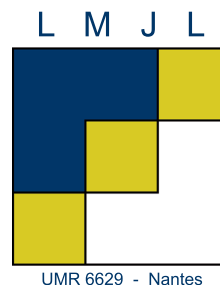




UNIVERSITÉ DE NANTES



UMR 6629 - Nantes

Master MFA

Rapports de stage M2

Parcours

Algèbre - Géométrie

- **Nil Garcés de Marcilla Escubedo** : Introduction à l'Analyse topologique des données et étude de l'algorithme ToMATo
- **Clovis Chabertier** : Dualités de Koszul algébrique, opéradique et propéradique
- **Lucas Darbas** : Cohomologie de Hochschild des schémas
- **Thibault Chailleux** : Décomposition des transformations birationnelles du plan projectif complexe
- **Mahmoud Elimam** : Lagrangian cobordisms and surgery
- **Jean Chartier** : Existence d'un 3-filet géodésique stationnaire sur une 2-sphère



UNIVERSITÉ DE NANTES

MÉMOIRE STAGE

Master 2 : Mathématiques et applications
(Algèbre et géométrie)

Faculté de Sciences - Université de Nantes

Introduction à l'Analyse topologique des données et étude de l'algorithme ToMATo

Auteur : Nil Garcés de Marcilla Escubedo

Tuteur : Bertrand Michel

Écrit à : Laboratoire de Mathématiques Jean Leray

Nantes, 28 juin 2020

Remerciements

Ce mémoire n'aurait pu se terminer sans l'aide des personnes qui m'ont accompagné pendant ces derniers mois si particuliers.

Je voudrais tout d'abord adresser toute ma gratitude à M. Bertrand MICHEL, professeur et chercheur à l'Université de Nantes, qui, en tant que tuteur de mémoire, m'a guidé pendant tout le travail et m'a aidé en tout moment pour continuer à avancer. Grâce toutes nos visioconférences et sa confiance et patience, surtout quand ma motivation était un peu plus faible, j'ai finalement appris plus de choses que je ne le pensais au début de ce travail.

J'aimerais aussi remercier M. Marc GLISSE, chercheur à INRIA-Saclay, qui a été disponible à tout moment pour répondre à mes interrogations sur le code de façon très précise et efficace.

Je désire aussi remercier les professeurs de l'Université de Nantes en général, qui m'ont fourni les outils nécessaires à la réussite de mes études universitaires en France.

Finalement, je voudrais exprimer ma reconnaissance envers ma famille, mes amis et collègues. Ces derniers m'ont apporté leur soutien moral et intellectuel tout au long de ma démarche. Plus spécialement, je remercie les résidents de la cité universitaire et alentours. Durant la période de confinement ils sont devenus comme une partie à part entière de ma famille.

Abstract

The goal of this memoir is to expose and manipulate some modern concepts and tools in the Data Science domain.

In the central part of the work, some basic notions and results of the emerging field of Topological Data Analysis (TDA) are explored, notably persistent homology and persistence diagrams, together with some stability results. Several effective algorithms to compute the homology groups and the persistent homology of a (filtration of a) simplicial complex are also given.

Together with that, following a more general approach, a brief survey of the Machine learning paradigm and some clustering algorithms are exposed in the first two chapters.

In the last chapter, the recently developed clustering method ToMATo is studied. This algorithm relies heavily on some of the concepts explained in the previous chapters. The theoretical study of this method is then followed by a more practical section in which programming takes the leading role : a (rather visual) exploration (in Python) of the implementation of this algorithm in the GUDHI library is carried out, as well as a little guide to understand its parameters and functionalities.

■

Table des matières

0 Introduction	ii
1 Introduction à l'apprentissage automatique	1
1.1 Machine Learning : idée générale et premiers exemples	1
1.2 Motivations	2
1.3 Méthodologie et différents types de systèmes d'apprentissage	3
1.4 Problèmes et challenges du Machine Learning	4
2 Plus en détail : algorithmes de clustering habituels	6
2.1 Considérations générales	6
2.2 Algorithmes de clustering combinatoires	7
2.2.1 K-means clustering	8
2.2.2 Clustering hiérarchique	9
3 Introduction à l'Analyse topologique des données	12
3.1 Idée générale et motivation	12
3.2 Complexes simpliciales, recouvrements et le Théorème du Neri	12
3.3 Inférence homologique	16
3.3.1 Homologie simpliciale et nombres de Betti	17
3.3.2 Filtrations	19
3.3.3 Un algorithme pour calculer les nombres de Betti	21
3.3.4 Homologie persistante : définitions et algorithmes	22
3.3.5 Diagrammes de persistance et stabilité	27
4 L'algorithme ToMATo	32
4.1 Introduction	32
4.1.1 L'intuition derrière l'algorithme : le cas continu	32
4.2 Les données d'entrée (input data)	34
4.2.1 Quelques constructions de graphes habituelles	34
4.2.2 Quelques estimateurs classiques de la fonction de densité	36
4.3 La procédure de l'algorithme	37
4.4 Information finale obtenue	38
4.5 Mise en œuvre de l'algorithme et exploration	40
Références	41
Annexe : A handy guide to using the ToMATo algorithm	41

0. Introduction

L'importance des domaines de la science des données (Data Science) et l'apprentissage automatique (Machine Learning) continue à croître dans le monde avec l'évolution technologique de notre époque. Dans ce contexte, de nouvelles idées et méthodes sont constamment développées pour traiter, analyser et exploiter la grande quantité de données qui nous entourent. Seule une bonne formulation mathématique peut justifier la pertinence des nouveaux algorithmes et son implémentation.

Parmi le grand nombre de méthodes existantes dans la science des données, nous trouvons les méthodes de *clustering*, ou segmentation des données. L'objectif de ces dernières est de trouver des sous-groupes "naturels" dans notre information de départ. Dans chaque groupe, les données seraient, sous une définition à préciser, plus "similaires" entre elles. Les problèmes de clustering sont particulièrement difficiles à traiter en raison de leur nature exploratoire et non-supervisée. Ainsi, la convenance d'un algorithme de clustering ou d'un autre dépend en grande partie des caractéristiques des données d'entrée.

En parallèle à l'apparition de nouvelles techniques pour réaliser des tâches spécifiques, différentes approches générales à la science des données sont aussi développées. Le domaine émergent de l'Analyse topologique des données (TDA en anglais) étudie les ensembles de données en utilisant des idées de la topologie et de la géométrie. Ce domaine illustre pleinement ce phénomène. L'intérêt pour ce champ d'étude, avec plein de nouveaux concepts et résultats, augmente de plus en plus, et actuellement de nombreux algorithmes très efficaces s'appuient sur la base théorique de la TDA.

Le récemment développé algorithme de clustering ToMATo (*Topological Mode Analysis Tool*) fait partie de ces nouvelles méthodes. Au coeur de son fonctionnement apparaissent les notions d'homologie persistante et les diagrammes de persistance, très habituels dans la TDA. De plus, une implémentation de cet algorithme a été récemment ajoutée dans la librairie GUDHI, l'un des outils de programmation de référence dans ce nouveau domaine. Il semble donc raisonnable de faire une exploration théorique mais aussi pratique / informatique de cette technique qui vient d'être conçue.

Structure de la mémoire

Tout d'abord, dans le premier chapitre, nous verrons une exposition générale du domaine de l'apprentissage automatique ; plusieurs concepts de base récurrents sont introduits dans cette partie. Nous essaierons aussi de répondre à trois questions significatives : qu'est-ce que le Machine Learning, pourquoi est-il utile, et comment les ordinateurs arrivent à "apprendre" et à améliorer sa performance. Les références principales de cette partie sont [3], [7] et [9].

Puis, au chapitre 2, nous développerons qu'est-ce que le clustering, ainsi que des notions clés dans l'étude de cette technique. Le contenu de ce chapitre est important afin de mieux comprendre l'algorithme ToMATo, ses innovations et ses particularités. Une partie des algorithmes les plus communs seront expliqués, et nous verrons aussi comment traiter les données pour appliquer au mieux ces méthodes. Les références les plus importantes de cette section sont principalement [9] et la documentation en ligne de la librairie Scikit-Learn [11], [12].

Le troisième chapitre constitue la partie la plus dense et mathématique du mémoire. Certains des concepts les plus fondamentaux de l'Analyse topologique des données y sont exposés. Nous

verrons d'abord comment construire un complexe simplicial sur un nuage de points, et pourquoi cette construction est intéressante. Ensuite, nous ferons l'inférence homologique de ces complexes simpliciaux. Cela implique une connaissance des notions d'homologie persistante ainsi que des diagrammes de persistance. Plusieurs algorithmes sont détaillés pour étudier ces informations dans la pratique. Nous terminerons le chapitre en étudiant la stabilité persistante des fonctions, qui est fondamentale pour justifier de façon théorique la performance de l'algorithme ToMATo. Plusieurs références ont été utilisés dans cette partie, dont : [2], [3], [6], [10] et [5].

Finalement, au chapitre 4 nous explorerons l'algorithme ToMATo. Nous nous appuierons sur les idées exposées aux chapitres précédents. D'un point de vue théorique, il convient d'expliquer certaines constructions de graphes sur des nuages de points, et comment estimer une hypothétique fonction de densité f à partir d'un échantillonnage. Les sources d'informations les plus importantes dans cette section sont [4] et sa version simplifiée, ainsi que [1] et [9].

Pour la pratique, nous avons produit un notebook de référence (en anglais) de l'implémentation de l'algorithme, qui vient d'être ajouté à la librairie GUDHI. L'objectif de cette partie était de tester la performance de cette implémentation, ainsi que mieux connaître le langage de programmation Python et certains outils habituels pour réaliser l'analyse de données.

1. Introduction à l'apprentissage automatique

1.1. Machine Learning : idée générale et premiers exemples

Même si le concept de *Machine Learning* (en français, *apprentissage statistique*, ou *apprentissage automatique*) a explosé en popularité ces dernières années, souvent on perçoit un peu de confusion en ce qui concerne la signification authentique de ce terme. Cette confusion est davantage aggravée lorsque d'autres expressions comme *data science*, *artificial intelligence* ou *data mining*, étroitement liées à la première, apparaissent sur la table. Inévitablement, avec le développement intensif de tous ces nouveaux domaines, un nouveau jargon est apparu, et il est indispensable de bien connaître les subtiles différences entre les mots pour encadrer convenablement les problèmes et les explications.

En termes généraux, le Machine Learning (ML) est le domaine d'étude qui cherche à donner aux ordinateurs la capacité d'apprendre sans être explicitement programmés, en utilisant des données (d'ici son interaction avec la *data science*) et/ ou des expériences antérieures. En voyant cette définition, qui d'ailleurs n'est pas très concrète, deux questions émergent naturellement par rapport au terme "apprendre" : qu'est-ce que cela veut dire, exactement, et comment obtient-on cet apprentissage ? De plus, il est naturel de se demander dans quelles situations ou pourquoi le Machine Learning peut être la meilleure option à considérer. Ce sont précisément ces trois questions que nous nous proposons de répondre tout de suite.

La première des trois est possiblement la plus générale : en effet, cette apprentissage peut prendre plusieurs formes, qui peuvent varier énormément en fonction du problème de départ. Ainsi, la manière la plus rapide de se faire une idée de quoi "apprendre" signifie véritablement est de regarder quelques exemples de situations où le Machine Learning s'est avéré être très efficace. Ces exemples vont apparaître plusieurs fois toute au long du chapitre :

- La classification du mail dans *spam* et *no-spam*. Dans ce cas, l'idée est de développer un algorithme pour choisir, en considérant plusieurs aspects (fréquence de quelques mots spécifiques, longueur, structure générale,...), si un courriel contient des informations qui nous intéressent ou pas. Donc, en somme, nous voulons que l'ordinateur apprenne à *classer* une série d'éléments.
- La prédiction de la valeur d'une maison, en sachant quelques aspects comme sa taille, emplacement et d'autres caractéristiques, ainsi comme celles des immeubles à proximité, y compris leur valeur. Dans cet exemple, on assume que tous ces facteurs peuvent être utilisés pour construire un modèle "réaliste" qui donne notre prix approximatif. Le résultat final du processus est une quantité, qui peut donc varier continuellement. Nous avons ici un problème typique de *régression*.
- Dans un magasin, on peut essayer de détecter des groupes de clients similaires selon leurs achats, ou selon leur genre, par exemple. En sachant cela, on peut élaborer des offres ou politiques commerciales plus dirigées vers ces groupes pour augmenter les ventes. Ici, nous avons de nouveau un problème de classification, mais d'une nature assez différente, car les groupes ne sont pas connus a priori, et ils pourraient même ne pas exister d'une façon évidente. Nous parlerons plus de ce type de procédure, appelé *clustering*, en peu plus tard.
- Le développement d'une application digitale de reconnaissance vocale. Par exemple, un programme de smartphone capable d'écrire et chercher sur Internet toute combinaison de mots

qu'on lui dicte. Dans ce cas, il y a aussi de quelque sorte un problème de classification (après tout, l'objectif du programme est de bien identifier chaque mot prononcé), mais les nuances du langage et la complexité de la prononciation humaine situent le défi beaucoup plus loin que d'autres problèmes de classification standards. Ce type de programmes qui cherchent à imiter (et dépasser) le comportement humain et réaliser des tâches plus complexes font partie de ce qu'on appelle *intelligence artificielle*. Ce domaine, de plus en plus actif et prometteur, a ses propres algorithmes et mécanismes, comme par exemple l'utilisation de réseaux neuronaux (neural networks) ou l'apprentissage par renforcement (reinforcement learning).

Donc, nous voyons que la variété de ce qu'on a appelé "apprentissage" est riche et considérable et, en fait, il y a beaucoup plus de situations et applications possibles : diagnostic médical guidé par ordinateur, séquençage d'ADN, vision par ordinateur,...

En tout cas, la plupart des méthodes et algorithmes ont pour objectif de classer des éléments, de prévoir ou d'estimer des résultats ou des valeurs pour prendre des décisions, trouver des relations entre variables, ou une combinaison de ces options.

1.2. Motivations

Mais pourquoi appeler tout cela apprentissage ? Après tout, tous les programmes informatiques visent la simplification des tâches et aider avec les calculs et la prise de décisions...

La différence essentielle avec le Machine Learning est la manière dans laquelle ces programmes arrivent à effectuer ces tâches : rappelons qu'un élément clé de notre brève définition est "sans être explicitement programmés, en utilisant des données et/ ou expériences antérieures". Avant d'expliquer, dans la section suivante, les idées générales qui présentent comment nous pouvons arriver à faire cela, il est naturel de se demander en premier lieu quelles sont les motivations de le faire.

Prenons-nous le premier exemple du mail. Si nous devons programmer nous-mêmes un détecteur de mail *spam* (pour bien le distinguer et séparer du mail "bon"), la manière la plus naturelle d'agir serait, d'abord, d'étudier un peu ses caractéristiques générales : quels types de mots ou d'expressions apparaissent le plus souvent dans ce type de courriels et ses fréquences en comparaison avec le mail ordinaire, sa longueur approximative, des régularités dans le nom ou dans l'adresse de l'émetteur, etc. Finalement, avec toute cette information, il faudrait programmer une par une les conditions ou les seuils à dépasser pour le considérer comme un courrier indésirable.

Ce n'est pas une chose facile ni rapide à faire ! Même si nous réussissons à trouver de bonnes conditions pour distinguer les deux types de mail, nous obtiendrions une liste énorme de règles à considérer. Ainsi, le résultat final serait un code très long et complexe : pas pratique à programmer ni facile à maintenir, modifier ou mettre à jour. Un algorithme plus "machine learning" chercherait lui-même les caractéristiques clés en comparant des exemples des deux types de courrier et associerait les poids convenables pour bien les classer.

Dans le dernier exemple de la reconnaissance vocale, la complexité d'un hypothétique programme codé à la main devient encore plus évidente : la quantité d'information et la variabilité dans un fichier audio est tellement énorme qu'il est simplement impossible d'analyser explicitement tous les cas où il sonne une "s" ou une "u". Seulement après avoir exposé à un bon algorithme milliers d'enregistrements des différents mots, nous pouvons espérer qu'il arrivera à les distinguer correctement.

Un autre avantage des algorithmes de Machine Learning est que souvent on peut les concevoir pour qu'ils soient adaptables à de possibles actualisations ou accroissements des données. Cette caractéristique peut être très utile dans toutes les situations, y comprises celles des exemples antérieurs.

Finalement, nous pouvons nous servir de toutes ces techniques et procédures pour améliorer l'apprentissage humaine même. En effet, quelques algorithmes ML peuvent être inspectés pour voir ce qu'ils ont appris, et ainsi mieux comprendre des corrélations et tendances non reconnues précédemment. Utiliser les techniques ML avec cet objectif s'appelle *data mining*.

1.3. Méthodologie et différents types de systèmes d'apprentissage

Alors, comment obtenir cet apprentissage ? Le principe de tous les systèmes de Machine Learning consiste dans le fait que la majorité des paramètres sont trouvés en utilisant des données et des exemples déjà existants, qu'on appelle "données d'entraînement" (*training data*). En tout cas, c'est pratique de classer ces systèmes de plusieurs manières en considérant quelques-unes de ses différences méthodologiques fondamentales.

Possiblement la distinction la plus importante au niveau méthodologique, car il affecte notamment les possibles algorithmes à appliquer, est celle d'apprentissage *supervisé* et *non-supervisé* (et quelques types "intermédiaires"). Cette classification prend en compte dans quelle mesure les données sur lesquelles on construit l'algorithme contiennent déjà des informations certaines, i.e. on a une connaissance préalable des types de solutions qu'on devrait obtenir.

Dans l'apprentissage supervisé, possiblement le plus naturel et intuitif, les données d'entraînement incluent les solutions souhaitées ; elles sont "étiquetées" (*labelled*). Par exemple, dans les deux premières situations expliquées précédemment, nous construirions le classificateur de mail à partir d'exemples de courriels "bons" et "spam" ; pareillement, on estimerait le prix de la maison en utilisant un modèle qui prend en compte les caractéristiques, mais aussi les prix (i.e. la "solution", ils sont donc étiquetés) des différentes maisons à proximité. Ces caractéristiques utilisées pour construire le modèle s'appellent *features*, ou *predictors*.

En résumé, les systèmes d'apprentissage supervisé sont conçus pour donner les résultats attendus sur les données d'entraînement, que nous connaissons. Les problèmes de régression et de classification en groupes spécifiques sont des exemples de ce type d'apprentissage.

Dans l'apprentissage non-supervisé, les données sur lesquelles nous travaillons ne sont pas étiquetées, et il n'y a pas une façon directe de vérifier ou mesurer la performance du système. Ce type d'apprentissage est plutôt lié à la visualisation des données et son exploration : corrélations inattendues, groupes avec des similitudes, détection des données mauvaises ou bizarres (*outliers*),... Par exemple, les méthodes de clustering sont de nature non-supervisée, y compris notre algorithme ToMATo, dont nous parlerons plus tard. Dans ce type d'apprentissage il y aurait aussi ces algorithmes de visualisation qui essaient de représenter les données en 2D et 3D en préservant au maximum sa structure. Finalement, nous y ajouterions aussi toutes les procédures de réduction de la dimensionnalité, qui ont pour objectif simplifier les données sans perdre trop d'information, par exemple en combinant plusieurs features corrélées entre elles.

D'autres types d'apprentissage sous ce critère seraient l'apprentissage semisupervisé, qui combine les deux types antérieurs, ou l'apprentissage par renforcement. Dans ce dernier, assez lié au

domaine de l'intelligence artificielle, l'algorithme observe continuellement les données et l'environnement, et sélectionne et réalise des actions qui peuvent être récompensées ou pénalisées ; au fil du temps, il apprend lui-même les stratégies le plus efficaces pour obtenir les meilleures récompenses.

Une autre manière de classer les systèmes ML est selon sa capacité d'adaptation aux nouvelles données. Les algorithmes qui ont besoin de tout l'ensemble de données pour être construits correctement font partie de ce qu'on appelle apprentissage *offline* ; ceux qui peuvent incorporer de nouvelles données et apprendre progressivement, une propriété en général désirable pour sa flexibilité et réduction du coût de calcul, sont de type *online*.

Finalement, une autre classification décisive au niveau méthodologique est celle qui prend en compte comment le système ML se généralise aux nouveaux cas ; c'est-à-dire de quelle façon on mesure sa performance en tant que prédicteur, avec de nouvelles observations.

Dans l'apprentissage *basé sur des instances*, l'algorithme apprend les exemples par coeur et étudie les nouveaux cas en utilisant une "mesure de similitude". Cette dernière compare quantitativement les nouveaux cas avec les données d'entraînement, afin de les étudier. En revanche, dans l'apprentissage *basé sur des modèles*, on essaie de construire un bon modèle ou formule à partir des exemples pour faire des prédictions. Normalement, dans le design de ce modèle, on utilise une fonction d'"aptitude" (*fitness function*, ou *cost function*) pour étudier quantitativement sa convenance sur les données d'entraînement.

Dans les deux cas, il faut toujours garder à l'esprit que tout ensemble de données d'entrée contient inévitablement du *bruit* : elles sont partiellement aléatoires, et l'information n'est jamais transparent. Donc, ajuster la flexibilité du modèle en fonction de chaque cas est toujours essentiel.

1.4. Problèmes et challenges du Machine Learning

En somme, dans tout processus d'apprentissage statistique nous trouvons deux étapes : la sélection d'un algorithme convenable et l'entraînement postérieur avec des données. Naturellement, il faut faire attention à ces deux choses si nous voulons obtenir un apprentissage effectif. Certains défis ou aspects à prendre en compte en ce qui concerne cela seraient :

- *Quantité insuffisante de données* : Dans la majorité des algorithmes, il faut disposer de beaucoup de données pour entraîner correctement le modèle et le faire fonctionner. En général, on a besoin de milliers d'exemples, ou des millions dans les problèmes les plus complexes. Dans certaines situations, il est possible de combiner ou extraire des nouvelles données à partir de celles déjà existantes, pour en avoir plus. Plusieurs études montrent que des algorithmes très différents peuvent accomplir des niveaux de succès similaires en utilisant suffisamment de données.
- *Données d'entraînement non représentatives* : Afin d'obtenir de bonnes généralisations, les données d'entraînement doivent être représentatives des nouveaux cas qu'on cherche à généraliser ; sinon, les prédictions du modèle difficilement s'ajusteront aux valeurs réelles. Par exemple, le caractère d'un modèle pour calculer quelque spécificité d'un pays peut changer largement en fonction de la richesse des pays utilisés pour le concevoir ; il faudrait se servir des pays avec un niveau économique similaire. Le même principe s'applique pour prédire les résultats d'une élection à partir des sondages. Quand les données utilisées ne sont pas représentatives, même si nous en avons une grande quantité, il s'agirait ici d'un "biais d'échantillonnage" (*sampling bias*).

- *Données de mauvaise qualité* : Naturellement, si les données d'entraînement contiennent beaucoup d'erreurs, outliers et bruit, les algorithmes auront plus de problèmes pour trouver des "patterns" et atteindre ses objectifs. Donc, en général, c'est recommandable d'investir du temps à détecter et écarter les outliers et traiter les valeurs manquantes ou incomplètes (*data cleaning*).
- *Features non pertinents* : Indépendamment de l'algorithme, celui-ci seulement apprendra si les données utilisées pendant l'étape d'entraînement ont un véritable lien avec ce que nous voulons estimer. Le complexe processus d'obtenir un ensemble de features pertinents s'appelle *feature engineering*. Il comprend, entre autres : sélectionner les features les plus utiles et écarter les autres, les combiner pour en obtenir des nouvelles d'une façon plus compacte (étroitement lié à la réduction de la dimensionnalité), en créer d'autres à partir de nouvelles données,...
- *Overfitting et underfitting* : Ces deux phénomènes, plus liés à l'algorithme lui-même qu'à la nature des données, se produisent quand le modèle obtenu se base trop ou respectivement trop peu sur les données d'entraînement.

Tout algorithme de Machine Learning essaie de trouver des régularités dans les données, mais celles-ci possèdent aussi de manière naturelle une variabilité qui peut empêcher l'algorithme de bien se généraliser à de nouveaux cas si nous nous y basons trop. Par exemple, il est presque toujours possible de trouver une fonction polynomiale qui passe pour n'importe quelle quantité de points dans \mathbb{R}^2 si son degré est suffisamment élevé (i.e. si nous augmentons suffisamment les degrés de liberté), mais un modèle si "courbé" ne sera possiblement pas le meilleur à prédire de futures observations. En somme, l'overfitting se produit quand le modèle est trop complexe par rapport au bruit et à la quantité de données d'entraînement.

Quelques possibles solutions dans ce cas seraient : recueillir plus de données, réduire le bruit des données (i.e. réparer les erreurs dans les données et écarter les outliers) ou simplifier le modèle, chose que nous pouvons faire en utilisant moins de paramètres, en considérant moins de features ou en "contraignant" le modèle. Ce dernier approche, appelé *regularization*, contient plein de méthodes et techniques : l'idée de base est d'utiliser des paramètres supplémentaires dans l'algorithme (les *hyperparamètres*), indépendants du modèle, fixés d'abord et constants pendant l'entraînement, qui "limitent" en quelque sorte la liberté des paramètres du modèle. Trouver de bons hyperparamètres est l'une des parties importantes de construire un bon système de Machine Learning.

Le underfitting est le problème contraire : il se produit quand le modèle est trop simple pour bien apprendre la structure sous-jacente des données. Trois stratégies pour améliorer rapidement cette situation sont : admettre plus de paramètres dans le modèle, réduire les contraintes s'il y en a, ou augmenter la pertinence des features.

- *Essai et validation* : Pour étudier l'efficacité du modèle, une bonne pratique consiste à diviser les données disponibles en plusieurs sous-groupes complémentaires et les entraîner, mesurer et vérifier les uns contre les autres. En somme, nous trouvons trois types de ces groupes : les données d'entraînement (*training set*), à partir duquel on construit le modèle ou mesure de similitude ; les données de validation (*validation set*), qui servent pour modifier le modèle ou les hyperparamètres jusqu'à obtenir une performance désirable ; et les données de vérification (*test set*), pour se faire une idée de l'erreur de généralisation (i.e. sa performance avec de nouveaux cas).

On appelle *cross-validation* la méthode, très commune à pratiquer, qui consiste à faire cette procédure plusieurs fois avec tout l'ensemble des données pour mieux choisir le modèle et les hyperparamètres.

2. Plus en détail : algorithmes de clustering habituels

2.1. Considérations générales

Le *cluster analysis*, ou segmentation des données, a pour objectif le regroupement d'un ensemble d'éléments en sous-groupes ou *clusters*. Ainsi, dans chaque cluster, les éléments sont plus "proches" entre eux à la différence des éléments classés dans des clusters différents. Dans cette branche du Machine Learning, à caractère non supervisé et exploratoire, les algorithmes cherchent à établir si les données peuvent être divisées dans des groupes différents avec des propriétés suffisamment distinctes. L'algorithme ToMATo, récemment développé, fait aussi partie de ces méthodes. Nous aborderons plus en détail cet algorithme dans le chapitre quatre.

La question fondamentale dans le cluster analysis est comment nous mesurons ce "degré de similarité" (ou dissimilarité) entre les données, donc c'est la définition sur laquelle les algorithmes se basent.

Une approche assez flexible consiste à utiliser ce qu'on appelle une *matrice de proximité*. Avec un ensemble de N éléments (ordonnées), $\{x_1, \dots, x_N\}$, on construit une matrice D de dimension $N \times N$, où le coefficient d_{ij} mesure quantitativement la proximité ou similarité de l'élément i à l'élément j . En général, plus le numéro est faible, plus des similitudes sont remarquées. De ce fait, la plupart des algorithmes assument $d_{ii} = 0, \forall i \in [1, N]$. De plus, certains algorithmes imposent notamment que la matrice soit symétrique ; sinon, D peut toujours être remplacée par $(D + D^T)/2$. Pour travailler avec "dissimilarités", on peut toujours convertir tous les valeurs avec une fonction monotone décroissante convenable.

Une des situations la plus habituelles est celle où chaque élément x_i consiste en p attributs de nature quantitative. Si ces attributs sont de nature qualitative (ou catégorique), on peut parfois les convertir facilement en numéros : par exemple, si nous avons une variable qualitative ordonnée avec M options, nous pouvons utiliser les valeurs $\frac{i - \frac{1}{2}}{M}, i = 1, \dots, M$, toutes entre 0 et 1. Si la variable n'est pas ordonnée et peut prendre M différentes valeurs, il faut préciser le "niveau de différence" entre les paires de valeurs en utilisant une matrice (comme évoqué dans le paragraphe précédent) : ses entrées, normalement 1s sauf 0s à la diagonale, jouent le rôle des d_j que nous expliquerons tout de suite.

Supposons que nous disposons de plusieurs données numériques $x_{ij}, i \in [1, N], j \in [1, p]$ (les cas catégoriques ont déjà été traités). En s'appuyant sur ces données, on construit une notion de "dissimilarité" entre les valeurs du j -ème attribut de deux éléments différents, $d_j(x_{ij}, x_{i'j})$. Le choix le plus commun pour d_j est la distance au carré,

$$d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2.$$

D'autres options existent aussi, comme par exemple la différence absolue $|x_{ij} - x_{i'j}|$, qui pénalise moins les grandes différences. Les résultats peuvent varier considérablement en fonction de la distance choisie.

Puis, nous définissons la "mesure de dissimilarité totale" $d(x_i, x_{i'})$ entre deux éléments en combinant ces p dissimilarités individuelles. Bien que la somme est l'option la plus naturelle à considérer, nous gagnons en flexibilité en travaillant avec une moyenne pondérée

$$d(x_i, x_{i'}) = \sum_{j=1}^p w_j \cdot d_j(x_{ij}, x_{i'j}), \quad \sum_{j=1}^p w_j = 1.$$

Cette dernière permet d'ajuster un poids convenable à chaque attribut (ces poids dépendent nécessairement de la nature du problème et des données concrètes). Pour bien adapter ces poids, il est important de remarquer que l'influence du j -ème attribut sur la dissimilarité totale $D(x_i, x_{i'})$ dépend de sa contribution relative à la moyenne des dissimilarités totales entre toutes les paires d'éléments de l'ensemble,

$$\bar{D} = \frac{1}{N^2} \sum_{i=1}^N \sum_{i'=1}^N d(x_i, x_{i'}) = \sum_{j=1}^p w_j \cdot \bar{d}_j,$$

où

$$\bar{d}_j = \frac{1}{N^2} \sum_{i=1}^N \sum_{i'=1}^N d_j(x_{ij}, x_{i'j})$$

est la dissimilarité moyenne du j -ème attribut. Ainsi, l'influence relative de la j -ème variable est $w_j \cdot \bar{d}_j$, et fixer $w_j \sim 1/\bar{d}_j$ (ou directement $w_j = 1/\bar{d}_j$, standardisé plus tard) donne à chaque attribut la même influence sur la dissimilarité totale.

Bien que cette dernière option semble appropriée, elle peut aussi être contre-productive. En effet, souvent les attributs ne contribuent pas de la même manière à la notion de similitude : certaines différences entre les valeurs peuvent refléter plus de dissimilarité que d'autres dans le contexte du problème, et devraient donc avoir plus de poids. Pour cela, il est important de préciser correctement toutes ces variables, ainsi que la fonction de similitude, chose qui dépend dans une large mesure du problème spécifique. En fait, tous ces paramètres peuvent avoir plus d'importance que l'algorithme lui-même pour réussir avec le clustering.

Finalement, il est aussi important de prêter attention à bien traiter les données manquantes (*missing values* en anglais) dans un ou plus des attributs. On peut faire cela en utilisant une moyenne (ou quelque autre valeur, processus appelé "imputation statistique"), en utilisant une nouvelle catégorie qualitative "missing", en omettant quelques dissimilarités concrètes ou en écartant directement ces éléments.

2.2. Algorithmes de clustering combinatoires

Pour résumer, nous trouvons trois types d'algorithmes de clustering :

- Les *algorithmes combinatoires* travaillent directement sur les données, sans avoir aucun type de modèle probabiliste sous-jacent, et assignent directement chaque élément à un group.
- Les *modèles de mélange* supposent que les données constituent un échantillon *i.i.d* d'une population décrite par une fonction de densité. Cette fonction de densité est caractérisée par un modèle paramétrique formé par un mélange/ somme de plusieurs fonctions de densité (habituellement gaussiennes) : chacune de ses fonctions décrirait un cluster.
- Les algorithmes *mode-seeking* ("chercheurs de modes"), aussi appelés *bump hunters*, ont une approche non paramétrique et tentent d'estimer directement les différentes modes (i.e. maximums locaux) d'une hypothétique fonction de densité de base. Les éléments les plus proches de chaque mode définissent ainsi les clusters individuels.

Ceux du premier type sont spécialement employés pour leur simplicité. Avec les données $\{x_1, \dots, x_N\}$, un numéro préspecifié de clusters $K < N$ est choisi, chacun étiqueté par un numéro $k \in \{1, \dots, K\}$. On assigne après à chaque élément i de l'ensemble un cluster $C : \{1, \dots, N\} \rightarrow \{1, \dots, K\}$, $i \mapsto C(i) = C_k$, en essayant de minimiser une fonction "de perte" qui prend en compte les dissimilarités $d(x_i, x_{i'})$ entre les données. Une fonction de perte naturelle à considérer serait

$$W(C) = \sum_{k=1}^K \sum_{i, i' \in C_k} d(x_i, x_{i'}), \quad (1)$$

qui quantifie de quelle manière les observations mises dans le même cluster sont proches entre elles. Il est facile de voir que minimiser $W(C)$ est équivalent à maximiser

$$B(C) = \sum_{k=1}^K \sum_{i \in C_k} \sum_{i' \notin C_k} d(x_i, x_{i'}),$$

car $T = \sum_{i, i'} d(x_i, x_{i'}) = W(C) + B(C)$ est constant.

Nous pourrions penser que cela réduit le problème au calcul de la valeur de la fonction de perte sur toutes les possibles combinaisons, mais dans la pratique le nombre d'allocations possibles pour tous les éléments augmente très rapidement avec N et k . De ce fait, tout algorithme de clustering efficace étudie seulement une fraction très petite des attributions $k = C(i)$ possibles, avec l'objectif d'identifier un sous-ensemble susceptible de contenir l'optimale, ou au moins une correspondance assez bonne.

La stratégie se base généralement sur ce qu'on appelle un "greedy descent" itérative : une partition initiale est choisie et, à chaque pas, les attributions sont changées de sorte que la valeur du critère est améliorée par rapport à l'antérieure. L'algorithme se termine par une partition lorsque aucune amélioration est possible.

Ces algorithmes, travaillant avec un sous-ensemble très petit des combinaisons possibles, convergent toujours à un maximum local, qui peut être très sub-optimal en comparaison avec le maximum global.

2.2.1. K-means clustering

Il est un des algorithmes les plus populaires en raison de sa vitesse et sa simplicité. Il a aussi des applications importantes dans la compression des images et signaux (*vector quantization*).

Cet algorithme suppose que toutes les variables sont de type quantitative, et il prend la distance euclidienne habituelle au carré, $d(x_i, x_{i'}) = \sum_{j=1}^p (x_{ij} - x_{i'j})^2 = \|x_i - x_{i'}\|^2$, pour mesurer la dissimilarité entre les observations. Avec ces conditions, nous remarquons que (1) est égal à

$$W(C) = \sum_{k=1}^K N_k \sum_{i \in C_k} \|x_i - \bar{x}_k\|^2, \quad (2)$$

où $\bar{x}_k = (\bar{x}_{1k}, \dots, \bar{x}_{pk})$ est le vecteur moyen associé aux observations du cluster k , et N_k est son nombre d'éléments.

