément car l'identification de deux points du bord fait perdre l'homéomorphisme entre cet espace et le disque unité plan, faisant perdre, de fait la possibilité de le considérer topologiquement comme un disque (qui est défini comme étant en homéomorphisme avec le disque plan unité).

Toutefois, le template proposé pour un cylindre quelconque est valide, si l'on prend la précaution de placer le label, correspondant au point commun, à la même position dans chacune des deux listes, en entrée de l'algorithme de la Figure 29.

Par mesure de simplicité, on placera ce label en première position de chacune des deux listes. On ôtera alors le premier et le dernier triplet de la liste $\mathcal{Q}$, retournée à l'issue de l'exécution de l'algorithme, afin d'obtenir une liste $\mathcal{Q}'$ minimale, d'ensembles de symboles engendrant un complexe simplicial, qui abstrait l'ensemble topologique considéré.

On notera que le premier et le dernier élément de $\mathcal{Q}$ sont redondants pour la définition du complexe simplicial recherché, puisque ceux-ci sont des paires de symboles incluses dans le complexe simplicial précédemment défini à partir de la liste de triplets $\mathcal{Q}'$.

5.9 Conclusion du chapitre et futurs travaux possibles

La contribution principale de ce chapitre réside dans la proposition d'une méthode et d'un algorithme permettant de décrire complètement le comportement d'un manipulateur robotique qui peut être représenté par une fonction géométrique entre deux espaces de dimension 2.

L'approche étudiée dans ce chapitre se propose de décrire les comportements de manipulateurs, au-delà des approches existantes qui fournissent usuellement des méthodes algébriques ou numériques afin de déterminer les singularités des manipulateurs. La méthode d'abstraction proposée est ainsi plus qualitative, tout en étant complémentaire, vis-à-vis des approches existantes puisqu'elle permet d'extraire des informations supplémentaires sur le comportement intrinsèque du manipulateur étudié.

Les principaux avantages de l'abstraction, comparée aux travaux existants sont les suivants :

- L'abstraction décrit complètement la topologie de la plupart des ensembles qui sont usuellement mis en avant pour caractériser une fonction géométrique.
Parmi ceux-ci on citera le contour apparent, le lieu singulier et les courbes caractéristiques.
La connaissance de la topologie de chacun de ces ensembles est de première importance pour la description des trajectoires non singulières de changement de posture (i.e. des trajectoires qui font passer le robot de l'une de ses solutions géométriques inverses à une autre) tel que détaillé dans (Wenger, 2004).
Les topologies du lieu singulier et des courbes caractéristiques permettent aussi de distinguer deux manipulateurs différents admettant des contours apparents de mêmes topologies.
- L'abstraction code l'action de la fonction géométrique sur les ensembles non singuliers, délimités par les courbes singulières et caractéristiques, permettant ainsi, entre autres, de connaître le nombre d'*aspects* (i.e. les ensembles articulaires connexes délimités par le lieu singulier) et le nombre de solutions géométriques inverses dans chacun de ces aspects.
- L'intégralité de l'information extraite est stockée dans l'abstraction, qui, en tant qu'objet combinatoire, peut être à nouveau traitée dans le cadre de comparaisons automatisées.
- La méthodologie d'abstraction n'est pas une méthodologie ad-hoc, restreinte à un type particulier de manipulateur, et est, au contraire, applicable à tout manipulateur dont la fonction géométrique est complètement définie par une fonction entre deux espaces de dimension 2. De plus, la méthodologie n'étant pas restreinte à des espaces de dimension 2, celle-ci peut être étendue aux manipulateurs admettant plus de degrés de liberté (voir ci-après).

L'abstraction proposée peut être réalisée géométriquement par une application entre unions de simplexes, qui est alors (topologiquement) conjuguée à la fonction géométrique étudiée.

L'utilisation des seuls complexes simpliciaux abstraits empêche de décrire les mouvements internes, qui sont définis par des courbes admettant un seul point comme image commune. Une extension de l'abstraction est donc proposée, de façon à gérer la présence de mouvements internes dans le comportement de la fonction géométrique d'intérêt. Cette extension utilise des objets combinatoires supplémentaires qui sont identifiés à des quadrilatères. Cette construction étendue a été mise en œuvre avec succès pour des manipulateurs admettant deux types différents de mouvements internes.

La première perspective immédiate réside dans la définition d'un invariant, en supplément de l'abstraction descriptive. En effet, il est à noter que plusieurs abstractions peuvent être construites pour décrire une même fonction. Cet invariant pourrait alors être formellement défini comme la classe des abstractions possibles pour la fonction considérée, mais il reste à trouver et à définir un objet simple, pour représenter cette classe.

La seconde perspective, qui apparaît naturelle, est d'implémenter la méthodologie proposée dans un langage informatique, de façon à construire automatiquement l'abstraction d'une fonction d'intérêt, entre espaces de dimension 2.

Une perspective de plus grande envergure serait de rendre la méthodologie proposée applicable à des fonctions entre espaces de dimensions plus élevées, puisque cette méthodologie n'est théoriquement pas restreinte à un certain type de fonction. L'applicabilité de la méthodologie d'abstraction est principalement restreinte par la difficulté que présente la définition d'un objet combinatoire, admettant des réalisations géométriques qui sont homéomorphes à un ensemble d'intérêt, de dimension élevée. Naturellement, l'augmentation de la dimension des espaces liés par la fonction d'intérêt, implique de devoir considérer un nombre croissant d'ensembles de spécificités $\mathcal{E}_k$ distincts non vides. Il est alors nécessaire de définir, pour chaque topologie admise par une composante connexe d'un ensemble $\mathcal{E}_k$, non vide, un template combinatoire représentant cette topologie, tout en prenant en compte la topologie de ses bords, qui réunissent alors des éléments dans un nombre, lui aussi croissant, d'ensembles $\mathcal{E}_k$ distincts. La stratégie consistant à définir récursivement le template de chaque élément topologique, à partir des templates de ses bords apparaît alors pertinente. Toutefois, certains ensembles admettant des topologies complexes peuvent mettre en difficulté cette stratégie par les prétraitements qu'ils imposent (tels que des subdivisions possiblement dépendantes des topologies de leurs bords).

Une personne intéressée par l'extension de l'applicabilité de la méthodologie proposée pourra aussi porter son intérêt sur la définition d'un objet, sur la base de $\mathcal{L}(\mathfrak{s}(f))$, qui représenterait, canoniquement, la classe des abstractions possibles pour la fonction f ou un invariant tel que $\mathcal{K}_0(f)$, de façon à s'affranchir de la nécessité de définir explicitement les méthodes de construction pour de nombreux templates, représentant chacun une topologie, comme enveloppe convexe de ses bords, et compatible avec chaque template représentant un des éléments topologique de $\mathfrak{s}(f)$ dans ces bords.

Comme nous l'avons déjà proposé dans la Remarque 5.4.1.3, on peut s'orienter vers un objet de la théorie des graphes pour représenter $\mathcal{K}_0(f)$, en associant à chaque élément topologique E un nœud, annoté de la topologie de E et de son appartenance à l'espace articulaire ou opérationnel de f , et connecté aux nœuds associés aux bords de E et à son image par f ou à ses pré-images par f . Toutefois, il apparaît que certaines propriétés que $\mathcal{K}_0(f)$ devrait conserver, mises en avant par la procédure d'abstraction, ne sont pas contenues dans cette première proposition de représentation.

Ces propriétés, telles que l'orientation relative des deux bords d'un cylindre, doivent nécessairement être conservée d'une certaine manière par la représentation choisie pour que celle-ci code l'invariant $\mathcal{K}_0(f)$ ou la classe des abstractions possibles de f , requérant alors de considérer une approche légèrement différente. Toutefois, l'objet proposé constitue bien un invariant pour l'équivalence $\sim$, qui, bien que, a priori, tronqué de certaines propriétés vis-à-vis de $\mathcal{K}_0(f)$, reste discriminant.

Chapitre 6 Conclusion du mémoire

La motivation générale des travaux présentés dans ce mémoire est l'étude de systèmes réels, et plus particulièrement de systèmes robotiques novateurs, tels que schématisé à la Figure 48.

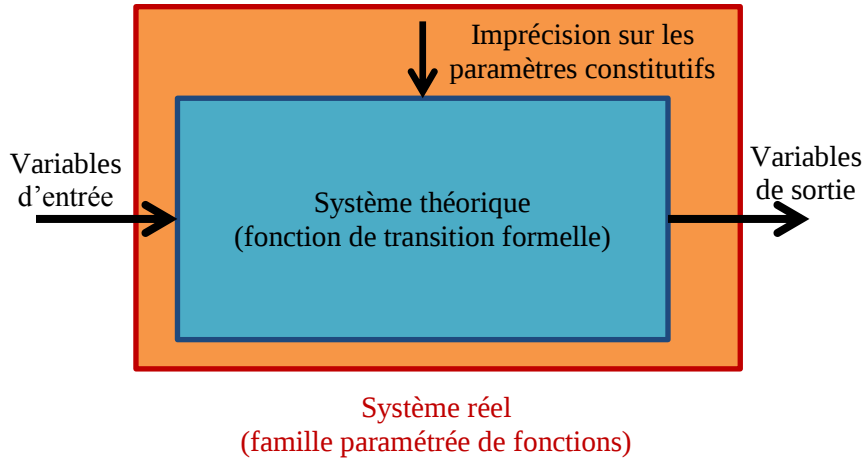


Figure 48 : Schéma général d'un système considéré dans ce mémoire

La capacité théorique de l'Analyse par Intervalles à gérer des fonctions dont les paramètres admettent une variabilité est l'une des motivations principale du choix de cette approche pour l'implémentation des méthodes algorithmiques proposées afin d'étudier les fonctions de transition d'intérêt.

La seconde motivation du choix de l'Analyse par Intervalles est son applicabilité à toute fonction, quelle que soit sa complexité formelle. Etant donné que les fonctions d'intérêt peuvent s'avérer formellement complexes, puisqu'elles définissent le comportement de systèmes novateurs, le choix de l'Analyse par Intervalles a cet avantage de ne pas se restreindre à certains systèmes plus simples.

Notre objectif principal était de classier les fonctions vis-à-vis d'une équivalence $\sim$ entre fonctions, qui correspond à l'existence de difféomorphismes entre les espaces d'entrée et les espaces de sortie des fonctions équivalentes, transformant l'une en l'autre.

Afin de ce faire, nous avons mis en évidence des ensembles caractéristiques d'une fonction f , pour l'équivalence $\sim$, qui sont les classes de singularités $\mathcal{S}_v(f)$ définies par :

$$\mathcal{S}_v(f) = \{x \in \mathcal{S}_{\mathcal{R}(v)} \mid \text{le rang de } f|_{\mathcal{S}_{\mathcal{R}(v)}} \text{ chute de } v_k\}$$

Chaque ensemble $\mathcal{S}_v(f)$ vérifie que sa topologie est partagée par $\mathcal{S}_v(g)$, si $f \sim g$.

Complémentairement, leurs images et contre images vérifient aussi cette propriété.

Nous faisons aussi remarquer que le partage d'image pour plusieurs points est aussi conservé par $\sim$.

Nous avons introduit, de plus, la notion de généricité dans l'étude des systèmes.

Cette notion permet de donner un sens aux affirmations colloquiales :

- Pour la plupart des fonctions, seul un nombre fini d'ensembles caractéristiques sont non vides
- Par suite, il est possible de caractériser formellement ces ensembles caractéristiques

De ceci nous avons extrait une méthodologie générale, au Chapitre 4, pour caractériser formellement, étant données les dimensions topologiques des espaces d'entrée et de sortie de la fonction f étudiée, les classes $\mathcal{S}_v(f)$ et leurs sous-ensembles partageant les mêmes images, génériquement non vides.

Le Chapitre 4 présente, complémentirement, des algorithmes basés sur l'Analyse par Intervalles, implémentant des méthodes de détection numériques garanties des sous-ensembles caractérisés.

Les algorithmes proposés pour chaque dimension d'espaces étudiée suivent des principes généraux, qui peuvent être appliqués pour la définition d'un algorithme de détection de tout point d'intérêt. Toutefois, pour chaque dimension prise en compte, les algorithmes proposés mettent en lumière des spécificités différentes de l'utilisation de méthodes basées sur l'Analyse par Intervalles.

Par souci de cohérence avec l'aspect garanti de l'Analyse par Intervalles, chaque algorithme proposé est implémenté de façon à pouvoir s'appliquer à toute fonction d'intérêt, y compris non générique.

A partir de la connaissance de la topologie de chacun des sous-ensembles d'intérêt et de leurs dispositions relatives, nous proposons une méthode algorithmique construisant un objet combinatoire, descripteur de la fonction étudiée.

La méthode proposée au Chapitre 5 est générale et requiert uniquement la définition préalable d'un template combinatoire pour chaque topologie extraite, afin de représenter celle-ci de façon cohérente avec les templates représentant la topologie de chaque composante de son bord.

La méthodologie proposée se résume à :

1. Isoler les ensembles caractéristiques pour $\sim$ afin de pouvoir en extraire :
 - a. Leurs topologies
 - b. Leurs dispositions relatives
2. Subdiviser les ensembles isolés de façon à ce qu'ils soient compatibles avec les templates définis et choisis avant d'exécuter la procédure.
3. Convertir les ensembles en objets combinatoires selon les templates et construire l'objet final, d'après le lien entre les ensembles isolés correspondants, par la fonction d'intérêt.

Les templates proposés au Chapitre 5 sont des complexes simpliciaux, dont la réalisation géométrique est homéomorphe à l'ensemble correspondant, pour chaque template considéré.

Une représentation généralisée permet de décrire les fonctions géométriques des robots admettant des mouvements internes, à l'aide de templates représentant l'image, par une fonction d'éclatement, d'ensembles dont la topologie peut déjà être représentée par un template.

L'objet final proposé admet ainsi une réalisation géométrique topologiquement conjuguée à la fonction d'intérêt considérée, sur chaque sous-espace où celle-ci est localement injective.

Je conclus ainsi ce mémoire de thèse sur un panorama de perspectives à développer que j'envisage :

- Définir des formules explicites caractérisant les ensembles $\mathcal{S}_v(f)$ pour une fonction f générale
- Recherche des templates représentant des topologies d'ensembles de dimension supérieure à 2
- Adapter la méthodologie proposée au Chapitre 5, pour construire un objet invariant pour $\sim$ et non seulement un objet descripteur de la fonction d'intérêt, relativement à l'équivalence $\sim$. (Une piste serait de construire un graphe annoté ou un objet dérivé d'une telle construction.)
- Réaliser une étude théorique comparant les invariants $\mathcal{K}_0(f)$ et $\mathcal{K}_1(f)$ et, éventuellement, délimiter des conditions, sur la fonction f , pour que $\mathcal{K}_0(f)$ et $\mathcal{K}_1(f)$ soient « égaux ». (Possiblement lorsque f admet un comportement différentiel trivial sur chaque ensemble $\mathcal{S}_v(f)$ vérifiant la propriété : $\exists k \in \mathbb{N} - \{0\}$ tel que $v_k = 0$)
- Optimisation de la détection des collisions et de la prise en compte des limites matérielles. Optimiser la méthode du Chapitre 5, pour prendre en compte plus efficacement ces limites. (Celles-ci peuvent être prises en compte par la méthode. Toutefois, la construction pratique des pré-images et images des espaces accessibles peut être délicate, notamment si l'on considère des encadrements par intervalles trop grossiers de ces espaces.)