Du fait que la moyenne des $\{y_1, \dots, y_m\}$ minimise la fonction $f(y) = \sum_{i=1}^m (y_i - y)^2$, nous

pouvons obtenir une méthode itérative descendant pour résoudre

$$C^* = \min_C \sum_{k=1}^K N_k \sum_{i \in C_k} \|x_i - \bar{x}_k\|^2, \quad (3)$$

notre problème original, en considérant le problème d'optimisation plus général

$$C^* = \min_{C, \{m_k\}_1^K} \sum_{k=1}^K N_k \sum_{i \in C_k} \|x_i - m_k\|^2. \quad (4)$$

L'algorithme est le suivant :

Algorithm 1: *K-means clustering*

Input: $\{x_1, \dots, x_N\}$ observations quantitatives (p features chacune)

K numéro de clusters souhaité

Output: Pour chaque observation, une étiquette $k \in [1, K]$ (cluster assigné)

- 1 On fait une première attribution $C(i)$ pour chaque observation, aléatoire ou avec une moyenne déjà établie.
 - 2 Avec notre partition C , on minimise la variance totale du clustering (4), obtenant ainsi les moyennes $\{m_1, \dots, m_K\}$ associées à chaque cluster.
 - 3 Avec ces valeurs $\{m_1, \dots, m_K\}$, on minimise davantage (4) en assignant à chaque observation le cluster avec la moyenne la plus proche : $C(i) = \arg \min_{1 \leq k \leq K} \|x_i - m_k\|^2$
 - 4 On répète 2 et 3 jusqu'à ce que les attributions C ne changent plus.
-

Étant donné que à les étapes 2 et 3 la quantité (4) diminue, la convergence de la méthode est assurée. Néanmoins, normalement on atteint un maximum local sub-optimal. De ce fait, c'est une bonne idée de courir l'algorithme avec différentes partitions initiales et prendre le meilleur résultat final.

Nous pouvons généraliser l'idée du clustering K-means à distances différentes à l'eulclidienne et features pas nécessairement quantitatives si nous travaillons directement avec les dissimilarités $d(x_i, x_{i'})$. Pour cela, nous pouvons utiliser l'algorithme décrit avant en changeant le m_k : au lieu de la moyenne des éléments du cluster k , nous prenons un de ces éléments ; en particulier, l'élément x_k qui minimise $\sum_{i \in C_k} d(x_k, x_i)$. Cette nouvelle méthode, qui s'appelle *clustering K-medoids*, a aussi un coût informatique considérable, et n'est souvent pas réalisable exhaustivement.

2.2.2. Clustering hiérarchique

Contrairement au clustering K-means/ K-medoids, qui part d'un nombre de clusters K préréglé et les cherchent, les méthodes de clustering hiérarchiques produisent une représentation "en échelle" qui passe pour tous les nombres possibles, et où les clusters à chaque niveau sont créés en unifiant ou divisant ceux du niveau inférieur. De cette façon, il est possible de voir plus facilement quel est le "bon" numéro de clusters de l'ensemble. Naturellement, il est encore nécessaire d'établir une "mesure de similitude" entre groupes, basée sur les dissimilarités entre paires d'éléments.

Il y a deux stratégies principales pour ce type de clustering : l'agglomérative (*bottom-up*), où nous commençons avec un cluster pour chaque observation et nous les unifions par paires à mesure que l'algorithme court ; et la divisive (*up-bottom*), qui part par un seul cluster et ensuite les divise

en deux peu à peu. Dans le deux cas, chaque niveau de la hiérarchie représente un regroupement spécifique des données en clusters disjoints, et la hiérarchie entière les différents "seuils" où ils apparaissent.

Ces agglomérations/ divisions binaires récursives peuvent être représentées sous forme d'arbre, qui commence avec une seule racine (le cluster avec toutes les données) et, à la fin, a une feuille pour chaque élément. De plus, une partie important de ces méthodes ont la propriété de la "monotonie", c'est à dire, la dissimilarité entre clusters (qui se mesure quantitativement) augmente de manière monotone à mesure qu'on les unifie. Ainsi, l'arbre peut être dessiné de sorte que les bifurcations entre les branches se produisent à des hauteurs qui reflètent la durée de tous les clusters de manière proportionnelle. Ce type de représentation graphique, assez complète et informative sur les données, s'appelle *dendrogram*.

Néanmoins, ces dendrograms sont assez sensibles aux données et à les particularités de la méthode choisie, et ils imposent sur les données une structure hiérarchique qui pourrait ne pas exister. Donc, plus qu'une "carte" infaillible de la structure des données elles-mêmes, le dendrogram devrait être vu plutôt comme une carte de la structure du clustering des ces données, obtenues avec un algorithme et une métrique spécifiques.

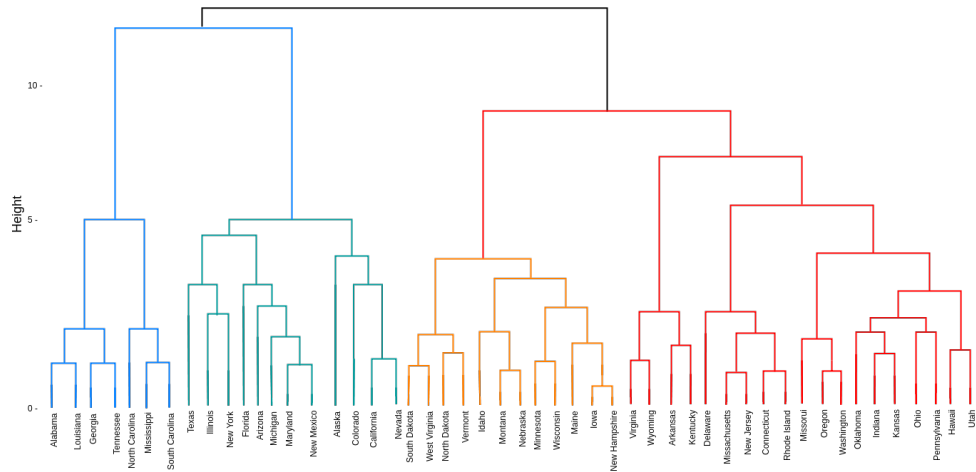


FIGURE 1 – Exemple de dendrogram (où on regroupe les états des États-Unis, critère inconnu)

Stratégies agglomératives

Ces méthodes commencent avec un singleton cluster pour chaque élément. Puis, à chacun des $N-1$ pas, les deux groupes les plus "proches" sont fusionnés, et on perd un cluster au niveau suivant.

Naturellement, la notion de "proximité" entre les groupes doit être définie en considérant les dissimilarités entre les paires d'observations. Soient G et H deux de ces groupes. Nous remarquons plusieurs options pour définir la dissimilarité $d(G, H)$:

- Le *saut minimum* (*single linkage* en anglais) est défini pour $d_{SL}(G, H) = \min_{i \in G, i' \in H} d_{ii'}$.
- Le *saut maximum* (*complete linkage* en anglais) se définit comme $d_{CL}(G, H) = \max_{i \in G, i' \in H} d_{ii'}$.
- Le *lien moyen* (*group average* en anglais) est défini pour $d_{GA}(G, H) = \frac{1}{N_G N_H} \sum_{i \in G} \sum_{i' \in H} d_{ii'}$.

En général, si les données montrent une structure claire, avec des clusters bien séparés les uns des autres et suffisamment compacts (i.e. ses éléments sont proches entre eux en comparaison avec ceux de clusters différents), les trois méthodes produisent des résultats similaires. Développons maintenant les particularités de chaque méthode :

Le type *saut minimum* a seulement besoin que deux éléments de groupes différents soient proches pour les fusionner, indépendamment des autres éléments ; cela résulte souvent en clusters peu compacts.

Le type *saut maximum* est le pôle opposé ; deux groupes G et H sont proches seulement si tous les éléments dans son union sont relativement similaires, et les clusters sont plus compacts. Pourtant, cela peut causer aussi une relaxation du "principe de proximité" : un élément assigné dans un cluster peut être beaucoup plus proche des éléments d'autres clusters que ceux de son propre groupe. Le type *lien moyen* permet un compromis entre ces deux extrêmes, mais il est aussi plus dépendant des valeurs spécifiques des $d_{ii'}$, à la différence des deux autres, qui dépendent uniquement de son ordre.

Stratégies divisives

Ces méthodes commencent avec toutes les données dans un unique groupe, et divisent à chaque itération un cluster existant en deux clusters plus petits.

Même si elles sont moins étudiées que les méthodes agglomératives, on peut toujours diviser n'importe quel cluster en appliquant une méthode combinatoire, comme K-means avec $K = 2$, à chaque itération. Cependant, en général ce processus ne produit pas une séquence de clusters avec la propriété de la monotonie nécessaire pour la représenter correctement en forme de dendrogram.

Un algorithme qui satisfait cela serait celui-ci :

Algorithm 2: *Clustering hiérarchique divisive monotone*

Input: $\{x_1, \dots, x_N\}$ observations quantitatives

Les dissimilarités $d_{ii'}$ entre toutes les paires d'observations

Output: Une séquence hiérarchique de clusters

- 1 On met toutes les observations dans un unique cluster, G .
 - 2 On trouve l'élément i dans G avec la dissimilarité moyenne avec les autres éléments de G , $\frac{1}{N_G} \sum_{j \in G} d_{ij}$, la plus élevée. Cet élément sera le premier membre d'un deuxième cluster H .
 - 3 On prend l'élément de G qui a la distance moyenne avec les éléments de G moins la distance moyenne avec les éléments de H la plus grande et le transfère à H .
 - 4 On continue à faire cela jusqu'à ce que cette différence devienne négative. En ce moment, il n'y a plus d'observations dans G qui sont, en moyenne, plus proches à celles du H qu'à celles de son groupe G . Nous avons alors deux nouveaux clusters.
 - 5 Nous continuons de répéter 2, 3 et 4 avec un cluster présent, nouveau ou pas, jusqu'à obtenir N singleton clusters. Pour choisir le groupe suivant à diviser, deux critères utiles seraient :
 - Le cluster C avec le diamètre $D_C = \max_{i,i' \in C} d_{ii'}$ le plus grand.
 - Celui avec la dissimilarité entre éléments moyenne, $\bar{d}_C = \frac{1}{N_C^2} \sum_{i,i' \in C} d_{ii'}$, la plus grande.
-

3. Introduction à l'Analyse topologique des données

3.1. Idée générale et motivation

L'Analyse topologique des données (*Topological Data Analysis* en anglais, souvent nommée *TDA*), commence à se développer dans les années 2000 à partir de quelques travaux dans la topologie appliquée et la géométrie algorithmique. Ce champ d'étude cherche à explorer et étudier les bases de données en utilisant des techniques et idées typiques du domaine de la topologie. Cette nouvelle approche de la science de données, qui s'est déjà avérée très utile dans plusieurs contextes, a pour objectif mieux comprendre la "forme" d'un ensemble de données. Cette question peut être spécialement compliquée quand on travaille en dimensions élevées, et avec des données incomplètes ou avec une forte présence de bruit.

En résumé, la TDA essaie de fournir des méthodes mathématiques, statistiques et algorithmiques pour révéler, analyser et utiliser des structures géométriques et topologiques non évidents dans un ensemble des données. Notamment, un de ses outils principales est celui de l'*homologie persistante*, une adaptation de l'homologie pour nuages de points, qui a besoin d'une solide formulation théorique et mathématique.

Le schéma de déroulement habituel en TDA est :

1. L'input est généralement un ensemble fini de points avec quelque type de similarité ou distance définie entre eux. Cette distance peut venir induite pour un hypothétique espace ambiant (par exemple, \mathbb{R}^d) ou être définie intrinsèquement entre paires de points, en fonction du cas.
2. Quelque type de structure géométrique de nature traitable et algorithmique est construite sur ces points, avec l'objectif de faire plus évidents quelques de ses caractéristiques. Souvent, nous faisons cela en utilisant un ou plusieurs *complexes simpliciaux*, qui peuvent être vus comme une généralisation des graphes en dimensions plus élevées.
3. Nous extrayons cette information géométrique et topologique en utilisant différents méthodes, et nous étudions sa pertinence et stabilité par rapport à possibles perturbations des données ou présence de bruit. Cette information est après souvent visualisée et combinée avec d'autres descripteurs pour guider les prochaines étapes de l'analyse des données ou tâches de ML.

Notre algorithme ToMATo fait usage de certains des concepts de ce nouveau champ d'étude, notamment de l'homologie persistante et les diagrammes de persistance. Donc, l'objectif de cette partie du travail est d'introduire avec rigueur et généralité les fondements de la TDA et les bases mathématiques de l'homologie persistante.

3.2. Complexes simpliciales, recouvrements et le Théorème du Nerf

Étant donné que la plupart des concepts de la topologie et la géométrie sont associés à des espaces continus, une pratique habituelle dans le TDA est de "connecter" de quelque sorte les données (représentées comme points) qui sont proches les unes des autres. On formalise souvent cette notion de proximité en utilisant une distance entre points, qui peut être définie entre paires directement (espace métrique discrète) ou en plongeant les données dans un espace métrique plus grand (typiquement, \mathbb{R}^d).

En tout cas, après avoir connecté les données proches, nous obtenons un graphe de voisinage, qui permet déjà appliquer plusieurs méthodes d'analyse. Pour aller au-delà de la connectivité,

nous pouvons associer pas seulement paires mais aussi $(k + 1)$ -tuples de points proches entre eux. Nous obtenons ainsi un *complexe simplicial*, qui permet identifier de nouvelles caractéristiques topologiques, comme cycles, "trous" et leurs généralisations en haute dimension.

Définition 3.1. Soit $\mathbb{X} = \{x_0, \dots, x_k\} \subset \mathbb{R}^d$ $(k + 1)$ points affines linéairement indépendants. Le simplexe k -dimensionnel $\sigma = [x_0, \dots, x_k]$ généré pour \mathbb{X} est l'enveloppe convexe de \mathbb{X} . Les points originales sont ses sommets, et les simplexes générés pour les sous-ensembles de ces points sont les faces de σ .

Remarque 3.2. Formellement, les sous-ensembles d'un simplexe sont appelés ses n -faces, où n est sa cardinalité moins 1. Cependant, pour les cas 0 et 1 normalement on utilise les mots *sommets* et *arêtes* respectivement, et le mot *face* pour le cas 2 et en général.

Définition 3.3. Un complexe simplicial géométrique $K \subset \mathbb{R}^d$ est une collection de simplexes telle que :

1. Toute face d'un simplexe de K est aussi un simplexe de K .
2. Toute intersection de deux simplexes de K est vide ou une face commune aux deux.

Encore plus généralement, un complexe simplicial abstrait avec des sommets V est une collection K de sous-ensembles finis de V telle que que les éléments de V appartient à K et, pour tout élément σ de K , tout sous-ensemble de σ appartient aussi à K .

Clairement, on peut utiliser la dernière définition, de nature plus combinatoire, pour étudier un complexe simplicial géométrique, mais la direction inverse fonctionne aussi : on peut mettre tout complexe simplicial abstrait dans \mathbb{R}^d pour quelque d , et le considérer comme un sous-espace avec la topologie induite. C'est cette structure, appelée *réalisation géométrique* de K , qui permet utiliser sans problèmes plein de notions géométriques et topologiques sur K .

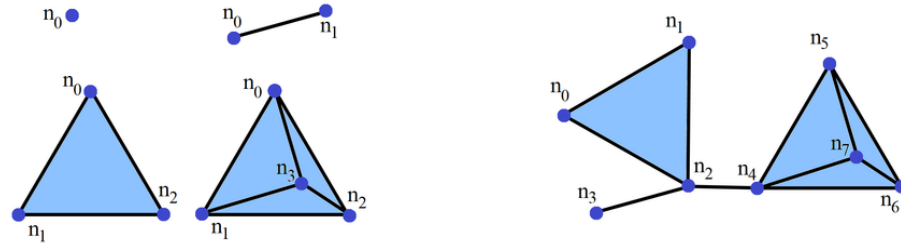


FIGURE 2 – Simplexes de dimension 0,1,2 et 3, et exemple d'un complexe simplicial géométrique

Définition 3.4. La dimension d'un simplexe est simplement sa cardinalité moins 1. La dimension d'un complexe simplicial est la dimension plus grande parmi les simplexes qui le constituent.

Remarque 3.5. Un graphe est un complexe simplicial de dimension 1.

Étant donnée un ensemble de points X (imaginons que dans un espace métrique (M, d)), nous pouvons construire des complexes simpliciaux au-dessus de plusieurs manières. Deux des constructions les plus habituelles seraient :

1. *Complexe de Vietoris-Rips, $Rips_\alpha(X)$* : La généralisation immédiate de la notion de graphe de voisinage. C'est le complexe simplicial qui a pour ensemble de faces les simplexes $[x_0, \dots, x_k]$ qui satisfont $d(x_i, x_j) \leq \alpha$ pour tout $0 \leq i, j \leq k$.

2. *Complexe de Čech, $Cech_\alpha(X)$* : Étroitement lié au Vietoris-Rips complexe, c'est le complexe simplicial formé pour les simplexes $[x_0, \dots, x_k]$ qui satisfont que l'intersection des $k+1$ boules $\overline{B}(x_i, \alpha)$ n'est pas vide.

Remarque 3.6. Même si X est un ensemble fini de points dans \mathbb{R}^d , $Rips_\alpha(X)$ et $Cech_\alpha(X)$ n'admettent pas toujours une réalisation géométrique dans \mathbb{R}^d , donc ses dimensions peuvent être plus élevées.

Remarque 3.7. C'est facile de voir qu'on a toujours $Rips_\alpha(X) \subseteq Cech_\alpha(X) \subseteq Rips_{2\alpha}(X)$, où les inclusions peuvent être strictes. Si $X \subset \mathbb{R}^d$, $Cech_\alpha(X)$ et $Rips_{2\alpha}(X)$ ont le même squelette 1-dimensionnel, i.e. le même ensemble de sommets et arêtes.

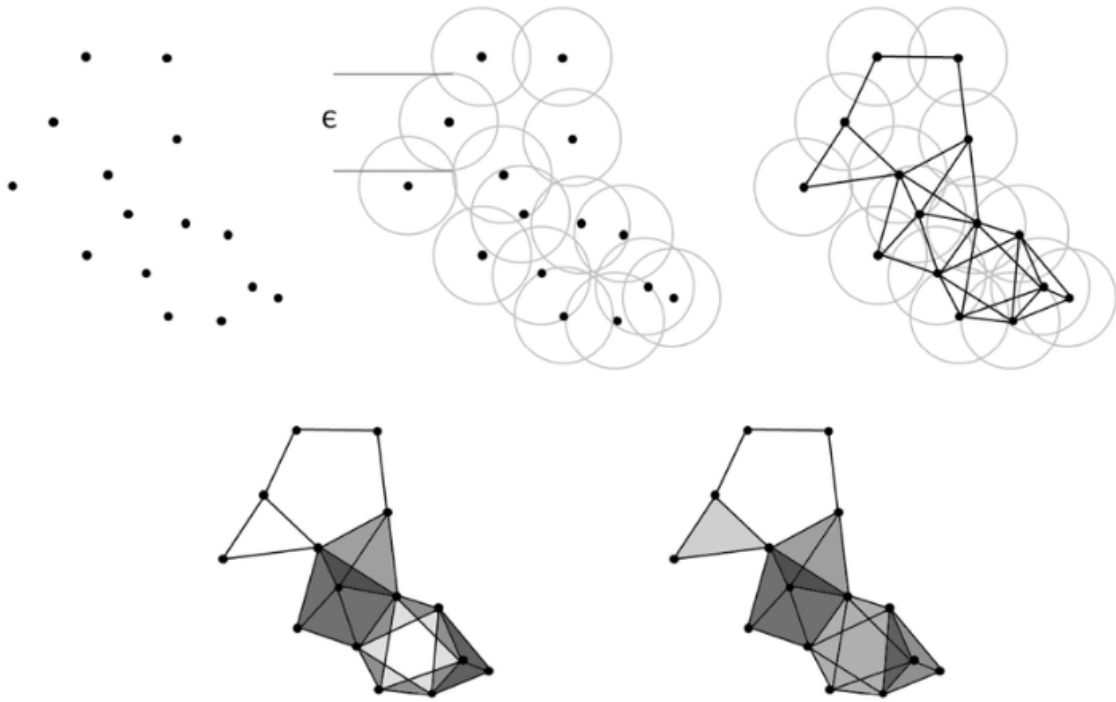


FIGURE 3 – Construction des complexes de $Cech_{\frac{\epsilon}{2}}$ (en bas à gauche) et de $Rips_\epsilon$ (en bas à droite). La troisième image montre comment les deux complexes ont le même squelette 1-dimensionnel.

En fait, le complexe de Čech est un cas particulier d'une construction de complexes plus générale en utilisant des recouvrements :

Définition 3.8. Soit M un espace topologique (ou un ensemble, en général). Un recouvrement U de M est une famille de sous-ensembles de M , $U = (U_i)_{i \in I}$, qui satisfait $\cup_{i \in I} U_i = M$. Le nerf d'un recouvrement U de M est le complexe simplicial abstrait $C(U)$ qui a U_i comme sommets et les faces

$$\sigma = [U_{i_0}, \dots, U_{i_k}] \in C(U) \iff \cap_{j=0}^k U_{i_j} \neq \emptyset$$

De cette façon, $Cech_\alpha(X)$ est le nerf du recouvrement $U = (\overline{B}(x_i, \alpha))_{x_i \in X}$ de l'ensemble $M = \cup_{x_i \in X} \overline{B}(x_i, \alpha)$, qui contient évidemment l'ensemble de points original. Mais un recouvrement d'un ensemble de données ne doit pas forcément être basé sur des boules centrées sur elles ;

par exemple, nous pourrions aussi en avoir un en faisant une subdivision des données en groupes de points (non nécessairement disjoints) avec des propriétés similaires.

En tout cas, le nerf d'un recouvrement offre une description de nature combinatoire assez globale et compacte de la relation entre les sous-ensembles du recouvrement en considérant ses plusieurs intersections. Le Théorème du Nerf (Nerve Theorem) est un résultat de topologie algébrique qui lie, avec quelques assumptions, la topologie du nerf d'un recouvrement avec la topologie du recouvrement lui-même. Ses implications dans l'Analyse topologique des données sont remarquables, et même si nous ne le démontrerons pas, le but de la dernière partie de cette section est introduire les notions nécessaires pour bien comprendre ce qu'il dit.

Dans la topologie, normalement nous considérons deux espaces topologiques X et Y comme égales quand ils sont *homéomorphes*, i.e. nous pouvons trouver deux applications continues et bijectives $f : X \rightarrow Y$ et $g : Y \rightarrow X$ qui satisfont $g \circ f = id_X$ et $f \circ g = id_Y$. Cependant, dans plusieurs situations, aussi dans la TDA, la notion d'homéomorphisme est trop rigide, et souvent il est convenable d'étudier des similitudes entre espaces topologiques un peu plus faibles. C'est ici où l'idée de l'homotopie apparaît :

Définition 3.9. *Soit X et Y deux espaces topologiques. Deux applications continues $f_0, f_1 : X \rightarrow Y$ sont homotopiques s'il existe une application continue $H : X \times [0, 1] \rightarrow Y$ telle que, $\forall x \in X$, $H(x, 0) = f_0(x)$ et $H(x, 1) = f_1(x)$. Dans ce cas, on écrit $f_0 \simeq f_1$. On dit que X et Y sont des espaces topologiques homotopiquement équivalents si on peut trouver deux applications $f : X \rightarrow Y$ et $g : Y \rightarrow X$ tels que $g \circ f \simeq id_X$ et $f \circ g \simeq id_Y$. Dans ce cas, on écrit $X \simeq Y$.*

La notion d'équivalence homotopique est plus faible que celle d'homéomorphisme, donc deux espaces homéomorphes sont toujours homotopiquement équivalents, mais le réciproque n'est pas vrai. En tout cas, l'intérêt principal derrière l'homotopie est que nous pouvons définir des objets (souvent de nature algébrique) sur les espaces topologiques qui sont effectivement des invariants homotopiques, c'est-à-dire qui sont conservés entre des espaces topologiques homotopiquement équivalents. Les exemples les plus notables seraient les *groupes d'homotopie* et les *groupes d'homologie* (singulaire, simpliciale). On parlera plus en détail de l'homologie dans la section suivante.

Définition 3.10. *Un espace X est contractile s'il est homotopiquement équivalent à un point.*

Exemple 3.11. Tout boule dans \mathbb{R}^d , ouverte ou fermée, est contractile. Plus généralement, tout sous-ensemble convexe X dans \mathbb{R}^d est contractile. En effet, si on suppose $0 \in X$, il y a les applications $f : X \rightarrow \{0\}$, $x \mapsto 0$, et $g : \{0\} \rightarrow X$, $0 \mapsto 0$. Clairement $f \circ g \simeq id_{\{0\}}$ (en fait, $f \circ g = id_{\{0\}}$), et $g \circ f \simeq id_X$, avec l'application continue $H : X \times [0, 1] \rightarrow X$, $H(x, t) = t \cdot x$.

Un recouvrement ouvert est celui où tous les éléments de la famille sont ouverts. Un recouvrement ouvert fini où tous les éléments et intersections entre éléments sont contractiles satisfait le résultat suivant, souvent nommé le Théorème du Nerf :

Théorème 3.12. (Théorème du Nerf) *Soit $U = (U_i)_{i \in I}$ un recouvrement ouvert fini d'un sous-ensemble $X \subseteq \mathbb{R}^d$ tel que toute intersection des U_i 's est vide ou contractile. Alors X et $C(U)$ sont homotopiquement équivalents.*

Ainsi, on a que le nerf défini par un "bon" recouvrement de X est homotopiquement équivalent à X , ce qui est remarquable pour des applications ; en effet, normalement un complexe simplicial possède une nature beaucoup plus traitable algorithmiquement qu'un espace topologique général.

En tout cas, le complexe de Čech se construit avec des boules fermées centrées sur chaque donnée, donc le recouvrement n'est pas ouvert dans ce cas. Heureusement, la version suivante du théorème est aussi vraie :

Théorème 3.13. (Théorème du Nerf pour un recouvrement convexe) *Soit $X \subseteq \mathbb{R}^d$ une union finie d'ensembles fermés convexes $F = (F_i)_{i \in I}$ dans \mathbb{R}^d . Alors X et $C(F)$ sont homotopiquement équivalents.*

De ce fait, on obtient que, en effet, si X est un nuage de points dans \mathbb{R}^d , alors $Cech_\alpha(X)$ est homotopiquement équivalent à l'union des boules $\bigcup_{x \in X} \overline{B}(x, \alpha)$.

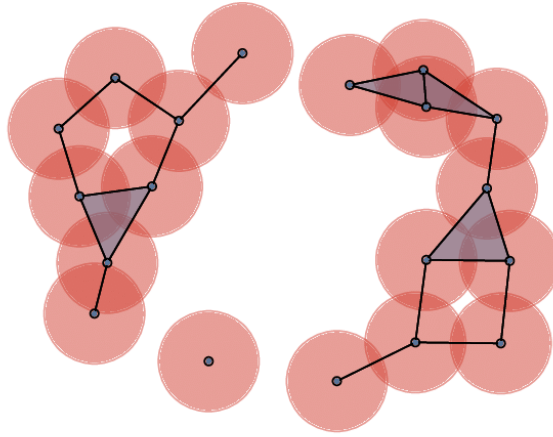


FIGURE 4 – Représentation de comment l'union des boules et le nerf associé (i.e. le complexe de Čech des données) sont homotopiquement équivalents.

3.3. Inférence homologique

Résumons la situation jusqu'à ici : pour aller plus loin dans notre étude de nos ensembles de données/ nuages de points, nous avons vu comment construire quelques structures géométriques au-dessus, appelées complexes simpliciaux, de nature plus algorithmique. Après, nous avons exposé le Théorème du Nerf, donc nous avons vu que, quand \mathbb{X} est un ensemble de points dans \mathbb{R}^d , $Cech_\alpha(\mathbb{X})$ est homotopiquement équivalent à l'union de boules $\bigcup_{x \in \mathbb{X}} \overline{B}(x, \alpha)$.

Dans toute situation avec des données numériques (supposons dans \mathbb{R}^d), et d'un point de vue statistique, il y a fondamentalement deux questions qui nous intéressent : d'une part, il y a l'"espace d'échantillonnage" de nos données, i.e. dans quelle région $M \subseteq \mathbb{R}^d$ toutes les possibles données "vivent" ; de l'autre part, il y a la mesure de probabilité μ sur cette région M , qui encode quelles zones de M sont plus probables d'avoir plus de points, et de quelle manière les données se regroupent. Normalement, nous supposons que M , le *support* de μ , est compact, et que nos données $\mathbb{X} = \{x_1, \dots, x_n\}$ ont été échantillonnées i.i.d. en suivant μ .

Évidemment, pour mieux comprendre nos données, faire des prédictions, etc., nous sommes intéressés à connaître μ et la "forme" de son support M . Le processus qui essaie de mieux caractériser

M s'appelle "reconstruction géométrique", et un schéma habituel pour inférer cette information géométrique et topologique est :

1. Nous recouvrons \mathbb{X} avec des boules $B(x, \alpha)$; sous certaines conditions de régularité sur M , nous pouvons lier la topologie de cette union avec celle de M .
2. Certaines propriétés topologiques de M sont inférées à partir du nerf de l'union de ces boules, en utilisant le Théorème du Nerf.

Des résultats mathématiquement rigoureux et importants existent avec cette approche de reconstruction. Néanmoins, ce n'est pas toujours possible, ni souhaitable, d'essayer de reconstruire complètement la forme de base à partir de nos données. De plus, dans le schéma que nous venons d'exposer, nous voyons que le choix du rayon des boules, qui souvent n'est pas du tout évident, joue un rôle clé dans les résultats obtenus.

Une autre manière de travailler les données est d'essayer de trouver des invariants topologiques plus faibles, mais plus faciles d'inférer. C'est ici que le concept d'*homologie*, un outil déjà classique dans la topologie algébrique, entre en scène. Plus notamment, nous pouvons faire usage de l'*homologie simpliciale* sur nos complexes simpliciaux pour mieux les comprendre et, finalement, élaborer davantage cette information homologique pour développer ce qu'on appelle *homologie persistante*, qui garde une trace de comme l'homologie des complexes simpliciaux obtenues évolue en variant le rayon. Une manière de représenter visuellement une bonne partie de toute cette information est avec ce qu'on appelle un *diagramme de persistance*.

3.3.1. Homologie simpliciale et nombres de Betti

L'idée intuitive derrière de l'homologie en général est de traiter et formaliser algébriquement la notion de "trou", ou "boucle", dans de différents contextes mathématiques, notamment dans les espaces topologiques. Pour toute dimension n , les "trous" n -dimensionnels sont représentés par un espace vectoriel H_n , et sa dimension serait le numéro de trous "indépendants" de ce type. Par exemple, H_0 représente les composantes connexes de notre espace, H_1 les "boucles unidimensionnelles", H_2 les "cavités 2-dimensionnelles", etc.

Le premier type de théorie d'homologie qui a été développé, il y a environ un siècle, est l'homologie simpliciale, qui se construit sur les complexes simpliciaux. Sur ces objets, c'est relativement simple d'imaginer la notion de trou k -dimensionnel. Même si les concepts que nous exposerons ensuite sont sensés avec tout corps k , nous travaillerons désormais avec $k = \mathbb{Z}/2\mathbb{Z} = \mathbb{Z}_2$, plus intuitif à niveau géométrique, et qui simplifie les arguments ; sinon, il faudrait considérer une orientation sur les sommets/ faces de notre complexe, et les formules deviendraient plus compliquées.

Soit K un complexe simplicial de dimension d :

Définition 3.14. Une n -chaîne est une somme formelle de simplexes n -dimensionnelles de K ; c'est à dire, si $\{\sigma_1, \dots, \sigma_p\}$ sont les n -faces de K , une n -chaîne c est une expression du type

$$c = \sum_{i=0}^p \lambda_i \sigma_i, \text{ avec } \lambda_i \in \mathbb{Z}_2$$

Pour chaque n , l'ensemble des n -chaînes $C_n(K)$ a une structure évidente de \mathbb{Z}_2 -espace vectoriel, où l'ensemble des n -faces de K est une base de $C_n(K)$. Les chaînes avec des coefficients dans \mathbb{Z}_2 ont une interprétation géométrique simple : du fait que toute n -chaîne peut être uniquement écrite comme $c = \sigma_{i_1} + \dots + \sigma_{i_m}$, c représente simplement l'union des n -simplexes σ_{i_j} .

Définition 3.15. Le bord $\partial(\sigma)$ d'un n -simplexe σ est la somme de ses $(n-1)$ -faces. Donc, pour le n -simplexe $\sigma = [v_0, \dots, v_n]$, on obtient la $(n-1)$ -chaîne

$$\partial(\sigma) = \sum_{i=0}^n [v_0, \dots, \hat{v}_i, \dots, v_n]$$

où $[v_0, \dots, \hat{v}_i, \dots, v_n]$ est le $(n-1)$ -simplexe formé pour les sommets originels sauf v_i .

Le bord d'un n -simplexe nous donne les $(n-1)$ -faces qui le constituent. Le bord ainsi défini sur les simplexes de K peut être étendue de manière naturelle à une (plusieurs) fonction entre les $C_i(K)$. Même si on devrait les distinguer ∂_i , souvent on écrit simplement ∂ pour éclaircir le texte :

Définition 3.16. La fonction bord est l'application linéaire définie par

$$\begin{aligned} \partial : C_n(K) &\longrightarrow C_{n-1}(K) \\ c &\mapsto \partial(c) = \sum_{\sigma \in c} \partial(\sigma) \end{aligned}$$

La propriété plus fondamentale de ∂ est celle-ci :

Proposition 3.17. $\partial\partial = \partial \circ \partial = 0$

Démonstration. Puisque la fonction bord est linéaire, il suffit de le vérifier simplement pour un seul simplexe $\sigma = [v_0, \dots, v_n]$, de dimension n :

$$\begin{aligned} \partial\partial\sigma &= \partial\left(\sum_{i=0}^n [v_0, \dots, \hat{v}_i, \dots, v_n]\right) = \sum_{i=0}^n \partial[v_0, \dots, \hat{v}_i, \dots, v_n] = \\ &= \sum_{j<i} [v_0, \dots, \hat{v}_j, \dots, \hat{v}_i, \dots, v_n] + \sum_{j>i} [v_0, \dots, \hat{v}_i, \dots, \hat{v}_j, \dots, v_n] = \sum_{\substack{j,i=0 \\ j \neq i}}^n 2[v_0, \dots, \hat{v}_i, \dots, \hat{v}_j, \dots, v_n] = 0 \end{aligned}$$

□

La fonction bord définit une séquence d'applications linéaires entre les $C_i(K)$:

Définition 3.18. Le complexe de chaînes associé au complexe simplicial K est la séquence d'espaces vectoriels et applications linéaires :

$$\{0\} \xrightarrow{\partial} C_d(K) \xrightarrow{\partial} C_{d-1}(K) \xrightarrow{\partial} \dots \xrightarrow{\partial} C_1(K) \xrightarrow{\partial} C_0(K) \xrightarrow{\partial} \{0\}$$

Pour $k \in \{0, \dots, d\}$, l'ensemble $Z_k(K)$ de k -cycles de K est le noyau de $\partial : C_k(K) \rightarrow C_{k-1}(K)$:

$$Z_k(K) = \{c \in C_k(K) \mid \partial(c) = 0\},$$

et l'ensemble $B_k(K)$ de k -bords de K sont les chaînes qui appartient à l'image de l'application ∂ :

$$B_k(K) = \{c \in C_k(K) \mid \exists b \in C_{k+1}(K) \text{ tel que } \partial(b) = c\},$$

De quelque sorte, Z_k encode quelles k -chaînes sont "fermées" (d'ici le nom "cycles"), et B_k quels ensembles de k -faces sont le bord d'une $(k+1)$ -chaîne.

Z_k et B_k sont évidemment des sous-espaces de C_k , et en vue de la Proposition 3.17, on a toujours

$B_k \subseteq Z_k$, où la inclusion peut être stricte. Ce dernier fait motive la définition des groupes d'homologie, qui essaient de trouver des "trous" dans notre complexe simplicial, i.e. des k -chaînes fermées qui ne sont la frontière d'aucune $(k+1)$ -chaîne du complexe :

Définition 3.19. *Le k -ème groupe d'homologie de K est l'espace vectoriel quotient*

$$H_k(K) = Z_k(K)/B_k(K);$$

ses éléments s'appellent les classes d'homologie de K . Deux cycles qui appartiennent à la même classe d'homologie sont appelés homologues.

La dimension $\beta_k(K)$ de $H_k(K)$ s'appelle le k -ème nombre de Betti de K .

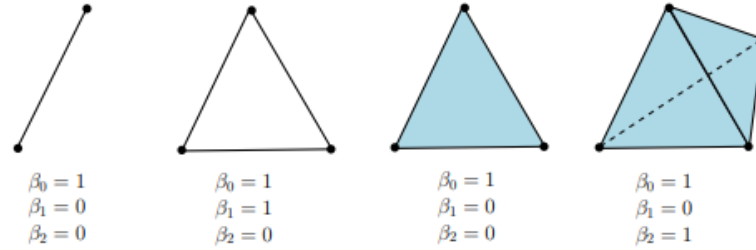


FIGURE 5 – Les nombres de Betti associés à quelques complexes simpliciaux simples : une arête (avec ses sommets), le périmètre d'un triangle, un triangle, et un tétraèdre.

On peut démontrer (ce qui n'est pas immédiat) que les groupes d'homologie et les nombres de Betti sont des invariants topologiques : si K_0 et K_1 sont deux complexes simpliciaux avec des réalisations géométriques homéomorphes, alors ses groupes d'homologie sont isomorphes et ses nombres de Betti sont égales. De plus, ces résultats sont aussi vraies si les réalisations géométriques sont seulement homotopiquement équivalents.

Ces résultats sont une conséquence de l'étroite relation entre l'homologie simpliciale et une autre type d'homologie, l'*homologie singulière*, beaucoup plus générale et qui peut être définie pour tout espace topologique. En fait, on peut démontrer que les groupes d'homologie simpliciales et singulières d'un complexe simplicial sont toujours isomorphes, et le résultat est une conséquence de l'invariance homotopique de l'homologie singulière. Bien que notre intention dans ce mémoire n'est pas d'introduire l'homologie singulière, le résultat suivant, pas difficile mais sans démonstration, nous sera utile dans les pages qui viennent :

Proposition 3.20. *Soit X un espace topologique (resp. un complexe simplicial). Alors, la dimension du premier groupe d'homologie singulière $H_0(X)$ (resp. homologie simpliciale) est égale au nombre de composantes connexes (par arcs) de X .*

3.3.2. Filtrations

Définition 3.21. *Une filtration d'un complexe simplicial K est une suite de subcomplexes $(K_r)_{r \in T}$, où $T \subseteq \mathbb{R}$ fini ou infini, telle que $\forall r_0, r_1 \in T, r_0 \leq r_1 \implies K_{r_0} \subseteq K_{r_1}$, et $K = \cup_{r \in T} K_r$. La définition peut être généralisée de la manière évidente à tout espace topologique.*

Dans des situations pratiques, les valeurs $r \in T$ souvent jouent le rôle de "paramètres d'échelle", qui ajustent la résolution du complexe. Deux filtrations habituelles dans le TDA seraient :

- *Filtrations construites sur des nuages de points* : Étant donné un sous-ensemble fini X d'un espace métrique compact (M, d) , les familles de complexes $(Rips_r(X))_{r \in \mathbb{R}}$ et $(Cech_r(X))_{r \in \mathbb{R}}$ sont des filtrations. Dans ces dernières, r peut être vu comme le paramètre de résolution, où, pour $r \leq 0$, nous considérons seulement les points. Par exemple, quand X est un nuage de points à \mathbb{R}^d , grâce au Théorème du Nerf, $(Cech_r(X))_{r \in \mathbb{R}}$ encode la topologie de la famille d'unions de boules $X_r = \cup_{x \in X} B(x, r)$ lorsque r varie de zéro à infini.
- *Filtrations associées aux ensembles de niveau* : Étant donné un espace topologique M et une fonction $f : M \rightarrow \mathbb{R}$, la famille $M_r = f^{-1}((-\infty, r])$, $r \in \mathbb{R}$ définit une filtration. On appelle les ensembles $M_r \subseteq M$ les *ensembles de sous-niveau* de f . On peut définir également les ensembles de super-niveau de f et sa filtration associé.
 Dans les cas où nous travaillons avec un complexe simplicial K , normalement la fonction est définie seulement sur son ensemble de sommets V . Nous pouvons étendre f à tout simplexe de K en prenant $f([v_0, \dots, v_k]) = \max_{0 \leq i \leq k} f(v_i)$ pour tout $\sigma = [v_0, \dots, v_k] \in K$. Ainsi, la famille de sous-complexes $K_r = \{\sigma \in K \mid f(\sigma) \leq r\}$ définit la filtration associée aux ensembles de sous-niveau de f .

Avec ces deux filtrations, dans des cas réels, même si T est infini, toutes les filtrations sont construites sur des nuages de points, qui sont des ensembles finis, donc elles sont aussi finies. Par conséquent, le complexe obtenu change seulement un numéro fini de fois, ce qui facilite son étude d'un point de vue algorithmique.

Nous exposons finalement un autre type de filtration sur les complexes simpliciales, facile de calculer et très pratique au niveau algorithmique, comme nous verrons toute de suite :

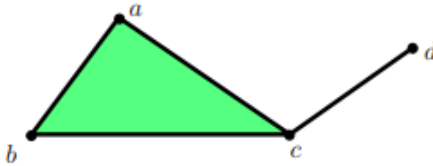
Définition 3.22. Une filtration de décomposition d'un complexe simplicial K est une suite de subcomplexes

$$\emptyset = K_0 \subset K_1 \subset K_2 \subset \dots \subset K_{m-1} \subset K_m = K$$

qui satisfait $K_i = K_{i-1} \cup \sigma_i$, où σ_i est un simplexe.

Intuitivement, dans une filtration de décomposition, nous ajoutons seulement un simplexe à chaque fois, et il faut que toutes ses faces appartiennent déjà au sous-complexe quand il est introduit.

Exemple 3.23. Avec le complexe simplicial de l'image suivant, une filtration de décomposition pourrait être $a, b, c, ab, ac, d, bc, abc, cd$.



Remarque 3.24. C'est facile de voir que toute filtration d'un complexe simplicial (y comprises les deux filtrations précédentes) peut être affinée à une filtration de décomposition : il faut seulement décomposer les nouveaux simplexes de K_i respect à K_{i-1} en sommets, arêtes, 2-faces,... et les ajouter un par un à chaque fois.

Cette dernière remarque ouvre tous les algorithmes typiques des filtrations de décomposition, comme ceux dans les sections suivantes, à toute filtration. C'est à cause de ce fait que désormais nous travaillerons plutôt avec ce type de filtrations.

3.3.3. Un algorithme pour calculer les nombres de Betti

Supposons un complexe simplicial K avec une filtration de décomposition. Dans cette situation, il existe un algorithme assez simple pour calculer les nombres de Betti associés à K , seulement en gardant une trace des nombres de Betti tout au long de la filtration ; en fait, l'algorithme calcule les nombres de Betti de chaque sous-complexe de la filtration, ce qui sera important dans l'homologie persistante.

Pour bien effectuer cette méthode, c'est indispensable de détecter quand le nouveau simplexe σ_i ajouté, de dimension k , appartient à quelque k -cycle ou non, ce qui motive la définition suivante :

Définition 3.25. *Si σ_i appartient à quelque k -cycle, on dit qu'il est un simplexe positif ; dans le cas contraire, c'est un simplexe négatif.*

Dans les sections suivantes, où plus d'algorithmes seront détaillés, nous expliquerons comment savoir si nous ajoutons un simplexe positif ou négatif. Pour l'instant, et pour expliquer l'algorithme, nous pouvons supposer que nous savons détecter quand σ_i est positif ou négatif :

Proposition 3.26. (Algorithme) : *Les nombres de Betti de K peuvent être calculés de manière inductive en faisant usage d'une filtration de décomposition.*

Démonstration. Évidemment, tous les nombres de Betti de $K_0 = \emptyset$ sont zéro.

Pour calculer les nombres de Betti de K_i , supposons que les nombres de Betti de K_{i-1} sont déjà calculés, et ajoutons le simplexe σ_i , de dimension k , pour obtenir K_i . Observons que, par définition de filtration de décomposition, σ_i ne peut pas faire partie du bord d'aucun $(k+1)$ -simplexe de K_i . Par conséquent, si σ_i est contenu dans un k -cycle de K_i (i.e. positif), ce cycle n'est pas le bord d'une $(k+1)$ -chaîne de K_i .

Il y a deux situations possibles :

Cas 1 : Si σ_i est positif et appartient à un k -cycle c de K_i , alors c ne peut pas être homologue à un cycle c' de K_{i-1} . En effet, dans ce cas $c + c'$ serait le bord d'une $(k+1)$ -chaîne d de K_i , et comme σ_i ne peut pas appartenir à c' (donc nous venons d'introduire cet nouveau simplexe à K_i), σ_i appartient à $c + c' = \partial d$, ce qui n'est pas possible comme nous avons déjà remarqué au début de la démonstration. Par conséquent, c crée une nouvelle classe d'homologie, qui est linéairement indépendant des classes générées par les cycles de K_{i-1} , donc $\beta_k(K_i) \geq \beta_k(K_{i-1}) + 1$.

Nous pouvons voir aussi que la dimension du k -ème groupe d'homologie ne peut pas augmenter plus que 1 : si c et c' sont deux k -cycles qui contiennent σ_i , $c + c'$ est un k -cycle de K_{i-1} , donc c' est inclus au sous-espace linéaire généré pour $Z_k(K_{i-1})$ et c . D'ici on a que $\dim Z_k(K_i) \leq \dim Z_k(K_{i-1}) + 1$ et, comme $B_k(K_{i-1}) = B_k(K_i)$, on a $\beta_k(K_i) \leq \beta_k(K_{i-1}) + 1$.

Il reste seulement pour montrer que $B_{k-1}(K_i) = B_{k-1}(K_{i-1})$, donc $H_{k-1}(K_i)$ est le seul autre group d'homologie de K_i qui peut changer en ajoutant σ_i , et clairement $Z_{k-1}(K_i) = Z_{k-1}(K_{i-1})$. Le résultat est une conséquence directe du fait que σ_i est positif, et il appartient donc à un k -cycle c de K_i : en effet, $0 = \partial c = \partial \sigma_i + \sum \partial(\text{autres } k\text{-simplexes de } K \text{ déjà ajoutés})$, et $\partial \sigma_i$ peut être écrit comme une combinaison linéaire de bords de k -chaînes de K_{i-1} .

Cas 2 : Si σ_i est négatif et n'appartient à aucun k -cycle de K_i , alors le $(k-1)$ -cycle $\partial \sigma_i$ n'est pas un bord à K_{i-1} . En effet, dans ce cas nous pourrions trouver une k -chaîne c' à K_{i-1} tel que $\partial c = \partial \sigma_i$, ou de façon équivalente, $\partial(c + \sigma_i) = 0$, ce qui implique que $c + \sigma_i$ est un k -cycle de K_i qui contient σ_i : contradiction. Par conséquent, comme le $(k-1)$ -cycle $\partial \sigma_i$, qui n'était pas un bord à K_{i-1} , devient un bord à K_i , nous avons $\beta_{k-1}(K_i) \leq \beta_{k-1}(K_{i-1}) - 1$. Avec un argument similaire

à celui de la fin du Cas 1, nous pouvons démontrer l'égalité.

Du fait que σ_i est négatif, c'est aussi évident que le groupe d'homologie $H_k(K_i)$ reste inchangé. \square

Nous voyons donc que, de quelque sorte, le processus se limite à trouver la différence entre les simplexes positifs et négatifs de la filtration ; les positifs créent de nouvelles k -classes d'homologie tandis que les négatifs effacent des $(k - 1)$ -classes. Voici un résumé de l'algorithme :

Algorithm 3: Calcul des nombres de Betti d'un complexe simplicial K

Input: Une filtration de décomposition de K , complexe simplicial d -dimensionnel avec m simplexes

$$\beta_0, \beta_1, \dots, \beta_d = 0$$

Output: Les nombres de Betti $\beta_0, \beta_1, \dots, \beta_d$ de K

1 **for** $i = 1$ jusqu'à m :

$k = \dim \sigma_i$

 Si σ_i est positif :

$$\beta_k = \beta_k + 1$$

 Si σ_i est négatif :

$$\beta_{k-1} = \beta_{k-1} - 1$$

3.3.4. Homologie persistante : définitions et algorithmes

Nous avons vu que l'algorithme précédant ne compute pas seulement les nombres de Betti d'un complexe simplicial, mais de tous les sous-complexes de la filtration (de décomposition). Intuitivement, l'objectif de l'homologie persistante est de garder une trace de toute cette information et enregistrer à quels moments chaque classe d'homologie est créée et détruite pendant le processus.

Avant d'expliquer les formalismes, montrons un petit exemple, en utilisant l'homologie singulière et la Proposition 3.20 :

Exemple 3.27. Soit $f : (0, 1) \rightarrow \mathbb{R}$ la fonction représentée dans l'image suivant :

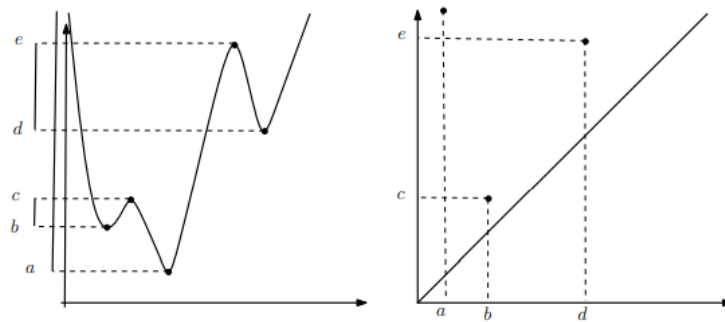


FIGURE 6 – Diagramme de persistance d'une fonction réelle, où seulement les composantes connexes (i.e. H_0) sont enregistrées.

Nous sommes intéressés à étudier l'évolution de la topologie de la filtration associé aux ensembles de sous-niveau de f , $f^{-1}((-\infty, t])$, à mesure que t augmente. La topologie de ces sous-ensembles change quand t atteint les valeurs critiques a, b, c, d et e :

Quand $t = a$, une nouvelle composante connexe apparaît, et pour $a \leq t \leq b$, $f^{-1}((-\infty, t))$ est un intervalle. Quand t atteint la valeur b , une deuxième composante connexe apparaît, et pour $b \leq t \leq c$, $f^{-1}((-\infty, t))$ a deux composantes connexes. Pour $t = c$, ces deux composantes connexes sont fusionnées : celle qui a été créée plus récemment, quand $t = b$, est unifiée à la première. Ainsi, on enregistre la paire (b, c) comme les temps de création et destruction de la composante ; cette paire est après représentée avec les coordonnées (b, c) au plan à droite. Intuitivement, le plus éloignée un point est de la diagonale, le plus relevant est la composante.

Si nous continuons à augmenter t , encore une nouvelle composante est créée à $t = d$, qui est finalement unifiée à la première quand t atteint la valeur e ; ainsi, un deuxième point est enregistré à droite, avec coordonnées (d, e) . La première valeur a ne peut pas être associée à aucune autre valeur finie, donc la composante connexe créée pour cette t ne meurt jamais ; par conséquent, elle est associée à $+\infty$.

À la fin, toutes ces paires peuvent être représentées comme une famille d'intervalles (*barcode*) ou comme un diagramme au plan, appelé *diagramme de persistance*. Pour des raisons qui deviendront claires plus tard, c'est aussi naturel d'ajouter la diagonale $\{y = x\}$ au diagramme.

Quand nous considérons des fonctions définies dans des espaces topologiques générales, atteindre certaines valeurs critiques peut changer ne pas seulement les composantes connexes des ensembles de sous-niveau, mais d'autres propriétés topologiques encodées dans les autres groupes d'homologie (i.e. les "trous" n -dimensionnels). De ce fait, il est aussi raisonnable de créer des paires de création/destruction pour chaque dimension.

Finalement, supposons une fonction g "proche" à f comme celle de l'image d'en bas. Nous pouvons observer que, même si g a plus de paires dans son diagramme de persistance, la majorité sont très proches à la diagonale, donc une durée de vie assez courte. En revanche, les paires associées à un intervalle plus long sont proches à celles de f . En d'autres termes, les propriétés topologiques qui ont une persistance élevée sont préservées, tandis que celles qui sont créées à cause de perturbations sur la fonction ont une persistance plus petite. Nous verrons que, en effet, deux fonctions "proches" ont toujours des diagrammes de persistance "proches". Cette notion de proximité est essentielle pour bien distinguer et traiter le bruit topologique dans nos données.

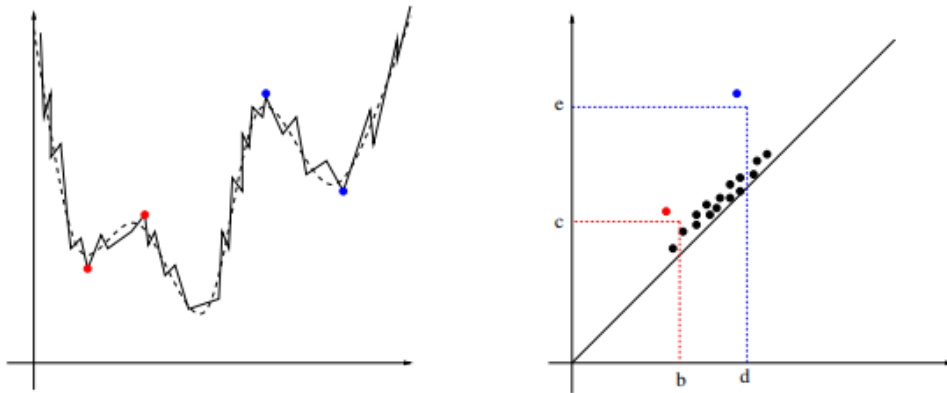


FIGURE 7 – Une approximation g de f , et les diagrammes de persistance respectifs.

Homologie persistante d'une filtration

Nous définissons d'abord la notion de persistance pour la filtration d'un complexe simplicial, qui a pour objectif étudier l'évolution de l'homologie des sous-complexes de la filtration.

Soit K un complexe simplicial d -dimensionnel avec une filtration de décomposition

$$\emptyset = K^0 \subset K^1 \subset \dots \subset K^m = K.$$

Pour tout $0 \leq n \leq m$, on dénote C_k^n l'ensemble de k -chaînes de K^n (avec des coefficients à \mathbb{Z}_2). Notons que, puisque $\partial^2 = 0$ et nous avons une filtration de décomposition, la restriction de la fonction bord sur C_k^n a toujours sa image contenue dans C_{k-1}^{n-1} . Dénnotons Z_k^n et B_k^n les cycles et bords k -dimensionnels de K^n respectivement, donc le k -ème groupe d'homologie de K^n est $H_k^n = Z_k^n / B_k^n$. Avec ces notations nous avons aussi

$$Z_k^0 \subset Z_k^1 \subset \dots \subset Z_k^m = Z_k(K), \quad B_k^0 \subset B_k^1 \subset \dots \subset B_k^m = B_k(K).$$

Définition 3.28. (Nombres de Betti persistants) : Pour $p \in \{0, \dots, m\}$ et $l \in \{0, \dots, m-p\}$, le k -ème nombre de Betti persistant de K^l est la dimension de l'espace vectoriel $H_k^{l,p} = Z_k^l / (B_k^{l+p} \cap Z_k^l)$.

Intuitivement, le k -ème nombre de Betti persistant de K^l représente le numéro de classes d'homologie indépendantes de k -cycles dans K^l qui ne sont pas de bords dans K^{l+p} ; par conséquent, de manière informelle, sa durée de vie est supérieure à p .

Nous avons vu, dans la section précédente, qu'une classe d'homologie est créée quand un simplexe positif est ajouté à la filtration, et qu'une classe d'homologie est détruite quand le simplexe est négatif. L'homologie persistante offre un cadre théorique pour associer des simplexes positifs et négatifs : quand un simplexe positif est ajouté à la filtration, il crée une classe d'homologie, qui disparaît quand on ajoute son simplexe négatif associé (s'il existe).

Nous sommes intéressés à trouver un algorithme pour calculer ces nombres de Betti persistantes. Pour l'obtenir, il faut mieux caractériser les classes d'homologie créées à chaque fois que nous ajoutons un simplexe positif à la filtration. Rappelons que, dans la définition 3.25 de simplexe positif et négatif, il faut seulement que σ_i appartient à un k -cycle, mais en général ce k -cycle n'est pas unique. Heureusement, pour chaque k -simplexe positif σ_i que nous ajoutons à la filtration, il y a un k -cycle associé "minimal", qui facilitera, à la fois, le calcul des nombres de Betti persistantes :

Lemme 3.29. Soit σ_i un k -simplexe positif ajouté à la filtration de K au pas i . Or, il n'y a qu'un seul k -cycle c qui n'est pas un bord dans K^i , qui contient σ_i et qui ne contient aucun autre k -simplexe positif.

Démonstration. Nous travaillons par induction sur l'ordre avec lequel les k -simplexes positifs sont ajoutés à la filtration. Pour le premier k -simplexe positif σ ajouté, ce k -cycle c existe par définition, est-il est nécessairement unique parce que s'il y en avait un autre c' de différent, $c + c' \neq 0$, qui ne contient pas σ , serait aussi un k -cycle et ses éléments seraient des k -simplexes positifs, contradiction.

Supposons maintenant que le résultat est vrai pour tous les k -simplexes positifs déjà ajoutés, et ajoutons σ_i . Comme σ_i est positif, il existe un k -cycle d qui n'est pas un bord dans K_i et qui contient σ_i . Soit σ_{i_j} , $j = 1, \dots, p$ les k -simplexes positifs différents de σ_i contenus dans d , et c_{i_j} ses

k -cycles respectifs associés, qui existent par hypothèse d'induction. Alors

$$c = d + c_{i_1} + \dots + c_{i_p} \quad (5)$$

est un k -cycle où σ_i est le seul k -simplexe positif. Du fait que σ_i est le dernier simplexe ajouté à K_i , il n'existe aucun $(k+1)$ -simplexe dans K_i où σ_i est une face. Par conséquent, c n'est pas un bord, et l'existence est démontrée.

Pour démontrer l'unicité de c , supposons deux k -cycles \hat{d}_1 et \hat{d}_2 qui ne sont pas un bord dans K_i et qui contient σ_i , et répétons la construction précédente pour obtenir $\hat{c}_1 \neq \hat{c}_2$. Alors $\hat{c}_1 - \hat{c}_2 \neq 0$ est un k -cycle qui ne contient aucun k -simplexe positif, et nous pourrions toujours le combiner avec un c_i antérieur pour obtenir un k -cycle avec les mêmes propriétés du lemme, ce qui contredit sa unicité et l'hypothèse d'induction. Donc, nous concluons que $\hat{c}_1 - \hat{c}_2 = 0$, et l'unicité est démontrée. \square

Bases des groupes d'homologie persistante et paires de persistance

Proposition 3.30. *Les k -cycles associés aux k -simplexes positifs décrits au Lemme 3.29 peuvent être utilisés pour calculer une base des différents groupes d'homologie k -dimensionnels de tous les sous-complexes de la filtration.*

Démonstration. Évidemment, au début toutes les bases de $H_n^0(K) = H_k(K_0)$ sont vides pour $k = 0, \dots, d$. Les bases des successives H_k^i sont construites de manière inductive :

- Supposons que nous avons déjà une base de H_k^{i-1} et que σ_i est positif et de dimension k . Si nous ajoutons à notre base la classe d'homologie définie par le cycle c_i associé à σ_i , nous obtenons une base de H_k^i grâce au Lemme précédent. En effet, du fait que c_i est une somme de σ_i et k -simplexes négatifs, il n'est homologue à aucune combinaison linéaire des cycles qui définissent la base de H_k^{i-1} . Du fait que $\dim H_k^i = \dim H_k^{i-1} + 1$, vu dans la Proposition 3.26, nous obtenons une base de H_k^i .

- Supposons maintenant qu'une base de H_k^{j-1} est déjà construite et que le simplexe σ_j est négatif et de dimension $k+1$. Soient c_{i_1}, \dots, c_{i_p} les k -cycles associés aux simplexes positifs déjà ajoutés, qui définissent les classes d'homologie qui forment notre base de H_k^{j-1} . Comme nous l'avons déjà expliqué, le bord $\partial\sigma_j$ est un k -cycle de K_{j-1} qui n'est pas un bord dans K_{j-1} , mais qui devient un bord dans K_j . Par conséquent, il peut être écrit de manière unique comme

$$\partial\sigma_j = \sum_{k=1}^p \varepsilon_k c_{i_k} + b, \quad (6)$$

où $\varepsilon_k \in \{0, 1\}$ et b est un bord. Soit $l(j) = \max\{i_k \mid \varepsilon_k = 1\}$.

Claim : Si on enlève la classe d'homologie associé à $c_{l(j)}$ de la base de H_k^{j-1} , on obtient une base de H_k^j .

En effet, comme $\dim H_k^{j-1} = \dim H_k^j + 1$ par la Proposition 3.26, il suffit de montrer que $c_{l(j)}$ est une combinaison linéaire d'un bord avec les autres cycles c_{i_k} dans Z_k^j . L'équation (6) antérieure montre une telle décomposition, ce qui finit la démonstration. \square

Définition 3.31. (Paires de persistance) *Les paires de simplexes $(\sigma_{l(j)}, \sigma_j)$ s'appellent les paires de persistance de la filtration de K .*

Intuitivement, la classe d'homologie créée pour $\sigma_{l(j)}$ dans $K_{l(j)}$ est détruite pour σ_j dans K_j . La persistance de cette paire est $j - l(j)$.

Le problème avec l'algorithme antérieur est de déterminer $l(j)$. N'oublions pas non plus qu'il faut aussi encore expliquer, en vue de l'algorithme du calcul des nombres de Betti de la Proposition 3.26, comment détecter si un nouveau k -simplexe est positif ou négatif. Toutes ces questions peuvent être répondues à la fois en étudiant la filtration sous une forme matricielle. D'idée derrière cette matrice est d'encoder le résultat de la fonction bord sur tous les simplexes de la filtration, ordonnés. Avec cette matrice, nous pouvons réélaborer la proposition antérieur pour obtenir un algorithme effectif pour calculer les paires de persistance, ce qui permet trouver les nombres de Betti persistantes. Elle proportionne aussi une manière de détecter quand le k -simplexe ajouté est positif ou négatif.

Soit K un complexe simplicial avec une filtration de décomposition. Soit $M = (m_{ij})_{i,j=1,\dots,m}$ la matrice associé au pas m de la filtration, où

$$m_{i,j} = 1 \text{ si } \sigma_i \text{ est une face de } \sigma_j, \text{ et } m_{i,j} = 0 \text{ autrement.}$$

Cette matrice augmente "à droite et en bas" à mesure que la filtration avance, et elle est toujours triangulaire supérieure puisqu'on a une filtration de décomposition.

Pour une colonne C_j , soit $l(j) = \max\{i \mid m_{i,j} = 1\}$, et non-assigné si la colonne contient seulement des zéros. Nous pouvons alors considérer l'algorithme suivant :

Algorithm 4: *Calcul des paires de persistance, version matricielle*

Input: Une filtration de décomposition de K , le sous-complexe K^m (qui contient m simplexes) et la matrice M associé au pas m

1 **for** $j = 0$ jusqu'à m :
 while qu'il existe $j' < j$ avec $l(j') == l(j)$
 $C_j = C_j + C_{j'} \pmod{2}$

Output: Les paires $(l(j), j)$

Proposition 3.32. *L'algorithme antérieur calcule les paires de persistance de la filtration de décomposition de K jusqu'au pas m , ainsi comme quels simplexes sont positifs et quels sont négatifs.*

Démonstration. Remarquons que, à chaque pas de l'algorithme, la colonne C_j représente une chaîne de la forme $\partial(\sigma_j + \sum_{i < j} \varepsilon_i \sigma_i)$, où $\varepsilon_i \in \{0, 1\}$.

- Si à la fin de l'algorithme j satisfait que $l(j)$ est assigné, alors $\sigma_{l(j)}$ est un simplexe positif. En effet, on a $\partial(\sigma_j + \sum_{i < j} \varepsilon_i \sigma_i) = \sigma_{l(j)} + \sum_{p < l(j)} \lambda_p \sigma_p$, où $\lambda_p \in \{0, 1\}$. Du fait que $\partial^2 = 0$, on a que $\sigma_{l(j)}$ appartient à un cycle et il est donc positif.

- Si à la fin de l'algorithme C_j contient seulement des zéros, σ_j est positif. Effectivement, $\partial(\sigma_j + \sum_{i < j} \varepsilon_i \sigma_i) = 0$, et σ_j appartient donc à un cycle.

-Finalement, si à la fin de l'algorithme la colonne C_j contient des termes non nuls, $(\sigma_{l(j)}, \sigma_j)$ est une paire de persistance, et σ_j est donc négatif. En effet, le bord de σ_j peut alors être écrit de la forme $\sigma_{l(j)} + \sum_{p < l(j)} \lambda_p \sigma_p + \partial(\sum_{i < j} \varepsilon_i \sigma_i)$. Or, $\sigma_{l(j)}$ est positif, donc il a créé une classe d'homologie au moment $l(j)$, et il reste non associé. Du fait que le dernier index non nul d'une colonne est unique et par la Proposition 3.30, on peut déduire que $(\sigma_{l(j)}, \sigma_j)$ est une paire de persistance. \square

3.3.5. Diagrammes de persistance et stabilité

Rappelons que, d'après le Remarque 3.24, tous les algorithmes que nous venons d'expliquer sont aussi applicables à des filtrations pas nécessairement de décomposition. En effet, nous pouvons toujours faire une affinage, et après prendre les coefficients de la filtration original. Également, avec des algorithmes qui facilitent les calculs ou pas, les notions de l'homologie persistante introduites jusqu'à ici, notamment les paires de persistance, peuvent aussi être appliquées avec des filtrations d'espaces topologiques plus générales, en prenant l'homologie singulière. Nous avons déjà observé ce fait informellement dans l'exemple 3.26, et nous en verrons aussi dans cette section.

Dans tous les cas, beaucoup des informations de l'homologie persistante, notamment la durée de vie des différentes classes d'homologie, peuvent être facilement représentées en forme de diagramme :

Avec k fixée, soient $(\sigma_{l(j)}, \sigma_j)$ les paires de persistance (calculées avec les algorithmes précédentes, par exemple), où $\sigma_{l(j)}$ et σ_j ont dimension k et $k+1$ respectivement. Nous représentons chacune de ces paires dans \mathbb{R}^2 avec le point de coordonnées $(l(j), j)$; pour les simplexes positifs σ_i qui restent non associés, nous créons les points $(i, +\infty)$.

Définition 3.33. *Nous appelons la représentation de cet ensemble de points dans \mathbb{R}^2 avec la diagonale $\Delta = \{x = y\}$ le diagramme de persistance k -dimensionnel de la filtration.*

Plus généralement, si la filtration est indexée par une suite croissante de numéros réels, comme par exemple avec les filtrations introduites dans la section 3.3.2,

$$\emptyset = K_{\alpha_0} \subset K_{\alpha_1} \subset \dots \subset K_{\alpha_{n-1}} \subset K_{\alpha_n} = K,$$

une paire de persistance de simplexes (σ_i, σ_j) est représentée par le point (α_i, α_j) , avec les indices d'apparition et mort ; si le simplexe σ_i reste non associé, nous ajoutons le point $(\alpha_i, +\infty)$.

Le même type de points peuvent être créés pour toute filtration d'un espace topologique et avec l'homologie singulière, où la coordonnée x enregistre l'apparition d'une classe d'homologie et la coordonnée y sa mort. En tout cas, dans ces cas plus générales, il faut faire attention au fait que plusieurs paires peuvent être associées au même point dans le plan. Donc, dans ces diagrammes de persistance il faut aussi considérer une multiplicité pour chaque point. Par convention, les points de la diagonale ont tous multiplicité infinie. Désormais, nous considérerons aussi une multiplicité pour chaque point dans la définition de diagramme de persistance.

Nous pouvons définir une distance entre diagrammes de persistance pour mieux les comparer :

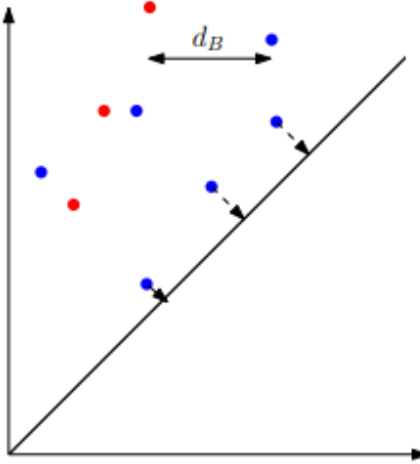
Définition 3.34. (Distance "bottleneck") *Soient D_1 et D_2 deux diagrammes de persistance. La distance goulot ("bottleneck" en anglais) entre D_1 et D_2 est définie comme*

$$d_B(D_1, D_2) = \inf_{\gamma} \sup_{p \in D_1} \|p - \gamma(p)\|_{\infty},$$

où γ est l'ensemble de bijections entre les points de D_1 et D_2 ; on prend m copies disjointes si un point a multiplicité $m > 1$. Par convention, si $p = (x_p, +\infty)$, $q = (x_q, +\infty)$, $\|p - q\|_{\infty} = |x_p - x_q|$.

Remarque 3.35. C'est précisément cette distance qui motive ajouter la diagonale aux diagrammes de persistance : elle permet de comparer des diagrammes qui n'ont pas le même nombre de points dehors la diagonale en les associant avec des points de la diagonale.

Nous omettrons la démonstration qu'il s'agit vraiment d'une distance. Dans l'image suivante il apparaît une représentation de la distance bottleneck entre deux diagrammes de persistance :



Stabilité persistante des fonctions

Dans cette dernière partie de la section, inévitablement plus avancée et sans toutes les démonstrations, nous allons lier la proximité de deux fonctions f et g à la proximité de ses diagrammes de persistance. Cette relation est fondamentale pour justifier la convenance des diagrammes de persistance associées à une fonction, ainsi comme pour identifier ses propriétés "proéminentes" et les distinguer du "bruit topologique", i.e. des caractéristiques topologiques de courte durée créées pour de petites perturbations. Ces résultats deviennent aussi importants pour justifier la procédure de l'algorithme ToMATo du prochain chapitre, où nous travaillons avec une estimation \hat{f} d'une densité f , et nous regardons son diagramme de persistance (0-dimensionnel).

Expliquons brièvement la situation : soit $f : X \rightarrow \mathbb{R}$ une fonction réelle continue définie sur un espace topologique X . Nous voulons étudier le diagramme de persistance k -dimensionnel associé à ses ensembles de sous-niveau $\{F_\alpha\}_{\alpha \in \mathbb{R}}$, où $F_\alpha = f^{-1}((-\infty, \alpha])$, avec l'homologie singulière. Une propriété fondamentale de l'homologie singulière est que toute application continue entre espaces topologiques $h : X \rightarrow Y$ induit un morphisme entre les respectives (k -èmes) groupes d'homologie, $h_* : H_*(X) \rightarrow H_*(Y)$. Plus concrètement, la (k -ème) homologie singulière est un foncteur (covariant) de la catégorie des espaces topologiques à la catégorie des groupes. Donc, il est toujours vrai que $(id_X)_* = id_{H_*(X)}$ et $(h \circ g)_* = h_* \circ g_*$.

Dans notre cas, on peut étudier les applications induites par les inclusions $F_a \subset F_b$ quand $a < b$, $f_a^b : H_*(F_a) \rightarrow H_*(F_b)$. Ces groupes et morphismes encodent toute l'information de l'homologie persistante : quelques références appellent *groupes d'homologie persistante* aux groupes $\text{Im } f_a^b$, qui suivent la même idée que les groupes qui apparaissent dans la Définition 3.28, dans le cas simplicial. Intuitivement, avec deux fonctions "proches" (avec la distance $\|f - g\|_\infty = \sup_{x \in X} |f(x) - g(x)|$), les moments de création et de mort de certaines caractéristiques topologiques (gardés dans les groupes d'homologie, et représentés graphiquement dans les diagrammes de persistance respectifs $D(f)$ et $D(g)$) devraient être similaires. Cette idée est formalisée dans le théorème suivant :

Théorème 3.36. (Théorème de la stabilité bottleneck des diagrammes de persistance) :
Soit X un espace topologique triangulable avec des fonctions tame $f, g : X \rightarrow \mathbb{R}$. Alors, les diagrammes de persistance satisfont $d_B(D(f), D(g)) < \|f - g\|_\infty$.

Nous démontrerons le théorème à la fin de la section, mais sans tous les pas intermédiaires. Avant, plusieurs définitions et résultats sont nécessaires :

Définition 3.37. Soit X un espace topologique, $f : X \rightarrow \mathbb{R}$. Une valeur critique homologique de f est un numéro réel a pour lequel il existe un entier k tel que, $\forall \varepsilon > 0$, l'application induite $f_{a-\varepsilon}^{a+\varepsilon} : H_k(F_{a-\varepsilon}) \rightarrow H_k(F_{a+\varepsilon})$ n'est pas un isomorphisme.

Lemme 3.38. (Lemme de la valeur critique) : Si $[a, b]$ ne contient aucune valeur critique homologique de f , f_a^b est un isomorphisme pour tout $k \in \mathbb{Z}$.