Complémentairement, toute perspective qui apparait au lecteur, de par son expertise, ses connaissances et ses savoirs faire, me semble toute aussi pertinente à être envisagée.

Annexes

A.1 Expressions formelles des fonctions décrivant les robots 3R

Nous reportons, ici, des expressions formelles des fonctions issues de la fonction géométrique associée à un robot positionneur 3R orthogonal, utilisées par les algorithmes de détection de cusps et de nodes.

Fonction géométrique simplifiée :

On considère ici $f : (\theta_2, \theta_3) \mapsto (f_{\rho^2}(\theta_2, \theta_3), f_z(\theta_2, \theta_3))$ où :

- $f_{\rho^2}(\theta_2, \theta_3) = (\cos \theta_2 (d_3 + d_4 \cos \theta_3) + d_2 - r_3 \sin \theta_2)^2 + (r_2 + d_4 \sin \theta_3)^2$
- $f_z(\theta_2, \theta_3) = \sin \theta_2 (d_3 + d_4 \cos \theta_3) + r_3 \cos \theta_2$

Matrice Jacobienne $Df : (\theta_2, \theta_3) \mapsto (Df_{i,j}(\theta_2, \theta_3))_{(i,j) \in \llbracket 1,2 \rrbracket}$

- $Df_{1,1}(\theta_2, \theta_3) = \frac{\partial f_{\rho^2}}{\partial \theta_2}(\theta_2, \theta_3) = -2(\sin \theta_2 (d_4 \cos \theta_3 + d_3) + r_3 \cos \theta_2)(d_2 + \cos \theta_2 (d_4 \cos \theta_3 + d_3) - r_3 \sin \theta_2)$
- $Df_{1,2}(\theta_2, \theta_3) = \frac{\partial f_{\rho^2}}{\partial \theta_3}(\theta_2, \theta_3) = -2d_4(\cos \theta_2 \sin \theta_3 (d_2 + \cos \theta_2 (d_4 \cos \theta_3 + d_3) - r_3 \sin \theta_2) - \cos \theta_3 (d_4 \sin \theta_3 + r_2))$
- $Df_{2,1}(\theta_2, \theta_3) = \frac{\partial f_z}{\partial \theta_2}(\theta_2, \theta_3) = \cos \theta_2 (d_4 \cos \theta_3 + d_3) - r_3 \sin \theta_2$
- $Df_{2,2}(\theta_2, \theta_3) = \frac{\partial f_z}{\partial \theta_3}(\theta_2, \theta_3) = -d_4 \sin \theta_2 \sin \theta_3$

Expressions déduites de celle de la Matrice Jacobienne Df :

Déterminant de Df :

$$\begin{aligned} (\det(Df))(\theta_2, \theta_3) &= 2d_4 \left((d_3 + d_4 \cos \theta_3)(d_2 \sin \theta_3 + \cos \theta_2 (d_3 \sin \theta_3 - r_2 \cos \theta_3)) \right. \\ &\quad \left. + r_3 \sin \theta_2 (r_2 \cos \theta_3 - d_3 \sin \theta_3) \right) \end{aligned}$$

Gradient du déterminant de Df :

$$\left(\nabla(\det(Df)) \right)(\theta_2, \theta_3) = \left(\frac{\partial(\det(Df))}{\partial \theta_2}(\theta_2, \theta_3), \frac{\partial(\det(Df))}{\partial \theta_3}(\theta_2, \theta_3) \right)$$

où :

- $\frac{\partial(\det(Df))}{\partial \theta_2}(\theta_2, \theta_3) = 2d_4((r_2 \cos \theta_3 - d_3 \sin \theta_3)(\sin \theta_2 (d_3 + d_4 \cos \theta_3) + r_3 \cos \theta_2))$
- $\frac{\partial(\det(Df))}{\partial \theta_3}(\theta_2, \theta_3) = 2d_4((d_3 \cos \theta_3 + r_2 \sin \theta_3)(d_2 + d_3 \cos \theta_2 + 2d_4 \cos \theta_2 \cos \theta_3 - r_3 \sin \theta_2) + d_4(d_2 - d_3 \cos \theta_2) - d_2 \sin \theta_3 (2d_4 \sin \theta_3 + r_2))$

Toute expression proportionnelle, par un facteur non nul, à $\det(Df)$ et au vecteur $\nabla(\det(Df))$, peut être substituée à l'expression correspondante pour les raisonnements effectués.

De ce fait, lorsque $d_4 \neq 0$, les algorithmes implémentés utilisent les expressions de :

- $\det(Df)/2d_4$ à la place de celle de $\det(Df)$
- $\nabla(\det(Df))/2d_4$ à la place de celle de $\nabla(\det(Df))$

Les autres expressions requises pour les algorithmes de détection proposés, présentent des complexités formelles relativement élevées, réduisant ainsi la pertinence de leur reproduction ici.

Ces autres expressions se déduisent de celles, ci avant, par dérivations et produits matriciels successifs réalisables à l'aide d'un logiciel de calcul formel d'après les formules du § 4.3.1.3 et du § 4.3.2.1.

A.2 Appendices au Chapitre 5

A.2.1. Conventions choisies pour les figures

Pour illustrer une application $\alpha : E \rightarrow F$, entre deux espaces (de dimension topologique 2) il est nécessaire de clairement définir la façon dont α transforme les éléments de E en éléments de F . De plus, certains espaces de dimension 2, tels que le tore ou le cône, ne peuvent être représentés, tels quels, dans le plan et donc sur une page.

Afin de pouvoir illustrer les applications considérées dans des figures et que la lecture de ces figures soit la plus claire et la plus naturelle possible, nous avons choisi de respecter certaines conventions *qui seront en vigueur pour la totalité de ce document*.

Ces conventions se classent selon deux catégories :

1. Codage du lien entre éléments sources et but
2. Représentation d'espaces à l'aide de patrons

A.2.1.1. Codage du lien entre éléments sources et but

Le codage du lien, par $\alpha : E \rightarrow F$, entre les éléments de l'espace source E et de l'espace but F vérifie :

- Des conventions de couleur :
 - Si une même couleur (autre que celle par défaut de la page) repère $E' \subseteq E$ et $F' \subseteq F$ alors $\alpha(E') = F'$ ou, autrement dit, $E' \subseteq \alpha^{-1}(F')$
 - Dans le cas où α est la réalisation de l'abstraction d'une fonction f (i.e. $\alpha = c_f$) la même couleur est utilisée pour les triangles ayant la même image et pour leur image
- Des conventions typographiques pour les désignations d'éléments :
 - Les désignations des éléments sources (dans E) utilisent des *lettres minuscules*
 - Les désignations des éléments buts (dans F) utilisent des *lettres majuscules*
 - La désignation d'une pré-image $E' \subseteq E$ d'un élément but $F' \subseteq F$ vérifie : Après retrait des caractères « ' » et « " » (apostrophe et guillemet simple) de la désignation de E' , celle-ci contient la désignation de F' où les lettres de celle-ci sont remplacées par leurs équivalents en minuscule, éventuellement accentués.

Pour illustrer les conventions typographiques, considérons, par exemple une application α telle que :

$$\alpha : \{n1a, n1b, a12, a1'2, a1''2, p\} \rightarrow \{N1, A12, P\}$$

On a alors, selon les conventions :

$$\alpha^{-1}(N1) = \{n1a, n1b\}, \alpha^{-1}(A12) = \{a12, a1'2, a1''2\} \text{ et } \alpha^{-1}(P) = \{p\}$$

A.2.1.2. Représentation d'espaces à l'aide de patrons

Tout espace de dimension topologique 2 peut être localement identifié à un sous-espace du plan $\mathbb{R}^2$. De ce fait, tout espace de dimension 2 peut être décrit par une liste de patrons et une méthode explicite de recollement entre ceux-ci. (Sous certaines hypothèses on prouve qu'il existe une telle liste finie.)

Dans le cas particulier de certains espaces compact de dimension 2 tels que le tore, il est ainsi possible de décrire cet espace à l'aide d'un unique patron, sous-espace compact du plan et d'une règle d'identification le long des bords du patron générateur.

Pour décrire de tels espaces, nous utiliserons alors la convention selon laquelle les bords du patron, doivent être identifiés au niveau des éléments ayant la même désignation, de même que au niveau des sous-ensembles des bords délimités par ces éléments de mêmes désignations.

A.2.2. Liste des abstractions présentées

A.2.2.1. Comportement 1: « Tore vu du dessus »

- 0 cusp et 0 node
- Paramètres géométriques : $d_2 = 1$, $d_3 = 0.7$, $d_4 = 0.3$, $r_2 = 0.3$, $r_3 = 0$

Abstraction du comportement 1 (c_{f_1}) :

- source :

$\langle s2, s3, l1h \rangle \cup \langle s1, s2, l4h \rangle \cup \langle s1, s4, l3h \rangle \cup \langle s3, s4, l2h \rangle \cup \langle s2, s3, l1b \rangle \cup \langle s1, s2, l4b \rangle \cup \langle s1, s4, l3b \rangle$
 $\cup \langle s3, s4, l2b \rangle \cup \langle r2, r3, l4h \rangle \cup \langle r1, r2, l3h \rangle \cup \langle r1, r4, l2h \rangle \cup \langle r3, r4, l1h \rangle \cup \langle r2, r3, l4b \rangle \cup \langle r1, r2, l3b \rangle$
 $\cup \langle r1, r4, l2b \rangle \cup \langle r3, r4, l1b \rangle \cup \langle l2h, l3h, s4 \rangle \cup \langle l1h, l2h, s3 \rangle \cup \langle l1h, l4h, s2 \rangle \cup \langle l3h, l4h, s1 \rangle$
 $\cup \langle l2b, l3b, s4 \rangle \cup \langle l1b, l2b, s3 \rangle \cup \langle l1b, l4b, s2 \rangle \cup \langle l3b, l4b, s1 \rangle \cup \langle l2h, l3h, r1 \rangle \cup \langle l1h, l2h, r4 \rangle$
 $\cup \langle l1h, l4h, r3 \rangle \cup \langle l3h, l4h, r2 \rangle \cup \langle l2b, l3b, r1 \rangle \cup \langle l1b, l2b, r4 \rangle \cup \langle l1b, l4b, r3 \rangle \cup \langle l3b, l4b, r2 \rangle$

- image :

$\langle S2, S3, L1 \rangle \cup \langle S1, S2, L4 \rangle \cup \langle S1, S4, L3 \rangle \cup \langle S3, S4, L2 \rangle \cup \langle S2, S3, W2 \rangle \cup \langle S1, S2, W1 \rangle$
 $\cup \langle S1, S4, W4 \rangle \cup \langle S3, S4, W3 \rangle \cup \langle R2, R3, L4 \rangle \cup \langle R1, R2, L3 \rangle \cup \langle R1, R4, L2 \rangle \cup \langle R3, R4, V \rangle$
 $\cup \langle R2, R3, V \rangle \cup \langle R1, R2, V \rangle \cup \langle R1, R4, V \rangle \cup \langle R3, R4, V \rangle \cup \langle L2, L3, S4 \rangle \cup \langle L1, L2, S3 \rangle$
 $\cup \langle L1, L4, S2 \rangle \cup \langle L3, L4, S1 \rangle \cup \langle L2, L3, R1 \rangle \cup \langle L1, L2, R4 \rangle \cup \langle L1, L4, R3 \rangle \cup \langle L3, L4, R2 \rangle$
 $\cup \langle W2, W3, S3 \rangle \cup \langle W1, W2, S2 \rangle \cup \langle W1, W4, S1 \rangle \cup \langle W3, W4, S4 \rangle$

- lien :

$\forall k \in \llbracket 1; 3 \rrbracket$, $c_{f_1}(rk) = Rk$, $c_{f_1}(sk) = Sk$, $c_{f_1}(lkh) = c_{f_1}(lkb) = Lk$

A.2.2.2. Comportement 2 : « Cusps seuls »

- 4 cusps et 0 node
- Paramètres géométriques : $d_2 = 1$, $d_3 = 2$, $d_4 = 1.5$, $r_2 = 1$, $r_3 = 0.5$

Abstraction du comportement 2 (c_{f_2}) :

- source :

$\langle l1, l2, \hat{a}34' \rangle \cup \langle l2, l3, c3 \rangle \cup \langle l3, l4, \hat{a}23 \rangle \cup \langle l4, l5, c2 \rangle \cup \langle l5, l6, \hat{a}1'2 \rangle \cup \langle l6, l7, c1' \rangle \cup \langle l7, l8, \hat{a}1'4' \rangle$
 $\cup \langle l8, l1, c4' \rangle \cup \langle l1, l2, \hat{a}3'4' \rangle \cup \langle l2, l3, c3' \rangle \cup \langle l3, l4, \hat{a}2'3' \rangle \cup \langle l4, l5, c2' \rangle \cup \langle l5, l6, \hat{a}1'2' \rangle \cup \langle l6, l7, c1' \rangle$
 $\cup \langle l7, l8, \hat{a}1'4' \rangle \cup \langle l8, l1, c4' \rangle \cup \langle c4', \hat{a}34', l1 \rangle \cup \langle \hat{a}34', c3, l2 \rangle \cup \langle c3, \hat{a}23, l3 \rangle \cup \langle \hat{a}23, c2, l4 \rangle \cup \langle c2, \hat{a}1'2, l5 \rangle$
 $\cup \langle \hat{a}1'2, c1', l6 \rangle \cup \langle c1', \hat{a}1'4', l7 \rangle \cup \langle \hat{a}1'4', c4', l8 \rangle \cup \langle c4', \hat{a}3'4, l1 \rangle \cup \langle \hat{a}3'4, c3', l2 \rangle \cup \langle c3', \hat{a}2'3', l3 \rangle$
 $\cup \langle \hat{a}2'3', c2', l4 \rangle \cup \langle c2', \hat{a}12', l5 \rangle \cup \langle \hat{a}12', c1, l6 \rangle \cup \langle c1, \hat{a}14, l7 \rangle \cup \langle \hat{a}14, c4, l8 \rangle \cup \langle c4', \hat{a}34', l1 \rangle$
 $\cup \langle \hat{a}34', c3, l1 \rangle \cup \langle c3, a23, l1 \rangle \cup \langle a23, c2, l1 \rangle \cup \langle c2, \hat{a}1'2, l1 \rangle \cup \langle \hat{a}1'2, c1', l1 \rangle \cup \langle c1', \hat{a}1'4', l1 \rangle$
 $\cup \langle \hat{a}1'4', c4', l1 \rangle \cup \langle c4, a34, l2 \rangle \cup \langle a34, c3, l2 \rangle \cup \langle c3, a23, l2 \rangle \cup \langle a23, c2, l2 \rangle \cup \langle c2, a12, l2 \rangle$
 $\cup \langle a12, c1, l2 \rangle \cup \langle c1, \hat{a}14, l2 \rangle \cup \langle \hat{a}14, c4, l2 \rangle \cup \langle c4, a34, l3 \rangle \cup \langle a34, c3, l3 \rangle \cup \langle c3, \hat{a}23, l3 \rangle \cup \langle \hat{a}23, c2, l3 \rangle$
 $\cup \langle c2, a12, l3 \rangle \cup \langle a12, c1, l3 \rangle \cup \langle c1, a14, l3 \rangle \cup \langle a14, c4, l3 \rangle \cup \langle c4, \hat{a}3'4, l4 \rangle \cup \langle \hat{a}3'4, c3', l4 \rangle$
 $\cup \langle c3', \hat{a}2'3', l4 \rangle \cup \langle \hat{a}2'3', c2', l4 \rangle \cup \langle c2', \hat{a}12', l4 \rangle \cup \langle \hat{a}12', c1, l4 \rangle \cup \langle c1, a14, l4 \rangle \cup \langle a14, c4, l4 \rangle$

- image :

$\langle C4, A34, L1 \rangle \cup \langle A34, C3, L2 \rangle \cup \langle C3, A23, L3 \rangle \cup \langle A23, C2, L4 \rangle \cup \langle C2, A12, L5 \rangle \cup \langle A12, C1, L6 \rangle$
 $\cup \langle C1, A14, L7 \rangle \cup \langle A14, C4, L8 \rangle \cup \langle L1, L2, A34 \rangle \cup \langle L2, L3, C3 \rangle \cup \langle L3, L4, A23 \rangle \cup \langle L4, L5, C2 \rangle$
 $\cup \langle L5, L6, A12 \rangle \cup \langle L6, L7, C1 \rangle \cup \langle L7, L8, A14 \rangle \cup \langle L8, L1, C4 \rangle \cup \langle C4, A34, I \rangle \cup \langle A34, C3, I \rangle$
 $\cup \langle C3, A23, I \rangle \cup \langle A23, C2, I \rangle \cup \langle C2, A12, I \rangle \cup \langle A12, C1, I \rangle \cup \langle C1, A14, I \rangle \cup \langle A14, C4, I \rangle$
 $\cup \langle L1, L2, W1 \rangle \cup \langle L2, L3, W2 \rangle \cup \langle L3, L4, W3 \rangle \cup \langle L4, L5, W4 \rangle \cup \langle L5, L6, W5 \rangle \cup \langle L6, L7, W6 \rangle$
 $\cup \langle L7, L8, W7 \rangle \cup \langle L1, L8, W8 \rangle \cup \langle W1, W2, L2 \rangle \cup \langle W2, W3, L3 \rangle \cup \langle W3, W4, L4 \rangle \cup \langle W4, W5, L5 \rangle$
 $\cup \langle W5, W6, L6 \rangle \cup \langle W6, W7, L7 \rangle \cup \langle W7, W8, L8 \rangle \cup \langle W1, W8, L1 \rangle$

- lien :

$\forall k \in \llbracket 1; 8 \rrbracket$, $c_{f_2}(lk) = Lk$; $\forall k \in \llbracket 1; 4 \rrbracket$, $c_{f_2}(ik) = I$, $c_{f_2}(ck) = c_{f_2}(ck') = Ck$;

$\forall (j, k) \in \llbracket 1; 4 \rrbracket \times \llbracket 1; 4 \rrbracket$, $c_{f_2}(ajk) = c_{f_2}(ajk') = c_{f_2}(aj'k) = c_{f_2}(aj'k') = Ajk$, de même que:

$\forall (j, k) \in \llbracket 1; 4 \rrbracket \times \llbracket 1; 4 \rrbracket$, $c_{f_2}(\hat{a}jk) = c_{f_2}(\hat{a}jk') = c_{f_2}(\hat{a}j'k) = c_{f_2}(\hat{a}j'k') = Ajk$

A.2.2.3. Comportement 3 : « Exemple complet »

- 4 cusps et 2 nodes
- Paramètres géométriques : $d_2 = 1$, $d_3 = 1.5$, $d_4 = 0.7$, $r_2 = 0.5$, $r_3 = 0.5$

Abstraction du comportement 3 (c_{f_3}) :

- source :

$\langle i2, c1, \hat{a}n11 \rangle \cup \langle i2, \hat{a}n11, n1 \rangle \cup \langle i2, n1, an12 \rangle \cup \langle i2, an12, c2 \rangle \cup \langle i2, c2, a12 \rangle \cup \langle i2, a12, c1 \rangle$
 $\cup \langle i3, c4, \hat{a}n24 \rangle \cup \langle i3, \hat{a}n24, n2 \rangle \cup \langle i3, n2, an23 \rangle \cup \langle i3, an23, c3 \rangle \cup \langle i3, c3, a34 \rangle \cup \langle i3, a34, c4 \rangle$
 $\cup \langle i2', c1, an1'1 \rangle \cup \langle i2', an1'1, n1' \rangle \cup \langle i2', n1', \hat{a}n1'2 \rangle \cup \langle i2', \hat{a}n1'2, c2' \rangle \cup \langle i2', c2', a12' \rangle \cup \langle i2', a12', c1' \rangle$
 $\cup \langle i3', c4, an2'4 \rangle \cup \langle i3', an2'4, n2' \rangle \cup \langle i3', n2', \hat{a}n2'3 \rangle \cup \langle i3', \hat{a}n2'3, c3' \rangle \cup \langle i3', c3', a34' \rangle \cup \langle i3', a34', c4' \rangle$
 $\cup \langle i2'', c1, an1''1 \rangle \cup \langle i2'', an1''1, n1'' \rangle \cup \langle i2'', n1'', \hat{a}n1''2 \rangle \cup \langle i2'', \hat{a}n1''2, c2'' \rangle \cup \langle i2'', c2'', a12'' \rangle \cup \langle i2'', a12'', c1'' \rangle$
 $\cup \langle i2'', \hat{a}12'', c1' \rangle \cup \langle i3'', c4, an2''4 \rangle \cup \langle i3'', an2''4, n2'' \rangle \cup \langle i3'', n2'', \hat{a}n2''3 \rangle \cup \langle i3'', \hat{a}n2''3, c3'' \rangle$
 $\cup \langle i3'', c3'', \hat{a}3'4 \rangle \cup \langle i3'', \hat{a}3'4, c4' \rangle \cup \langle i2''', c1', \hat{a}n11' \rangle \cup \langle i2''', \hat{a}n11', n1' \rangle \cup \langle i2''', n1', an12' \rangle$
 $\cup \langle i2''', an12, c2' \rangle \cup \langle i2''', c2', \hat{a}1'2 \rangle \cup \langle i2''', \hat{a}1'2, c1' \rangle \cup \langle i3''', c4', \hat{a}n24' \rangle \cup \langle i3''', \hat{a}n24', n2' \rangle$
 $\cup \langle i3''', n2, an23' \rangle \cup \langle i3''', an23, c3' \rangle \cup \langle i3''', c3, \hat{a}34' \rangle \cup \langle i3''', \hat{a}34', c4' \rangle \cup \langle u6, c1, \hat{a}n11 \rangle \cup \langle u7, \hat{a}n11, n1 \rangle$
 $\cup \langle u2, n1', \hat{a}n1'2' \rangle \cup \langle u3, \hat{a}n1'2', c2' \rangle \cup \langle u4, c2', \hat{a}12' \rangle \cup \langle u5, \hat{a}12', c1' \rangle \cup \langle u11, c4, \hat{a}n24 \rangle \cup \langle u10, \hat{a}n24, n2 \rangle$
 $\cup \langle u15, n2', \hat{a}n2'3' \rangle \cup \langle u14, \hat{a}n2'3, c3' \rangle \cup \langle u13, c3', \hat{a}3'4' \rangle \cup \langle u12, \hat{a}3'4, c4' \rangle \cup \langle an2n1, n2', u16 \rangle$
 $\cup \langle n1', an2n1, u1' \rangle \cup \langle an1n2, n2, u9' \rangle \cup \langle n1, an1n2, u8' \rangle \cup \langle u1, u2, n1' \rangle \cup \langle u2, u3, \hat{a}n1'2' \rangle \cup \langle u3, u4, c2' \rangle$
 $\cup \langle u4, u5, \hat{a}12' \rangle \cup \langle u5, u6, c1' \rangle \cup \langle u6, u7, \hat{a}n11' \rangle \cup \langle u7, u8, n1' \rangle \cup \langle u8, u9, an1n2' \rangle \cup \langle u9, u10, n2' \rangle$
 $\cup \langle u10, u11, \hat{a}n24' \rangle \cup \langle u11, u12, c4' \rangle \cup \langle u12, u13, \hat{a}3'4' \rangle \cup \langle u13, u14, c3' \rangle \cup \langle u14, u15, \hat{a}n2'3' \rangle$
 $\cup \langle u15, u16, n2' \rangle \cup \langle u16, u1, an1n2' \rangle \cup \langle \hat{u}6, c1', \hat{a}n11' \rangle \cup \langle \hat{u}7, \hat{a}n11', n1' \rangle \cup \langle \hat{u}2, n1', \hat{a}n1'2' \rangle$
 $\cup \langle \hat{u}3, \hat{a}n1'2, c2' \rangle \cup \langle \hat{u}4, c2', \hat{a}1'2' \rangle \cup \langle \hat{u}5, \hat{a}1'2, c1' \rangle \cup \langle \hat{u}11, c4', \hat{a}n24' \rangle \cup \langle \hat{u}10, \hat{a}n24', n2' \rangle$
 $\cup \langle \hat{u}15, n2', \hat{a}n2'3' \rangle \cup \langle \hat{u}14, \hat{a}n2'3, c3' \rangle \cup \langle \hat{u}13, c3', \hat{a}3'4' \rangle \cup \langle \hat{u}12, \hat{a}3'4, c4' \rangle \cup \langle an2n1, n2', \hat{u}16 \rangle$
 $\cup \langle n1', an2n1, \hat{u}1' \rangle \cup \langle an1n2, n2, \hat{u}9' \rangle \cup \langle n1, an1n2, \hat{u}8' \rangle \cup \langle \hat{u}1, \hat{u}2, n1' \rangle \cup \langle \hat{u}2, \hat{u}3, \hat{a}n1'2' \rangle \cup \langle \hat{u}3, \hat{u}4, c2' \rangle$
 $\cup \langle \hat{u}4, \hat{u}5, \hat{a}1'2' \rangle \cup \langle \hat{u}5, \hat{u}6, c1' \rangle \cup \langle \hat{u}6, \hat{u}7, \hat{a}n11' \rangle \cup \langle \hat{u}7, \hat{u}8, n1' \rangle \cup \langle \hat{u}8, \hat{u}9, an1n2' \rangle \cup \langle \hat{u}9, \hat{u}10, n2' \rangle$
 $\cup \langle \hat{u}10, \hat{u}11, \hat{a}n24' \rangle \cup \langle \hat{u}11, \hat{u}12, c4' \rangle \cup \langle \hat{u}12, \hat{u}13, \hat{a}3'4' \rangle \cup \langle \hat{u}13, \hat{u}14, c3' \rangle \cup \langle \hat{u}14, \hat{u}15, \hat{a}n2'3' \rangle$
 $\cup \langle \hat{u}15, \hat{u}16, n2' \rangle \cup \langle \hat{u}16, \hat{u}1, an2n1' \rangle \cup \langle u1, u2, l2' \rangle \cup \langle u2, u3, l3' \rangle \cup \langle u3, u4, l4' \rangle \cup \langle u4, u5, l5' \rangle$
 $\cup \langle u5, u6, l6' \rangle \cup \langle u6, u7, l7' \rangle \cup \langle u7, u8, l8' \rangle \cup \langle u8, u9, l9' \rangle \cup \langle u9, u10, l10' \rangle \cup \langle u10, u11, l11' \rangle$
 $\cup \langle u11, u12, l12' \rangle \cup \langle u12, u13, l13' \rangle \cup \langle u13, u14, l14' \rangle \cup \langle u14, u15, l15' \rangle \cup \langle u15, u16, l16' \rangle \cup \langle u16, u1, l1' \rangle$
 $\cup \langle l1, l2, u1' \rangle \cup \langle l2, l3, u2' \rangle \cup \langle l3, l4, u3' \rangle \cup \langle l4, l5, u4' \rangle \cup \langle l5, l6, u5' \rangle \cup \langle l6, l7, u6' \rangle \cup \langle l7, l8, u7' \rangle$
 $\cup \langle l8, l9, u8' \rangle \cup \langle l9, l10, u9' \rangle \cup \langle l10, l11, u10' \rangle \cup \langle l11, l12, u11' \rangle \cup \langle l12, l13, u12' \rangle \cup \langle l13, l14, u13' \rangle$
 $\cup \langle l14, l15, u14' \rangle \cup \langle l15, l16, u15' \rangle \cup \langle l16, l1, u16' \rangle \cup \langle \hat{u}1, \hat{u}2, l2' \rangle \cup \langle \hat{u}2, \hat{u}3, l3' \rangle \cup \langle \hat{u}3, \hat{u}4, l4' \rangle$
 $\cup \langle \hat{u}4, \hat{u}5, l5' \rangle \cup \langle \hat{u}5, \hat{u}6, l6' \rangle \cup \langle \hat{u}6, \hat{u}7, l7' \rangle \cup \langle \hat{u}7, \hat{u}8, l8' \rangle \cup \langle \hat{u}8, \hat{u}9, l9' \rangle \cup \langle \hat{u}9, \hat{u}10, l10' \rangle \cup \langle \hat{u}10, \hat{u}11, l11' \rangle$
 $\cup \langle \hat{u}11, \hat{u}12, l12' \rangle \cup \langle \hat{u}12, \hat{u}13, l13' \rangle \cup \langle \hat{u}13, \hat{u}14, l14' \rangle \cup \langle \hat{u}14, \hat{u}15, l15' \rangle \cup \langle \hat{u}15, \hat{u}16, l16' \rangle \cup \langle \hat{u}16, \hat{u}1, l1' \rangle$
 $\cup \langle l1, l2, \hat{u}1' \rangle \cup \langle l2, l3, \hat{u}2' \rangle \cup \langle l3, l4, \hat{u}3' \rangle \cup \langle l4, l5, \hat{u}4' \rangle \cup \langle l5, l6, \hat{u}5' \rangle \cup \langle l6, l7, \hat{u}6' \rangle \cup \langle l7, l8, \hat{u}7' \rangle$
 $\cup \langle l8, l9, \hat{u}8' \rangle \cup \langle l9, l10, \hat{u}9' \rangle \cup \langle l10, l11, \hat{u}10' \rangle \cup \langle l11, l12, \hat{u}11' \rangle \cup \langle l12, l13, \hat{u}12' \rangle \cup \langle l13, l14, \hat{u}13' \rangle$
 $\cup \langle l14, l15, \hat{u}14' \rangle \cup \langle l15, l16, \hat{u}15' \rangle \cup \langle l16, l1, \hat{u}16' \rangle$