Démonstration. Soit $m = \frac{x+y}{2}$, or $f_a^b = f_m^b \circ f_a^m$. Si f_a^b n'est pas un isomorphisme, au moins une des deux fonctions antérieures ne l'est pas non plus. Ainsi, on peut construire de manière inductive une suite d'intervalles fermés décroissants, l'intersection desquels est une valeur critique homologique dans $[x, y]$, ce qui est une contradiction. \square

Le lemme antérieur formalise l'idée que c'est seulement quand on atteint des valeurs critiques homologiques que nouvelles caractéristiques topologiques sont créées ou détruites. Notre résultat requiert une condition de finitude sur notre fonction, qui se satisfait dans la plupart des cas d'étude :

Définition 3.39. Une fonction $f : X \rightarrow \mathbb{R}$ s'appelle tame si elle a un numéro fini de valeurs critiques homologiques et tous les groupes d'homologie $H_k(F_a)$ ont dimension finie $\forall k \in \mathbb{Z}$, $a \in \mathbb{R}$.

Rappelons maintenant la définition de la distance de Hausdorff, très habituelle dans la TDA, pour des nuages de points :

Définition 3.40. Soient X et Y des ensembles de points (avec multiplicité). Alors la distance Hausdorff est $d_H(X, Y) = \max\{\sup_x \inf_y \|x - y\|_\infty, \sup_y \inf_x \|y - x\|_\infty\}$.

Un résultat important, mais sans démonstration, que nous utiliserons plus tard est celui-ci :

Proposition 3.41. Soit X un espace topologique triangulable avec des fonctions tame $f, g : X \rightarrow \mathbb{R}$. Alors $d_H(D(f), D(g)) < \|f - g\|_\infty$.

Remarque 3.42. La distance Hausdorff entre deux diagrammes de persistance n'excède jamais la distance bottleneck, car elle ne fait pas attention aux multiplicités, ou regroupements de points. Le résultat pour la distance bottleneck est plus fort, et clé, pour quelques applications.

Voyons avant le résultat du Théorème 3.36 pour un cas spécial, et plus simple. Nous nous en servirons plus tard pour prouver le cas général.

Étant donnée une fonction tame $f : X \rightarrow \mathbb{R}$, soit δ_f la distance minimale entre deux points dehors la diagonale, ou entre un point dehors la diagonale et un autre dedans :

$$\delta_f = \min\{\|p - q\|_\infty \mid D(f) \setminus \Delta \ni p \neq q \in D(f)\}.$$

Si on dessine des carrés de rayon $\varepsilon = \delta_f/2$ centrés sur les points de $D(f)$, on obtient une diagonale plus "grosse", et une collection finie de carrés disjoints entre eux et avec la diagonale.

Définition 3.43. Une autre fonction tame $g : X \rightarrow \mathbb{R}$ est appelée très proche à f si $\|f - g\|_\infty < \frac{\delta_f}{2}$.

Ici un autre lemme nécessaire mais sans démonstration, de nature plus technique :

Lemme 3.44. Soient $f, g : X \rightarrow \mathbb{R}$ des fonctions tames, g très proche à f . Soient $p \in D(f) \setminus \Delta$, m_p sa multiplicité et \square_ε le carré centré en p de rayon $\varepsilon = \|f - g\|_\infty$. Alors $|D(g) \cap \square_\varepsilon| = m_p$.

Lemme 3.45. (Lemme de la bijection) : Soit X un espace topologique triangulable, $f, g : X \rightarrow \mathbb{R}$ des fonctions tames, g très proche à f . Alors $d_B(D(f), D(g)) \leq \|f - g\|_\infty$.

Démonstration. Soient $p \in D(f) \setminus \Delta$ et \square_ε le carré centré en p de rayon $\varepsilon = \|f - g\|_\infty$, comme avant. Du lemme précédant, tous les points de $D(g) \cap \square_\varepsilon$ peuvent être associés à p . Nous pouvons suivre cette procédure pour tous les points dehors la diagonale de $D(f)$. Après, les seuls points de $D(g)$ qui restent sans image ont une distance supérieure à ε de $D(f) \setminus \Delta$. Du fait que $d_H(D(f), D(g)) \leq \varepsilon$ (Proposition 3.41), ces points de $D(g)$ sont au plus à distance ε de la diagonale. Si nous leur associons respectivement les points les plus proches de la diagonale, nous obtenons une bijection entre $D(f)$ et $D(g)$ (rappelons que les points de la diagonale ont multiplicité infinie). Cette bijection déplace les points au plus ε , ce qui finit la démonstration. \square

Rappelons qu'un espace topologique est appelé *triangulable* s'il existe un complexe simplicial fini avec une réalisation géométrique homéomorphe. Nous pouvons montrer notre théorème dans toute sa généralité en faisant des approximations successives de notre fonction originelle $f : X \rightarrow \mathbb{R}$ par des fonctions linéaires par morceaux définies sur un complexe simplicial :

Définition 3.46. Soit K un complexe simplicial avec des valeurs réels spécifiées sur chaque sommet x_i , $f(x_i)$. Sa fonction linéaire par morceaux (LPM) associée est $\hat{f} : K \rightarrow \mathbb{R}$, $\hat{f}(x) = \sum_i b_i(x) f(x_i)$, où $b_i(x)$ sont les coordonnées barycentriques de x . On obtient une fonction linéaire sur chaque simplexe du complexe.

Remarquons que, à cause de sa nature finie et linéaire, une fonction LPM sur un complexe simplicial fini est toujours tame. Ce fait permet de démontrer le Théorème 3.36 pour deux fonctions LPM \hat{f}, \hat{g} définies sur un complexe simplicial K fini. Avant, une dernière définition :

Définition 3.47. Une combinaison convexe de \hat{f} et \hat{g} est une fonction du type $h_\lambda = (1 - \lambda)\hat{f} + \lambda\hat{g}$, avec $\lambda \in [0, 1]$. Cette famille de combinaisons convexes entre les deux fonctions, où $h_0 = \hat{f}$ et $h_1 = \hat{g}$, s'appelle interpolation linéaire de \hat{f} à \hat{g} .

Lemme 3.48. (Lemme d'interpolation) : Soient \hat{f}, \hat{g} deux fonctions LPM définies sur un complexe simplicial K fini. Alors $d_B(D(\hat{f}), D(\hat{g})) \leq \|\hat{f} - \hat{g}\|_\infty$

Démonstration. L'idée de base de la démonstration est de décomposer l'interpolation linéaire de \hat{f} à \hat{g} en petites sections pour utiliser le Lemme de la bijection, et ainsi obtenir une bijection dans chaque section.

Soit $\varepsilon = \|\hat{f} - \hat{g}\|_\infty$, et observons que, pour tout $\lambda \in [0, 1]$, h_λ est tame (car elle est aussi une fonction LPM) et $\delta(\lambda) = \delta_{h_\lambda}$ est strictement positif quand au moins \hat{f} ou \hat{g} ont un point dehors la diagonale (sinon, l'inégalité du lemme est triviale). Donc, la famille $C = \{J_\lambda\}_{\lambda \in \mathbb{Q} \cap [0, 1]}$ d'intervalles ouverts $J_\lambda = (\lambda - \frac{\delta(\lambda)}{4\varepsilon}, \lambda + \frac{\delta(\lambda)}{4\varepsilon})$ forme un recouvrement ouvert de l'intervalle $[0, 1]$. Prenons un sous-recouvrement fini C' de C , qui existe pour être $[0, 1]$ compact, et minimal. Soient $\lambda_1 < \dots < \lambda_n$ les points médians des intervalles de C' . Du fait que C' est minimale, deux intervalles consécutifs J_{λ_i} et $J_{\lambda_{i+1}}$ ont toujours intersection non-vide, et

$$\lambda_{i+1} - \lambda_i \leq \frac{\delta(\lambda_i) + \delta(\lambda_{i+1})}{4\varepsilon} \leq \frac{\max\{\delta(\lambda_i), \delta(\lambda_{i+1})\}}{2\varepsilon}$$

Par définition de ε , on a aussi $\|h_{\lambda_i} - h_{\lambda_{i+1}}\|_\infty = \|(\lambda_{i+1} - \lambda_i)(\hat{g} - \hat{f})\|_\infty = \varepsilon(\lambda_{i+1} - \lambda_i)$. Par conséquent,

$$\|h_{\lambda_i} - h_{\lambda_{i+1}}\|_\infty \leq \frac{\max\{\delta(\lambda_i), \delta(\lambda_{i+1})\}}{2},$$

ce qui implique que h_{λ_i} est très proche à $h_{\lambda_{i+1}}$, ou à l'inverse. Nous pouvons alors appliquer le Lemme de la bijection, qui dit que $d_B(D(h_{\lambda_{i+1}}), D(h_{\lambda_i})) \leq \|h_{\lambda_{i+1}} - h_{\lambda_i}\|_\infty$ pour $1 \leq i \leq n-1$. Observons que, si nous ajoutons $\lambda_0 = 0$ et $\lambda_{n+1} = 1$ (donc $h_0 = \hat{f}$ et $h_1 = \hat{g}$), ces derniers raisonnements sont encore vraies, car 0 et 1 font aussi partie du recouvrement, et h_λ varie continuellement avec λ . Donc \hat{f} est très proche à h_{λ_1} (ou à l'inverse), et \hat{g} est très proche à h_{λ_n} (ou à l'inverse).

Maintenant, l'inégalité triangulaire donne

$$d_B(D(\hat{f}), D(\hat{g})) \leq \sum_{i=0}^n d_B(D(h_{\lambda_i}), D(h_{\lambda_{i+1}})) \leq \sum_{i=0}^n \|h_{\lambda_i} - h_{\lambda_{i+1}}\|_\infty.$$

Du fait que les h_{λ_i} forment une interpolation linéaire de \hat{f} à \hat{g} et leurs valeurs changent linéairement entre les deux, la dernière somme est égale à $\|\hat{f} - \hat{g}\|_\infty$, ce qui finit la démonstration. \square

Avec ce dernier résultat, nous pouvons démontrer le Théorème 3.36 :

Théorème de la stabilité bottleneck des diagrammes de persistance : *Soit X un espace topologique triangulable avec des fonctions tame $f, g : X \rightarrow \mathbb{R}$. Alors, les diagrammes de persistance satisfont $d_B(D(f), D(g)) < \|f - g\|_\infty$.*

Démonstration. (du Théorème 3.36 :) Par définition de triangulabilité, il existe un complexe simplicial fini L et un homéomorphisme $\phi : L \rightarrow X$. Notons que, du fait que ϕ est un homéomorphisme, $\phi^{-1}(f^{-1}((-\infty, a])) \cong f^{-1}((-\infty, a]) \forall a \in \mathbb{R}$, et les groupes d'homologie singulière sont aussi tous isomorphes à cause de sa fonctorialité. Par conséquent, le diagramme de persistance reste non altéré par ce changement de variables : $f \circ \phi : L \rightarrow \mathbb{R}$ est aussi tame et a le même diagramme de persistance que f .

Soit $\delta > 0$ suffisamment petit. Du fait que f et g sont continues et L est compact, il existe une sous-division K de L telle que

$$|f \circ \phi(x) - f \circ \phi(y)| \leq \delta, \quad |g \circ \phi(x) - g \circ \phi(y)| \leq \delta \quad (7)$$

pour x, y dans le même simplexe de K .

Soient $\hat{f}, \hat{g} : \text{Sd}K \rightarrow \mathbb{R}$ les fonctions linéaires par morceaux qu'on obtient à partir de $f \circ \phi$ et $g \circ \phi$ sur les sommets de $\text{Sd}K$, où $\text{Sd}K$ dénote la sous-division barycentrique de K . Par construction de K , ces fonctions satisfont $\|\hat{f} - f \circ \phi\|_\infty \leq \delta$ et $\|\hat{g} - g \circ \phi\|_\infty \leq \delta$. En faisant usage du Lemme d'Interpolation, le fait que \hat{f} et \hat{g} diffèrent au maximum δ de $f \circ \phi$ et $g \circ \phi$ respectivement, et $\|f - g\|_\infty = \|f \circ \phi - g \circ \phi\|_\infty$, on obtient

$$d_B(D(\hat{f}), D(\hat{g})) \leq \|\hat{f} - \hat{g}\|_\infty \leq \|f \circ \phi - g \circ \phi\|_\infty + 2\delta = \|f - g\|_\infty + 2\delta. \quad (8)$$

Si nous supposons de plus que $\delta < \frac{\delta_f}{2}$, nous obtenons une bijection du Lemme de la Bijection. Du fait que le changement de variables n'affecte pas le diagramme de persistance, on a

$$d_B(D(f), D(\hat{f})) = d_B(D(f \circ \phi), D(\hat{f})) \leq \delta. \quad (9)$$

Si nous supposons pareillement que $\delta < \frac{\delta_g}{2}$, l'inégalité triangulaire appliqué plusieurs fois avec (8) et (9) donne

$$d_B(D(f), D(g)) \leq \|f - g\|_\infty + 4\delta,$$

ce qui montre le résultat, donc δ peut être arbitrairement petit. \square

4. L'algorithme ToMATo

4.1. Introduction

L'exposition et exploration que nous ferons maintenant de l'algorithme ToMATo (*Topological Mode Analysis Tool*) et son implementation dans la librairie GUDHI constituent la partie la plus innovante de notre travail. Cette méthode, récemment conçue, se situe dans les techniques de clustering, donc dans l'apprentissage non supervisé. Bien que le fonctionnement ne soit pas spécialement complexe, il se base sur des idées de l'analyse topologique de données exposées antérieurement qu'il faut bien comprendre, notamment les complexes simpliciales ("graphes" désormais) et les diagrammes de persistance.

Un autre principe important de notre algorithme est qu'il est sensé fonctionner avec des sous-variétés de \mathbb{R}^d (ou variétés riemanniennes en général), indépendamment de sa "forme". Cela est remarquable puisqu'une bonne partie des algorithmes de clustering existants (par exemple, la méthode K -means, déjà exposée) ne parviennent pas à identifier les clusters lorsque ces derniers s'éloignent d'une structure convexe. Par exemple, certains algorithmes sont incapables de bien regrouper un ensemble de données échantillonnées à partir de deux anneaux concentriques dans \mathbb{R}^2 . Du ce fait, dans la section 4.1.1., nous exposerons les constructions et les arguments en prenant une variété riemannienne X , le cas le plus général. Cependant, dans la pratique nos données sont presque toujours dans \mathbb{R}^d , et seulement dans certains cas particuliers ils présentent une forme clairement semblable à une sous-variété de \mathbb{R}^d .

En nous appuyant sur la classification des techniques de clustering faite au début de la section "Algorithmes de clustering combinatoires", au deuxième chapitre, on pourrait affirmer que l'algorithme ToMATo combine une partie "mode-seeking" et une partie de nature plus combinatoire. En plus de cela, son innovation principale est que, pour guider la fusion des différents mini-clusters tout au long de la méthode, il utilise la notion de "persistance topologique", introduite au chapitre précédent. En plus d'étiqueter les données dans de différents groupes, l'algorithme produit aussi un diagramme de persistance, qui permet de choisir des paramètres précis afin d'obtenir le nombre de clusters souhaité.

4.1.1. L'intuition derrière l'algorithme : le cas continu

L'idée de base de la méthode est que, si les données sont obtenues en suivant une fonction de densité f , les clusters les plus logiques sont ces régions où la fonction fait des "bosses significatives". C'est dans ces dernières où les points seront plus probablement situés et regroupés.

Soit X une variété riemannienne de dimension m et f une fonction $f : X \rightarrow \mathbb{R}$ \mathcal{C}^2 -continue sans points critiques dégénérés. Supposons aussi que f a un nombre fini de points critiques. La *région ascendante* d'un point critique m est le sous-ensemble de points $A(m) \subseteq X$ qui parviennent finalement à m en suivant le flux induit pour le champ de vecteurs gradient de f . On appelle m la *racine* de $x \in A(m)$.

On peut démontrer que les régions ascendantes des pics de f forment des sous-ensembles de X disjoints et homéomorphes à \mathbb{R}^m . De plus, si f est bornée et propre, les régions ascendantes de ces pics couvrent X sauf un sous-ensemble de mesure de Lebesgue zéro. Il est donc logique d'utiliser ces régions pour découper X p.p. en régions d'influence.

Considérons maintenant la famille de sous-espaces $\{F_\alpha\}_{\alpha \in \mathbb{R}}$, où $F_\alpha = f^{-1}([\alpha, +\infty))$ et α varie de $+\infty$ à $-\infty$ (i.e. la filtration de X associée aux ensembles de super-niveau de f). Pour $\alpha \in \mathbb{R}$

et $x \in F_\alpha$, appelons $C(x, \alpha) \subseteq F_\alpha$ la composante connexe par arcs de F_α contenant x . Selon la Théorie de Morse, lorsqu'un maximum local m_p de f entre dans la filtration au moment $\alpha = b(m_p)$ (b de "birth"), une nouvelle composante connexe par arcs $C(m_p, \alpha)$ apparaît dans F_α . Puis, cette dernière cesse d'être indépendante quand elle se connecte avec une autre composante générée pour un pic m_q plus haut, pour quelque autre $\alpha = d(m_p)$ (d de "death"). Dans ce cas, on nomme m_q la racine de m_p , et on écrit $m_q = r(m_p)$. Dans le diagramme de persistance 0-dimensionnel de f , la durée de vie de m_p comme racine est encodée pour le point $p = (b(m_p), d(m_p))$, et on appelle la différence $dp = p_x - p_y$ la *proéminence* de m_p , ou que m_p est *dp-proéminent*.

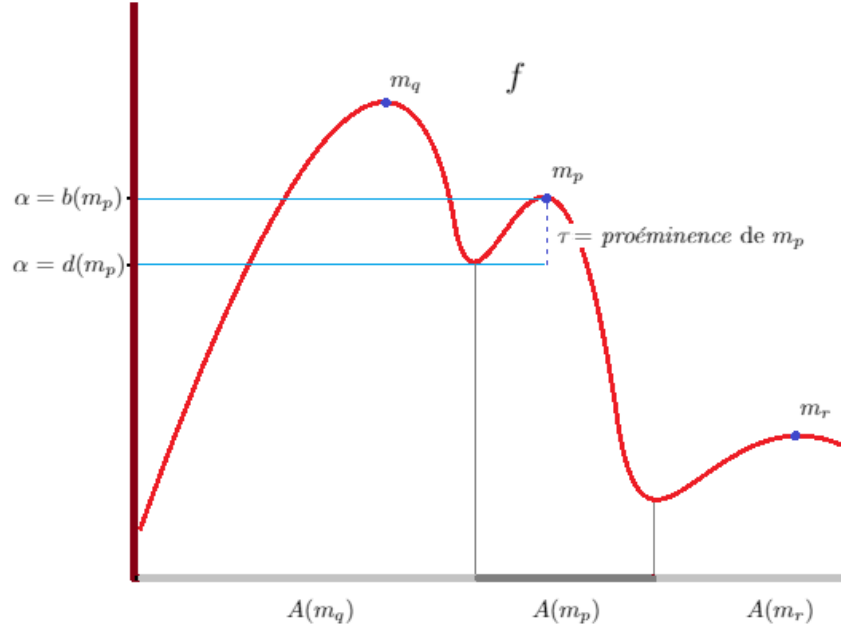


FIGURE 8 – Représentation graphique, avec f une fonction réelle d'une variable, de toutes les idées exposées jusqu'à ici : pics de f (points critiques/ maximums locaux), régions ascendantes de ces pics et proéminence du pic m_p .

En nous appuyant sur un "paramètre de fusion" $\tau \geq 0$, on peut seulement considérer les pics de f de proéminence au moins τ . Pour tout pic m_q de f , on itère l'"application racine" $m_q \mapsto r(m_q)$ jusqu'à ce qu'un pic de proéminence τ soit obtenu. Ce processus finit toujours, donc f a un nombre fini de points critiques, et on a toujours $f(m_q) \leq f(r(m_q))$. Appelons cette fonction itérée r_τ^* . Observons que tout pic de proéminence au moins τ est un point fixe de r_τ^* . L'union des régions ascendantes de tous les pics qui arrivent finalement à m_p avec r_τ^* est appelée le *bassin d'attraction* de m_p (de paramètre τ), $B_\tau(m_p)$:

$$\forall m_p \text{ tel que } p_x - p_y \geq \tau, B_\tau(m_p) = \bigcup_{r_\tau^*(m)=m_p} A(m).$$

Clairement, $B_\tau(m_p)$ contient $A(m_p)$, donc m_p est un point fixe de r_τ^* . De plus, Ces bassins d'attraction forment une partition de l'union de toutes les régions ascendantes. Ce sont précisément ces bassins d'attraction qui constituent nos candidats à clusters.

4.2. Les données d'entrée (input data)

Dans tous les cas, pour fonctionner, la méthode ToMATo a besoin de deux informations sur nos données. Tout d'abord, l'algorithme requiert un poids pour chaque point, c'est à dire, une valeur $\hat{f}(i)$ associée à chaque donnée i , qui représente une estimation d'une hypothétique fonction de densité f , sur laquelle les données ont été obtenues. Puis, il est aussi nécessaire d'avoir un graphe de voisinage, qui "connecte" de quelque sorte ces données entre elles, et qui encode la proximité des points.

De plus, une autre information clé doit être aussi transmise à l'algorithme pour guider la fusion des clusters intermédiaires : un *paramètre de fusion* τ . En somme, ce paramètre détermine à partir de quelle persistance un point ou région de points se mélange avec d'autres ou pas. Sa valeur "idéale" change selon le cas, donc elle dépend de \hat{f} et le numéro de clusters que l'on souhaite obtenir. Le fonctionnement et la détermination de ce paramètre, très important dans la méthode, deviendront beaucoup plus claires après avoir étudié la procédure de l'algorithme et le diagramme qu'il produit.

Développons à présent les deux premières informations nécessaires pour l'algorithme. Nous remarquons que, dans la pratique, nous ne les avons presque jamais directement. En effet, dans le cas le plus simple, nous avons seulement un nuage de points dans \mathbb{R}^d avec n observations, ou, dans des cas plus élaborés et "théoriques", un ensemble de points dans une variété riemannienne, qui permet également de définir des distances entre eux. Avec une base de données réelle, nous avons généralement une quantité n de données avec p attributs quantitatives et/ou catégoriques, où nous pouvons définir distances entre paires, ou les plonger dans \mathbb{R}^p "convenablement" avec un métrique (voir l'introduction de la Section 2).

Indépendamment de la façon dont les calculs sont réalisés ou si on utilise un espace métrique ambiant (normalement \mathbb{R}^d), utiliser des distances entre paires de données est très pratique : elles permettent de construire assez rapidement les graphes de voisinage les plus naturels, et notre algorithme a besoin d'un graphe entre les données pour bien fonctionner. De plus, elles sont aussi pratiques pour calculer certaines estimations de la fonction de densité de base f .

4.2.1. Quelques constructions de graphes habituelles

Développons à présent certaines constructions de graphes habituelles sur des nuages de points, qui peuvent naturellement être utilisées dans notre situation. Nous assumons qu'il n'y a jamais la même distance entre toutes les paires de points. Si ce n'est pas le cas, nous pouvons adapter notre démarche en fonction de la situation :

- *Graphe α -Rips* : Il unit toutes les paires de points x, y qui satisfont $d(x, y) \leq \alpha$. Il est donc le squelette 1-dimensionnel de $Rips_\alpha(X)$, ou $Cech_{\frac{\alpha}{2}}(X)$.
Il constitue, en quelque sorte, le graphe le plus naturel pour connecter les points proches entre eux, et il est aussi très facile à construire. Néanmoins, le nombre d'arêtes peut beaucoup varier entre sommets différents, et le paramètre α n'est pas toujours évident pour obtenir les résultats souhaités : si c'est trop petit, il peut y avoir un numéro excessif de composantes connexes ; cependant, s'il est trop grand, la structure de proximité se dilue aussi.
- *Graphe des k plus proches voisins (k -NN)* : Il connecte chaque sommet avec ses k autres sommets les plus proches. De cette façon, chaque sommet est l'extrémité d'au moins k arêtes. C'est à priori un graphe orienté, donc cette relation de proximité n'est pas symétrique : par exemple, avec $k = 1$, un sommet 1 peut avoir le sommet 2 comme le sommet le plus proche,

mais ce dernier avoir un sommet 3 plus proche que le sommet 1. Parfois, dans la pratique, on ignore cette directionnalité et on accepte que quelques sommets aient plus d'arêtes incidentes. Ce graphe est intéressant et utile puisque, en général, le numéro d'arêtes incidentes à chaque sommet reste assez similaire, et il n'y a jamais des points isolés. Il est aussi un peu plus exigeant a niveau de calcul, donc il faut ordonner à chaque pas les distances d'un sommet aux autres, mais certains algorithmes pour trouver des approximations du graphe k -NN existent qui sont beaucoup plus rapides. Son désavantage principal est que parfois il connecte de points qui ne sont pas spécialement proches.

- *Graphe de Delaunay* : C'est le graphe qu'on obtient si on triangule les points de façon à ce qu'aucun des points reste à l'intérieur du cercle circonscrit d'un des triangles. Normalement, on obtient ainsi une triangulation sans beaucoup d'angles pointus. Il y a des algorithmes assez rapides pour le calculer, et il est aussi généralisable aux dimensions supérieures. Son principal avantage est que, à la différence des deux algorithmes précédents, ce dernier n'a pas besoin d'un paramètre pour être défini. Pareillement au graphe k -NN, le numéro d'arêtes incidentes à chaque sommet est souvent similaire, mais parfois il unit des points qui ne sont pas spécialement proches entre eux.

Ci-dessous, un exemple de chacun de ces trois graphes.

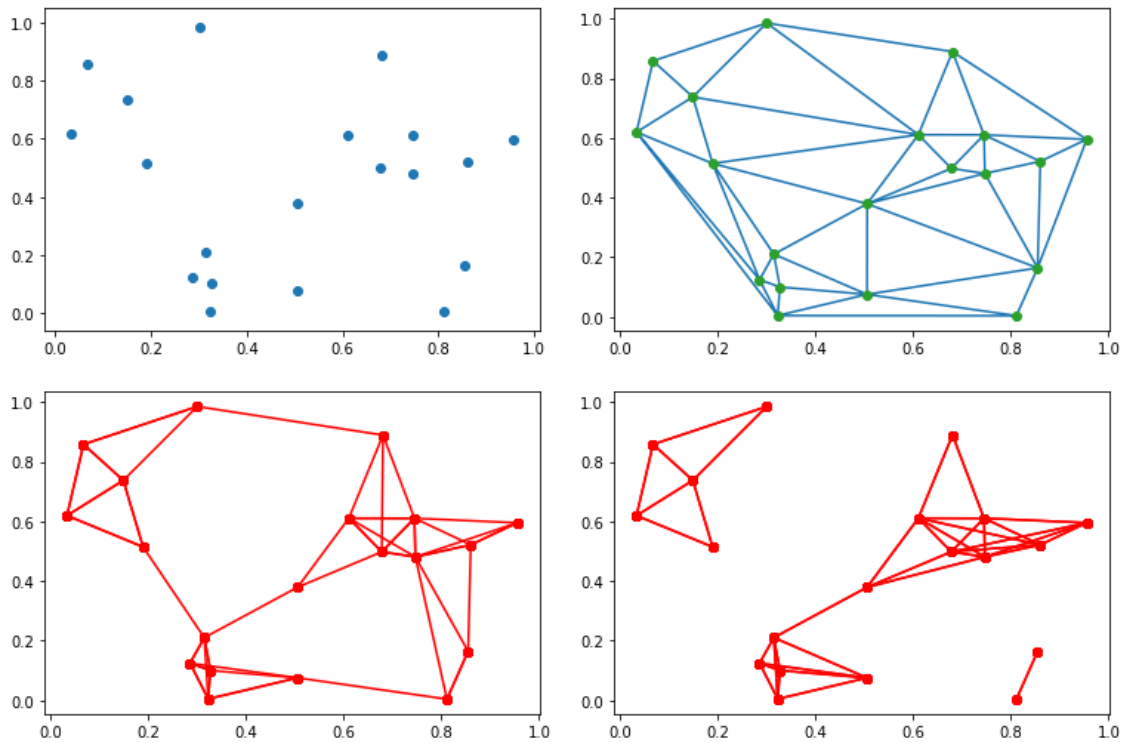


FIGURE 9 – Représentation des trois constructions exposées antérieurement avec un ensemble de 20 points échantillonnés dans le carré 1x1 (distribution uniforme). En haut à droite, le graphe de Delaunay. En bas à gauche : le graphe k -NN avec $k = 4$ (chaque point est aussi son propre voisin le plus proche) ; nous observons qu'une partie importante des sommets a plus de trois arêtes incidentes. En bas à droite ; le graphe α -rayon avec $\alpha = 0.3$.

4.2.2. Quelques estimateurs classiques de la fonction de densité

Exposons à présent deux manières (non-paramétriques) d'estimer la fonction de densité f sur laquelle on suppose que les données ont été obtenues. L'idée est toujours de construire une fonction \hat{f} , estimation de la "véritable" fonction de densité f , en utilisant la disposition des points, qui nous donne des informations sur f . Nous ne détaillerons pas les arguments théoriques qui justifient la justesse (asymptotique) vers f de ces méthodes, et nous ignorerons aussi les possibles généralisations sur des sous-variétés : nous supposons que f est simplement définie sur \mathbb{R}^d (et souvent, seulement \mathbb{R}).

- *Estimation par les k plus proches voisins* : Rappelons avant que, par définition de fonction de densité, un vecteur aléatoire X dans \mathbb{R}^d satisfait, pour tout borélien $A \subseteq \mathbb{R}^d$, $\mathbb{P}(X \in A) = \int_A f$. Donc, si A est une boule petite centrée sur x , la probabilité que X tombe dans A est approximativement $f(x)$ multiplié par le volume de A . En fait, avec des hypothèses assez faibles sur f continue, on a

$$\lim_{\alpha \rightarrow 0} \frac{\int_{B(x_0, \alpha)} f(x) dx}{|B(x_0, \alpha)|} = f(x_0), \quad (10)$$

où $|\cdot|$ dénote le volume dans \mathbb{R}^d . Rappelons aussi que le volume de la boule unité dans \mathbb{R}^d satisfait la formule

$$V_d = \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2} + 1)},$$

où Γ est la fonction Gamma $\Gamma(m) = \int_0^{+\infty} x^{m-1} e^{-x} dx$, et si on varie le rayon r de la boule, le volume change en suivant la formule $V_d \cdot r^d$.

En vue de l'équation (10), avec les données $\{x_1, \dots, x_n\}$, on peut estimer $f(x)$ de la manière "naturelle" suivante : soit k un entier avec $1 \leq k \leq n$, $R_{(k)}(x) = \|x_{(k)}(x) - x\|$ la distance de x à son k plus proche voisin, et μ_n la fonction de répartition empirique, où pour tout borélien $A \subseteq \mathbb{R}^d$, $\mu_n(A) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{x_i \in A\}}$. Alors, l'estimateur par les k plus proches voisins est défini comme

$$f_n(x) = \frac{\mu_n(B(x, R_{(k)}))}{|B(x, R_{(k)})|} = \frac{k}{n V_d \|x_{(k)}(x) - x\|^d}, \quad x \in \mathbb{R}^d. \quad (11)$$

- *Estimation par noyau (Kernel density estimation)* : C'est possiblement la méthode d'estimation la plus habituelle et étudiée. En résumé, c'est une généralisation de la notion d'histogramme, mais facilement réalisable en dimensions plus élevées, et (souvent) aussi continue et différentiable.

L'idée est de construire \hat{f} en additionnant plusieurs petites fonctions centrées chacune sur une donnée. On appelle ces petites fonctions *noyaux*, qui sont toujours réelles, non-négatives et intégrables. De plus, en général on assume aussi, pour notre fonction noyau $K(x)$, que $\int_{\mathbb{R}^d} K(x) dx = 1$ (i.e. $K(x)$ est une fonction de densité) et que K est radiale ($K(-x) = K(x)$ quand $d = 1$, K constant sur $S_r = \{x \in \mathbb{R}^d \mid \|x\| = r\}$ en général).

Prenons maintenant nos données (x_1, \dots, x_n) (indépendantes et identiquement distribuées, obtenues à partir de f). Nous supposons désormais que $d = 1$ pour simplifier les notations, bien que pour d général les constructions suivants sont aussi valides avec quelques légères mo-

difications. En choisissant une fonction noyau $K(x)$, nous construisons la fonction \hat{f} comme :

$$\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right). \quad (12)$$

Ici, h est un *paramètre d'échelle* à déterminer, mais qui a une grande influence sur l'estimation finale. Ce paramètre détermine de quelque sorte la largeur de la fonction noyau autour de chaque donnée. C'est aussi immédiat de vérifier que quand $\int_{\mathbb{R}} K(x)dx = 1$, on a $\int_{\mathbb{R}} \hat{f}_h(x)dx = 1$.

Dans la pratique, la meilleure valeur de ce paramètre est difficile à déterminer, donc il y a toujours un compromis entre biais et variance. Différents travaux essaient d'étudier les meilleures valeurs de h en fonction de chaque situation. En tout cas, bien que certaines indications existent (par exemple, avec un noyau gaussienne, il est habituel de prendre $h \approx 1.06 \cdot \min(\hat{\sigma}, \frac{EIQ}{1.34}) \cdot n^{-\frac{1}{5}}$, où $\hat{\sigma}$ est l'estimateur de l'écart-type habituel et EIQ est l'écart interquartile), normalement l'estimation est faite avec plusieurs valeurs de h et on prend celle qui donne le meilleur résultat.

En ce qui concerne les fonctions noyau, nous remarquons différentes options. Nous montrons, pour $d = 1$ (mais facilement généralisables à d supérieure en prenant $\|x\|$ au lieu de x , et en changeant légèrement quelques coefficients en fonction de la dimension), certaines des plus utilisées, mais sans entrer dans les détails et particularités de chacune :

1. Noyau gaussienne : $K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$.
2. Noyau uniforme : $K(x) = \frac{1}{2} \mathbb{1}(x)_{\{-1 \leq x \leq 1\}}$.
3. Noyau triangulaire : $K(x) = (1 - |x|) \mathbb{1}(x)_{\{-1 \leq x \leq 1\}}$.
4. Noyau de Epanechnikov (parabolique) : $K(x) = \frac{3}{4}(1 - x^2) \mathbb{1}(x)_{\{-1 \leq x \leq 1\}}$.
5. Noyau tricubique : $K(x) = \frac{70}{81}(1 - |x|^3)^3 \mathbb{1}(x)_{\{-1 \leq x \leq 1\}}$.

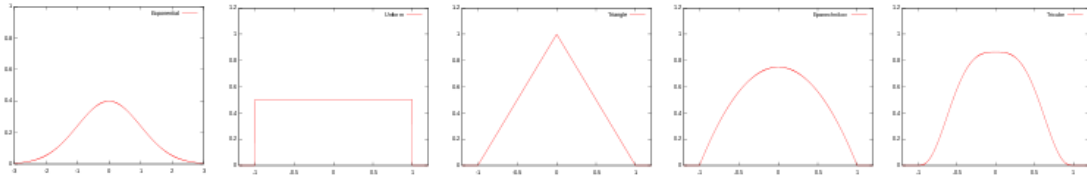


FIGURE 10 – Toutes les fonctions noyaux mentionnées, dans le même ordre.

4.3. La procédure de l'algorithme

Expliquons maintenant comment l'algorithme obtient les différents clusters. Supposons que nous avons un graphe de voisinage G entre les points, des valeurs $\hat{f}(i)$ pour chaque sommet i , et le paramètre de fusion τ . L'algorithme de base se compose de deux parties :

1. (Recherche de modes) Pour calculer les clusters de départ, ToMATo ordonne d'abord tous les sommets de manière décroissante en fonction de sa valeur \hat{f} . Avec cet ordre, il passe par

chaque sommet i et il simule un effet de gradient de la fonction de densité de base : il connecte i avec son voisin dans G avec la \hat{f} -valeur la plus haute. Si tous ses voisins ont des valeurs inférieures, i est étiqueté comme un "pic" (ou mode) de \hat{f} .

En regardant les arêtes "de gradient" à la fin de ce processus, on obtient ce qu'on appelle une "forêt couvrante" du graphe G , une notion similaire à celle de "arbre couvrant" (i.e. un sub-graphe de G connexe et acyclique qui passe pour tous les sommets de G) ; cependant, dans ce cas, la connectivité n'est pas exigée. Chaque composante connexe dans une forêt est un arbre avec la définition usuelle dans la théorie de graphes ; d'ici provient le nom.