- image :

$\langle I2, C1, AN11 \rangle \cup \langle I2, AN11, N1 \rangle \cup \langle I2, N1, AN12 \rangle \cup \langle I2, AN12, C2 \rangle \cup \langle I2, C2, A12 \rangle \cup \langle I2, A12, C1 \rangle$
 $\cup \langle I3, C4, AN24 \rangle \cup \langle I3, AN24, N2 \rangle \cup \langle I3, N2, AN23 \rangle \cup \langle I3, AN23, C3 \rangle \cup \langle I3, C3, A34 \rangle$
 $\cup \langle I3, A34, C4 \rangle \cup \langle U6, C1, AN11 \rangle \cup \langle U7, AN11, N1 \rangle \cup \langle U2, N1, AN12 \rangle \cup \langle U3, AN12, C2 \rangle$
 $\cup \langle U4, C2, A12 \rangle \cup \langle U5, A12, C1 \rangle \cup \langle U11, C4, AN24 \rangle \cup \langle U10, AN24, N2 \rangle \cup \langle U15, N2, AN23 \rangle$
 $\cup \langle U14, AN23, C3 \rangle \cup \langle U13, C3, A34 \rangle \cup \langle U12, A34, C4 \rangle \cup \langle AN2N1, N2, U16 \rangle \cup \langle N1, AN2N1, U1 \rangle$
 $\cup \langle AN1N2, N2, U9 \rangle \cup \langle N1, AN1N2, U8 \rangle \cup \langle U1, U2, N1 \rangle \cup \langle U2, U3, AN12 \rangle \cup \langle U3, U4, C2 \rangle$
 $\cup \langle U4, U5, A12 \rangle \cup \langle U5, U6, C1 \rangle \cup \langle U6, U7, AN11 \rangle \cup \langle U7, U8, N1 \rangle \cup \langle U8, U9, AN1N2 \rangle$
 $\cup \langle U9, U10, N2 \rangle \cup \langle U10, U11, AN24 \rangle \cup \langle U11, U12, C4 \rangle \cup \langle U12, U13, A34 \rangle \cup \langle U13, U14, C3 \rangle$
 $\cup \langle U14, U15, AN23 \rangle \cup \langle U15, U16, N2 \rangle \cup \langle U16, U1, AN2N1 \rangle \cup \langle U1, U2, L2 \rangle \cup \langle U2, U3, L3 \rangle$
 $\cup \langle U3, U4, L4 \rangle \cup \langle U4, U5, L5 \rangle \cup \langle U5, U6, L6 \rangle \cup \langle U6, U7, L7 \rangle \cup \langle U7, U8, L8 \rangle \cup \langle U8, U9, L9 \rangle$
 $\cup \langle U9, U10, L10 \rangle \cup \langle U10, U11, L11 \rangle \cup \langle U11, U12, L12 \rangle \cup \langle U12, U13, L13 \rangle \cup \langle U13, U14, L14 \rangle$
 $\cup \langle U14, U15, L15 \rangle \cup \langle U15, U16, L16 \rangle \cup \langle U16, U1, L1 \rangle \cup \langle L1, L2, U1 \rangle \cup \langle L2, L3, U2 \rangle \cup \langle L3, L4, U3 \rangle$
 $\cup \langle L4, L5, U4 \rangle \cup \langle L5, L6, U5 \rangle \cup \langle L6, L7, U6 \rangle \cup \langle L7, L8, U7 \rangle \cup \langle L8, L9, U8 \rangle \cup \langle L9, L10, U9 \rangle$
 $\cup \langle L10, L11, U10 \rangle \cup \langle L11, L12, U11 \rangle \cup \langle L12, L13, U12 \rangle \cup \langle L13, L14, U13 \rangle \cup \langle L14, L15, U14 \rangle$
 $\cup \langle L15, L16, U15 \rangle \cup \langle L16, L1, U16 \rangle \cup \langle L1, L2, W1 \rangle \cup \langle L2, L3, W2 \rangle \cup \langle L3, L4, W3 \rangle \cup \langle L4, L5, W4 \rangle$
 $\cup \langle L5, L6, W5 \rangle \cup \langle L6, L7, W6 \rangle \cup \langle L7, L8, W7 \rangle \cup \langle L8, L9, W8 \rangle \cup \langle L9, L10, W9 \rangle \cup \langle L10, L11, W10 \rangle$
 $\cup \langle L11, L12, W11 \rangle \cup \langle L12, L13, W12 \rangle \cup \langle L13, L14, W13 \rangle \cup \langle L14, L15, W14 \rangle \cup \langle L15, L16, W15 \rangle$
 $\cup \langle L1, L16, W16 \rangle \cup \langle W1, W2, L2 \rangle \cup \langle W2, W3, L3 \rangle \cup \langle W3, W4, L4 \rangle \cup \langle W4, W5, L5 \rangle \cup \langle W5, W6, L6 \rangle$
 $\cup \langle W6, W7, L7 \rangle \cup \langle W7, W8, L8 \rangle \cup \langle W8, W9, L9 \rangle \cup \langle W9, W10, L10 \rangle \cup \langle W10, W11, L11 \rangle$
 $\cup \langle W11, W12, L12 \rangle \cup \langle W12, W13, L13 \rangle \cup \langle W13, W14, L14 \rangle \cup \langle W14, W15, L15 \rangle \cup \langle W15, W16, L16 \rangle$
 $\cup \langle W1, W16, L1 \rangle \cup \langle V, N1, AN1N2 \rangle \cup \langle V, AN1N2, N2 \rangle \cup \langle V, N2, AN2N1 \rangle \cup \langle V, AN2N1, N1 \rangle$

- lien :

$\forall k \in \llbracket 1; 4 \rrbracket, c_{f_3}(ck) = c_{f_3}(ck') = Ck ; \forall k \in \llbracket 1; 16 \rrbracket, c_{f_3}(lk) = Lk, c_{f_3}(uk) = c_{f_3}(\hat{u}k) = Uk ;$
 $\forall (j, k) \in \llbracket 1; 4 \rrbracket \times \llbracket 1; 4 \rrbracket, c_{f_3}(ajk) = c_{f_3}(ajk') = c_{f_3}(aj'k) = c_{f_3}(aj'k') = Ajk, \text{ de même que :}$
 $\forall (j, k) \in \llbracket 1; 4 \rrbracket \times \llbracket 1; 4 \rrbracket, c_{f_3}(\hat{a}jk) = c_{f_3}(\hat{a}jk') = c_{f_3}(\hat{a}j'k) = c_{f_3}(\hat{a}j'k') = Ajk$
 $\forall (j, k) \in \llbracket 1; 2 \rrbracket \times \llbracket 1; 4 \rrbracket, c_{f_3}(anj k) = c_{f_3}(anj k') = c_{f_3}(anj'k) = c_{f_3}(anj'k') = ANjk, \text{ de même :}$
 $\forall (j, k) \in \llbracket 1; 2 \rrbracket \times \llbracket 1; 4 \rrbracket, c_{f_3}(\hat{a}nj k) = c_{f_3}(\hat{a}nj k') = c_{f_3}(\hat{a}nj'k) = c_{f_3}(\hat{a}nj'k') = ANjk ;$
 $\forall k \in \llbracket 1; 2 \rrbracket, c_{f_3}(nk) = c_{f_3}(nk') = Nk ; c_{f_3}(an1n2) = AN1N2 ; c_{f_3}(an2n1) = AN2N1$

A.2.2.4. Comportement 4 : « Mouvements internes isolés »

- 4 cusps, 0 node et 2 mouvements internes isolés
- Paramètres géométriques : $d_2 = 1, d_3 = 0.7, d_4 = 0.3, r_2 = 0.3, r_3 = 0$

Abstraction du comportement 4 (c_{f_4}) :

- source :

$\langle r'1, r'2, p'1, p'1 \rangle \cup \langle r'2, r'3, p'1, p'1 \rangle \cup \langle r'3, i3, p'1, p'1 \rangle \cup \langle i3, r'1, p'1, p'1 \rangle \cup \langle r'3, r'4, p'2 \rangle$
 $\cup \langle r'4, r'1, p'2 \rangle \cup \langle r'3, i3, p'2 \rangle \cup \langle i3, r'1, p'2 \rangle \cup \langle r1, r2, p'1, p'1 \rangle \cup \langle r2, r3, p'1, p'1 \rangle \cup \langle r1, i4, p'1, p'1 \rangle$
 $\cup \langle i4, r3, p'1, p'1 \rangle \cup \langle r3, r4, p'2 \rangle \cup \langle r4, r1, p'2 \rangle \cup \langle r3, i4, p'2 \rangle \cup \langle i4, r1, p'2 \rangle \cup \langle r3, r4, p'2, p'2 \rangle$
 $\cup \langle r4, r1, p'2, p'2 \rangle \cup \langle r3, i1, p'2, p'2 \rangle \cup \langle i1, r1, p'2, p'2 \rangle \cup \langle r1, r2, p'1 \rangle \cup \langle r2, r3, p'1 \rangle \cup \langle r1, i1, p'1 \rangle$
 $\cup \langle i1, r3, p'1 \rangle \cup \langle r'3, r'4, p'2, p'2 \rangle \cup \langle r'4, r'1, p'2, p'2 \rangle \cup \langle r'3, i2, p'2, p'2 \rangle \cup \langle i2, r'1, p'2, p'2 \rangle$
 $\cup \langle r'1, r'2, p'1 \rangle \cup \langle r'2, r'3, p'1 \rangle \cup \langle r'1, i2, p'1 \rangle \cup \langle i2, r'3, p'1 \rangle \cup \langle a1, a2, r'2 \rangle \cup \langle a2, a3, r'3 \rangle \cup \langle a3, a4, r'4 \rangle$
 $\cup \langle a4, a5, r'1 \rangle \cup \langle r'1, r'2, a1 \rangle \cup \langle r'2, r'3, a2 \rangle \cup \langle r'3, r'4, a3 \rangle \cup \langle r'4, r'1, a4 \rangle \cup \langle a1, a2, r'2 \rangle \cup \langle a2, a3, r'3 \rangle$
 $\cup \langle a3, a4, r'4 \rangle \cup \langle a4, a5, r'1 \rangle \cup \langle r'1, r'2, a1 \rangle \cup \langle r'2, r'3, a2 \rangle \cup \langle r'3, r'4, a3 \rangle \cup \langle r'4, r'1, a4 \rangle$

- image :

$\langle A1, A2, R2 \rangle \cup \langle A2, A3, R3 \rangle \cup \langle A3, A4, R4 \rangle \cup \langle A4, A5, R1 \rangle \cup \langle R1, R2, A1 \rangle \cup \langle R2, R3, A2 \rangle$
 $\cup \langle R3, R4, A3 \rangle \cup \langle R4, R1, A4 \rangle \cup \langle R1, R2, P1 \rangle \cup \langle R2, R3, P1 \rangle \cup \langle R3, R4, P2 \rangle \cup \langle R4, R1, P2 \rangle$
 $\cup \langle R1, I, P1 \rangle \cup \langle I, R3, P1 \rangle \cup \langle R3, I, P2 \rangle \cup \langle I, R1, P2 \rangle \cup \langle A2, A3, W2 \rangle \cup \langle A1, A2, W1 \rangle \cup \langle A1, A4, W4 \rangle$
 $\cup \langle A3, A4, W3 \rangle \cup \langle W2, W3, A3 \rangle \cup \langle W1, W2, A2 \rangle \cup \langle W1, W4, A1 \rangle \cup \langle W3, W4, A4 \rangle$

- lien :

$\forall k \in \llbracket 1; 4 \rrbracket, c_{f_4}(rk) = c_{f_4}(r'k) = c_{f_4}(r''k) = Rk, c_{f_4}(ak) = Ak, c_{f_4}(ik) = I ;$
 $\forall k \in \llbracket 1; 2 \rrbracket, c_{f_4}(pk) = c_{f_4}(p'k) = c_{f_4}(p''k) = c_{f_4}(p'''k) = c_{f_4}(pk') = c_{f_4}(pk'') = Pk$

A.2.2.5. Comportement 5 : « Mouvements internes non isolés »

- 2 cusps, 2 nodes, 1 mouvement interne isolé et 1 mouvement interne non isolé
- Paramètres géométriques : $d_2 = 1$, $d_3 = 3$, $d_4 = 4$, $r_2 = 2$, $r_3 = 0$

Abstraction du comportement 5 (c_{f_5}) :

- source :