Chaque arbre dans cette forêt couvrant peut être vu comme l'équivalent discret d'une région ascendante de f dans le cas continu, expliqué en 4.1.1, donc un cluster initial de notre nuage de points. Pour conserver toute cette information, on peut numéroter les q arbres obtenus et étiqueter chaque sommet avec son arbre correspondant. Dans un arbre i , on appelle $r(i)$ (r de "racine") son sommet avec la \hat{f} -valeur la plus haute, et on appelle aussi $a(j)$ l'arbre associé à un sommet j .

2. (Fusion des arbres) Bien que l'idée de la première étape soit logique, donc elle regroupe des données dans des "bosses" de \hat{f} , elle est aussi un peu aléatoire et inévitablement insuffisant. Dans cette deuxième étape, l'objectif est de fusionner les arbres "similaires", en adaptant la notion de persistance topologique introduite à 4.1.1.

Pour cela, ToMATo passe plusieurs fois sur les sommets de G dans le même ordre. Ici, tous les sommets sont déjà étiquetés dans un arbre. Dans cette itération, étant donné un sommet i , deux cas sont possibles :

- (a) i est déjà un pic d'un arbre, et donc aussi sa racine, et tous les voisins de i ont des \hat{f} -valeurs inférieures. Dans ce cas, les correspondances entre arbres et sommets restent inaltérées.
- (b) i n'est pas le pic de $a(i)$, et on cherche des fusions potentielles entre $a(i)$ et d'autres arbres "à côté".

Pour cela, on prend les voisins k de i dans G (aussi de manière ordonnée) qui satisfont $\hat{f}(k) \geq \hat{f}(i)$, et on regarde si $\hat{f}(r(a(k))) \leq \min\{\hat{f}(r(a(i))), \hat{f}(i) + \tau\}$; ainsi, on étudie si le pic de $a(k)$ est inférieur à celui de $a(i)$ et si sa proéminence est inférieure à τ . Si c'est le cas, toutes les sommets appartenant à $a(k)$ sont réétiquetés à $a(i)$. De la même manière, nous pouvons vérifier si $\hat{f}(r(a(i))) \leq \min\{\hat{f}(r(a(k))), \hat{f}(k) + \tau\}$, et réétiqueter les sommets de $a(i)$ à $a(k)$ si c'est le cas.

À la fin de cette deuxième étape, tous les arbres (mini-clusters) de départ avec des pics de proéminence moins de τ et avec des sommets "connectés" à d'autres arbres ont été unifiés les uns avec les autres pour créer des arbres plus grands, et avec une proéminence d'au moins τ (nos clusters finaux). De plus, on a enregistré dans quel arbre/ cluster chaque donnée appartient.

4.4. Information finale obtenue

Avec le processus expliqué précédemment, l'information finale obtenue semble claire : pour chaque donnée i , une étiquette $a(i)$, son cluster final associé. Néanmoins, la méthode précédente n'est pas la plus utile pour travailler avec le type de problèmes que l'on retrouve avec des données réelles. On peut donc utiliser les notions expliquées au troisième chapitre pour obtenir un algorithme plus flexible et informatif.

En effet, en reprenant la deuxième étape exposée précédemment, il est évident que la valeur τ joue un rôle essentiel dans l'algorithme ; c'est ce numéro qui décide quelle doit être la proéminence

minimale d'un pic-cluster pour ne pas être fusionné avec d'autres pics-clusters "proches". Cependant, dans la pratique, normalement nous n'avons pas connaissance de la valeur τ "idéale" pour obtenir le meilleur résultat de clustering. Par exemple, nous avons déjà remarqué que les valeurs \hat{f} associées à chaque donnée dépendent de l'estimation choisie et, par conséquent, un "bon" paramètre τ , s'il existe, possiblement aussi.

C'est au moment de choisir une valeur de τ convenable que les diagrammes de persistance introduits au troisième chapitre deviennent utiles. L'idée est de créer une représentation graphique de la prééminence de tous les différents clusters pour mieux détecter quels sont spécialement prééminents. Avec cette information, nous pouvons ajuster τ pour obtenir un nombre de clusters plus naturel, avec les étiquettes correspondantes.

Notre diagramme de persistance peut être obtenu de la manière suivant :

- Au début, on crée un point (x, y) pour chaque arbre-cluster initial, qui a toujours un pic associé : son sommet avec l'estimation de f la plus élevée, un mode de f . La coordonnée x stocke cette valeur, tandis que la coordonnée y reste non-attribuée.
- Puis, on commence à fusionner ces clusters initiaux, en suivant la deuxième étape expliquée dans la section précédente et en gardant une trace de ces fusions. Intuitivement, on peut imaginer le paramètre de fusion τ qui vaut 0 au début, et qui augmente progressivement. Chaque fois que deux clusters sont fusionnés, on enregistre la mort du plus "petit" (i.e. moins prééminent, i.e. avec un pic associé moins haut) dans la coordonnée y , qui prend la valeur $y = x - \tau$, tandis que le plus "grand" continue d'exister.
- Ce processus continue jusqu'à ce que toutes les fusions possibles aient lieu. À ce moment, seulement les clusters associés aux composantes connexes du graphe de voisinage restent en vie, et on leur assigne la coordonnée $y = -\infty$.

Enfin, on obtient un ensemble de points qui encode d'une manière assez complète les prééminences relatives de tous les clusters/bassins de f , où les distances (verticales) entre les (x, y) et la diagonale sont leur prééminence. Il est recommandable de dessiner les points avec $y = -\infty$ avec une couleur différente, pour mieux identifier dans le diagramme le nombre de composantes connexes existantes.

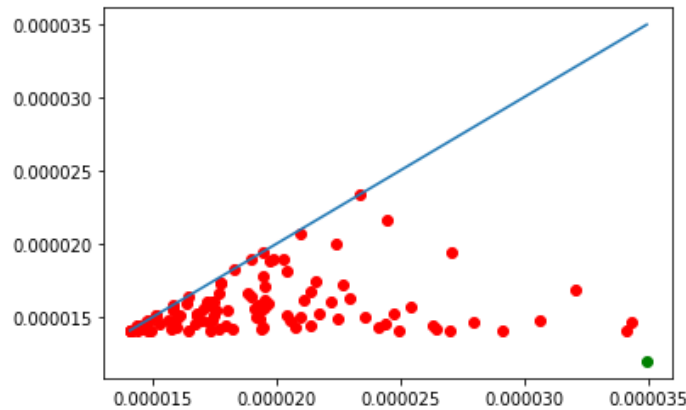


FIGURE 11 – Exemple de diagramme de persistance du type que nous venons d'exposer.

On voit rapidement que le diagramme de persistance obtenu n'est qu'une variation du diagramme de persistance 0-dimensionnel associé aux ensembles de super-niveau d'une fonction f ,

comme celui décrit à l'exemple 3.26 (où on prenait les ensembles de sous-niveau). Cependant, la "connectivité" est encodée dans un graphe de voisinage, et les points habitent la moitié inférieure de la diagonal. Néanmoins, les différents résultats sur les diagrammes de persistance, notamment ceux liés à sa stabilité (section 3.5), peuvent être appliqués avec de petites variations dans notre cas discret.

Nous remarquons que, même si nous avons exposé la méthode en imaginant que la valeur τ augmente progressivement, au niveau algorithmique tous les points du diagramme de persistance peuvent être calculés d'une façon directe : au début, et aussi après chaque fusion, on parcourt tous les sommets ordonnés en fonction de \hat{f} en cherchant, pour un sommet i avec son pic correspondant p_i , un voisin k dans un cluster différent qui satisfait $\hat{f}(k) \geq \hat{f}(i)$ et $\hat{f}(p_k) \leq \hat{f}(p_i)$ (resp. $\hat{f}(p_k) \leq \hat{f}(p_i)$). Dans ce cas, on assigne la valeur $y = \hat{f}(p_k) - \hat{f}(i)$ au cluster du sommet k , et tous ses sommets sont étiquetés avec le cluster du sommet i , et p_i devient son nouveau pic (resp. on assigne la valeur $y = \hat{f}(p_i) - \hat{f}(i)$ au cluster du sommet i , et tous ses sommets sont étiquetés au cluster du sommet k , et p_k devient son nouveau pic).

Dans tous les cas, ce diagramme de persistance devient très utile pour choisir une bonne valeur de τ pour l'algorithme original, exposé à la section précédant : il convient de regarder quels points sont de manière naturel plus éloignés de la diagonale (et combien il y en a) ; puis, nous choisissons une valeur τ inférieure à sa proéminence, les laissant intacts à la fin. En fait, après avoir calculé toutes les proéminences relatives pour dessiner le diagramme de persistance, on peut coder l'algorithme d'une manière encore plus intuitive : au lieu de donner une valeur τ d'entrée, on donne le numéro de clusters final souhaité m , et les fusions continuent de se produire jusqu'à ce que seulement les m clusters les plus proéminents restent. Cependant, il faut prendre en compte que l'algorithme ne peut pas fusionner des composantes connexes différentes (qui ont une proéminence "infinie").

4.5. Mise en œuvre de l'algorithme et exploration

L'algorithme ToMATo exposé à ce chapitre vient d'être implémenté à Python/ C++ et ajouté à la librairie GUDHI [8], une des bibliothèques de référence de la TDA. Cette librairie open-source, codée en C++ mais avec une interface Python, offre des méthodes et ressources pour construire des complexes simpliciales et d'autres structures sur des nuages de points, et calculer les différents types d'homologie persistante.

La partie la plus pratique de ce travail a été de bien comprendre cette implémentation, réalisée par le chercheur Marc Glisse. Puis, nous avons essayé de tester ses limites et possibles erreurs. Cela a été fait par correspondance virtuelle avec plusieurs Jupyter notebooks. Cela a impliqué un apprentissage continu de Python et d'autres outils de programmation qui sont très pratiques et habituels dans le monde de la science des données et sur le marché du travail en général.

Finalement, avec le code déjà définitif, il paraissait approprié de préparer aussi un tutoriel de référence (en anglais) montrant toutes les options du code. Dans ce dernier, plusieurs exemples illustratifs aideraient et guideraient les utilisateurs potentiels. Le tutoriel final est annexé en PDF à la fin de ce travail. Il peut aussi être consulté en version HTML (de façon temporaire) avec le lien nilgarces.com/tomato.html.

Références

- [1] Gérard BIAU et Luc DEVROYE. *Lectures on the Nearest Neighbor Method*. Springer Series in Data Sciences. Springer New York Inc.
- [2] Jean-Daniel BOISSONANT, Frédéric CHAZAL et Mariette YVINEC. « Geometric and Topological Inference ». In : (2018).
- [3] Frédéric CHAZAL et Bertrand MICHEL. « An introduction to Topological Data Analysis : fundamental and practical aspects for data scientists ». In : (oct. 2017).
- [4] Frédéric CHAZAL et al. « Persistence-Based Clustering in Riemannian Manifolds ». In : *Journal of the ACM* 60 (juin 2011). DOI : [10.1145/1998196.1998212](https://doi.org/10.1145/1998196.1998212).
- [5] David COHEN-STEINER, Herbert EDELSBRUNNER et John HARER. « Stability of Persistence Diagrams ». In : *Discrete Computational Geometry* 37 (2007), p. 103-120.
- [6] Herbert EDELSBRUNNER, David LETSCHER et Afra ZOMORODIAN. « Topological Persistence and Simplification ». In : *Discrete Computational Geometry* 28 (2002), p. 511-533.
- [7] Aurélien GÉRON. *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O'Reilly Media, Inc., 2017. ISBN : 9781492032632.
- [8] *GUDHI Library*. URL : <https://gudhi.inria.fr/>. (accessed : 20.06.2020).
- [9] Trevor HASTIE, Robert TIBSHIRANI et Jerome FRIEDMAN. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., 2001.
- [10] Allen HATCHER. *Algebraic topology*. Cambridge : Cambridge Univ. Press, 2000. URL : <https://cds.cern.ch/record/478079>.
- [11] F. PEDREGOSA et al. « Scikit-learn : Machine Learning in Python ». In : *Journal of Machine Learning Research* 12 (2011), p. 2825-2830.
- [12] *Scikit-Learn documentation : Clustering*. URL : <https://scikit-learn.org/stable/modules/clustering.html>. (accessed : 20.05.2020).

A handy guide to using the ToMATo algorithm

Introduction

This code is an implementation of the ToMATo algorithm exposed in [1], a clustering method based on the idea of topological persistence. In short, the algorithm needs a density estimation (so to each point x we associate a value $\hat{f}(x)$) and a neighborhood graph. First, it starts with a mode-seeking phase (naive hill-climbing) to build the initial clusters (each with its own mode), following the connected points in the neighborhood graph. Finally, it merges these initial clusters based on their prominence. This merging has a hierarchical nature, i.e. we always obtain the successive new clusters by merging two existing ones.

The merging phase depends on a parameter, which is the minimum prominence a cluster needs to avoid getting merged into another, adjacent, bigger cluster (i.e. with a higher associated mode); thus, it determines to a great extent the obtained number of clusters. In practice, the convenience of this parameter depends on the input graph and the density estimation, and it can be hard to choose it properly. This is why, in our implementation, we allow instead the option to choose the "desired" final number of clusters n , and the algorithm itself, after computing the initial clusters as well as their prominences, keeps merging them "parameterless-ly" until only the n clusters with highest prominence remain (if possible).

Along with the clustering itself, the algorithm also produces the *persistence diagram* of the merge tree of the initial clusters. This is a really convenient graphical tool to help decide the "natural" number of clusters in our input data. We explain its interpretation briefly in the section "Output information".

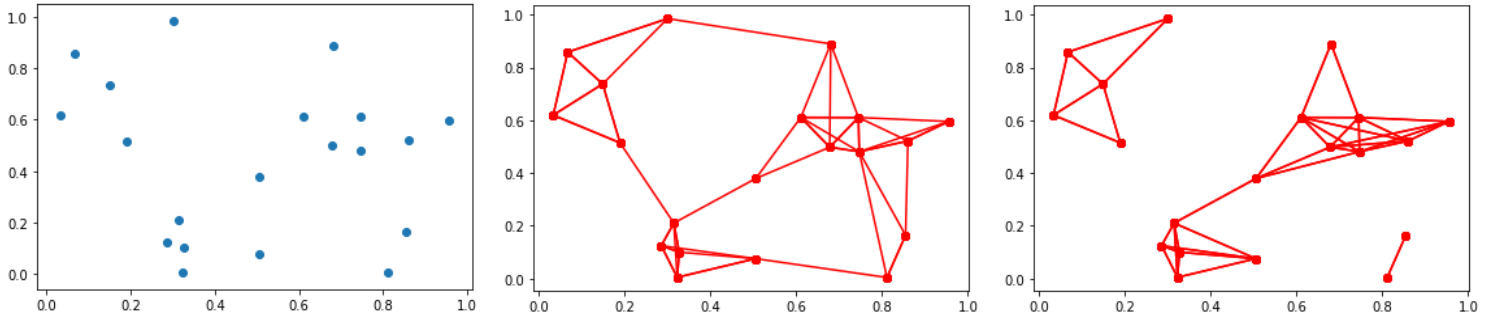
Input data format

As mentioned, the algorithm needs a neighborhood graph of the data and a value associated each entry (an estimation of f over it). Given that, in many situations, the input data is a point cloud (i.e. a set of n observations each with p numerical features), the code provides a few density estimators and graph constructions over them for convenience, but advanced users may provide their own graph and density estimates instead of point coordinates.

Since the algorithm essentially computes basins of attraction, it is also encouraged to use it on functions that do not represent densities at all.

For an input point cloud, the density estimation and graph construction methods that have been implemented are:

- For density estimation, the ubiquitous Kernel Density Estimation (KDE for short) can be used (using the scikit-learn library), and also the Distance-to-a-Measure method (DTM), a bit more experimental and recently developed to face more efficiently the potential presence of outliers; more information about it can be found in the tutorial [2] and the paper [3]. The logarithmic versions of both estimation methods are also implemented.
- Regarding the building of the graph, there is the option to construct the k -NN graph (where, for each vertex, an edge is created between it and its k nearest neighbors), and the r -radius graph (where an edge is created whenever two vertices lay in a distance less than r). Obviously, both parameters can (and should) be properly chosen. In the following image we can see both constructions over a point cloud in the square 1×1 (first image); in the second one, we have the k -NN graph (with $k=4$), while in the third we have the r -radius graph (with $r=0.3$):



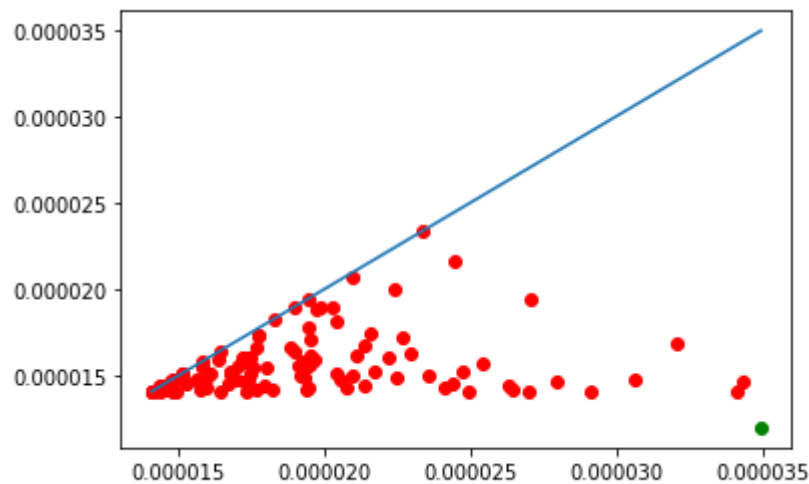
Output information

At the end, the algorithm outputs basically two informations of interest:

In all cases, it produces the (0-dimensional) *persistence diagram* of the merging process of the initial clusters. In short, this is a graphical representation of the lifespan of the different clusters as we keep diminishing the prominence threshold.

At the beginning, we have a point for each initial cluster, which also has an associated peak (the vertex with the highest estimate of f , a "mode" of f). Then, we start looking for merges of these clusters, by melding them with neighboring clusters with higher associated peaks. To do so, we basically keep checking, for the different vertices i (in decreasing order), which "neighboring" peaks p_j lower than p_i satisfy $\hat{f}(p_j) < \hat{f}(i) + \tau$, where τ is our prominence value. When this happens, we merge the whole cluster associated to that peak p_j to the one in which i belongs, forming a new, bigger cluster, still with peak p_i . The higher τ needs to be before this happens, the more prominent is p_j and its associated cluster.

In a persistence diagram, all this information is encoded in the following way: there is a point (x, y) for each initial cluster. The x coordinate is the value of its associated peak p . The y coordinate is the value $\hat{f}(p) - \tau$ from which we can find a "neighboring point" of that peak, but belonging to a different cluster, with equal or greater \hat{f} ; equivalently, it is the highest neighbor of p not belonging to the cluster it defines. Thus, the length of vertical line connecting (x, y) with the diagonal, or equivalently $x - y$, is the prominence of the peak. In consequence, to get an idea of the real number of clusters, it is natural to look for the number of points in the persistence diagram further away from the diagonal. The points associated to a peak of a cluster which never dies (i.e. it never gets merged, so it forms a connected component at the end) are colored in green.



In view of the persistence diagram obtained, it is then natural to ask for a specific number of clusters at the end, or to specify a certain persistence threshold. After this has been stipulated, the algorithm also outputs a numerical "label" for each entry in the input data (in the same order they have been introduced, whatever the format): the cluster it has been assigned to. This labelling is saved in the attribute "labels_" as an ordered vector, so it can be easily used to plot the data in different colors or formats depending on their assigned cluster.

THE TOMATO CLASS

The code now

This is the current version of the code in the Gudhi Library:


```

In [63]: # This file is part of the Gudhi Library - https://gudhi.inria.fr/ - which is release
d under MIT.
# See file LICENSE or go to https://gudhi.inria.fr/licensing/ for full license detail
s.
# Author(s):          Marc Glisse
#
# Copyright (C) 2020 Inria
#
# Modification(s):
#   - YYYY/MM Author: Description of the modification

import numpy
from ..point_cloud.knn import KNearestNeighbors
from ..point_cloud.dtm import DTMDensity
from ._tomato import *

# The fit/predict interface is not so well suited...

class Tomato:
    """
    This clustering algorithm needs a neighborhood graph on the points, and an estima
    tion of the density at each point.
    A few possible graph constructions and density estimators are provided for conven
    ience, but it is perfectly natural
    to provide your own.

    :Requires: `SciPy <installation.html#scipy>`, `Scikit-learn <installation.html#s
    cikit-learn>` or others
    (see :class:`~gudhi.point_cloud.knn.KNearestNeighbors`) in function of the op
    tions.

    Attributes
    -----
    n_clusters_: int
        The number of clusters. Writing to it automatically adjusts `labels_`.
    merge_threshold_: float
        minimum prominence of a cluster so it doesn't get merged. Writing to it autom
        atically adjusts `labels_`.
    n_leaves_: int
        number of leaves (unstable clusters) in the hierarchical tree
    leaf_labels_: ndarray of shape (n_samples,)
        cluster labels for each point, at the very bottom of the hierarchy
    labels_: ndarray of shape (n_samples,)
        cluster labels for each point, after merging
    diagram_: ndarray of shape (`n_leaves_`, 2)
        persistence diagram (only the finite points)
    max_weight_per_cc_: ndarray of shape (n_connected_components,)
        maximum of the density function on each connected component. This corresponds
        to the abscissa of infinite
        points in the diagram
    children_: ndarray of shape (`n_leaves_`-n_connected_components, 2)
        The children of each non-leaf node. Values less than `n_leaves_` correspond t
        o leaves of the tree.
        A node i greater than or equal to `n_leaves_` is a non-leaf node and has chil
        dren children_[i - `n_leaves_`].
        Alternatively at the i-th iteration, children[i][0] and children[i][1] are me
        rged to form node `n_leaves_` + i
    weights_: ndarray of shape (n_samples,)
        weights of the points, as computed by the density estimator or provided by th
        e user
    params_: dict

```

```

        Parameters like metric, etc
        """

def __init__(
    self,
    graph_type="knn",
    density_type="logDTM",
    n_clusters=None,
    merge_threshold=None,
    #         eliminate_threshold=None,
    #         eliminate_threshold (float): minimum max weight of a cluster so i
t doesn't get eliminated
    **params
):
    """
    Args:
        graph_type (str): 'manual', 'knn' or 'radius'. Default is 'knn'.
        density_type (str): 'manual', 'DTM', 'logDTM', 'KDE' or 'logKDE'. When yo
u have many points,
            'KDE' and 'logKDE' tend to be slower. Default is 'logDTM'.
        metric (str/Callable): metric used when calculating the distance between
instances in a feature array.
            Defaults to Minkowski of parameter p.
        kde_params (dict): if density_type is 'KDE' or 'logKDE', additional param
eters passed directly to
            sklearn.neighbors.KernelDensity.
        k (int): number of neighbors for a knn graph (including the vertex itsel
f). Defaults to 10.
        k_DTM (int): number of neighbors for the DTM density estimation (includin
g the vertex itself).
            Defaults to k.
        r (float): size of a neighborhood if graph_type is 'radius'. Also used as
default bandwidth in kde_params.
        eps (float): (1+eps) approximation factor when computing distances (ignor
ed in many cases).
        n_clusters (int): number of clusters requested. Defaults to None, i.e. no
merging occurs and we get
            the maximal number of clusters.
        merge_threshold (float): minimum prominence of a cluster so it doesn't ge
t merged.
        symmetrize_graph (bool): whether we should add edges to make the neighbor
hood graph symmetric.
            This can be useful with k-NN for small k. Defaults to false.
        p (float): norm  $L^p$  on input points. Defaults to 2.
        q (float): order used to compute the distance to measure. Defaults to di
m.
            Beware that when the dimension is large, this can easily cause overfl
ows.
        dim (float): final exponent in DTM density estimation, representing the d
imension. Defaults to the
            dimension, or 2 when the dimension cannot be read from the input (met
ric is "precomputed").
        n_jobs (int): Number of jobs to schedule for parallel processing on the C
PU.
            If -1 is given all processors are used. Default: 1.
        params: extra parameters are passed to :class:`~gudhi.point_cloud.knn.KNe
arestNeighbors` and
            :class:`~gudhi.point_cloud.dtm.DTMDensity`.
        """
    # Should metric='precomputed' mean input_type='distance_matrix'?
    # Should we be able to pass metric='minkowski' (what None does currently)?
    self.graph_type_ = graph_type
    self.density_type_ = density_type

```

```

self.params_ = params
self.__n_clusters = n_clusters
self.__merge_threshold = merge_threshold
# self.eliminate_threshold_ = eliminate_threshold
if n_clusters and merge_threshold:
    raise ValueError("Cannot specify both a merge threshold and a number of clusters")

def fit(self, X, y=None, weights=None):
    """
    Args:
        X ((n,d)-array of float|(n,n)-array of float|Sequence[Iterable[int]]): coordinates of the points,
            or distance matrix (full, not just a triangle) if metric is "precomputed", or list of neighbors
            for each point (points are represented by their index, starting from 0) if graph_type is "manual".
        weights (ndarray of shape (n_samples)): if density_type is 'manual', a density estimate at each point
        y: Not used, present here for API consistency with scikit-learn by convention.
    """
    # TODO: First detect if this is a new call with the same data (only threshold changed?)
    # TODO: Less code duplication (subroutines?), less spaghetti, but don't compute neighbors twice if not needed. Clear error message for missing or contradictory parameters.
    if weights is not None:
        density_type = "manual"
    else:
        density_type = self.density_type_
        if density_type == "manual":
            raise ValueError("If density_type is 'manual', you must provide weights to fit()")

    if self.graph_type_ == "manual":
        self.neighbors_ = X
        # FIXME: uniformize "message 'option'" vs 'message "option"'
        assert density_type == "manual", 'If graph_type is "manual", density_type must be as well'
    else:
        metric = self.params_.get("metric", "minkowski")
        if metric != "precomputed":
            self.points_ = X

    # Slight complication to avoid computing knn twice.
    need_knn = 0
    need_knn_ngb = False
    need_knn_dist = False
    if self.graph_type_ == "knn":
        k_graph = self.params_.get("k", 10)
        # If X has fewer than k points...
        if k_graph > len(X):
            k_graph = len(X)
        need_knn = k_graph
        need_knn_ngb = True
    if self.density_type_ in ["DTM", "logDTM"]:
        k = self.params_.get("k", 10)
        k_DTM = self.params_.get("k_DTM", k)
        # If X has fewer than k points...
        if k_DTM > len(X):
            k_DTM = len(X)
        need_knn = max(need_knn, k_DTM)

```

```

        need_knn_dist = True
        # if we ask for more neighbors for the graph than the DTM, getting the di
stances is a slight waste,
        # but it looks negligible
    if need_knn > 0:
        knn_args = dict(self.params_)
        knn_args["k"] = need_knn
        knn = KNearestNeighbors(return_index=need_knn_ngb, return_distance=need_k
nn_dist, **knn_args).fit_transform(
            X
        )
        if need_knn_ngb:
            if need_knn_dist:
                self.neighbors_ = knn[0][:, 0:k_graph]
                knn_dist = knn[1]
            else:
                self.neighbors_ = knn
        elif need_knn_dist:
            knn_dist = knn
    if self.density_type_ in ["DTM", "logDTM"]:
        dim = self.params_.get("dim")
        if dim is None:
            dim = len(X[0]) if metric != "precomputed" else 2
        q = self.params_.get("q", dim)
        weights = DTMDensity(k=k_DTM, metric="neighbors", dim=dim, q=q).fit_trans
form(knn_dist)
        if self.density_type_ == "logDTM":
            weights = numpy.log(weights)

    if self.graph_type_ == "radius":
        if metric in ["minkowski", "euclidean", "manhattan", "chebyshev"]:
            from scipy.spatial import cKDTree

            tree = cKDTree(X)
            # TODO: handle "l1" and "l2" aliases?
            p = self.params_.get("p")
            if metric == "euclidean":
                assert p is None or p == 2, "p=" + str(p) + " is not consistent w
ith metric='euclidean'"
                p = 2
            elif metric == "manhattan":
                assert p is None or p == 1, "p=" + str(p) + " is not consistent w
ith metric='manhattan'"
                p = 1
            elif metric == "chebyshev":
                assert p is None or p == numpy.inf, "p=" + str(p) + " is not cons
istent with metric='chebyshev'"
                p = numpy.inf
            elif p is None:
                p = 2 # the default
            eps = self.params_.get("eps", 0)
            self.neighbors_ = tree.query_ball_tree(tree, r=self.params_["r"], p=p
, eps=eps)

        # TODO: sklearn's NearestNeighbors.radius_neighbors can handle more metri
cs efficiently via its BallTree
        # (don't bother with the _graph variant, it just calls radius_neighbors).
        elif metric != "precomputed":
            from sklearn.metrics import pairwise_distances

            X = pairwise_distances(X, metric=metric, n_jobs=self.params_.get("n_j
obs"))
            metric = "precomputed"

```

```

        if metric == "precomputed":
            # TODO: parallelize? May not be worth it.
            X = numpy.asarray(X)
            r = self.params_["r"]
            self.neighbors_ = [numpy.flatnonzero(1 <= r) for l in X]

    if self.density_type_ in {"KDE", "logKDE"}:
        # Slow...
        assert (
            self.graph_type_ != "manual" and metric != "precomputed"
        ), "Scikit-learn's KernelDensity requires point coordinates"
        kde_params = dict(self.params_.get("kde_params", dict()))
        kde_params.setdefault("metric", metric)
        r = self.params_.get("r")
        if r is not None:
            kde_params.setdefault("bandwidth", r)
        # Should we default rtol to eps?
        from sklearn.neighbors import KernelDensity

        weights = KernelDensity(**kde_params).fit(self.points_).score_samples(self
f.points_)

        if self.density_type_ == "KDE":
            weights = numpy.exp(weights)

        # TODO: do it at the C++ level and/or in parallel if this is too slow?
        if self.params_.get("symmetrize_graph"):
            self.neighbors_ = [set(line) for line in self.neighbors_]
            for i, line in enumerate(self.neighbors_):
                line.discard(i)
                for j in line:
                    self.neighbors_[j].add(i)

        self.weights_ = weights
        # This is where the main computation happens
        self.leaf_labels_, self.children_, self.diagram_, self.max_weight_per_cc_ = h
ierarchy(self.neighbors_, weights)
        self.n_leaves_ = len(self.max_weight_per_cc_) + len(self.children_)
        assert self.leaf_labels_.max() + 1 == len(self.max_weight_per_cc_) + len(self
.children_)
        # TODO: deduplicate this code with the setters below
        if self.__merge_threshold:
            assert not self.__n_clusters
            self.__n_clusters = numpy.count_nonzero(
                self.diagram[:, 0] - self.diagram[:, 1] > self.__merge_threshold
            ) + len(self.max_weight_per_cc_)
        if self.__n_clusters:
            # TODO: set corresponding merge_threshold?
            renaming = merge(self.children_, self.n_leaves_, self.__n_clusters)
            self.labels_ = renaming[self.leaf_labels_]
            # In case the user asked for something impossible.
            # TODO: check for impossible situations before calling merge.
            self.__n_clusters = self.labels_.max() + 1
        else:
            self.labels_ = self.leaf_labels_
            self.__n_clusters = self.n_leaves_
        return self

def fit_predict(self, X, y=None, weights=None):
    """
    Equivalent to fit(), and returns the `labels_`.
    """
    return self.fit(X, y, weights).labels_

```

TODO: add argument k or threshold? Have a version where you can click and it shows the line and the corresponding k?

```
def plot_diagram(self):
    """
    """

    import matplotlib.pyplot as plt

    l = self.max_weight_per_cc_.min()
    r = self.max_weight_per_cc_.max()
    if self.diagram_.size > 0:
        plt.plot(self.diagram_[:, 0], self.diagram_[:, 1], "ro")
        l = min(l, self.diagram_[:, 1].min())
        r = max(r, self.diagram_[:, 0].max())
    if l == r:
        if l > 0:
            l, r = 0.9 * l, 1.1 * r
        elif l < 0:
            l, r = 1.1 * l, 0.9 * r
        else:
            l, r = -1.0, 1.0
    plt.plot([l, r], [l, r])
    plt.plot(
        self.max_weight_per_cc_, numpy.full(self.max_weight_per_cc_.shape, 1.1 *
l - 0.1 * r), "ro", color="green"
    )
    plt.show()

# Use set_params instead?
@property
def n_clusters_(self):
    return self.__n_clusters

@n_clusters_.setter
def n_clusters_(self, n_clusters):
    if n_clusters == self.__n_clusters:
        return
    self.__n_clusters = n_clusters
    self.__merge_threshold = None
    if hasattr(self, "leaf_labels_"):
        renaming = merge(self.children_, self.n_leaves_, self.__n_clusters)
        self.labels_ = renaming[self.leaf_labels_]
        # In case the user asked for something impossible
        self.__n_clusters = self.labels_.max() + 1

@property
def merge_threshold_(self):
    return self.__merge_threshold

@merge_threshold_.setter
def merge_threshold_(self, merge_threshold):
    if merge_threshold == self.__merge_threshold:
        return
    if hasattr(self, "leaf_labels_"):
        self.n_clusters_ = numpy.count_nonzero(self.diagram_[:, 0] - self.diagram
_[:, 1] > merge_threshold) + len(
        self.max_weight_per_cc_
    )
    else:
        self.__n_clusters = None
        self.__merge_threshold = merge_threshold
```

Description

Parameters

By "parameters" we mean the information we (must) provide to construct a specific instance of the class. They are given as arguments in the constructor function "`__init__`":

- **graph_type** (str): 'manual', 'knn' (default) or 'radius'.
- **density_type** (str): 'manual', 'DTM', 'logDTM' (default), 'KDE' or 'logKDE'. With many points, 'KDE' and 'logKDE' tend to be slower.
- **n_clusters** (int): number of clusters requested. Defaults to None, i.e. no merging occurs and we get the maximal number of clusters.
- **merge_threshold** (float): minimum prominence of a cluster so it doesn't get merged.

(Naturally, both `n_clusters` and `merge_threshold` cannot be provided simultaneously, as it can be deduced from the explanation of the algorithm)

- **metric** (str|Callable): metric used to compute the pairwise distances between points (if we don't input them). If None, use Minkowski of parameter `p`.
- **kde_params** (dict): if `density_type` is 'KDE' or 'logKDE', additional parameters passed directly to `sklearn.neighbors.KernelDensity`.
- **k** (int): number of neighbors for a k-NN graph (including the vertex itself). Defaults to 10.
- **k_DTM** (int): number of neighbors for the DTM density estimation (including the vertex itself). Defaults to `k`.
- **r** (float): size of a neighborhood if `graph_type` is 'radius'. Also used as default bandwidth in `kde_params`.
- **eps** (float): approximation factor when computing nearest neighbors (ignored in many cases).
- **symmetrize_graph** (bool): whether we should add edges to make the neighborhood graph symmetric. This can be useful with k-NN for small `k`. Defaults to false.
- **p** (float): norm L^p on input points (`numpy.inf` is supported without gpu). Defaults to 2.
- **dim** (float): final exponent in DTM density estimation, representing the dimension. Defaults to the dimension, or 2 when the dimension cannot be read from the input (metric is "precomputed").
- **q** (float): order used to compute the distance to measure. Defaults to `dim`. Beware that when the dimension is large, this can easily cause overflows.
- **n_jobs** (int): Number of jobs to schedule for parallel processing on the CPU. If -1 is given all processors are used. Default: 1.
- **params**: extra parameters are passed to the classes `gudhi.point_cloud.knn.KNearestNeighbors` and `gudhi.point_cloud.dtm.DTMDensity`, for example `'implementation="keops"'` for the first one.

Attributes

By "attributes" we mean the properties, or variables, created within a class: they store its information, allow it to run some of its methods and functionalities, etc... We recall also that, as a common practice, the attributes of a class (those defined with `self`.) usually have some `"_"` in its name to make them more distinguishable within the code.

Naturally, the values of most of the attributes depend on the instance itself, and, depending on it, some of them will be present or not. Actually, many of the previous parameters have their corresponding attribute, as for example `n_clusters_` and `merge_threshold_` (which, when modified, can alter the values of other attributes, as the `.setter` property shows), or they are stored inside the `"params_"` dictionary; `input_type`, `metric`,...