$\langle i1, n1', ann1'2' \rangle \rightarrow \langle i1, n2', ann1'2' \rangle \rightarrow \langle i1, n1', \hat{a}n1'n2' \rangle \rightarrow \langle i1, n2', \hat{a}n1'n2' \rangle \rightarrow \langle i1', n1', ann1'2' \rangle$
 $\rightarrow \langle i1', n2', ann1'2' \rangle \rightarrow \langle i1', n1', \hat{a}n1'n2' \rangle \rightarrow \langle i1', n2', \hat{a}n1'n2' \rangle \rightarrow \langle p1', n1', \hat{a}n1'n2' \rangle \rightarrow \langle p1', n2', \hat{a}n1'n2' \rangle$
 $\rightarrow \langle p1', n1', i2' \rangle \rightarrow \langle p1', n2', i2' \rangle \rightarrow \langle ap2''''n2', n2', i2' \rangle \rightarrow \langle ap2''''n1', n1', i2' \rangle \rightarrow \langle ap2''''n2', p2''', p2', i2' \rangle$
 $\rightarrow \langle ap2''''n1', p2''', p2', i2' \rangle \rightarrow \langle p1', p1', n1', \hat{a}n1'n2' \rangle \rightarrow \langle p1', p1', n2', \hat{a}n1'n2' \rangle \rightarrow \langle p1', p1', n1', i2' \rangle$
 $\rightarrow \langle p1', p1', n2', i2' \rangle \rightarrow \langle ap2''''n2', n2', i2' \rangle \rightarrow \langle ap2''''n1', n1', i2' \rangle \rightarrow \langle ap2''''n2', p2''', p2', i2' \rangle$
 $\rightarrow \langle ap2''''n1', p2''', p2', i2' \rangle \rightarrow \langle p1', p1', n1, an1n2 \rangle \rightarrow \langle p1', p1', n2, an1n2 \rangle \rightarrow \langle p1', p1', n1, i2' \rangle$
 $\rightarrow \langle p1', p1', n2, i2' \rangle \rightarrow \langle \hat{a}p2''n2, n2, i2' \rangle \rightarrow \langle \hat{a}p2''n1, n1, i2' \rangle \rightarrow \langle \hat{a}p2''n2, p2'', i2' \rangle \rightarrow \langle \hat{a}p2''n1, p2'', i2' \rangle$
 $\rightarrow \langle p1'', n1, an1n2 \rangle \rightarrow \langle p1'', n2, an1n2 \rangle \rightarrow \langle p1'', n1, i2' \rangle \rightarrow \langle p1'', n2, i2' \rangle \rightarrow \langle \hat{a}p2''n2, n2, i2' \rangle$
 $\rightarrow \langle \hat{a}p2''n1, n1, i2' \rangle \rightarrow \langle \hat{a}p2''n2, p2'', i2' \rangle \rightarrow \langle \hat{a}p2''n1, p2'', i2' \rangle \rightarrow \langle p2'', \hat{a}p2''2', i3' \rangle \rightarrow \langle p2'', \hat{a}p2''1', i3' \rangle$
 $\rightarrow \langle c2', \hat{a}p2''2', i3' \rangle \rightarrow \langle c1', \hat{a}p2''1', i3' \rangle \rightarrow \langle c2', \hat{a}1'2', i3' \rangle \rightarrow \langle c1', \hat{a}1'2', i3' \rangle \rightarrow \langle p2'', \hat{a}p2''2, i3' \rangle$
 $\rightarrow \langle p2'', \hat{a}p2''1, i3' \rangle \rightarrow \langle c2, \hat{a}p2''2, i3' \rangle \rightarrow \langle c1, \hat{a}p2''1, i3' \rangle \rightarrow \langle c2, a12, i3' \rangle \rightarrow \langle c1, a12, i3' \rangle$
 $\rightarrow \langle p2, p2''', ap2''''2, i3' \rangle \rightarrow \langle p2, p2''', ap2''''1, i3' \rangle \rightarrow \langle c2, ap2''''2, i3' \rangle \rightarrow \langle c1, ap2''''1, i3' \rangle \rightarrow \langle c2, a12, i3' \rangle$
 $\rightarrow \langle c1, a12, i3' \rangle \rightarrow \langle p2'', p2''', ap2''''2, i3' \rangle \rightarrow \langle p2'', p2''', ap2''''1, i3' \rangle \rightarrow \langle c2, ap2''''2, i3' \rangle \rightarrow \langle c1, ap2''''1, i3' \rangle$
 $\rightarrow \langle c2, \hat{a}12, i3' \rangle \rightarrow \langle c1, \hat{a}12, i3' \rangle \rightarrow \langle \hat{a}p2''n2, p2'', \hat{a}p2''2' \rangle \rightarrow \langle \hat{a}p2''n1, p2'', \hat{a}p2''1' \rangle$
 $\rightarrow \langle c2', \hat{a}p2''2', \hat{a}p2''n2 \rangle \rightarrow \langle c1', \hat{a}p2''1', n1 \rangle \rightarrow \langle c2', \hat{a}1'2', n2 \rangle \rightarrow \langle c1', \hat{a}1'2', an2n1 \rangle \rightarrow \langle \hat{a}p2''n2, n2, c2' \rangle$
 $\rightarrow \langle \hat{a}p2''n1, n1, \hat{a}p2''1' \rangle \rightarrow \langle n1, c1', an2n1 \rangle \rightarrow \langle n2, \hat{a}1'2', an2n1 \rangle \rightarrow \langle \hat{a}p2''n2, p2'', \hat{a}p2''2' \rangle$
 $\rightarrow \langle \hat{a}p2''n1, p2'', \hat{a}p2''1' \rangle \rightarrow \langle c2, \hat{a}p2''2, \hat{a}p2''n2 \rangle \rightarrow \langle c1, \hat{a}p2''1, n1 \rangle \rightarrow \langle c2, \hat{a}12, n2 \rangle \rightarrow \langle c1, \hat{a}12, an2n1 \rangle$
 $\rightarrow \langle \hat{a}p2''n2, n2, c2' \rangle \rightarrow \langle \hat{a}p2''n1, n1, \hat{a}p2''1' \rangle \rightarrow \langle n1, c1, an2n1 \rangle \rightarrow \langle n2, \hat{a}12, an2n1 \rangle$

- image :

$\langle I1, N1, ANN12 \rangle \rightarrow \langle I1, N2, ANN12 \rangle \rightarrow \langle I1, N1, AN1N2 \rangle \rightarrow \langle I1, N2, AN1N2 \rangle \rightarrow \langle P1, N1, AN1N2 \rangle$
 $\rightarrow \langle P1, N2, AN1N2 \rangle \rightarrow \langle P1, N1, I2 \rangle \rightarrow \langle P1, N2, I2 \rangle \rightarrow \langle AP2N2, N2, I2 \rangle \rightarrow \langle AP2N1, N1, I2 \rangle$
 $\rightarrow \langle AP2N2, P2, I2 \rangle \rightarrow \langle AP2N1, P2, I2 \rangle \rightarrow \langle P2, AP22, I3 \rangle \rightarrow \langle P2, AP21, I3 \rangle \rightarrow \langle C2, AP22, I3 \rangle$
 $\rightarrow \langle C1, AP21, I3 \rangle \rightarrow \langle C2, A12, I3 \rangle \rightarrow \langle C1, A12, I3 \rangle \rightarrow \langle AP2N2, P2, AP22 \rangle \rightarrow \langle AP2N1, P2, AP21 \rangle$
 $\rightarrow \langle C2, AP22, AP2N2 \rangle \rightarrow \langle C1, AP21, N1 \rangle \rightarrow \langle C2, A12, N2 \rangle \rightarrow \langle C1, A12, AN2N1 \rangle \rightarrow \langle AP2N2, N2, C2 \rangle$
 $\rightarrow \langle AP2N1, N1, AP21 \rangle \rightarrow \langle N1, C1, AN2N1 \rangle \rightarrow \langle N2, A12, AN2N1 \rangle \rightarrow \langle N1, AN2N1, W2 \rangle$
 $\rightarrow \langle N1, ANN12, W1 \rangle \rightarrow \langle ANN12, N2, W4 \rangle \rightarrow \langle N2, AN2N1, W3 \rangle \rightarrow \langle W2, W3, AN2N1 \rangle \rightarrow \langle W1, W2, N1 \rangle$
 $\rightarrow \langle W1, W4, ANN12 \rangle \rightarrow \langle W3, W4, N2 \rangle$

- lien :

$c_{f_5}(an1n2) = c_{f_5}(\hat{a}n1'n2') = c_{f_5}(\hat{a}n1'n2') = AN1N2$; $c_{f_5}(an2n1) = AN2N1$;
 $c_{f_5}(ap2''''n1') = c_{f_5}(\hat{a}p2''n1) = c_{f_5}(\hat{a}p2''n1) = AP2N1$;
 $c_{f_5}(ap2n2') = c_{f_5}(\hat{a}p2''n2) = c_{f_5}(\hat{a}p2''n2) = AP2N2$;
 $c_{f_5}(ap2'2) = c_{f_5}(\hat{a}p2''2) = c_{f_5}(\hat{a}p2''2') = AP22$;
 $c_{f_5}(ap2''''1) = c_{f_5}(\hat{a}p2''1) = c_{f_5}(\hat{a}p2''1') = AP21$;
 $\forall k \in \llbracket 1; 2 \rrbracket, c_{f_5}(ck) = c_{f_5}(ck') = Ck$; $c_{f_5}(a12) = c_{f_5}(\hat{a}12) = c_{f_5}(\hat{a}1'2') = A12$;
 $\forall k \in \llbracket 1; 3 \rrbracket, c_{f_5}(ik) = c_{f_5}(ik') = c_{f_5}(ik'') = c_{f_5}(ik''') = Ik$

A.3 Nomenclature

Avec, dans l'ordre d'apparition dans ce mémoire :

- A : l'espace articulaire du robot considéré actuellement
- $\mathcal{P}$: l'espace opérationnel du robot considéré actuellement
- W : l'espace de travail du robot considéré actuellement
- C : espace des configurations admissible du robot considéré actuellement
- C_s : système des contraintes sur la configuration du robot considéré actuellement
- $\#E$: le nombre d'éléments de E
- S_f : lieu singulier de la fonction (différentiable) f et du robot défini par cette fonction
- $Df(z)$: matrice associée à df_z
- df_z : la différentielle de f , en z
- « variété » : « variété différentielle » (synonyme du terme anglais « manifold »)
- $\pitchfork$: intersection transverse (détail de la transversalité précisée en indice)
- $\mathcal{C}^\infty$ ou « lisse » : « indéfiniment différentiable »
- Pour $\alpha = (k_j)_{j \in J}$ où, $\forall j \in J, k_j \in \mathbb{N}^*$ (α est un multi-indice)
 - $|\alpha| = \#J \in \mathbb{N}$
 - $z_\alpha = \prod_{j \in J} z_{k_j}$ (en particulier $\partial x_\alpha = \prod_{j \in J} \partial x_{k_j}$)
- $\mathcal{T}_M E$: espace tangent à l'espace E , au point M voir aussi la Définition 4.3.1.2
- $J^k(X, Y)$: espace des k -jets des fonctions de $\mathcal{C}^\infty(X, Y)$
- $j^k f$: k -jet de f (identifiable à ses développements de Taylor à l'ordre k)
- $\mathcal{T}_E$: topologie associée à l'espace E
- $\mathcal{T}^k(X, Y)$: topologie $\mathcal{C}^k$ de Whitney (usuellement associée à $\mathcal{C}^\infty(X, Y)$)
- $\llbracket a; b \rrbracket = [a, b] \cap \mathbb{Z}$
- $\mathfrak{Z}^m(X) = \{(x_i)_{i \in \llbracket 1; m \rrbracket} \mid \exists (i, j) \in \llbracket 1; m \rrbracket^2, i \neq j \text{ et } x_i = x_j\} \subset X^m$ (Définition 2.2.4.1)
- $\mathfrak{X}_m(X) = X^m - \mathfrak{Z}^m(X)$: X^m privé de sa « diagonale étendue » $\mathfrak{Z}^m(X)$ (Définition 2.2.4.1)
- $\mathfrak{X}_m(f) : \mathfrak{X}_m(X) \rightarrow Y^m$: produit cartésien de m f restreint à $\mathfrak{X}_m(X)$ (Définition 2.2.4.1)
- $j_m^k(f) = j^k(\mathfrak{X}_m(f))$: le m -multi- k -jet de f (Définition 2.2.4.1)
- $\mathbb{I}_{\mathcal{T}_E}$: l'ensemble des éléments connexes de $\mathcal{T}_E$ (Définition 2.3.1.1)
- $\mathbb{I}E = \mathbb{I}(E) = \mathbb{I}_{\mathcal{T}_E}$: ensemble des intervalles de E (où $\mathcal{T}_E$ est sans ambiguïté la topologie de E)
- w_Ω : une fonction de taille sur Ω (Définition 2.3.1.3)
- w : une fonction de taille (sur un ensemble quelconque ou sans ambiguïté)
- $[f]_{\Omega_A; \Omega_P}$: une fonction d'inclusion de f , où $[f]_{\Omega_A; \Omega_P} : \Omega_A \rightarrow \Omega_P$ (Définition 2.3.1.4)
- $[f]$: une fonction $[f]_{\Omega_A; \Omega_P}$ lorsque (Ω_A, Ω_P) est fixé où quelconque
- $\langle V \rangle_\Omega$: un intervalle minimal de V dans Ω (Définition 2.3.1.5)
- $\langle f \rangle_{\Omega_A; \Omega_P}$: une fonction d'inclusion minimale de f (Définition 2.3.1.6)
- $\mathbb{I}f$: extension par intervalles de f (i.e. $\mathbb{I}f = \langle f \rangle_{\mathbb{I}A; \mathbb{I}P}$ si $f : A \rightarrow P$) (Définition 2.3.1.7)
- $\mathbb{I}\mathbb{R}$: les éléments de $\mathbb{R}$ représentable par l'index informatique utilisé
- $\mathbb{i}\mathbb{R}$: l'ensemble des intervalles de $\mathbb{R}$ dont les bornes appartiennent à $\mathbb{I}\mathbb{R} \cup \{-\infty, +\infty\}$
- $\mathbb{X}$: fonction retournant le vecteur des « différences des bornes » d'un élément de $(\mathbb{I}\mathbb{R})^k$
- $\mathcal{F}$: une famille de fonctions
- g : une application associant à chaque fonction, d'une famille, une de ses fonctions d'inclusion
- $[f]_{\mathcal{F}, g}$: une fonction d'inclusion naturelle de f , selon $\mathcal{F}$ et g (Définition 2.3.2.1)
- $\pi(v)$: le vecteur v privé de son dernier élément
- f_v : une application d'une décomposition de f (Définition 2.3.2.1)
- $f_{v\text{hw}}$: la « composition itérative » des applications f_w où $v = \pi(w)$ (Définition 2.3.2.1)
- $f_{v\text{hwn}}$: la « composition n -récursive » des applications f_w où $v = \pi(w)$ (Définition 2.3.2.1)