Other important attributes which are created specifically to run the desired methods and are not given as parameters are:

- `n_leaves_` (int): Number of leaves (unstable clusters) in the hierarchical tree. Basically, the number of "temporary" clusters (or mini-clusters) we have along the way.
- `leaf_labels_` (ndarray of shape (n_samples)): Cluster labels for each point, at the very bottom of the hierarchy.
- `labels_` (ndarray of shape (n_samples)): Cluster labels for each point, after merging. Writing to `n_clusters_` and `merge_threshold_` automatically adjusts it.
- `diagram_` (ndarray of shape (n_leaves_, 2)): Persistence diagram (only the finite points).
- `weights_` (ndarray of shape (n_samples,)): Weights of the points, as computed by the density estimator or provided by the user.
- `max_weight_per_cc_` (ndarray of shape (n_connected_components,)): Maximum of the density function on each connected component. This corresponds to the abscissa of infinite points in the diagram.

Methods

The Tomato class contains, in essence, two methods:

- The first one is the `.fit` method, which does basically everything: it processes the input data taking into account its format and the given arguments, it does the merging process depending on them, does the labelling of the entries and stores the points that will eventually form the persistence diagram. The method `.fit_predict` is identical, but it returns the labels vector. Both of them take as the input the coordinates of the points/ distance matrix/ neighborhood matrix, and possibly a "weights" vector, the estimate of f on each entry.
- The second one is the `.plot_diagram` method, without arguments, that plots the persistence diagram (after the `fit` method).

EXAMPLES AND TESTS

Example 1

We start with a really simple example with a few hundreds points to get used to manipulating the Tomato class.


```
In [32]: import matplotlib.pyplot as plt

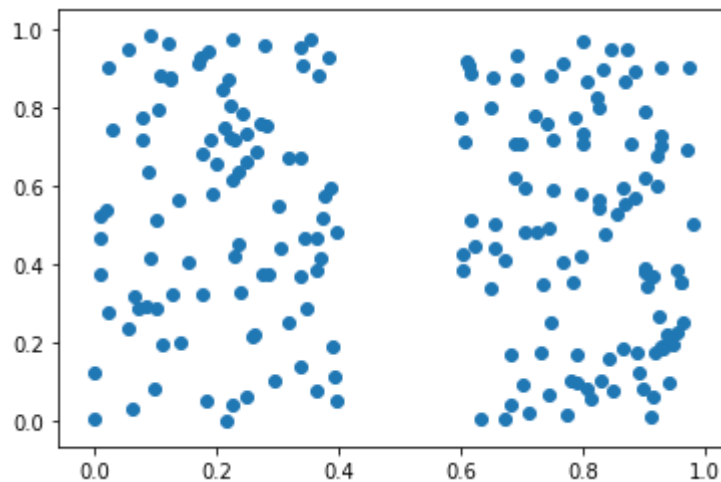
cmap = plt.cm.Spectral;
fig, ax = plt.subplots();

import random as rd
import numpy as np

# Simple function to get random values for x uniformly but within intervals (0,a) U
(b, 1)
def x_var(x):
    if x > 0.5:
        return rd.uniform(0.6, 1)
    else:
        return rd.uniform(0, 0.4)

p1 = np.zeros((200,2))
for i in range(200):
    p1[i,0] = x_var(rd.uniform(0,1))
    p1[i,1] = rd.uniform(0,1)

ax.cla()
ax.scatter(*zip(*p1));
```



There are "clearly" two main groups of points.

Let's suppose we don't know that, so we run the Tomato algorithm blindly. We use the KDE (without specifying extra parameters, thus using the default parameters in Scikit-Learn) and the radius graph with $r=0.1$. We want to take a look at the persistence diagram:

```
In [57]: import gudhi

from gudhi.clustering.tomato import Tomato

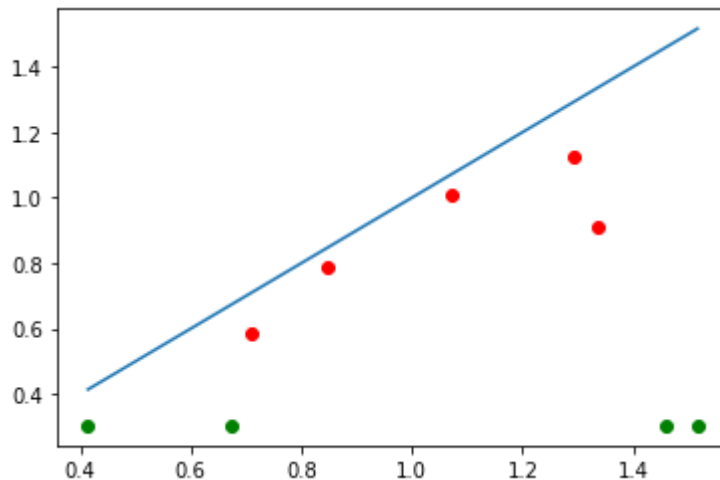
ex1 = Tomato(
    input_type="points",
    metric="euclidean",
    graph_type="radius",
    density_type="KDE",
    #n_clusters=2,
    r=0.1,
)

labels = ex1.fit_predict(p1)
print(labels)

print("\nThere are " + str(ex1.n_clusters_) + " initial clusters")
ex1.plot_diagram()
```

```
[1 4 1 6 3 1 0 2 3 1 3 0 3 0 0 4 1 4 1 4 1 2 0 3 1 4 3 1 4 1 2 1 3 3 1 6 0
 3 2 0 0 0 6 1 0 2 2 6 0 1 8 2 7 4 2 6 1 1 1 4 4 5 2 1 0 0 1 6 2 3 4 3 2 2
 3 1 1 2 1 0 6 2 4 1 0 3 2 1 1 3 2 0 4 3 2 3 0 3 3 0 1 4 0 0 3 0 3 6 3 4 5
 1 0 0 0 6 1 0 1 0 0 2 3 1 1 2 4 0 2 0 4 3 2 1 4 2 2 0 2 1 6 2 0 2 4 4 0 2
 0 1 3 5 2 4 4 1 1 1 2 4 1 1 3 3 0 5 1 0 1 4 3 1 4 2 1 1 4 1 3 1 6 1 2 2 0
 2 0 1 0 2 2 3 0 1 0 4 1 0 6 4]
```

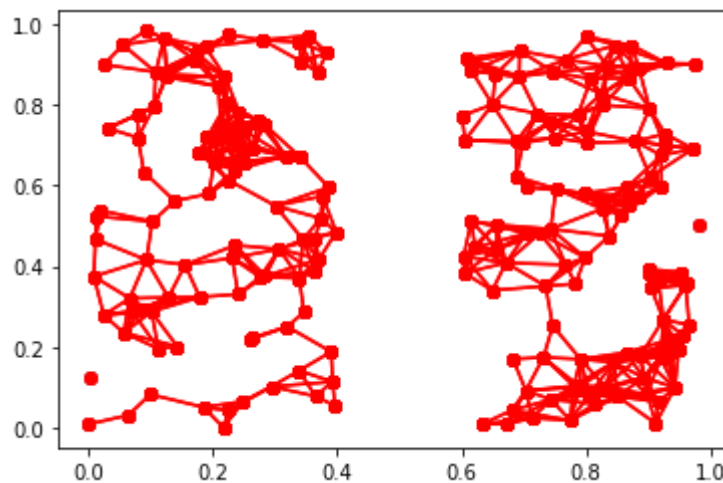
There are 9 initial clusters



Even if `n_clusters_` gives us 9 initial clusters (when we don't specify the parameter `n_clusters` in `Tomato` no merging occurs), we can see from the bottom-right that there are clearly two more prominent groups, but four connected components. Indeed, let's output the graph built on top of our data:

```
In [58]: from gudhi.point_cloud.knn import KNearestNeighbors
X = np.array(p1)
nbrs = KNearestNeighbors(k=30, return_distance= True)
indices, distances = nbrs.fit_transform(X)
plt.plot(X[:,0], X[:,1], 'o')
for i in indices:
    Y = np.zeros((2,2))
    for j in range(len(i)):
        if distances[int(i[0]), j] < 0.1:
            Y[0][0]= X[int(i[0])][0]
            Y[1][0]= X[int(i[0])][1]
            Y[0][1]= X[int(i[j])][0]
            Y[1][1]= X[int(i[j])][1]
            plt.plot(Y[0], Y[1], 'ro-')

plt.show()
```



Even if we know that "there are" two main clusters, we cannot force the algorithm to output them, because there is no way the algorithm can merge disconnected components. We don't have problems if we ask for a bigger number of clusters:

```
In [61]: ex1.n_clusters_ = 6
print(ex1.n_clusters_)
print(ex1.labels_)

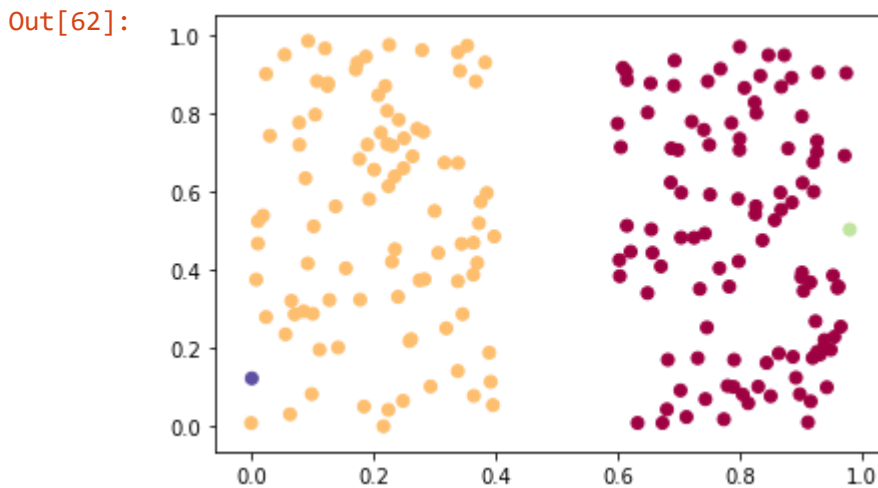
ex1.n_clusters_ = 2
print(ex1.n_clusters_)
print(ex1.labels_)
```

```
6
[1 1 1 1 3 1 0 2 3 1 3 0 3 0 0 1 1 1 1 1 2 0 3 1 1 3 1 1 1 2 1 3 3 1 1 0
 3 2 0 0 0 1 1 0 2 2 1 0 1 5 2 4 1 2 1 1 1 1 1 1 2 1 0 0 1 1 2 3 1 3 2 2
 3 1 1 2 1 0 1 2 1 1 0 3 2 1 1 3 2 0 1 3 2 3 0 3 3 0 1 1 0 0 3 0 3 1 3 1 1
 1 0 0 0 1 1 0 1 0 0 2 3 1 1 2 1 0 2 0 1 3 2 1 1 2 2 0 2 1 1 2 0 2 1 1 0 2
 0 1 3 1 2 1 1 1 1 2 1 1 1 3 3 0 1 1 0 1 1 3 1 1 2 1 1 1 1 3 1 1 1 2 2 0
 2 0 1 0 2 2 3 0 1 0 1 1 0 1 1]

4
[1 1 1 1 0 1 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1 0 0 0 1 1 0 1 1 1 0 1 0 0 1 1 0
 0 0 0 0 0 1 1 0 0 0 1 0 1 3 0 2 1 0 1 1 1 1 1 1 1 0 1 0 0 1 1 0 0 1 0 0 0
 0 1 1 0 1 0 1 0 1 1 0 0 0 1 1 0 0 0 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 1
 1 0 0 0 1 1 0 1 0 0 0 0 1 1 0 1 0 0 0 1 0 0 1 1 0 0 0 0 1 1 0 0 0 1 1 0 0
 0 1 0 1 0 1 1 1 1 1 0 1 1 1 0 0 0 1 1 0 1 1 0 1 1 1 0 1 1 1 1 0 1 1 1 0 0 0
 0 0 1 0 0 0 0 0 1 0 1 1 0 1 1]
```

Unsurprisingly, if we plot the points with different colors according to their labels, we don't get a very satisfying result:

```
In [62]: n = ex1.n_clusters_  
labels = ex1.labels_  
  
norm = plt.Normalize(vmin=0, vmax=n-1)  
  
ax.cla()  
ax.scatter(*zip(*p1), c=cmap(norm(labels)))  
fig
```



This is the reason why running the algorithm for different values of the parameters is a good idea, specially if the algorithm produces persistence diagrams with several green dots (i.e. connected components) near the bottom-left part (i.e. low, isolated peaks).

Here is the situation when we increase r to 0.15:

```

In [63]: ex1 = Tomato(
            input_type="points",
            metric="euclidean",
            graph_type="radius",
            density_type="KDE",
            n_clusters=2,
            r=0.13,
        )

n = ex1.n_clusters_
print("We obtain " + str(n) + " clusters.")
labels = ex1.fit_predict(p1)
print(ex1.labels_)

print("\nThe persistence diagram looks better, with just two connected components, and two prominent regions:")
ex1.plot_diagram()

print("\nThe graph over which the algorithm runs is:")

plt.plot(X[:,0], X[:,1], 'o')
for i in indices:
    Y = np.zeros((2,2))
    for j in range(len(i)):
        if distances[i[0]][j] < 0.15:
            Y[0][0]= X[i[0]][0]
            Y[1][0]= X[i[0]][1]
            Y[0][1]= X[i[j]][0]
            Y[1][1]= X[i[j]][1]
            plt.plot(Y[0], Y[1], 'ro-')

plt.show()

print("\nAnd the plot of the points according to their label is:")

norm = plt.Normalize(vmin=0, vmax=n-1)

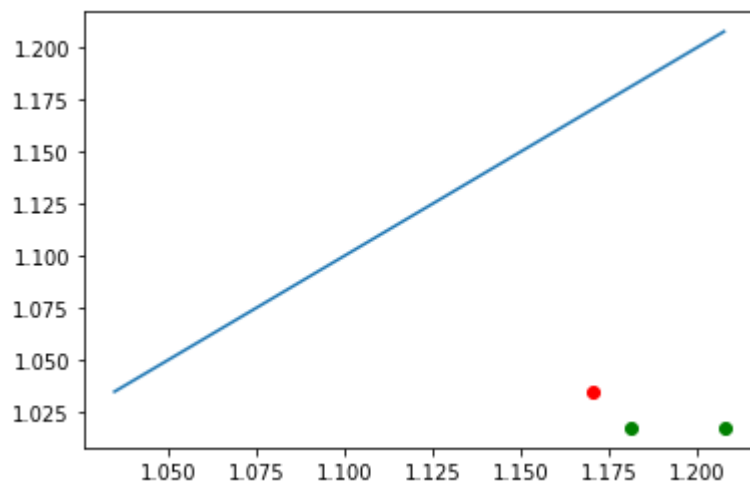
ax.cla()
ax.scatter(*zip(*p1), c=cmap(norm(labels)))
fig

```

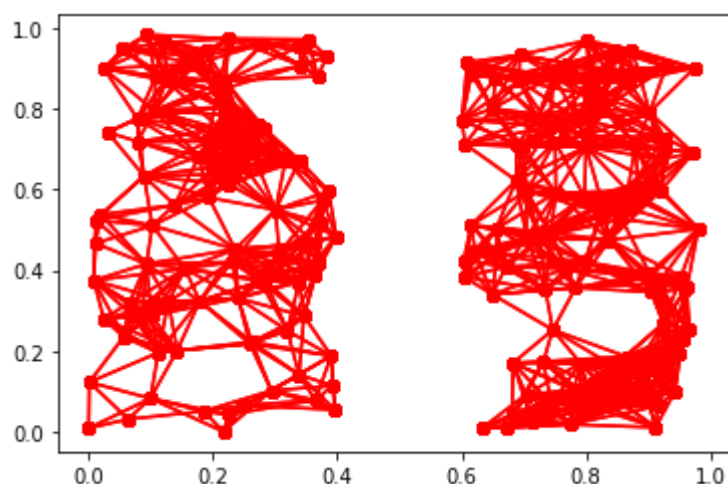
We obtain 2 clusters.

```
[1 1 1 1 0 1 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1 0 0 0 1 1 0 1 1 1 0 1 0 0 1 1 0
 0 0 0 0 0 1 1 0 0 0 1 0 1 1 0 0 1 0 1 1 1 1 1 1 1 0 1 0 0 1 1 0 0 1 0 0 0
 0 1 1 0 1 0 1 0 1 1 0 0 0 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 1
 1 0 0 0 1 1 0 1 0 0 0 0 1 1 0 1 0 0 0 1 0 0 1 1 0 0 0 0 1 1 0 0 0 1 1 0 0
 0 1 0 1 0 1 1 1 1 1 0 1 1 1 0 0 0 1 1 0 1 1 0 1 1 0 1 1 1 1 0 1 1 1 0 0 0
 0 0 1 0 0 0 0 0 1 0 1 1 0 1 1]
```

The persistence diagram looks better, with just two connected components, and two prominent regions:

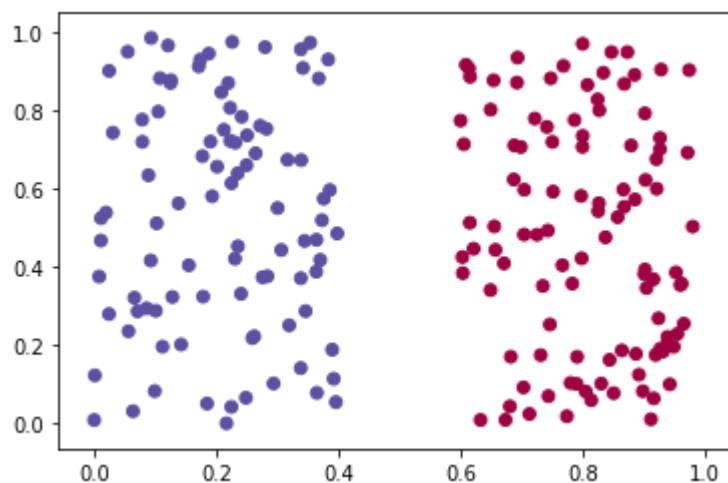


The graph over which the algorithm runs is:



And the plot of the points according to their label is:

Out[63]:



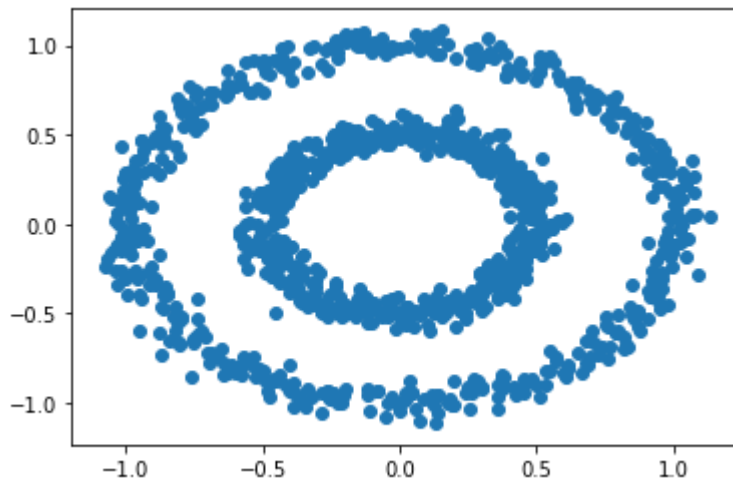
Example 2

We use now a rather typical example to test clustering algorithms: a point cloud sampled from two concentric circles:

```
In [97]: from sklearn import manifold, datasets
p2, y = datasets.make_circles(n_samples=1000, factor=.5, noise=.05)

ax.cla()
ax.scatter(*zip(*p2))
fig
```

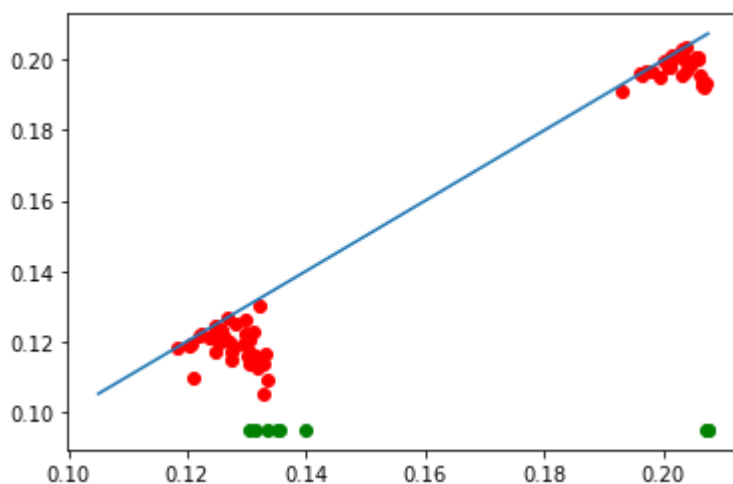
Out[97]:



It is well known that many clustering methods perform poorly with non-convex groupings of data, as the one above. This is not the case with the Tomato algorithm, which relies just on looking for "nearby" modes. We use now the k -NN graph construction, with $k=7$, and the KDE again, specifying some of its parameters now (for more information, check the Scikit-learn documentation):

```
In [98]: ex2 = Tomato(
    input_type="points",
    metric="euclidean",
    graph_type="knn",
    density_type="KDE",
    kde_params = {"bandwidth": 1.3, "kernel": "epanechnikov"},
    #n_clusters=2,
    k=7,
    eps=0.05,
)

ex2.fit_predict(p2)
ex2.plot_diagram()
```



The diagram is not specially obvious; if this happens, it is in general a good idea to run the algorithm with different values in the parameters.

We also see that there are several connected components, more specifically 9; a quick way to know how many of them we have is check the size of the attribute "max_weight_per_cc_":

```
In [99]: n = len(ex2.max_weight_per_cc_)
print("There are " + str(n) + " connected components")
```

There are 9 connected components

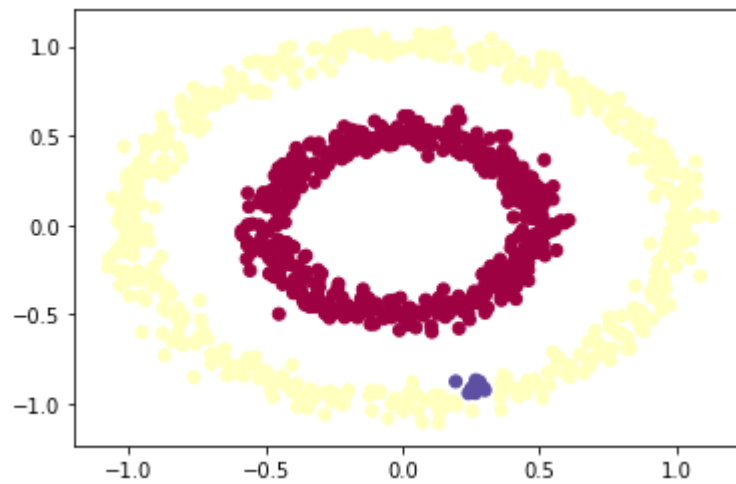
Let's plot these components:


```
In [103]: ex2.n_clusters_ = n
labels = ex2.labels_

norm = plt.Normalize(vmin=0, vmax=n-1)

ax.cla()
ax.scatter(*zip(*p2), c=cmap(norm(labels)))
fig
```

Out[103]:



A bit frustrating; this is "natural" consequence of the the k -NN graph being directed. We can "solve" this by symmetrizing the graph, although its effectiveness is uncertain. In this case it also makes sense to reduce k , as we add more edges:

```

In [102]: ex2 = Tomato(
            input_type="points",
            metric="euclidean",
            graph_type="knn",
            density_type="KDE",
            kde_params = {"bandwidth": 1.3, "kernel": "epanechnikov"},
            #n_clusters=2,
            k=5,
            symmetrize_graph = True,
            eps=0.05,
        )

ex2.fit_predict(p2)
ex2.plot_diagram()

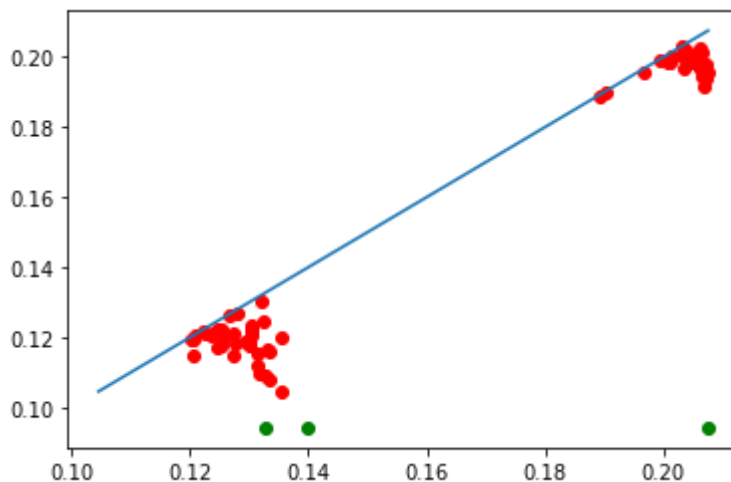
n = len(ex2.max_weight_per_cc_)
print("There are " + str(n) + " connected components")

ex2.n_clusters_ = n
labels = ex2.labels_

norm = plt.Normalize(vmin=0, vmax=n-1)

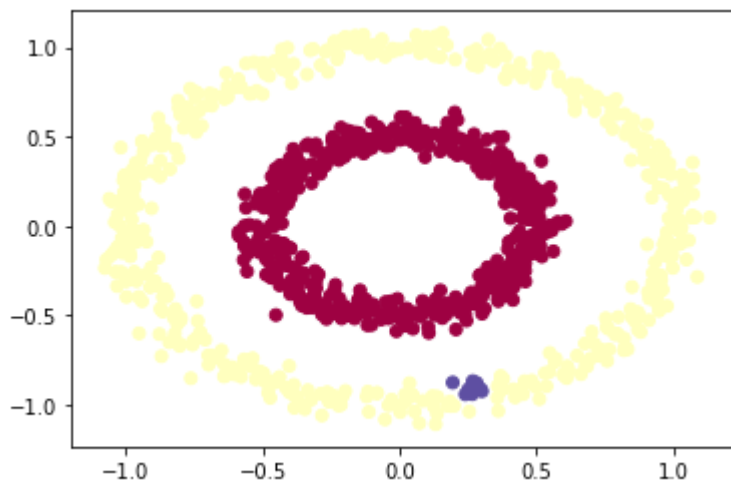
ax.cla()
ax.scatter(*zip(*p2), c=cmap(norm(labels)))
fig

```



There are 3 connected components

Out[102]:



In general, and intelligent way to proceed would be to run the algorithm for different values of k and the bandwidth λ , and see for which values we obtain "good" persistence diagrams, with "clearly prominent clusters". This is what we do below, where, for a fixed k and different values of λ , we compute the prominence of each point of the persistence diagram ($x - y$), and we plot the information, as well as the number of connected components (number under every vertical bar) in each case:

```

In [148]: for n_neigh in range(6,12):
            n_diagram = []
            x_diagram = []
            y_diagram = []
            cc = []
            y_cc = []
            bandwidth_values = [0.1, 2, 0.1]
            bandwidth = bandwidth_values[0]

            while bandwidth < bandwidth_values[1]:
                ex2 = Tomato(
                    input_type="points",
                    metric="euclidean",
                    graph_type="knn",
                    density_type="KDE",
                    kde_params = {"bandwidth": bandwidth, "kernel": "epanechnikov"},
                    #n_clusters=2,
                    k=n_neigh,
                    eps=0.05,
                )
                ex2.fit(p2)
                cc.append(str(len(ex2.max_weight_per_cc_)))
                init_clusters = len(ex2.diagram_)
                prominences = np.zeros(init_clusters)
                for i in range(init_clusters):
                    prominences[i] = ex2.diagram_[i,0] - ex2.diagram_[i,1]

                ##"Normalizing" prominences
                max_prom = np.max(prominences)
                for i in range(init_clusters):
                    prominences[i] /= max_prom

                n_diagram.append(prominences)
                bandwidth += bandwidth_values[2]

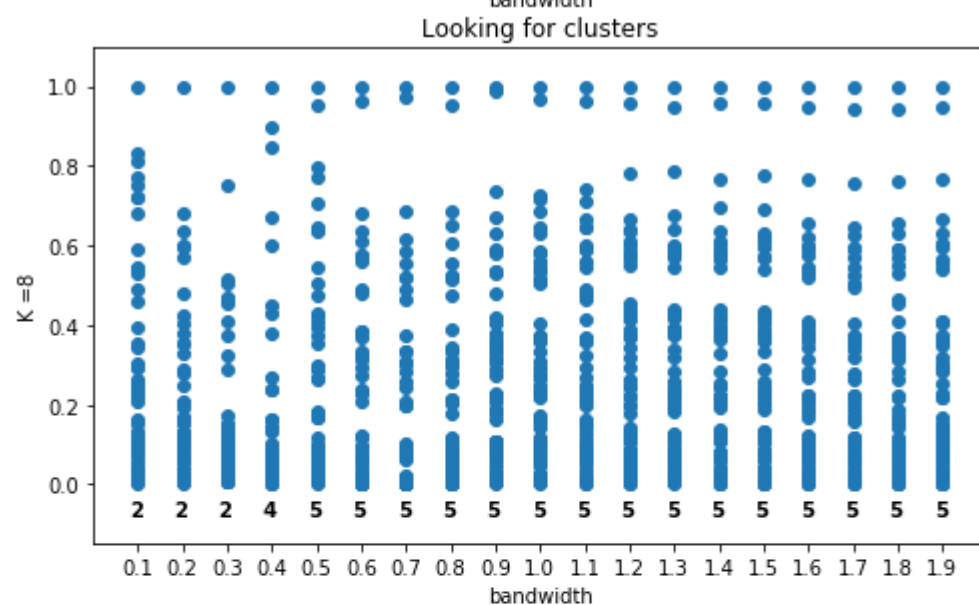
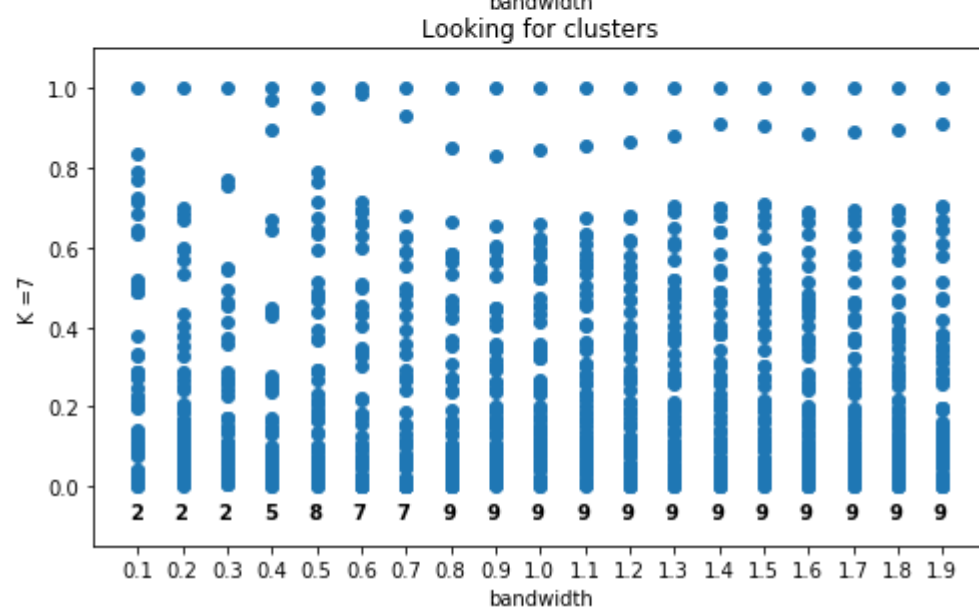
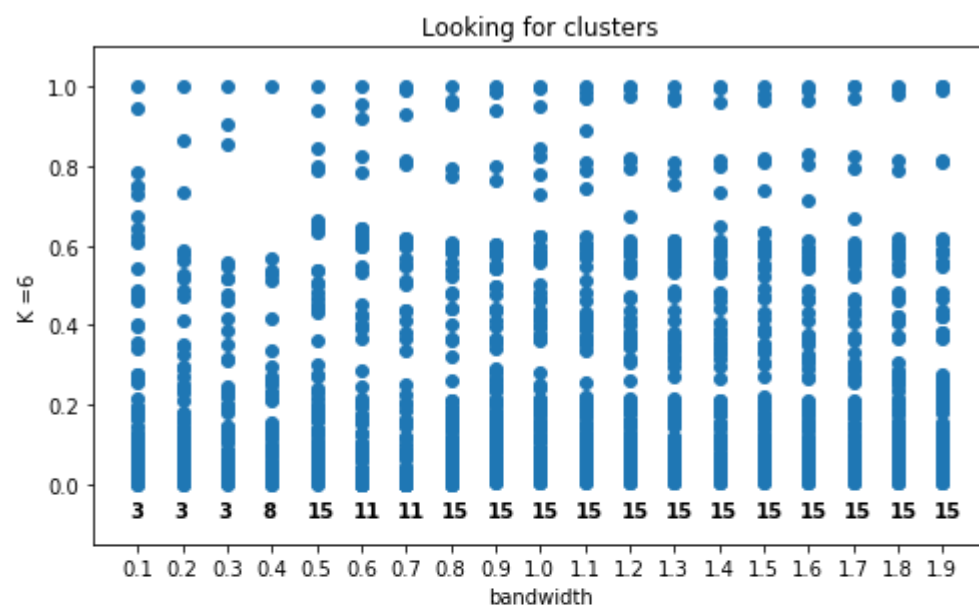
            for i in range(len(n_diagram)):
                for j in range(len(n_diagram[i])):
                    x_diagram.append(bandwidth_values[0] + i*bandwidth_values[2])
                    y_diagram.append(n_diagram[i][j])
                    y_cc.append(-0.08)

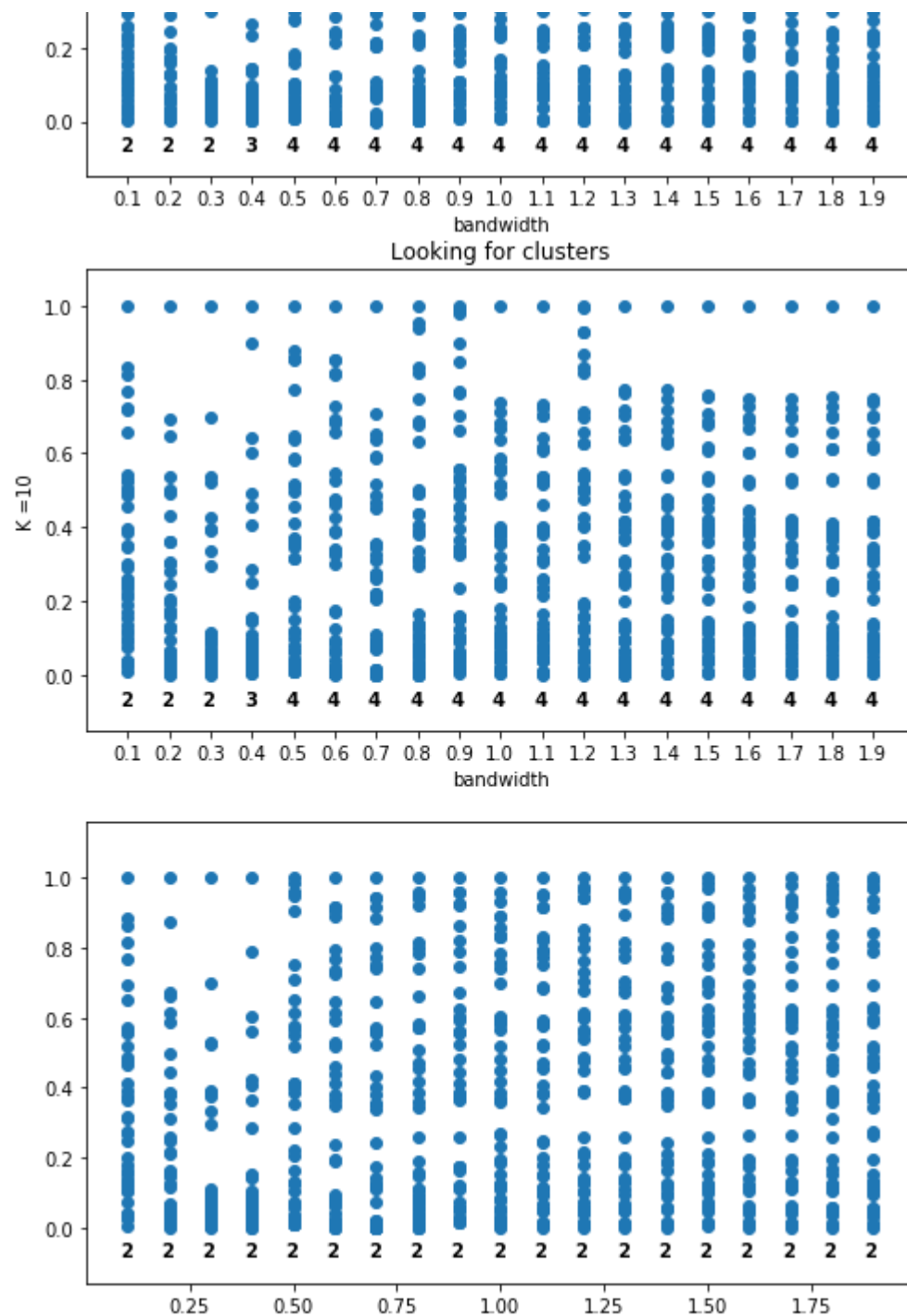
            plt.title('Looking for clusters')
            plt.axis('tight')
            plt.ylabel('K = ' + str(n_neigh-1))
            plt.ylim((-0.15, 1.1))
            plt.xlabel('bandwidth')
            plt.xticks(np.arange(bandwidth_values[0], bandwidth_values[1], bandwidth_values[2]
]))
            plt.subplot(6, 1, n_neigh-5)
            n = int((bandwidth_values[1]-bandwidth_values[0])/bandwidth_values[2]) + 1
            for i in range(n):
                plt.text(-0.02 + bandwidth_values[0] + i*bandwidth_values[2], y_cc[i], cc[i],
fontdict={'weight': 'bold', 'size': 10})

            plt.scatter(x_diagram, y_diagram)

fig = plt.gcf()
fig.set_size_inches(8, 32)
plt.show()