- $\text{SIVIA}_{f,V,E,\varepsilon}$: la méthode SIVIA, de paramètres f , V , E et ε qui retourne :
 - $E_{f,V,\varepsilon} = E_{f,V,\varepsilon,\subseteq} \cup E_{f,V,\varepsilon,\emptyset} \cup E_{f,V,\varepsilon,\gamma}$: un recouvrement de V par des intervalles tel que :
 - $E_{f,V,\varepsilon,\subseteq} = \{I \in E_{f,V,\varepsilon} \mid f(I) \subseteq V\}$
 - $E_{f,V,\varepsilon,\emptyset} = \{I \in E_{f,V,\varepsilon} \mid f(I) \cap V = \emptyset\}$
 - $E_{f,V,\varepsilon,\gamma} = \{I \in E_{f,V,\varepsilon} \mid I \notin (E_{f,V,\varepsilon,\emptyset} \cup E_{f,V,\varepsilon,\subseteq})\} \subseteq \{I \in E_{f,V,\varepsilon} \mid \omega(I) \leq \varepsilon\}$
- $\text{IVIA}_{f,E,\varepsilon}$ la méthode IVIA, de paramètres f , E et ε qui retourne :
 - $E_{f,\varepsilon,\supseteq}$: Un ensemble d'intervalles tel que $(E \cap f^{-1}(\{0\})) \subseteq \{x \in I \mid I \in E_{f,\varepsilon,\supseteq}\}$
- N_f : un opérateur de Newton associé à la fonction f
- x_j : une application qui retourne un élément d'un ensemble
- $\text{NpI}_{f,E,\varepsilon}$: un algorithme de Newton par Intervalles de paramètres f , E et ε qui retourne :
 - $\mathfrak{R}_{E,f,\varepsilon}$: un ensemble d'intervalles tel que $(E \cap f^{-1}(\{0\})) \subseteq \{x \in I \mid I \in E_{f,\varepsilon,\supseteq}\}$
 - $\mathfrak{R}_{E,f,\varepsilon} \subseteq \mathfrak{R}_{E,f,\varepsilon}$ tel que $(\forall I \in \mathfrak{R}_{E,f,\varepsilon}, \exists! x \in I \text{ tel que } f(x) = 0)$
- $\sim$: équivalence (par difféomorphismes) (Définition 3.1.1.3)
- $\sim_0$: conjugaison (par homéomorphismes) (Définition 3.1.1.4)
- $f|_E$: la restriction de la fonction f à l'espace E (Définition 3.2.1.1)
- $\mathcal{S}_v$: classe de singularités (v caractérise le niveau de singularité) (Définition 3.2.1.2)
- $\mathcal{K}_1$: invariant « différentiel » pour $\sim$ (construit à partir des $\mathcal{S}_v$) (Définition 3.2.2.1)
- $\mathcal{K}_0$: invariant « continu » pour $\sim$ (construit à partir des $\mathcal{S}_v$) (Définition 3.2.2.2)
- δE : bord (topologique) de l'espace E
- $\bar{E}$: adhérence (topologique) de l'espace E
- $\mathcal{V}_E^F(V)$: espace vectoriel engendré par V , dans le F -espace vectoriel E (Définition 4.3.1.1)
- ∇ : opérateur gradient (Définition 4.3.1.3)
- c_f : abstraction combinatoire de la fonction f
- c_E : abstraction combinatoire de l'espace E
- $|c_z|$: réalisation géométrique et topologique de l'abstraction c_z
- $f_{\#}(S_f)$: contour apparent « amélioré » de f (par le nombre de solutions inverses) voir § 5.4.1
- $\mathcal{Z}(f)$: trace singulière de f ($\mathcal{Z}(f) = (f^{-1}(f(S_f)), f(S_f))$) (Définition 5.4.1.1)
- $\mathcal{Z}_n(f) = (\mathcal{Z}_{n_a}(f), \mathcal{Z}_{n_p}(f))$: trace spécifique d'ordre $n \in \mathbb{N} \cup \{-1\}$ (Définition 5.4.1.2)
- $\mathcal{E}_k(f) = (\mathcal{E}_{k_a}(f), \mathcal{E}_{k_p}(f))$: ensemble de spécificités d'ordre $k \in \mathbb{N}$ (Définition 5.4.1.4)
- $p(E)$: une partition de E
- $\mathfrak{s}(f) = (\mathfrak{s}_a(f), \mathfrak{s}_p(f))$: stratification tracée de f (Définition 5.4.1.4)
- $\mathcal{L} : E = \{(z, l_z) \mid z \in E\}$ où $\forall z \in E, l_z$ est un label : fonction de labélisation d'un ensemble
- $\mathcal{L}(\mathfrak{s}(f)) = (\mathcal{L}(\mathfrak{s}_a(f)), \mathcal{L}(\mathfrak{s}_p(f)))$: stratification tracée labélisée de f (Définition 5.4.1.5)
(Cette notation définit implicitement une extension de $\mathcal{L}$ aux vecteurs d'ensembles.)
- s : un simplexe abstrait (Définition 5.4.2.2)
- $\mathcal{Z}$: un ensemble de symboles
- ξ : un symbole
- $\langle\!\langle \mathcal{Z} \rangle\!\rangle$: le simplexe abstrait engendré par $\mathcal{Z}$
- c : un complexe simplicial abstrait (Définition 5.4.2.2)
- $\mathcal{P}(E)$: l'ensemble des parties de E
- $|z|$: réalisation géométrique de l'objet z , voir sous-sections 5.4.2 et 5.7.3
- $\mathcal{Y}(z)$: ensemble des symboles apparaissant dans au moins un élément de z

- $\mathfrak{t}(f) = (\mathfrak{t}_a(f), \mathfrak{t}_p(f))$: une stratification tracée adaptée de f voir § 5.5.3.1
 $(\mathcal{L}(\mathfrak{t}(f)) = (\mathcal{L}(\mathfrak{t}_a(f)), \mathcal{L}(\mathfrak{t}_p(f)))$ est une stratification tracée adaptée labélisée de f)
- $\ell = \ell_{\mathcal{L}(E)}$: associe à chaque élément de $\mathcal{L}(E)$ son label, utilisée § 5.5.3.7
- $\mathfrak{t} = \mathfrak{t}_{\mathcal{L}(E)}$: associe à un label, l'élément de E qui lui est associé dans $\mathcal{L}(E)$, utilisée § 5.5.3.7
- $\#E$: le cardinal mathématique de E (i.e. le « nombre d'éléments » de E)
- $\langle \xi_1, \dots, \xi_n \rangle$: le simplexe abstrait engendré par le n -uplet de symboles $\{\xi_1, \dots, \xi_n\}$
On notera que : $\langle \xi_1, \dots, \xi_n \rangle = \langle \{\xi_1, \dots, \xi_n\} \rangle$
- $c_1 \cup c_2$: complexe simplicial abstrait généré par l'union de c_1 et de c_2
- ϑ : une fonction d'éclatement (pour un éclatement topologique) (Définition 5.7.1.1)
- q : une quadrilatation abstraite (Définition 5.7.3.1)
- $s = \langle z \rangle_q$: simplexe généralisé dans q engendré par z (Définition 5.7.3.1)
- $\langle \xi_1, \xi_2, \xi_3, \xi_4 \rangle = \{\xi_1, \xi_2, \xi_3, \xi_4\} \cup \langle \xi_1, \xi_2 \rangle \cup \langle \xi_2, \xi_3 \rangle \cup \langle \xi_3, \xi_4 \rangle \cup \langle \xi_4, \xi_1 \rangle$
- Pour $f : A \rightarrow \mathcal{P}$ admettant $\{\mathcal{I}_i | i \in I\}$, des courbes telles que $\forall i \in I, \forall z \in \mathcal{I}_i, \{f(z)\} = f(\mathcal{I}_i)$, on définit, voir sous-section 5.7.5 :
 - $\vartheta_f : A \rightarrow \vartheta_f(A)$: fonction d'éclatement telle que :
 - $\forall i \in I, \forall z \in \mathcal{I}_i, \vartheta_f(z) = O_i$, où O_i est un point
 - $\vartheta_f|_{A - \{\mathcal{I}_i | i \in I\}} : A - \{\mathcal{I}_i | i \in I\} \rightarrow \vartheta_f(A) - \{O_i | i \in I\}$ est un homéomorphisme
 - $f_\vartheta : \vartheta_f(A) \rightarrow \mathcal{P}$ définie par $f = f_\vartheta \circ \vartheta_f$

A.4 Reproductions plus uniformisées des algorithmes

Pour faciliter la lecture et la mise en page des algorithmes proposés tout au long de ce mémoire, ceux-ci ont été proposés sous forme d'organigrammes dont la structure est librement adaptée des normes existantes, usuellement utilisées pour représenter les processus, sans ambiguïté.

Afin de rendre, la moins ambiguë possible, la compréhension des algorithmes proposés, ceux-ci sont donc reproduits ici, à une taille réduite, selon une présentation plus conventionnelle et plus uniforme.

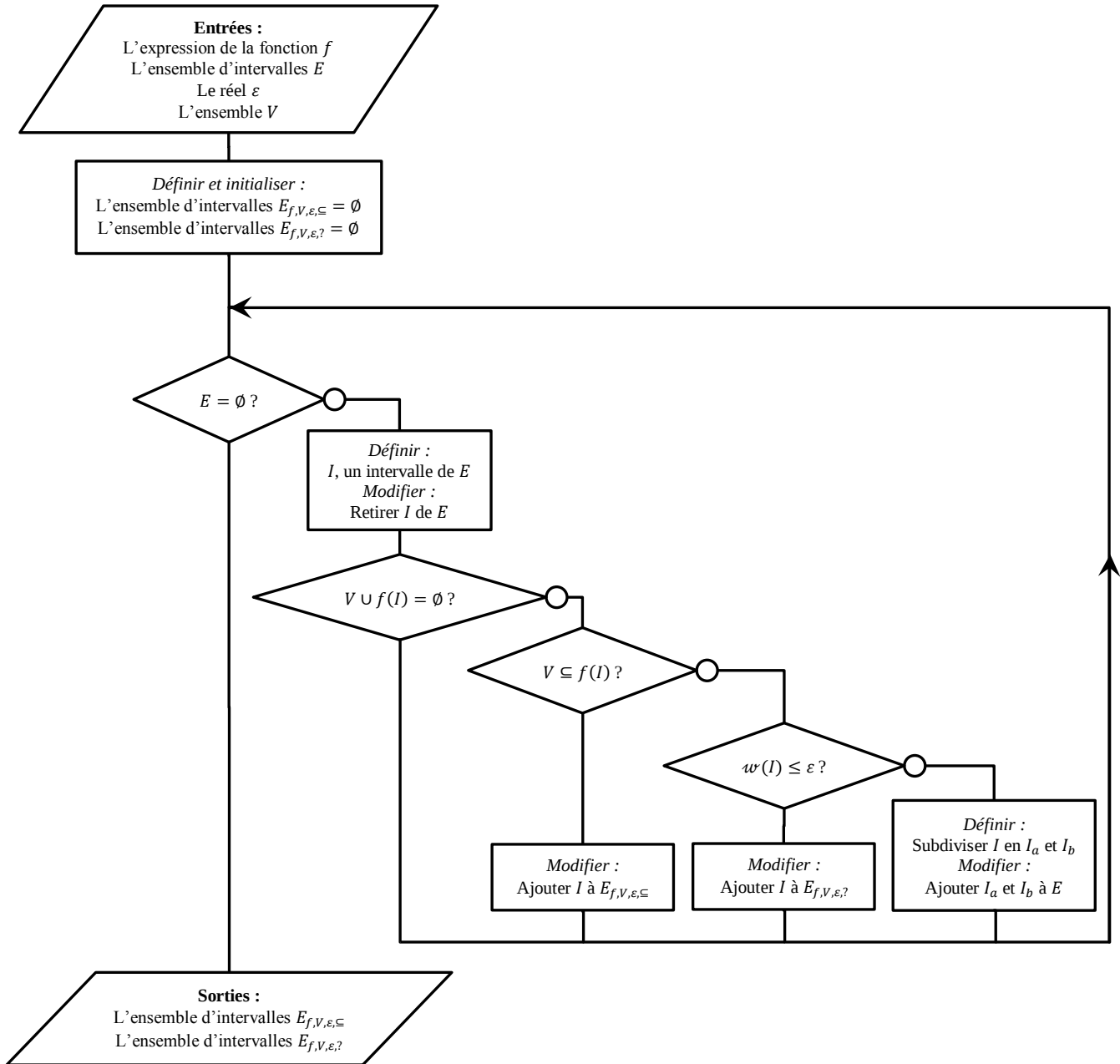


Figure 49 : Présentation plus uniformisée de l'algorithme $SIVIA_{f,V,E,\varepsilon}$ de la Figure 5

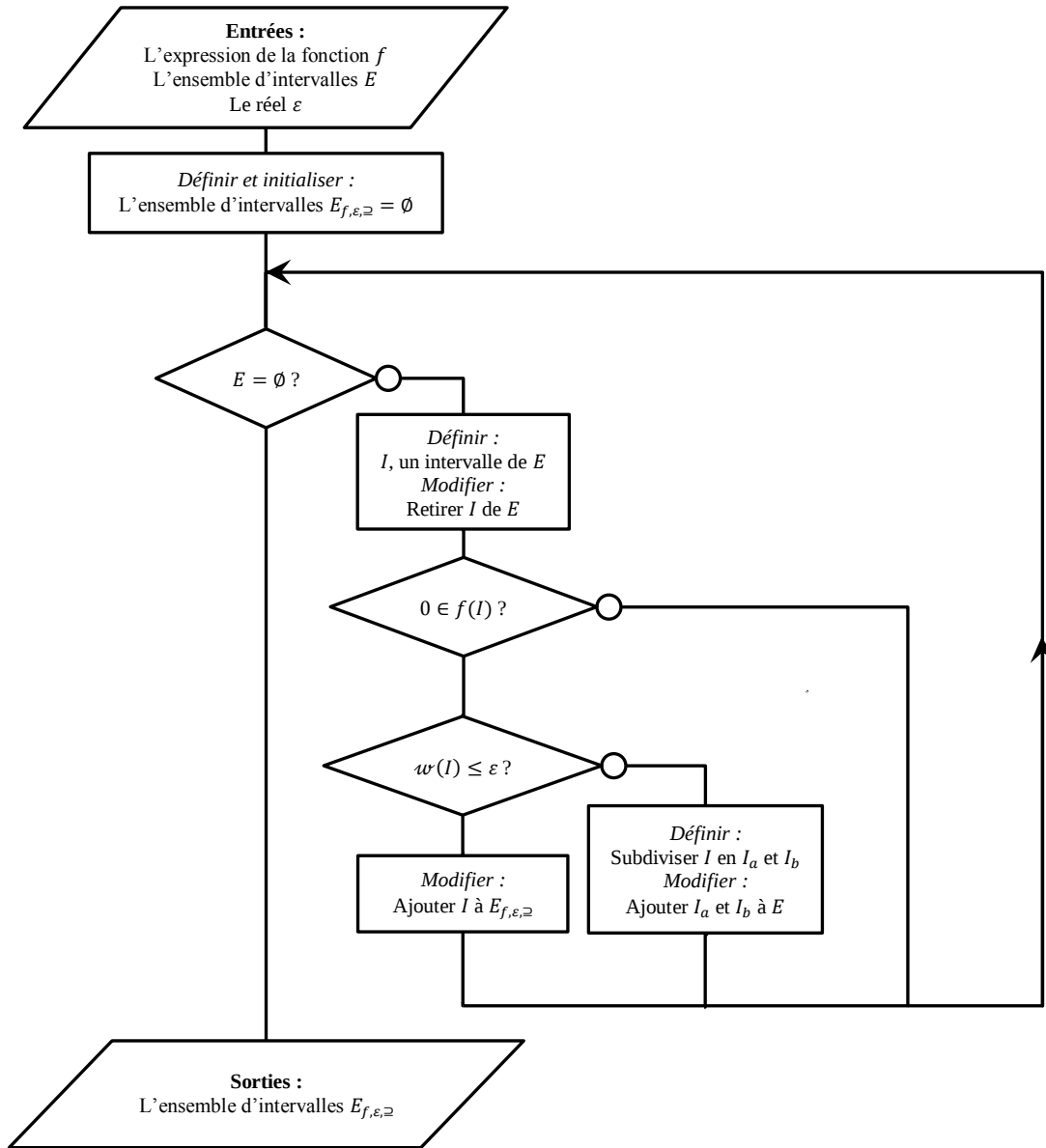


Figure 50 : Présentation plus uniformisée de l'algorithme $\text{IVIA}_{f,E,\varepsilon}$ de la Figure 5