```





One can see, for example, that when the bandwidth is $\lambda = 0.3$, two more prominent clusters appear consistently, for all the last values of k , and we always get two connected components. If we run Tomato with these parameters, we obtain the "desired" result:

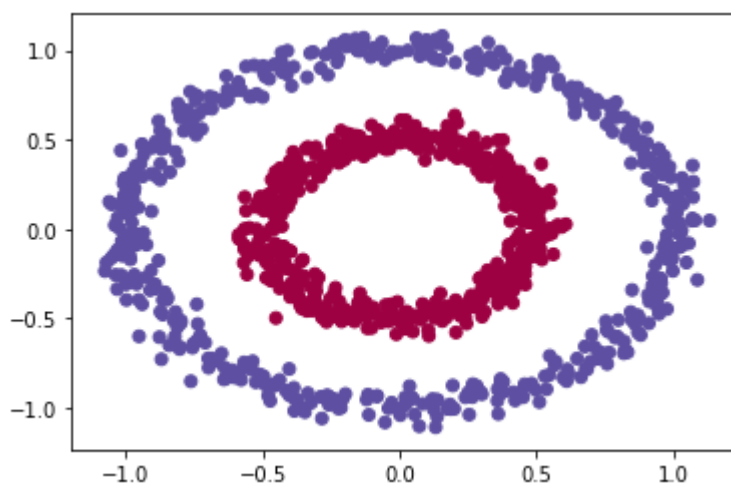
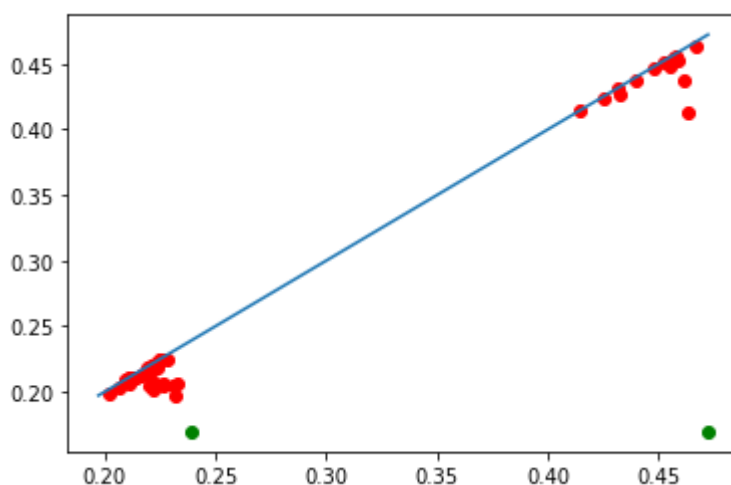
```
In [185]: ex2 = Tomato(
    input_type="points",
    metric="euclidean",
    graph_type="knn",
    density_type="KDE",
    kde_params = {"bandwidth": 0.3, "kernel": "epanechnikov"},
    n_clusters=2,
    k=9,
    eps=0.05,
)

labels = ex2.fit_predict(p2)
ex2.plot_diagram()

norm = plt.Normalize(vmin=0, vmax=1)

fig, ax = plt.subplots()

ax.cla()
ax.scatter(*zip(*p2), c=cmap(norm(labels)));
```



Example 3

We do now a rather spectacular example in 3D just to show the effectiveness of the algorithm to separate clusters with different shapes. We will generate, using points, a cube, a sphere, and a "swiss roll", together with some noise:

```

In [157]: import mpl_toolkits.mplot3d.axes3d as plt3
from sklearn.datasets import make_swiss_roll

fig3 = plt.figure()
ax = plt3.Axes3D(fig3)
ax.view_init(7, -70)

points_cube = 1000
points_sphere = 800
#points_line = 700
points_sr = 8000
points_noise = 2000

X1 = np.zeros((points_cube, 3))
for i in range(points_cube):
    X1[i,0], X1[i,1], X1[i,2] = rd.uniform(-2,2), rd.uniform(-2,2), rd.uniform(-2,2)

X2 = np.zeros((points_sphere, 3))
for i in range(points_sphere):
    X2[i,0], X2[i,1], X2[i,2] = rd.uniform(-1,1), rd.uniform(-1,1), rd.uniform(-1,1)
    X2[i,0], X2[i,1], X2[i,2] = 12 + 3*X2[i,0]/np.sqrt(X2[i,0]**2 + X2[i,1]**2 + X2[i,2]**2), 15 + 3*X2[i,1]/np.sqrt(X2[i,0]**2 + X2[i,1]**2 + X2[i,2]**2), -4 + 3*X2[i,2]/np.sqrt(X2[i,0]**2 + X2[i,1]**2 + X2[i,2]**2)

"""
X3 = np.zeros((points_line, 3))
for i in range(points_line):
    param = rd.uniform(-15, 15)
    X3[i,0], X3[i,1], X3[i,2] = 2 - param*0.7, 4 + param*0.7, 2 - param*0.6
    X3[:,0] += 0.02*np.random.randn(points_line)
    X3[:,1] += 0.02*np.random.randn(points_line)
    X3[:,2] += 0.02*np.random.randn(points_line)
"""

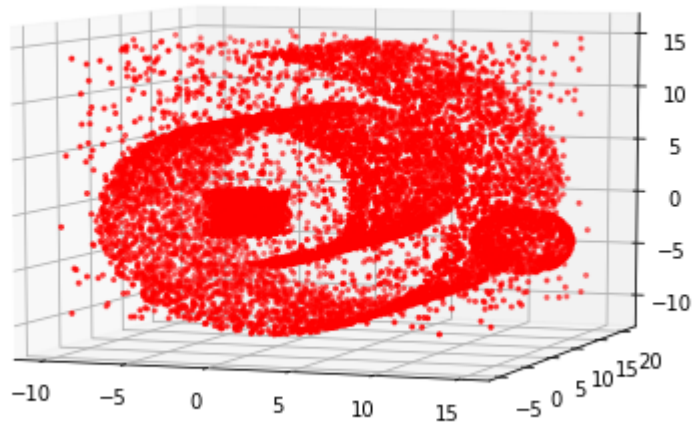
X4, _ = make_swiss_roll(n_samples=points_sr, noise=.05)

X5 = np.zeros((points_noise, 3))
for i in range(points_noise):
    X5[i,0], X5[i,1], X5[i,2] = rd.uniform(-10,15), rd.uniform(-5,20), rd.uniform(-10,15)

X = np.concatenate((X1,X2,X4,X5))
X = np.array(X)

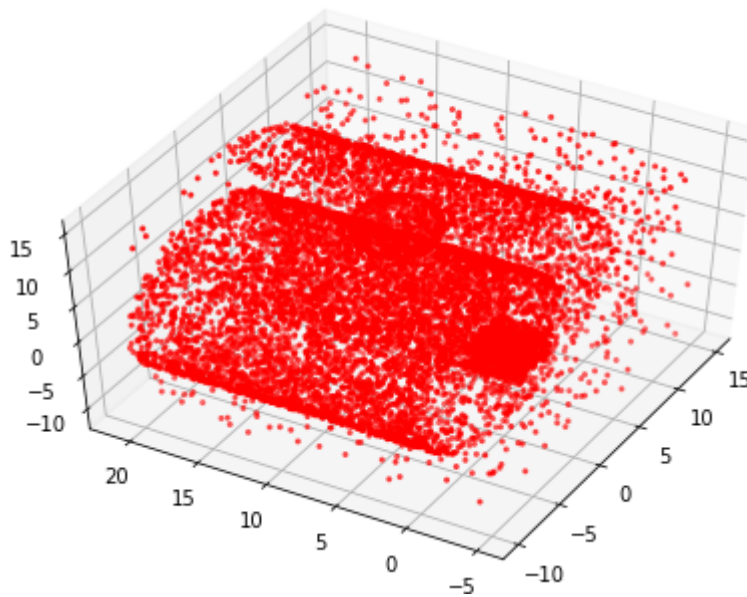
ax.scatter(X[:, 0], X[:, 1], X[:, 2], color="red", s=4);

```

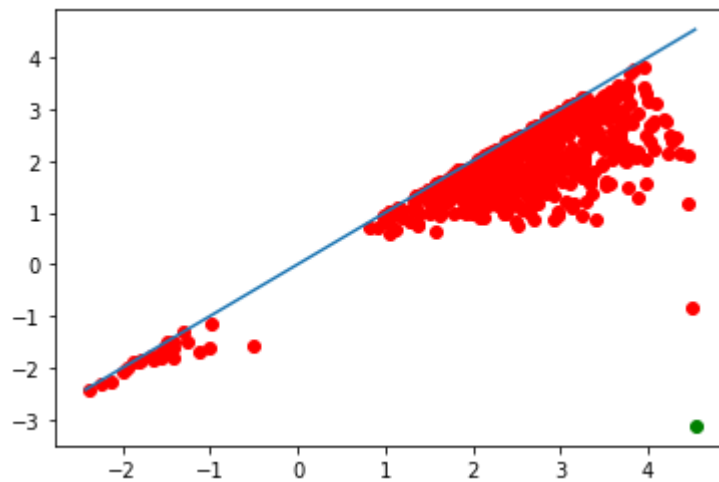
```
In [158]: ax.view_init(50, -150)  
fig3
```

Out[158]:



Let's run the algorithm with k -NN and the logDTM estimation. We also use the parameter `n_jobs=-1`, which becomes useful to increase the computational power when the size of our dataset becomes large, even though in our case we don't have an specially high number of points:

```
In [159]: ex3 = Tomato(  
    input_type="points",  
    metric="euclidean",  
    graph_type="knn",  
    density_type="logDTM",  
    #n_clusters=2,  
    #symmetrize_graph= True,  
    k=9,  
    n_jobs=-1,  
    )  
  
ex3.fit(X)  
ex3.plot_diagram()  
print(ex3.labels_)
```



```
[158  88 158 ... 436 382 174]
```

We see 2-3 prominent clusters in the persistence diagram. We can "identify" the noise by checking which points have a low estimate, and creating a new label. We plot the result at the end:

```

In [180]: ex3.n_clusters_ = 3
          label = ex3.labels_

          for i in range(len(X)):
              if ex3.weights_[i] < 0.5:
                  label[i] = 3

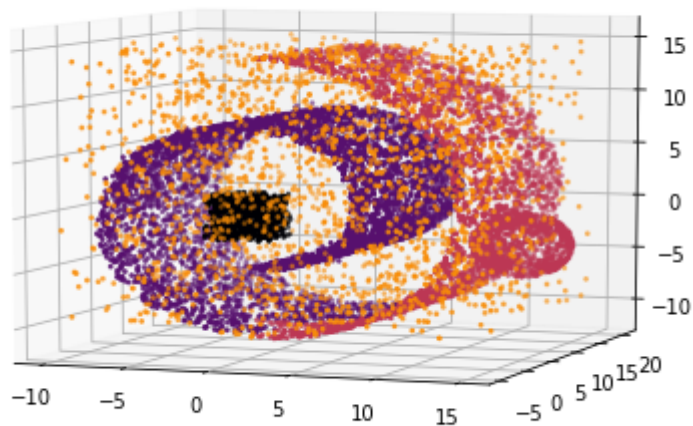
          print(label)

          fig3 = plt.figure()
          ax = plt3.Axes3D(fig3)
          ax.view_init(7, -70)

          for l in np.unique(label):
              ax.scatter(X[label == l, 0], X[label == l, 1], X[label == l, 2],
                          color=plt.cm.inferno(np.float(l) / np.max(label + 1)),
                          s=3)

```

[0 0 0 ... 3 3 3]

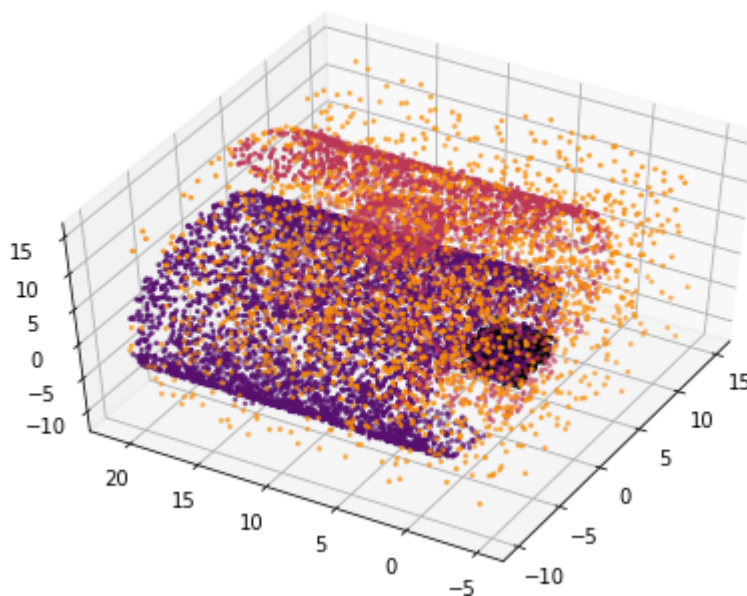


```

In [181]: ax.view_init(50, -150)
          fig3

```

Out[181]:



The swiss roll is not completely clustered and it gets separated into two regions due to the presence of the sphere; we cannot expect our algorithm to distinguish them properly with an intersection so noticeable. The result with two clusters is also quite satisfactory and more realistic, with the whole spiral and the sphere clustered together. We also see that, in both cases, the noise is quite properly identified:

```
In [182]: ex3.n_clusters_ = 2
          label = ex3.labels_

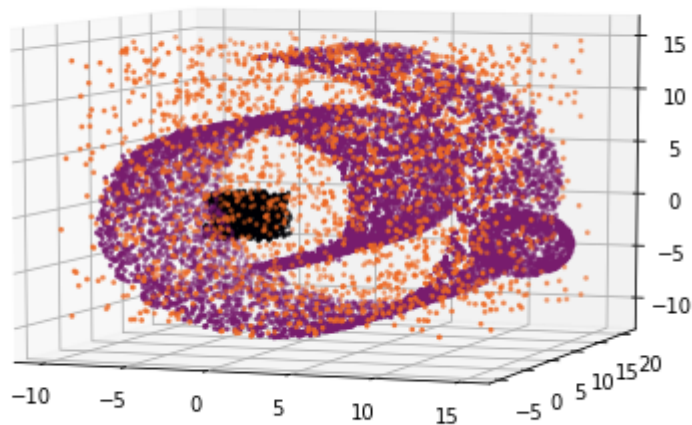
          for i in range(len(X)):
              if ex3.weights_[i] < 0.5:
                  label[i] = 2

          print(label)

          fig3 = plt.figure()
          ax = plt3.Axes3D(fig3)
          ax.view_init(7, -70)

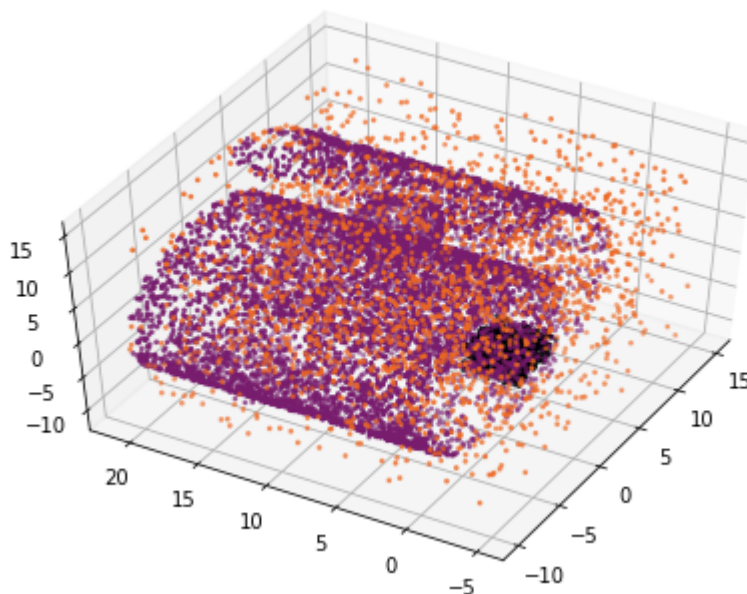
          for l in np.unique(label):
              ax.scatter(X[label == l, 0], X[label == l, 1], X[label == l, 2],
                          color=plt.cm.inferno(np.float(l) / np.max(label + 1)),
                          s=3)
```

```
[0 0 0 ... 2 2 2]
```



```
In [183]: ax.view_init(50, -150)
fig3
```

Out[183]:



Example 4

In this example we explore the case in which we don't give the coordinates of the points directly, but the distances between them.

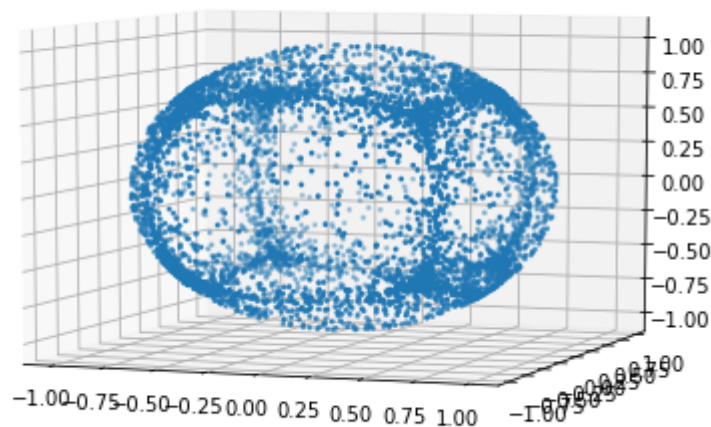
To do so, we sample a set of points over the unit sphere, but not uniformly: we sample them first in the cube $1 \times 1 \times 1$ using a sigmoid function in each variable to concentrate them near the vertices and edges of the cube, and then we normalize them. This creates naturally regions of the sphere with more points, more specifically the directions pointing towards the vertices and edges of the cube:

```
In [188]: def sample_spherical(npoints):
    vec = []
    vec.append(-0.5 + 1/(1 + np.exp(-5*np.random.uniform(-1,1, npoints))))
    vec.append(-0.5 + 1/(1 + np.exp(-5*np.random.uniform(-1,1, npoints))))
    vec.append(-0.5 + 1/(1 + np.exp(-5*np.random.uniform(-1,1, npoints))))
    vec /= np.linalg.norm(vec, axis=0)
    return vec

npoints = 6000
points = sample_spherical(npoints)

fig3 = plt.figure()
ax = plt3.Axes3D(fig3)
ax.view_init(7, -70)

ax.scatter(points[0,:], points[1,:], points[2:], s=3);
```



We compute now the pairwise distances between all the points, using the "spherical" distance d_S : the distance between two points on the surface of a unit sphere with coordinates $a = (a_1, a_2, a_3)$ and $b = (b_1, b_2, b_3)$ is given by the formula:

$$d_S(a, b) = \arccos(a_1b_1 + a_2b_2 + a_3b_3)$$

As we don't have many points, we can compute all pairwise distances without much problem:

```
In [189]: distance_matrix = np.zeros((npoints, npoints))

for i in range(npoints):
    distance_matrix[i,i]= 0
    for j in range(i+1, npoints):
        distance_matrix[i,j] = np.arccos(points[0,i]*points[0,j] + points[1,i]*points
[1,j] + points[2,i]*points[2,j])
        distance_matrix[j,i] = distance_matrix[i,j]

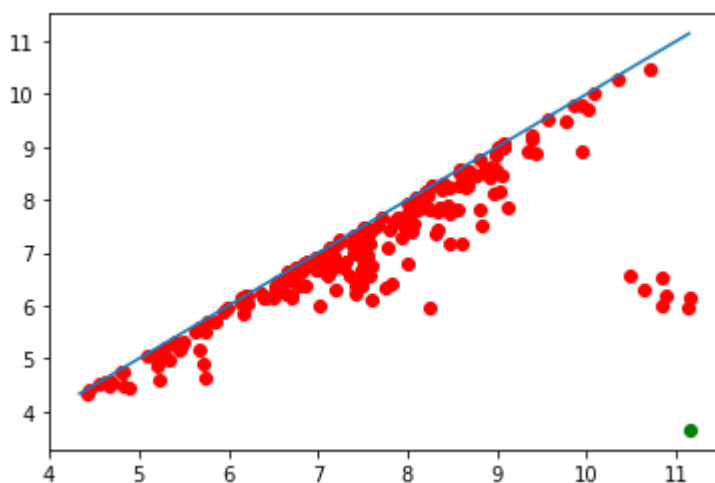
print(distance_matrix)

[[0.          1.1558255  1.47491536 ... 2.68105662 0.94665096 0.8865952 ]
 [1.1558255  0.          1.83518165 ... 1.77024725 1.71115835 1.14674659]
 [1.47491536 1.83518165 0.          ... 2.12567266 2.05734047 0.7744797 ]
 ...
 [2.68105662 1.77024725 2.12567266 ... 0.          1.88764992 2.57444621]
 [0.94665096 1.71115835 2.05734047 ... 1.88764992 0.          1.80283258]
 [0.8865952  1.14674659 0.7744797  ... 2.57444621 1.80283258 0.          ]]
```

KDE and logKDE use the already-built Scikit-learn library and we cannot use them for a precomputed distance matrix. We use logDTM insted of DTM to make the persistance diagram look more clear:

```
In [190]: ex4 = Tomato(
    input_type="points",
    metric="precomputed",
    graph_type="knn",
    density_type="logDTM",
    #n_clusters=2,
    k=10,
)

ex4.fit(distance_matrix)
ex4.plot_diagram()
```

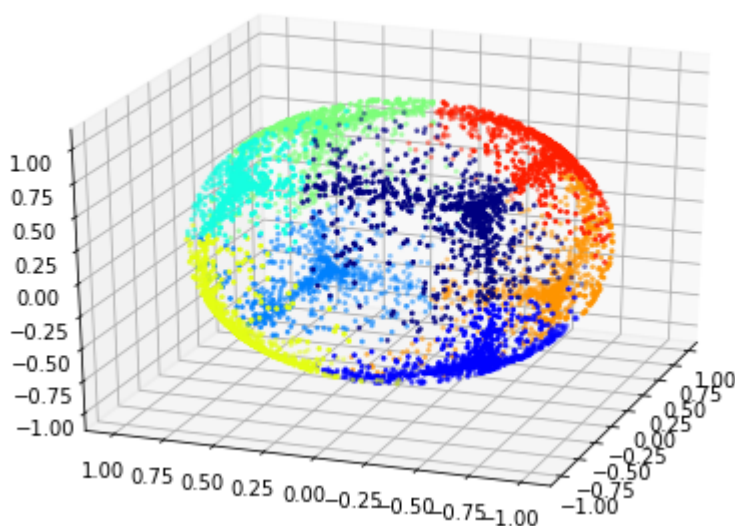


There are 8 clear clusters, a quite expected result:

```
In [191]: ex4.n_clusters_ = 8
label = ex4.labels_

fig3 = plt.figure()
ax = plt3.Axes3D(fig3)
ax.view_init(25, -160)

for l in np.unique(label):
    ax.scatter(points[0, label == l], points[1, label == l], points[2, label == l],
               color=plt.cm.jet(np.float(l) / np.max(label + 1)),
               s=3)
```



Example 5

We do another easy example just to get used to other input formats to our algorithm. In this one we will input ourselves the weights of the points as well as a neighboring graph, which will just be a rectangular mesh in the square 10x10. For the weights, we will be using the function:

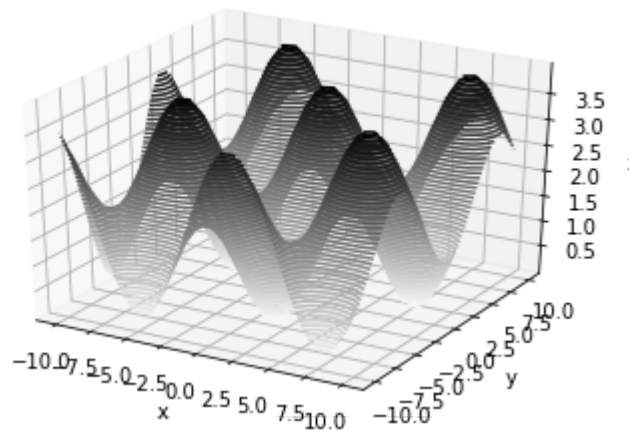
$$f(x, y) = \sin\left(\frac{x+y}{2}\right) + \cos\left(\frac{x-y}{2}\right),$$

plotted below. In this setting, our algorithm will be just looking for basins of attraction of our function.


```
In [208]: def f(x, y):
            return 2+ np.sin(0.5*(x+y)) + np.cos(0.5*(x-y))

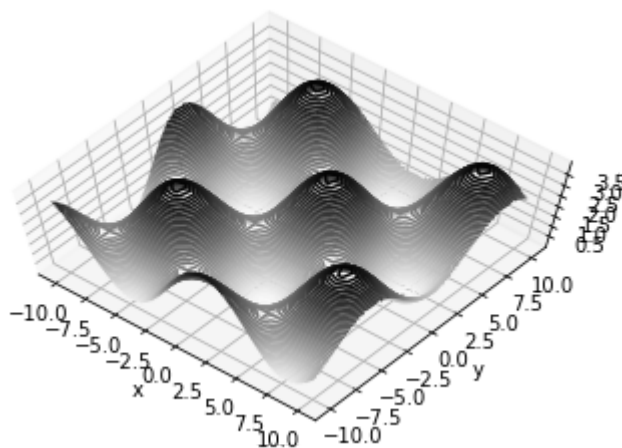
x = np.linspace(-10, 10, 30)
y = np.linspace(-10, 10, 30)
X, Y = np.meshgrid(x, y)
Z = f(X, Y)

fig = plt.figure()
ax = plt.axes(projection='3d')
ax.contour3D(X, Y, Z, 50, cmap='binary')
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z');
```



```
In [198]: ax.view_init(70, -50)
fig
```

Out[198]:



And now the points, with the neighboring graph:

```
In [200]: size_mesh = 30
points = np.zeros((2, size_mesh**2))
arange = np.linspace(-10., 10., size_mesh)

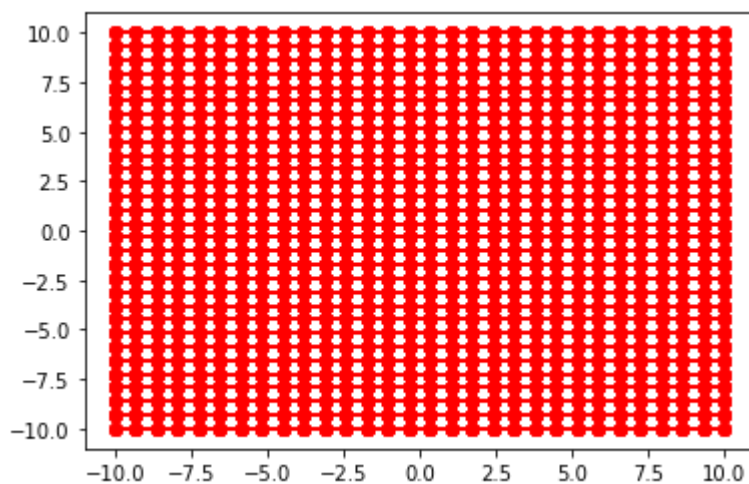
#Coordinates of the points
for i in range(size_mesh):
    for j in range(size_mesh):
        points[0][i*size_mesh + j] = arange[i]
        points[1][i*size_mesh + j] = arange[j]

#Neighboring graph
neigh_graph = []
for i in range(size_mesh):
    for j in range(size_mesh):
        neigh = []
        if i > 0:
            neigh.append((i-1)*size_mesh + j)
        if i < size_mesh -1:
            neigh.append((i+1)*size_mesh + j)
        if j > 0:
            neigh.append(i*size_mesh + j-1)
        if j < size_mesh -1:
            neigh.append(i*size_mesh + j+1)
        neigh_graph.append(neigh)
```

```
In [201]: #Drawing the graph
plt.plot(points[0,:], points[1,:], 'o', markersize=2)

for i in range(len(neigh_graph)):
    Y = np.zeros((2,2))
    for j in neigh_graph[i]:
        Y[0][0]= points[0][i]
        Y[1][0]= points[1][i]
        Y[0][1]= points[0][j]
        Y[1][1]= points[1][j]
        plt.plot(Y[0], Y[1], 'ro-', linewidth=2)

plt.show()
```



We now associate the weights to the different points according to f , and run the Tomato algorithm to compute the basins of attraction:

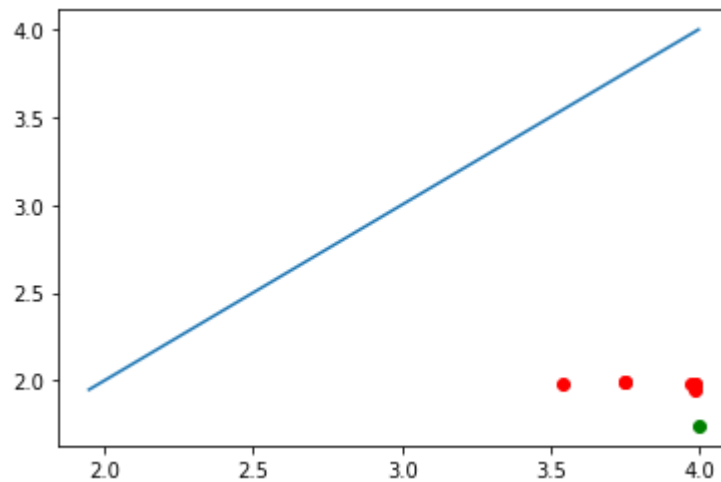
```

In [202]: #We associate the weights
weights = np.zeros(size_mesh**2)
for i in range(size_mesh**2):
    weights[i] = f(points[0][i], points[1][i])

#We run Tomato
ex5 = Tomato(
    graph_type = "manual",
    density_type = "manual"
)

ex5.fit(neigh_graph, weights= weights)
ex5.plot_diagram()
print(ex5.diagram_)

```



```

[[3.75212014  1.99409954]
 [3.75212014  1.99409954]
 [3.98535762  1.98244992]
 [3.98535762  1.98244992]
 [3.9737506   1.9824371 ]
 [3.54402111  1.97677169]
 [3.9882662   1.94776161]]

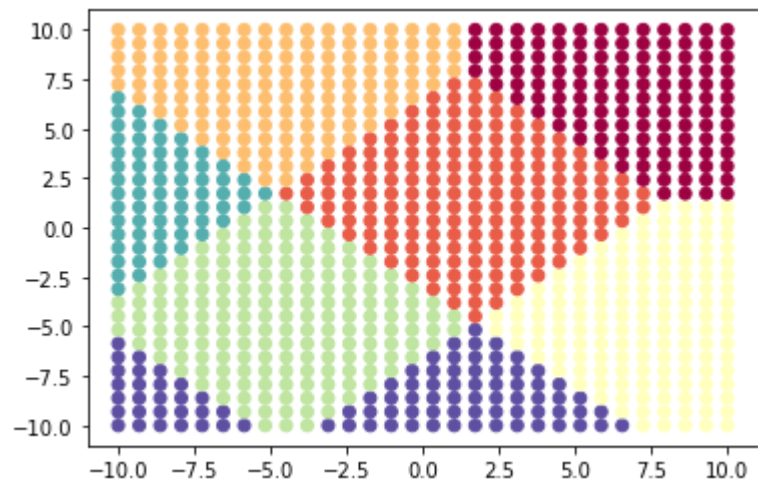
```

```
In [206]: ex5.n_clusters = 7
labels = ex5.fit_predict(neigh_graph, weights= weights)

norm = plt.Normalize(vmin=0, vmax=6)

fig, ax = plt.subplots();

ax.cla()
ax.scatter(points[0,:], points[1,:], c=cmap(norm(labels)));
```



```

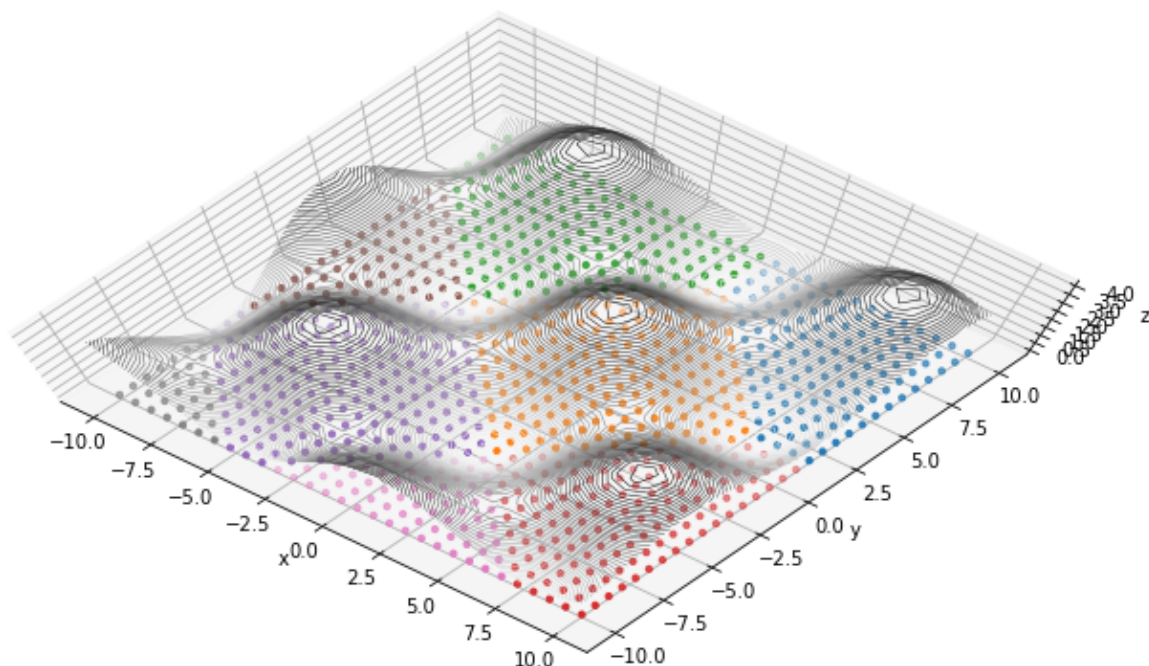
In [209]: fig = plt.figure()
ax = plt.axes(projection='3d')
ax.contour3D(X, Y, Z, 50, cmap='binary', linewidths=0.5);
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z');

points3d = np.zeros((3, size_mesh**2))
for i in range(size_mesh**2):
    points3d[0, i] = points[0,i]
    points3d[1, i] = points[1,i]

z_coord = np.zeros(size_mesh**2)
for l in np.unique(labels):
    ax.scatter(points3d[0, labels == l], points3d[1, labels == l], points3d[2, labels
== l], s=10)

ax.view_init(80, -50)
fig = plt.gcf()
fig.set_size_inches(12,7)
plt.show()