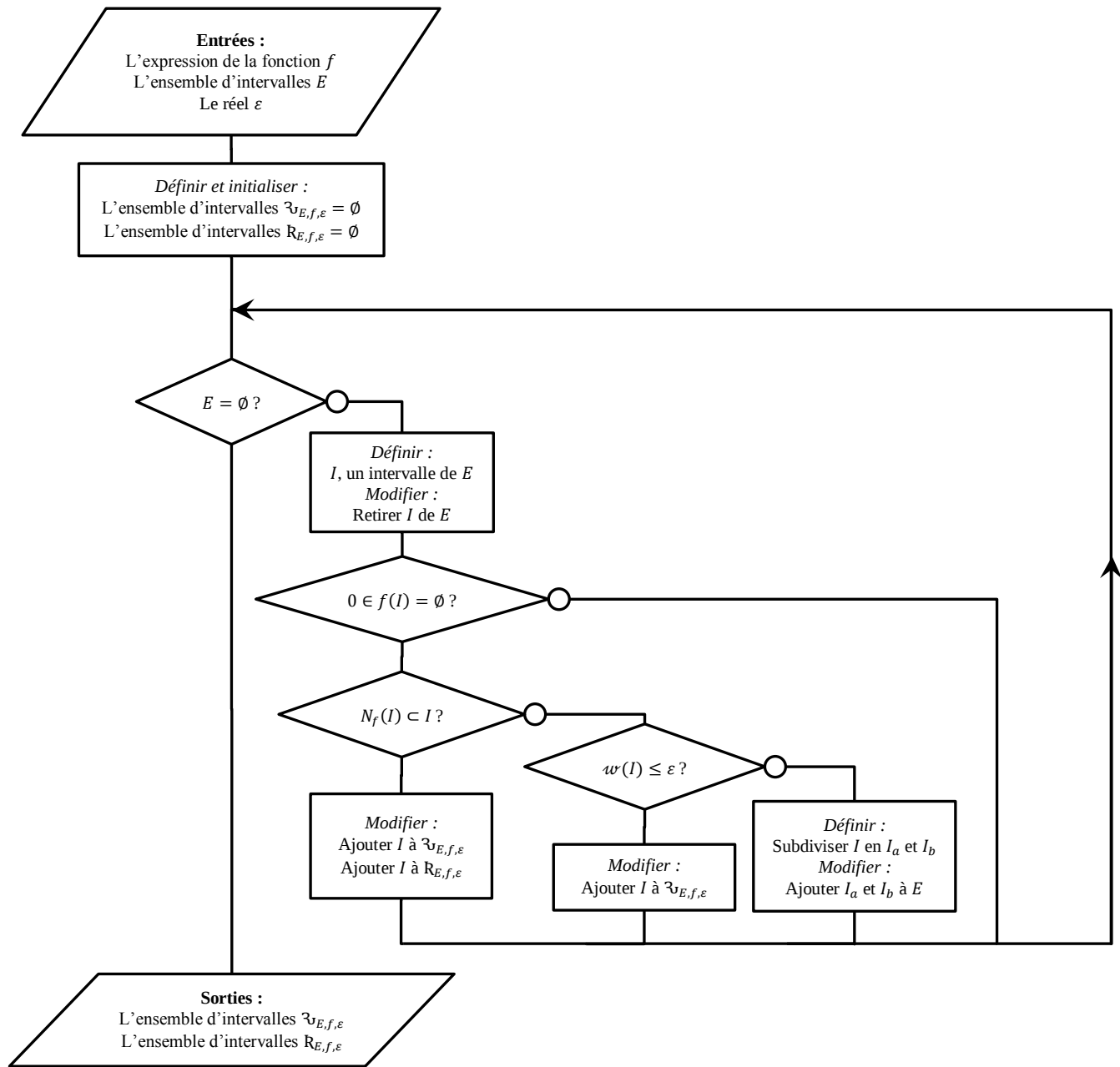


Figure 51 : Présentation plus uniformisée de l'algorithme $\text{NpI}_{f,E,\epsilon}$ de la Figure 6

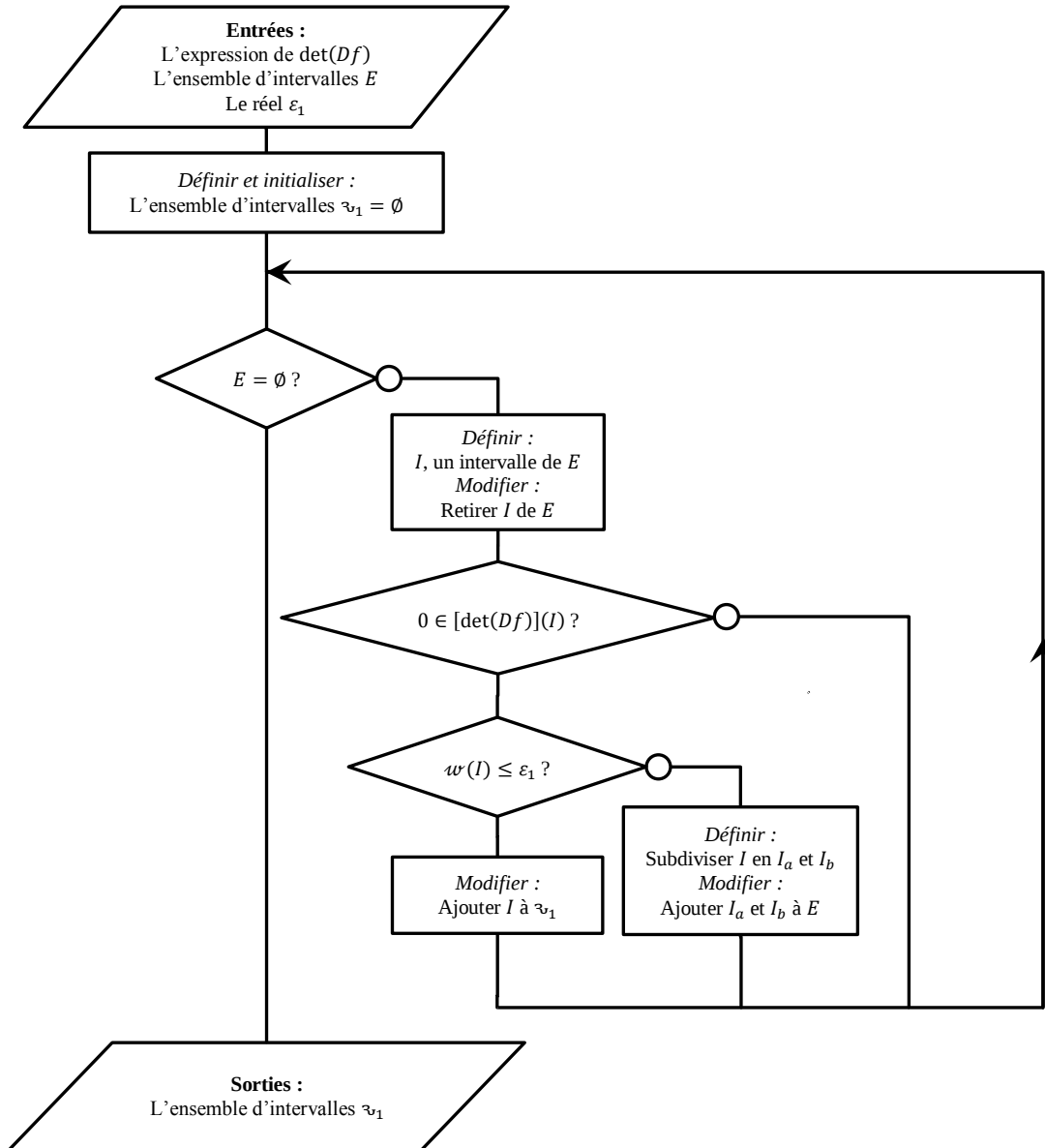


Figure 52 : Présentation plus uniformisée de l'algorithme d'encadrement de S_f de la Figure 15

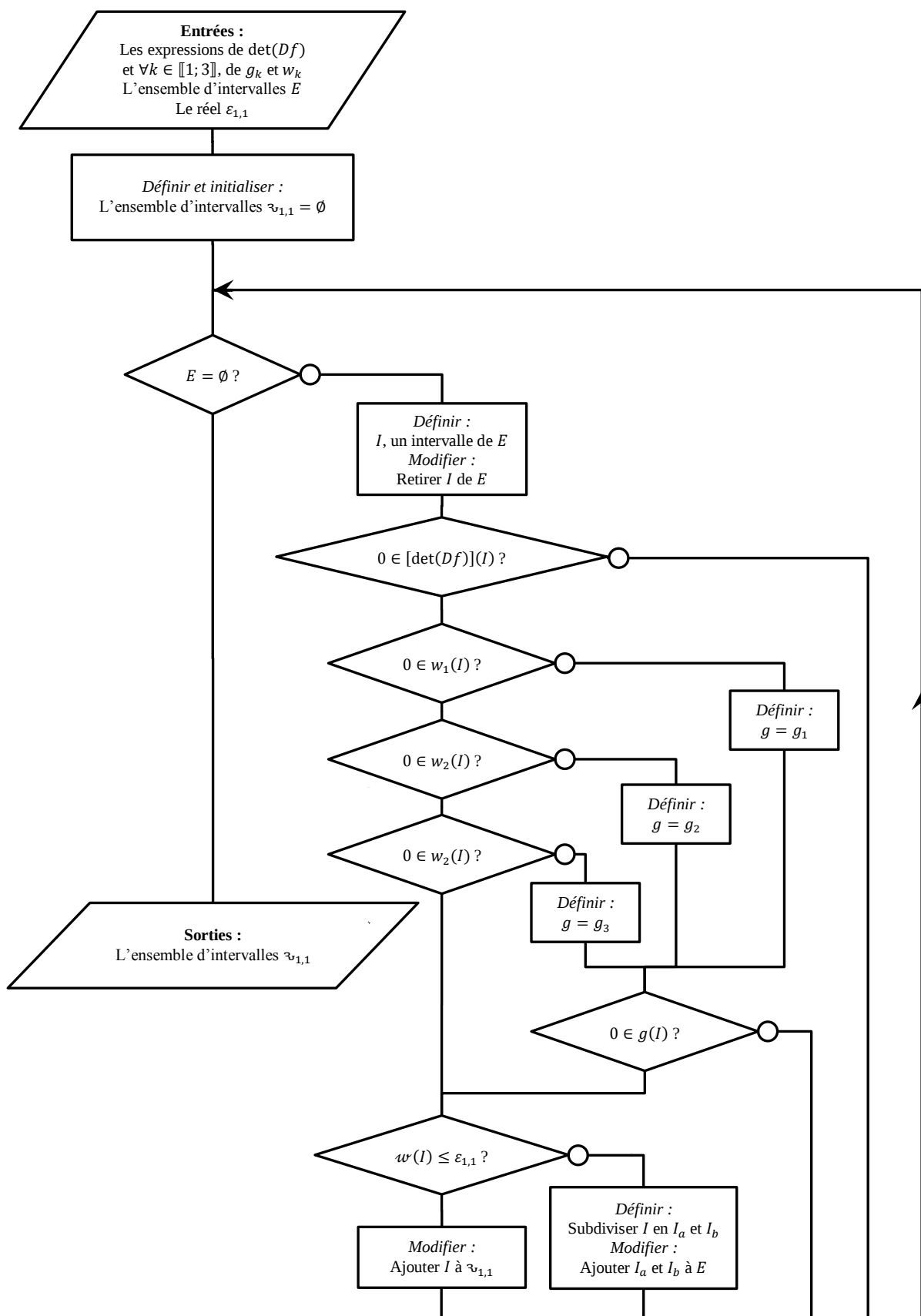


Figure 53 : Présentation plus uniformisée de l'algorithme, encadrant $\mathcal{S}_{1,1}(f)$, de la Figure 16

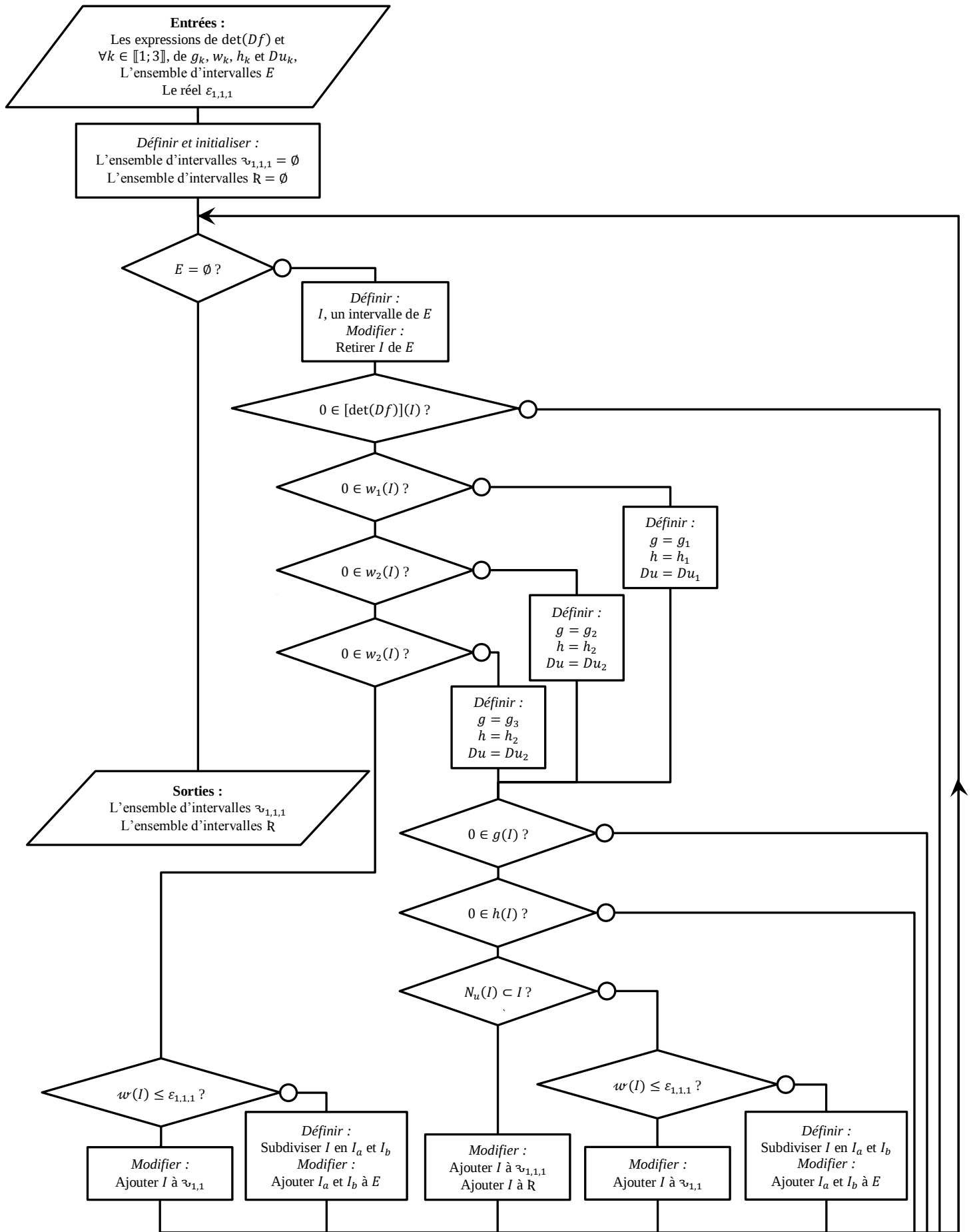


Figure 54 : Présentation plus uniformisée de l'algorithme, encadrant $\mathcal{S}_{1,1,1}(f)$, de la Figure 17

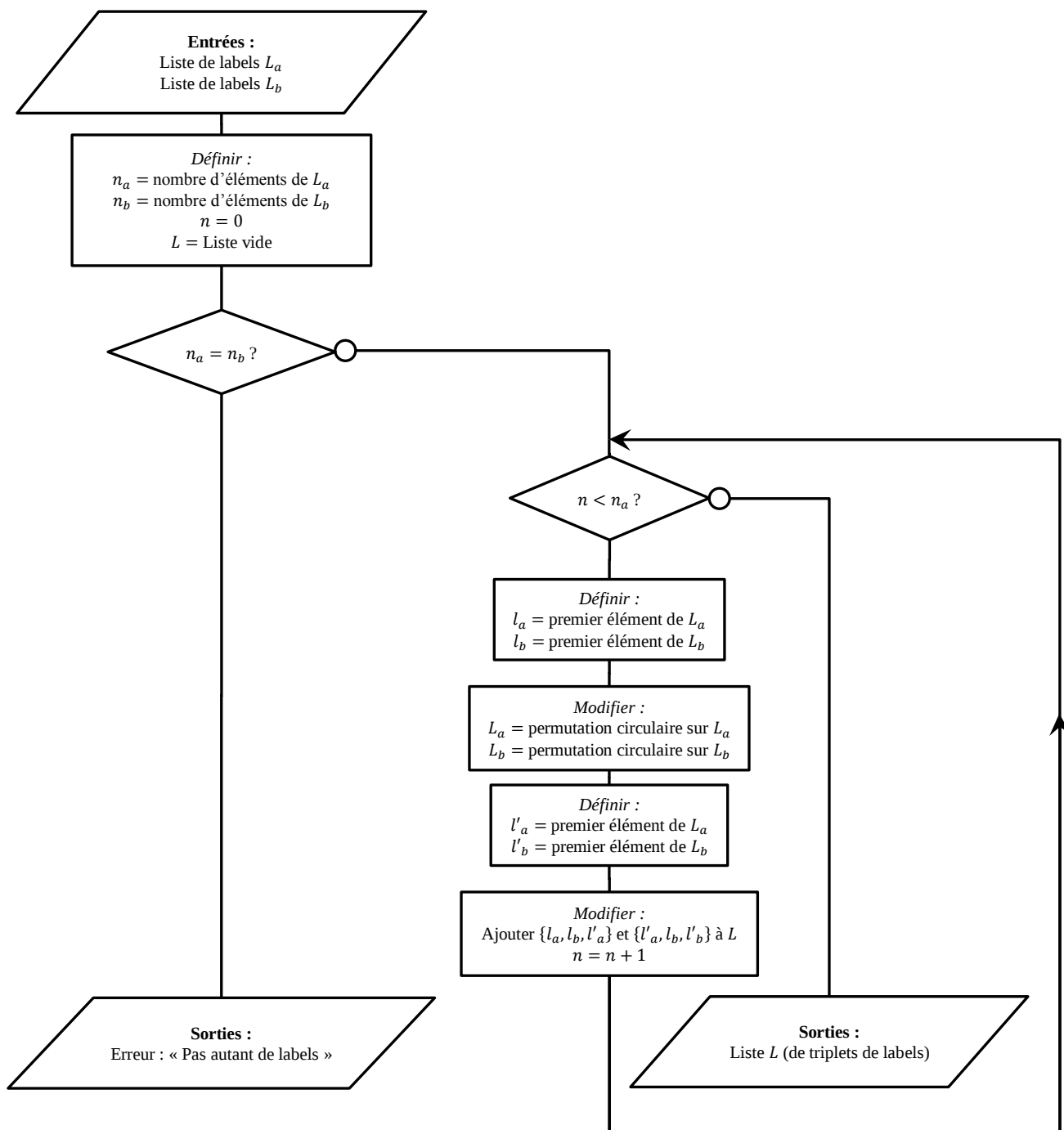


Figure 55 : Présentation plus uniformisée de l'algorithme de la Figure 29

Bibliographie

Arnold V.I. Geometrical methods in the theory of ordinary differential equations [Ouvrage] ; New York : Springer, 1988.

Arnold V.I., Varchenko A.N. et Guisen-Zade S.M. Singularities of Differentiable Maps: Volume I: The Classification of Critical Points Caustics and Wave Fronts [Ouvrage] ; Boston : Birkhäuser, 1985.

Baili M. Analyse et classification de manipulateurs 3R à axes orthogonaux [Rapport] : Thèse de doctorat / Ecole Centrale de Nantes, Université de Nantes ; Nantes : [s.n.], 2004.

Baili M., Wenger P. et Chablat D. A Classification of 3R Orthogonal Manipulators by the Topology of their Workspace [Conférence] : IEEE International Conference On Robotics And Automation ; Barcelone : IEEE, mai 2004 ; pages 1933-1938.

Benoit R., Delanoue N., Lagrange S. et Wenger P. Analyse par Intervalles pour la Détection Garantie de Points Singuliers Spécifiques de Robots [En ligne] ; 23 juin 2014 ; http://www.ibex-lib.org/sites/default/files/workshop/workshop_ctr_geom_2014_06_benoit_slides.pdf ; Contribution au Workshop "Constraints & Geometry".

Benoit R., Delanoue N., Lagrange S. et Wenger P. Guaranteed Detection of the Singularities of 3R Robotic Manipulators [Section] : New Trends in Mechanism and Machine Science / éd. Flores P. et Viadero F. ; [s.l.] : Springer International Publishing, 2015.

Benoit R., Delanoue N., Lagrange S. et Wenger P. Objets combinatoires et invariants de fonctions lisses. Applications à la robotique [En ligne] ; 16 décembre 2014 ; <https://project.inria.fr/singcast/files/2014/12/benoit-romain.pdf> ; Contribution au séminaire "Méthodes de subdivisions pour les systèmes singuliers".

Bohigas O., Manubens M. et Ros L. Singularities of non-redundant manipulators: A short account and a method for their computation in the planar case [Revue] : Mechanism and Machine Theory ; 2013 ; Volume 68 ; pages 1-17.

Bohigas O., Zlatanov D., Ros L., Manubens M. et Porta J.M. Numerical computation of manipulator singularities [Conférence] : IEEE International Conference on Robotics and Automation ; Saint Paul, Minnesota : IEEE, mai 2012 ; pages 1351-1358.

Bop C. Traité de robotique. 1. Les architectures [Ouvrage] ; [s.l.] : Ellipses, 2010.

Burdick J.W. A classification of 3R regional manipulator singularities and geometries [Revue] : Mechanisms and Machine Theory ; 1995 ; Volume 30 ; pages 71-89.

Burdick J.W. Kinematic analysis and design of redundant manipulators [Rapport] : Thèse de doctorat / Université de Stanford ; Stanford : [s.n.], 1988.

Chabert G. Workshop "Constraints & Geometry" [En ligne] : [ibex-lib.org](http://www.ibex-lib.org/workshop-23-06-14) ; 23 juin 2014 ; <http://www.ibex-lib.org/workshop-23-06-14>.

- Corvez S.** Etude de systèmes polynomiaux : contributions à la classification d'une famille de manipulateurs et au calcul des intersections de courbes A - splines [Rapport] : Thèse de doctorat / Université de Rennes 1 ; Rennes : [s.n.], 2005.
- Coste M., Chablat D. et Wenger P.** Nonsingular Change of Assembly Mode Without any Cusp [Section] : Advances in Robot Kinematics / éd. Lenarčič Jadran et Khatib Oussama ; [s.l.] : Springer International Publishing, 2014.
- Delanoue N. et Lagrange S.** A numerical approach to compute the topology of the Apparent Contour of a smooth mapping from $\mathbb{R}^2$ to $\mathbb{R}^2$ [Revue] : Journal of Computational and Applied Mathematics ; December 2014 ; Volume 271 ; pages 267 - 284.
- El Omri J. et Wenger P.** How to recognize simply a nonsingular posture changing 3-DOF manipulator [Conférence] : 7th International Conference on Advanced Robotics ; Sant Feliu de Guíxols : Universidad Politécnica de Cataluña. Servei de Publicacions, septembre 1995 ; pages 215-222.
- Golubitsky M. et Guillemin V.** Stable Mappings and Their Singularities [Ouvrage] ; New York : Springer-Verlag, 1986.
- Hartshorne R.** Algebraic Geometry [Ouvrage] ; New York : Springer-Verlag, 1977.
- Herrmann M., Ikeda S. et Orbanz U.** Equimultiplicity and Blowing up, An Algebraic Study [Ouvrage] ; Berlin Heidelberg : Springer-Verlag, 1988.
- Hu B.** Blow-up Theories for Semilinear Parabolic Equations [Ouvrage] ; Berlin Heidelberg : Springer-Verlag, 2011.
- Imbach R., Moroz G. et Pouget M.** Numeric certified algorithm for the topology of resultant and discriminant curves [Rapport] : Rapport de recherche / Inria ; 2015.
- Jaulin L., Kieffer M., Didrit O. et Walter E.** Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics [Ouvrage] ; Londres : Springer-Verlag, 2001.
- Mather J.N.** How to stratify mappings and jet spaces [Section] : Singularités d'Applications Différentiables / éd. Burlet O. et Ronga F. ; Berlin Heidelberg : Springer-Verlag, 1976.
- Maunder C.R.F.** Algebraic Topology [Ouvrage] ; [s.l.] : Dover Publications, 1996.
- Merlet J.P.** Interval analysis and robotics [Section] : Robotics Research / éd. Kaneko M. et Nakamura Y. ; Berlin Heidelberg : Springer, 2011.
- Merlet J.P.** Les Robots Parallèles [Ouvrage] ; Paris : Hermès, 1997.
- Moore R.E.** Interval Analysis [Ouvrage] ; Englewood Cliffs, New Jersey : Prentice-Hall, 1996.
- Moroz G., Rouiller F., Chablat D. et Wenger P.** On the determination of cusp points of 3-RPR parallel manipulators [Revue] : Mechanism and Machine Theory ; 2010 ; Numéro 11 : Volume 45 ; pages 1555-1567.
- Neumaier A.** Interval methods for systems of equations [Ouvrage] ; [s.l.] : Cambridge University Press, 1990 ; Volume 37 ; Encyclopedia of mathematics and its applications.

Pai D.K. et Leu M.C. Genericity and singularities of robot manipulators [Revue] : IEEE Transactions on Robotics and Automation ; Octobre 1992 ; Numéro 5 : Volume 8 ; pages 545-559.

Parenti C.V et Innocenti C. Position Analysis of Robot Manipulators : Regions and Sub-regions [Conférence] : International Conference on Advances in Robot Kinematics ; Ljubljana : J. Stefan Institute, septembre 1988 ; pages 150-158.

Poujet M. et Moroz G. Méthodes de subdivisions pour les systèmes singuliers [En ligne] : SingCAST ; 16 décembre 2014 ; <https://project.inria.fr/singcast/reunions/nantes-20141214-15-methodes-de-subdivisions-pour-les-systemes-singuliers/>.

Smith D.R. et Lipkin H. Higher order singularities of regional manipulators [Conférence] : IEEE International Conference on Robotics and Automation ; Atlanta : IEEE, mai 1993 ; pages 194-199.

Tsai K.-Y. Admissible motions in manipulator's workspace [Rapport] : Thèse de doctorat / Université du Wisconsin ; Milwaukee : [s.n.], 1990.

Wenger P. Classification of 3R positioning manipulators [Revue] : Journal of mechanical design ; [s.l.] : American Society of Mechanical Engineers, 1998 ; Numéro 2 : Volume 120 ; pages 327-332.

Wenger P. Cuspidal and noncuspidal robot manipulators [Revue] ; Cambridge UK : Cambridge University Press, November 2007 ; Numéro 06 : Volume 25 ; pages 677--689 ; 1469-8668.

Wenger P. Design of cuspidal and noncuspidal manipulators [Conférence] : IEEE International Conference on Robotics and Automation ; Albuquerque : IEEE, avril 1997 ; Volume 3 ; pages 2172-2177.

Wenger P. Uniqueness domains and regions of feasible continuous paths for cuspidal manipulators [Revue] : IEEE Transactions on Robotics ; Aout 2004 ; Volume 20.

Thèse de Doctorat

Romain BENOIT

Analyse qualitative de robots

Qualitative analysis of robots

Résumé

Cette thèse s'inscrit dans la problématique générale de la caractérisation et de la classification des systèmes. Plus précisément, nos contributions s'orientent vers les applications robotiques et, plus particulièrement, vers la classification de manipulateurs. Les algorithmes et méthodologies proposés dans ce document s'appuient sur plusieurs théories mathématiques dont la théorie des singularités et l'Analyse par Intervalles, qui sont formalisés dans la première partie de cette Thèse. Classifier des systèmes induit deux objectifs que sont la formalisation des éléments communs aux systèmes dans la même classe ainsi que leur détection formelle et pratique. Ainsi, ce mémoire compile, pour chacun dans un chapitre dédié, nos contributions vis-à-vis de ces deux objectifs associés à la classification. Une méthodologie générale est proposée, afin de distinguer plusieurs sous-ensembles singuliers caractéristiques et génériquement non vides. Cette méthodologie est illustrée pour des fonctions entre espaces de même dimension, allant de 1 à 3. Chaque application met en lumière des propriétés différentes de la méthodologie générale proposée. Complémentairement, chacune de ces applications mène à la conception d'algorithmes, basé sur l'Analyse par Intervalles, mettant en œuvre cette méthodologie. Complémentairement à la détection de points d'intérêt, nous présentons une méthodologie générale pour définir un objet combinatoire codant le comportement d'une fonction. Cette méthodologie s'appuie sur la détection préalable d'ensembles d'intérêt dont les topologies sont invariantes pour l'équivalence choisie. Ces développements ouvrent vers de nombreuses perspectives futures qui concluent ce mémoire.

Mots Clés :

Analyse par Intervalles, Classification, Singularités, Robotique, Cusps, Nodes

Abstract

This thesis is within the scope of the general problematic of characterizing and classifying systems. More precisely, our contributions are aimed toward robotic applications and, more particularly, at the classification of manipulators. The algorithms and methodologies proposed in this document are based upon several mathematical theories including Singularity Theory and Interval Analysis, which are formalized in the first part of this Phd Thesis report. Classifying systems induces two objectives that are the formalization of the elements commons to systems in the same class as well as their formal and practical detection. Then, this report compiles, for each one in a dedicated chapter, our contributions regarding these two objectives associated with the classification. A general methodology is proposed, with the objective to distinguish several characteristic singular subsets that prove to be generically not empty. This methodology is illustrated for functions between spaces of the same dimension, ranging from 1 to 3. Each application highlights different properties that are admitted by the proposed general methodology. Furthermore, each of these applications leads to the conception of several algorithms, based on Interval Analysis, that enforce the proposed methodology. Complementarily to the detection of interest points, we introduce a general methodology to define a combinatorial object encoding the behavior of a function. This methodology is based on the preliminary detection of interest sets whose topologies are invariant for the chosen equivalence. These developments open up to numerous future perspectives that conclude this report.

Keywords :

Interval Analysis, Classification, Singularities, Robotics, Cusps, Nodes