```

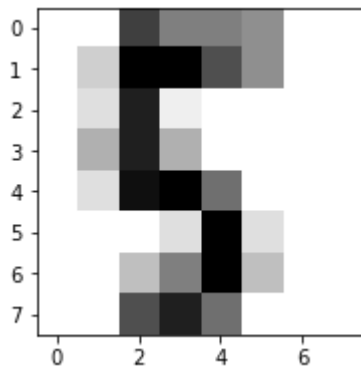


Example 6

In this last example, closer to the kind of datasets we could find in real life, we will work with the famous "Digits dataset", containing 1797 observations each with 64 features: each entry represents a (highly compressed) hand-written digit in a 8x8 grid, where each cell can vary from 0 to 16, representing its opacity. Naturally, the dataset also contains the correct labels of each instance: a number from 0 to 9, the one written in the grid.

```
In [987]: #Load the digits dataset
digits = datasets.load_digits()

#Display the 25th digit
plt.figure(1, figsize=(3, 3))
plt.imshow(digits.images[25], cmap=plt.cm.gray_r, interpolation='nearest')
plt.show()
```



It is well known the difficulty of performing data science algorithms in high dimension, and clustering is not an exception; in fact, it is a process particularly sensitive to numerical data being sparse. Thus, even with dimensionality reduction techniques, it's not a good idea to expect a brilliant performance of our algorithm in this setting. In any case, it is interesting to see what kind of results we get. The results of other clustering methods over this dataset can be found in [4].

```
In [1056]: digits, real_label = datasets.load_digits(return_X_y=True)

print(digits)
print(real_label)

[[ 0.  0.  5. ...  0.  0.  0.]
 [ 0.  0.  0. ... 10.  0.  0.]
 [ 0.  0.  0. ... 16.  9.  0.]
 ...
 [ 0.  0.  1. ...  6.  0.  0.]
 [ 0.  0.  2. ... 12.  0.  0.]
 [ 0.  0. 10. ... 12.  1.  0.]]
[0 1 2 ... 8 9 8]
```

We can embed the dataset in the plane by using PCA dimensionality reduction. We observe that, with that reduction level, the different clusters of numbers are somewhat distinguishable, but there is also considerable overlapping:

```
In [1009]: from sklearn.decomposition import PCA

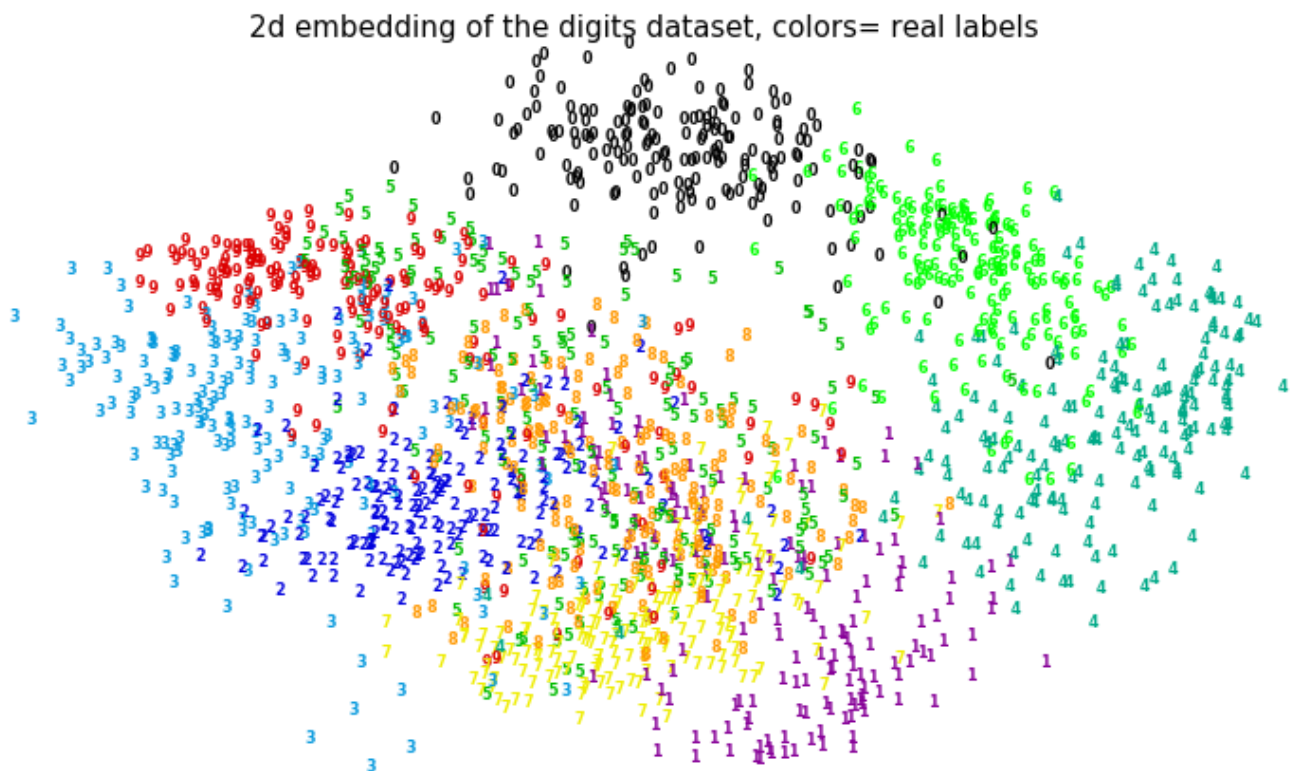
pca = PCA(n_components=2)
digits_red = pca.fit_transform(digits)

def plot_clustering(X_red, labels, title=None):
    x_min, x_max = np.min(X_red, axis=0), np.max(X_red, axis=0)
    X_red = (X_red - x_min) / (x_max - x_min)

    plt.figure(figsize=(6, 4))
    for i in range(X_red.shape[0]):
        plt.text(X_red[i, 0], X_red[i, 1], str(y[i]),
                color=plt.cm.nipy_spectral(labels[i] / 10.),
                fontdict={'weight': 'bold', 'size': 8})

    plt.xticks([])
    plt.yticks([])
    if title is not None:
        plt.title(title, size=15)
    plt.axis('off')
    fig = plt.gcf()
    fig.set_size_inches(12,7)

plot_clustering(digits_red, real_label, title = "2d embedding of the digits dataset,
colors= real labels")
```

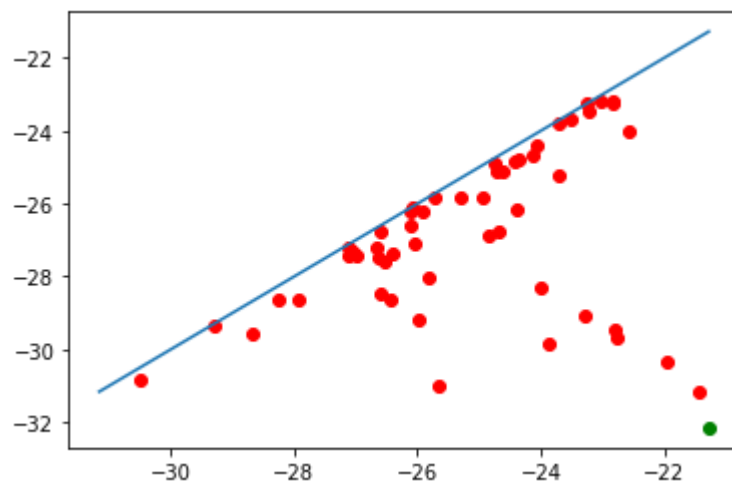


It's useless to try to run the algorithm without doing any kind of dimensionality reduction first: accurate density estimation is almost always unsuccessful with highly sparse data. We can try to use our algorithm after killing some dimensions first. With our dataset, after some experimentation, when there are 11 dimensions left DTM density estimation looks quite well:

```
In [1110]: pca = PCA(n_components=11)
digits_red = pca.fit_transform(digits)

ex6 = Tomato(
    input_type="points",
    metric="euclidean",
    graph_type="knn",
    density_type="logDTM",
    n_clusters=10,
    k=9,
)

ex6.fit(digits_red)
ex6.plot_diagram()
```

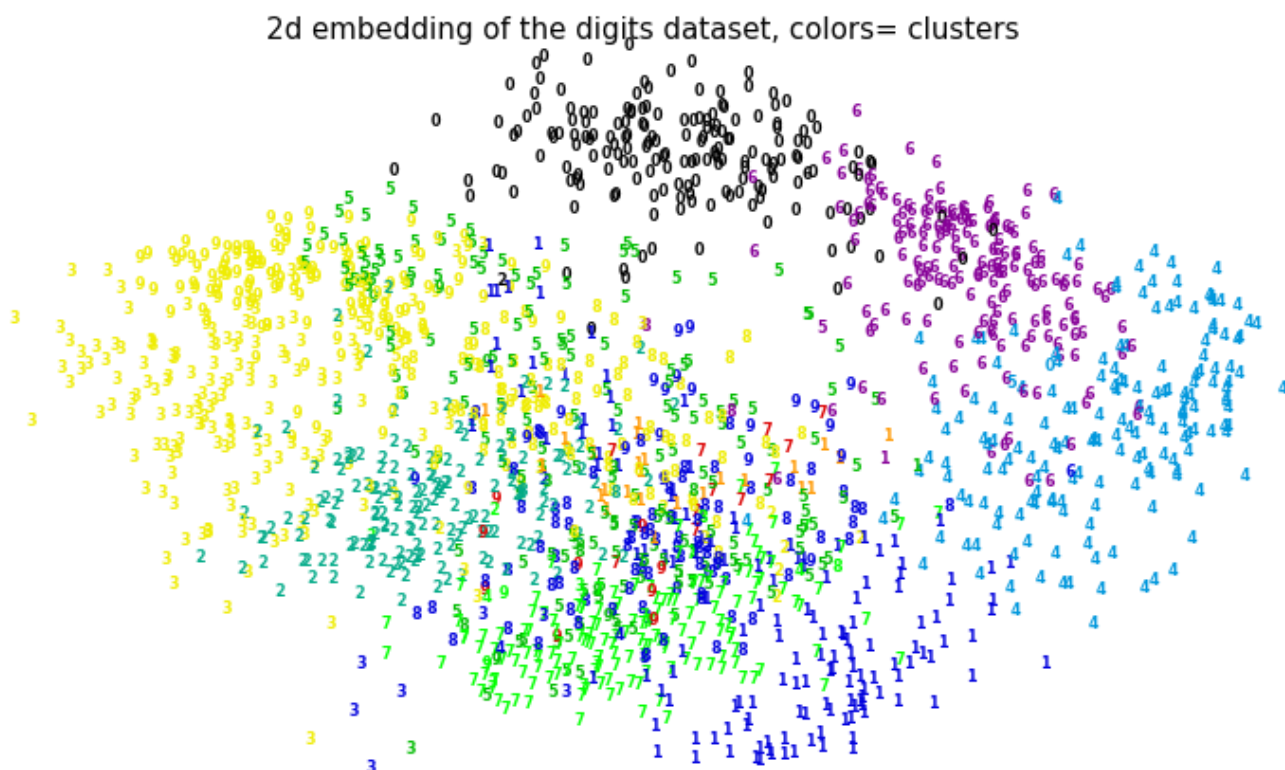


It looks like the algorithm found "naturally" 9-10 clusters, let's plot these 10 groups in 2D:


```
In [1100]: labels = ex6.fit_predict(digits_red)

pca = PCA(n_components=2)
digits_red = pca.fit_transform(digits_red)

plot_clustering(digits_red, labels, title = "2d embedding of the digits dataset, colors= clusters")
```



The result looks surprisingly good, actually.

A way to measure the matching degree consists in computing the vector $10 \cdot \text{real_labels} + \text{clustering_labels}$, which takes values $\in \{0, \dots, 99\}$, and then counting the number of times each number appears. In a perfect classification, only 10 values would appear, more specifically $10 \cdot i + \text{label}_i$, with $i \in \{0, \dots, 10\}$; in a decent clustering, we should at least see some clearly more prominent values, which is indeed what happens in our case!

```

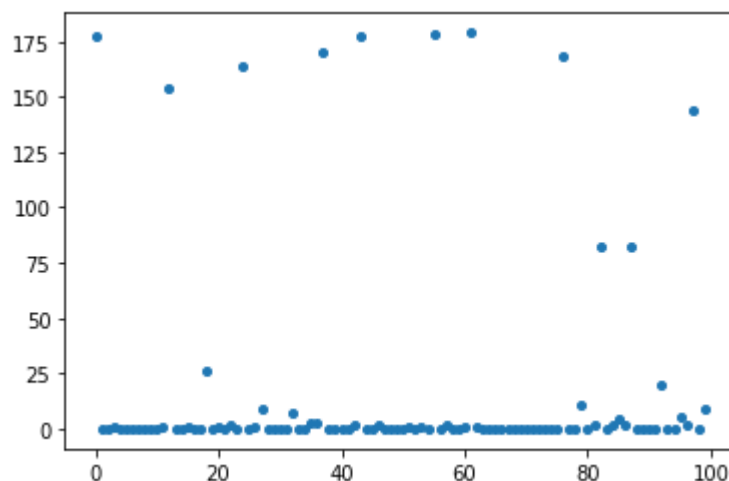
In [1111]: vect_count = 10*real_label + labels
print(vect_count)

count = np.zeros((2,100))
for i in range(100):
    count[0,i]= i
for i in vect_count:
    count[1,i] += 1

fig, ax = plt.subplots();
ax.cla()
ax.scatter(count[0,:], count[1,:], s=16);

```

```
[ 0 12 27 ... 82 97 87]
```



From this graph it looks like most of the values have been properly grouped. Only the value 8 looks more mismatched. We can also get an idea about how the numbers have been grouped or labeled using a table, and checking the columns and rows: each of them should only contain one "big" value:

```
In [1102]: import pandas as pd
data = [[1, 2], [3, 4]]
pd.DataFrame(data, columns=["Foo", "Bar"])

table = []
for i in range(10):
    row = []
    #row.append(i)
    for i in range(i*10, i*10 +10):
        row.append(count[1, i])
    table.append(row)

pd.DataFrame(table, columns=[ 'Label 0', 'Label 1', 'Label 2', 'Label 3', 'Label 4',
'Label 5', 'Label 6', 'Label 7', 'Label 8', 'Label 9'])
```

Out[1102]:

	Label 0	Label 1	Label 2	Label 3	Label 4	Label 5	Label 6	Label 7	Label 8	Label 9
0	177.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	1.0	154.0	0.0	0.0	1.0	0.0	0.0	26.0	0.0
2	1.0	0.0	2.0	0.0	164.0	0.0	1.0	9.0	0.0	0.0
3	0.0	0.0	7.0	0.0	0.0	3.0	3.0	170.0	0.0	0.0
4	0.0	0.0	2.0	177.0	0.0	0.0	2.0	0.0	0.0	0.0
5	0.0	1.0	0.0	1.0	0.0	178.0	0.0	2.0	0.0	0.0
6	1.0	179.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	168.0	0.0	0.0	11.0
8	0.0	2.0	82.0	0.0	2.0	4.0	2.0	82.0	0.0	0.0
9	0.0	0.0	20.0	0.0	0.0	5.0	2.0	144.0	0.0	9.0

The clustering has been quite successful. In any case, the numbers 3's and 9's have been almost completely clustered together (which is not that suprising, given the low resolution of the dataset), and the number 8 is almost evenly divided between labels 2 (with the 1's) and 7 (with the 3's and 9's), again not very surprising.

References

[1]. Frédéric CHAZAL et al., *Persistence-Based Clustering in Riemannian Manifolds* . In :Journal of the ACM 60 (juin 2011).doi:10.1145/1998196.1998212 : URL: <https://hal.inria.fr/inria-00389390/document> (<https://hal.inria.fr/inria-00389390/document>)

[2]. Tutorial for DTM: <https://github.com/GUDHI/TDA-tutorial/blob/master/Tuto-GUDHI-DTM-filtrations.ipynb> (<https://github.com/GUDHI/TDA-tutorial/blob/master/Tuto-GUDHI-DTM-filtrations.ipynb>)

[3]. Gérard BIAU, Frédéric CHAZAL, David COHEN-STEINER, Luc DEVROYE, Carlos RODRIGUEZ. *A Weighted k-Nearest Neighbor Density Estimate for Geometric Inference* . 2011. ffinria-00560623v1 : URL: <http://luc.devroye.org/BiauChazalCohenDevroyeRodriguez-kNN-2EJS-2011.pdf> (<http://luc.devroye.org/BiauChazalCohenDevroyeRodriguez-kNN-2EJS-2011.pdf>)

[4] Various Agglomerative Clustering on a 2D embedding of digits: https://scikit-learn.org/stable/auto_examples/cluster/plot_digits_linkage.html#sphx-glr-auto-examples-cluster-plot-digits-linkage-py (https://scikit-learn.org/stable/auto_examples/cluster/plot_digits_linkage.html#sphx-glr-auto-examples-cluster-plot-digits-linkage-py).

Dualités de Koszul algébrique, opéradique et propéradique

Mémoire de M2

Clovis Chabertier

Sous la direction de Salim Rivière

2020

Dans les catégories abéliennes [Gro57] possédant assez de projectifs (resp. injectifs), une façon d'obtenir des invariants, dits homologiques, est de trouver un foncteur exact à droite (resp. gauche) et de regarder l'homologie du foncteur évalué sur une résolution projective (resp. injective), ce sont les foncteurs dérivés. Cependant, $As-alg$ n'est pas abélienne, mais elle possède une structure de modèle [Hin97], qui permet quand même d'obtenir des invariants homologiques. Ce sont les résolutions cofibrantes dans les catégories de modèle qui jouent le rôle des résolutions projectives dans les catégories abéliennes, et les adjonctions de Quillen qui jouent celui des foncteurs exacts d'un côté.

Si on veut obtenir de façon systématique une résolution cofibrante $X_A \xrightarrow{\sim} A$, on peut chercher un endofoncteur $\mathcal{F} : \mathcal{C} \rightarrow \mathcal{C}$, muni d'une transformation naturelle $\eta_\bullet : \mathcal{F} \rightarrow Id_{\mathcal{C}}$ telle que les η_X soient des équivalences faibles cofibrantes. A nouveau, une façon d'obtenir un endofoncteur et une transformation naturelle de celui-ci vers l'identité, est de chercher une paire de foncteurs adjoints L et R puis de considérer la counité de l'adjonction $\epsilon : L \circ R \rightarrow Id$.

C'est ce qu'on s'attachera à faire en détail dans la première partie (puis dans la seconde partie) en construisant une adjonction :

$$Aug - dga - alg \xrightleftharpoons[B]{\Omega} Conil - dga - coalg$$

Table des matières

1	Une adjonction cobar-bar pour les algèbres associatives	6
1.1	Rappels et conventions	6
1.1.1	Algèbres	6
1.1.2	Coalgèbres	8
1.1.3	(Co)Algèbres différentielles graduées	9
1.1.4	Convolution	11
1.2	Adjonction cobar-bar	12
1.2.1	Résolution cobar-bar	15
1.3	Une résolution plus économe	20
1.3.1	Complexe de Koszul associé à une algèbre quadratique	21
1.3.2	Construction de $A^i \hookrightarrow B^0 A \subset BA$	23
2	Cobar-bar pour les opérades algébriques	26
2.1	Introduction aux opérades algébriques	26
2.1.1	\mathbb{S} -modules	26
2.1.2	Opérades	29
2.1.3	Algèbre universelle	32
2.1.4	Définition d'une opérade par compositions partielles	32
2.1.5	Opérade libre	33
2.2	Constructions d'algèbre homologique pour les opérades	33
2.2.1	Linéarisation	33
2.2.2	Extension au cadre différentiel gradué	35

<i>TABLE DES MATIÈRES</i>	5
2.3 Adjonction cobar-bar opéradique	36
2.3.1 Morphismes tordants	36
2.3.2 Constructions cobar et bar	37
2.4 Opérades quadratiques et dualité de Koszul	38
3 Propérades	40
3.1 Introduction	40
3.2 Calculs de duals	44
A Rappels	50

Chapitre 1

Une adjonction cobar-bar pour les algèbres associatives

Ce chapitre est très largement inspiré de [LV12]. On fixe un corps \mathbb{K} de caractéristique nulle. Dans ce qui suit, les espaces vectoriels seront sur \mathbb{K} et les applications entre espaces vectoriels seront supposées linéaires (sauf mention du contraire). Les algèbres et coalgèbres seront toujours supposées (co-)associatives et (co-)augmentées et l'idéal d'augmentation sera surmonté d'une barre, sauf mention du contraire.

1.1 Rappels et conventions

On introduit ici les définitions et propriétés permettant d'atteindre rapidement l'adjonction voulue.

1.1.1 Algèbres

On rappelle la construction de l'algèbre libre (unitaire) sur un espace vectoriel V :

Construction 1.1. Soit $\bar{T}(V) := \bigoplus_{n \in \mathbb{N}^*} V^{\otimes n}$ le \mathbb{K} -module tensoriel et $i_V := i_1 : V \hookrightarrow$

$T(\bar{V})$ l'inclusion canonique. $T(\bar{V})$ muni du produit de concaténation μ :

$$\forall x_1 \otimes \dots \otimes x_n \forall x_{n+1} \otimes \dots \otimes x_k : (x_1 \otimes \dots \otimes x_n) \cdot (x_{n+1} \otimes \dots \otimes x_k) := x_1 \otimes \dots \otimes x_k$$

est une algèbre associative libre sur V , notée $\mathcal{F}_u(V)$, ou juste $\mathcal{F}(V)$ si aucune confusion n'est possible. L'algèbre unitaire augmentée sur un espace vectoriel V doit quant à elle vérifier l'adjonction :

$$\begin{array}{ccc} \text{Hom}_{uAs-alg}(\mathcal{F}(V), A) & \rightarrow & \text{Hom}_{\mathbb{K}}(V, \bar{A}) \\ f & \mapsto & f \circ i_V \end{array}$$

Soit $T(V) := \bigoplus_{n \in \mathbb{N}} V^{\otimes n}$ le module tensoriel unitaire muni du produit de concaténation μ décrit précédemment et de l'inclusion canonique $i_V : V \hookrightarrow T(V)$. Alors ce triplet $\mathcal{F}(V) := (T(V), \mu, i_V)$ est une algèbre unitaire augmentée sur V .

Lemme 1.1. Soit M un $T(V)$ -bimodule et $\text{Der}(T(V), M)$ l'espace des dérivations de $T(V)$ dans M ([LV12] 1.1.6), alors

$$\begin{array}{ccc} \text{Der}(T(V), M) & \rightarrow & \text{Hom}_{\mathbb{K}}(V, M) \\ d & \mapsto & d \circ i_V \end{array}$$

est un isomorphisme. Autrement dit toute application linéaire $f : V \rightarrow M$ s'étend de façon unique en une dérivation $d_f : T(V) \rightarrow M$:

$$\forall v_1 \otimes \dots \otimes v_n : d_f(v_1 \otimes \dots \otimes v_n) = \sum_{1 \leq i \leq n} v_1 \otimes \dots \otimes f(v_i) \otimes \dots \otimes v_n$$

1.1.2 Coalgèbres

Définition 1.1. On rappelle les notations de Sweedler :

$$\Delta(x) = \sum_i x_1^i \otimes x_2^i$$

et la coassociativité :

$$\Delta^2(x) = \sum_{i,k} x_1^i \otimes x_{2,1}^{i,k} \otimes x_{2,2}^{i,k} = \sum_{i,j} x_{1,1}^{i,j} \otimes x_{1,2}^{i,j} \otimes x_2^i, \text{ que l'on écrira : } \sum_i x_1^i \otimes x_2^i \otimes x_3^i.$$

$$\Delta^n(x) = \sum_i x_1^i \otimes \dots \otimes x_{n+1}^i.$$

Définition 1.2. Une coalgèbre coaugmentée (C, Δ) est dite *conilpotente* si

$$\forall x \in \bar{C}, \exists n, \forall m \geq n : \Delta^m(x) = 0$$

Un morphisme de coalgèbres conilpotentes est un morphisme de coalgèbres coaugmentées. La catégorie des coalgèbres conilpotentes est notée *Conil – Coalg*.

Une coalgèbre colibre conilpotente sur un espace vectoriel V dans la catégorie des coalgèbres conilpotentes est la donnée d'une coalgèbre conilpotente $\mathcal{F}^c(V)$ et d'une application linéaire $p_V : \mathcal{F}^c(V) \rightarrow V$ vérifiant la propriété que toute application linéaire du coidéal d'augmentation d'une coalgèbre conilpotente \bar{C} dans V se relève en un morphisme de coalgèbres coaugmentées $C \rightarrow \mathcal{F}^c(V)$. Une construction est donnée par ce qui suit.

Construction 1.2. Soit $T^c(V) := \bigoplus_{n \in \mathbb{N}} V^{\otimes n}$ le module tensoriel, $p_V := \pi_1 : T^c(V) \rightarrow V$ la première projection et le coproduit de déconcaténation¹ :

$$\forall x_1 \otimes \dots \otimes x_n \in T(V) : \Delta(x_1 \otimes \dots \otimes x_n) := \sum_{0 \leq i \leq n} x_1 \otimes \dots \otimes x_i \boxtimes x_{i+1} \otimes \dots \otimes x_n \in T^c(V) \boxtimes T^c(V)$$

Alors $(T^c(V), p_V, \Delta)$ est une coalgèbre conilpotente colibre sur V , notée $\mathcal{F}^c(V)$. Ob-

1. \boxtimes est le produit tensoriel au dessus de \mathbb{K} , noté différemment pour le discerner du produit tensoriel \otimes interne à $T(V)$.

servons que le coproduit réduit est donné par :

$$\bar{\Delta}(x_1 \otimes \dots \otimes x_n) := \sum_{1 \leq i \leq n-1} x_1 \otimes \dots \otimes x_i \boxtimes x_{i+1} \otimes \dots \otimes x_n \in T^c(\bar{V}) \boxtimes T^c(\bar{V})$$

Définition 1.3. Soit (C, Δ) une coalgèbre conilpotente. Une *codérivation* de C est une application linéaire $d : C \rightarrow C$ vérifiant

$$\Delta \circ d = (d \otimes Id) \circ \Delta + (Id \otimes d) \circ \Delta$$

L'espace des dérivations de C est noté $Coder(C)$.

Lemme 1.2. *L'application linéaire suivante*

$$\begin{array}{ccc} Coder(T^c(V)) & \rightarrow & Hom_{\mathbb{K}}(T^c(V), V) \\ d & \mapsto & p_V \circ d \end{array}$$

est un isomorphisme.

Si $f \in Hom_{\mathbb{K}}(T^c(V), V)$, alors la codérivation correspondante est donnée par

$$\forall x : d(x)^{(n)} = \sum_{1 \leq i \leq n} \sum_j x_1^j \otimes \dots \otimes f(x_i^j) \otimes \dots \otimes x_n^j$$

Où $d(x)^n$ est la composante dans $V^{\otimes n}$ de $d(x)$ et $\bar{\Delta}^{n-1}(x) = \sum_j x_1^j \otimes \dots \otimes x_n^j$.

1.1.3 (Co)Algèbres différentielles graduées

Pour une introduction détaillée au cadre différentiel gradué, le chapitre 1 de [LV12] est suffisant.

Si $V = \{V_i\}$ et $W = \{W_j\}$ sont des espaces vectoriels gradués, alors $V \otimes W := \{(V \otimes W)_n\}$ où

$$(V \otimes W)_n := \bigoplus_{i+j=n} V_i \otimes W_j$$

Le cas $W = V$ induit par récurrence une graduation sur $V^{\otimes n}$ et donc sur $T(V)$.

Soit $\mathbb{K}s$ l'espace vectoriel gradué engendré par s , concentré en degré 1. La *suspension* de l'espace vectoriel gradué V est définie par : $sV := \mathbb{K}s \otimes V$, et sa *dé-suspension* $s^{-1}V$ par $\mathbb{K}s^{-1} \otimes V$, où $\mathbb{K}s^{-1}$ est concentré en degré -1.

L'espace des morphismes d'espaces gradués de degré r est noté $Hom(V, W)_r$. Le *bord* $\partial(f)$ d'un morphisme d'espaces gradués $f : V_\bullet \rightarrow W_{\bullet+r}$ de degré r entre complexes de chaînes est donné par :

$$\partial(f) := d_W \circ f - (-1)^r f \circ d_V$$

Ainsi f est un morphisme de complexes de chaînes si, et seulement si $\partial(f) = 0$.

Jusqu'à la fin, on utilisera les définitions de Koszul : pour V et W gradués, l'isomorphisme de symétrie τ est donné par :

$$\begin{aligned} \tau_{V,W} : V \otimes W &\rightarrow W \otimes V \\ v \otimes w &\mapsto (-1)^{|v||w|} w \otimes v \end{aligned}$$

Pour $f : V \rightarrow V'$ et $g : W \rightarrow W'$ des applications linéaires de degré $|f|$ et $|g|$, leur produit tensoriel² est

$$\forall v \otimes w : (f \otimes g)(v \otimes w) := (-1)^{|g||v|} f(v) \otimes g(w)$$

Définition 1.4. Si (V, d_V) et (W, d_W) sont des complexes de chaînes, leur produit tensoriel est le complexe de chaîne $(V \otimes W, d_{V \otimes W})$ où

$$d_{V \otimes W} := d_V \otimes Id + Id \otimes d_W$$

Définition 1.5. Un complexe de chaînes *gradué par le poids* M est un complexe de chaînes M , muni d'une décomposition en somme directe de sous complexes de

2. La tensorisation $- \otimes -$ reste associative.

chaînes $M^{(d)}$:

$$M = \bigoplus_{d \in \mathbb{N}} M^{(d)}$$

Une algèbre différentielle graduée par le poids A , ou *wdg-algèbre*, est la donnée d'une structure d'algèbre sur un complexe de chaînes gradué par le poids A , telle que le produit préserve le poids et le degré. On note $A_n^{(d)}$ la composante de poids d et degré n de A . Elle est de plus dite *connexe* si $A^{(0)} = \mathbb{K}.1_A$

1.1.4 Convolution

Définition 1.6. Soit (C, Δ, ϵ) une dg-coalgèbre et (A, μ, u) une dg-algèbre, on note $Hom(C, A) := \bigoplus_{r \in \mathbb{Z}} Hom(C, A)_r$.

La convolution :

$$Hom(C, A) \otimes Hom(C, A) \rightarrow Hom(C, A)$$

$$f \otimes g \mapsto f * g := \mu \circ f \otimes g \circ \Delta$$

fait de $(Hom(C, A), *, \partial, u \circ \epsilon)$ une dg-algèbre unitaire.

Définition 1.7. Soient A une algèbre munie d'une dérivation $d_A : A \rightarrow A$ et M un A -module à droite. Une *A-dérivation* sur M est une application linéaire $d : M \rightarrow M$ telle que

$$\forall a, \forall m : d(m.a) = d(m).a + m.d_A(a)$$

Proposition 1.1. *Toute application linéaire $\alpha : C \rightarrow A$ d'une coalgèbre graduée vers une algèbre graduée, induit une application linéaire $d_\alpha : C \otimes A \rightarrow C \otimes A$ qui est une A-dérivation et C-codérivation, par :*

$$d_\alpha^r := (Id \otimes \mu) \circ (Id_C \otimes \alpha \otimes Id_A) \circ (\Delta \otimes Id_A)$$

Toute application linéaire $\alpha : C \rightarrow A$ induit une dérivation sur $C \otimes A$ par $d_\alpha := d_{C \otimes A} + d_\alpha^r$. On aura besoin du lemme suivant dans la section suivante.

Lemme 1.3.

$$d_\alpha^2 = d_{\partial(\alpha) + \alpha * \alpha}^r$$

1.2 Adjonction cobar-bar

On rappelle qu'on a les paires d'adjonctions suivantes (où les flèches opposées sont adjointes) :

$$\begin{array}{ccccc}
 & & \mathbb{K} - Mod & & \\
 & \swarrow T & & \nwarrow \bar{U} & \\
 Aug - alg & \xleftarrow{\bar{U}} & & \xrightarrow{T^c} & Conil - coalg \\
 & \xleftarrow{T \circ \bar{U}} & & \xrightarrow{T^c \circ \bar{U}} &
 \end{array}$$

qui fournit une adjonction entre la catégorie des algèbres augmentées $Aug - alg$ et la catégorie des coalgèbres conilpotentes $Conil - coalg$, qui s'étend au cas gradué³. Cette adjonction, se factorisant par $\mathbb{K} - Mod$, oublie les structures d'algèbres et coalgèbres. Dans la section qui suit, on s'attache à construire une adjonction entre les catégories des algèbres différentielles graduées augmentées $dg - Aug - alg$ et des coalgèbres conilpotentes différentielles graduées $dg - conil - coalg$ qui n'oubliera ni les structures de (co-)algèbre ni les différentielles internes.

Soit C une dg -coalgèbre conilpotente. Un point technique intervenant dans la section suivante nous invite à plutôt considérer $\mathcal{F}(s^{-1}\bar{C})$ et non simplement $\mathcal{F}(\bar{C})$. On peut alors munir l'algèbre $\Omega C := \mathcal{F}(s^{-1}\bar{C})$ de deux dérivations d_{int} et d_{ext} :

- $d_{int} := \sum_{n \in \mathbb{N}} d_{(s^{-1}\bar{C})^{\otimes n}}$, explicitement donnée pour tous $s^{-1}x_1 \otimes \dots \otimes s^{-1}x_n \in \Omega C$ par :

$$d_{int}(s^{-1}x_1 \otimes \dots \otimes s^{-1}x_n) = \sum_{1 \leq i \leq n} (-1)^{i+|x_1|+\dots+|x_{i-1}|} s^{-1}x_1 \otimes \dots \otimes s^{-1}d_C(x_i) \otimes \dots \otimes s^{-1}x_n$$

- d_{ext} , obtenue en prolongeant (voir lemme 1.1) la composée $f := (id \otimes \tau \otimes id) \circ$

3. L'isomorphisme naturel en A et C : $\text{Hom}_{g-Aug-alg}(T(s^{-1}\bar{C}), A) \simeq \text{Hom}_{g\mathbb{K}}(\bar{C}, \bar{A})_{-1}$ est noté $\psi_{C,A}$

$\Delta_s \otimes \Delta_{\bar{C}} :$

$$\begin{array}{ccc} \mathbb{K}s^{-1} \otimes \bar{C} & \xrightarrow{\Delta_s \otimes \Delta_{\bar{C}}} & \mathbb{K}s^{-1} \otimes \mathbb{K}s^{-1} \otimes \bar{C} \otimes \bar{C} \\ & \searrow f & \downarrow \simeq id \otimes \tau \otimes id \\ & & (\mathbb{K}s^{-1} \otimes \bar{C})^{\otimes 2} \end{array}$$

en une dérivation sur $T(s^{-1}\bar{C})$.

Et Δ_s est le morphisme de degré -1,

$$\begin{array}{ccc} \mathbb{K}s^{-1} & \rightarrow & \mathbb{K}s^{-1} \otimes \mathbb{K}s^{-1} \\ s^{-1} & \mapsto & s^{-1} \otimes s^{-1} \end{array}$$

Explicitement, f est donnée par la formule :

$$\forall s^{-1}x \in s^{-1}\bar{C} : f(s^{-1}x) = \sum_i (-1)^{|x_1^i|} s^{-1}x_1^i \otimes s^{-1}x_2^i$$

Lemme 1.4. — $d_{ext}^2 = 0$.

$$— d_{ext} \circ d_{int} + d_{int} \circ d_{ext} = 0$$

Esquisse de démonstration 1.1. La coassociativité de Δ_C implique que $d_{ext}^2 = 0$.

Quant à l'égalité $d_{ext} \circ d_{int} + d_{int} \circ d_{ext} = 0$, elle est conséquence du caractère morphisme de complexe de chaînes de Δ_C .

ΩC munit de la différentielle $d_{\Omega C} := d_{int} + d_{ext} :$

$$(\Omega C, d_{\Omega C})$$

est appelée *construction*⁴ *cobar* de C .

Ayant le foncteur d'oubli, fidèle

$$U : dg - Aug - alg \rightarrow g - Aug - alg$$

4. Qui est fonctorielle en C

on va caractériser l'image de la composée :

$$\mathrm{Hom}_{dg-Aug-alg}(\Omega C, A) \xrightarrow{U} \mathrm{Hom}_{g-Aug-alg}(\Omega C, A) \xrightarrow{\psi_{C,A}} \mathrm{Hom}_{g\mathbb{K}}(\bar{C}, \bar{A})_{-1}$$

Si $f : \Omega C \rightarrow A$ est un morphisme d'algèbres graduées augmentées, alors

$$f \in \mathrm{Hom}_{dg-Aug-alg}(\Omega C, A) \Leftrightarrow f \circ d_{\Omega C} = d_A \circ f$$

Or ΩC est quasi-libre et f est un morphisme d'algèbres, donc :

$$f \circ d_{\Omega C} = d_A \circ f \Leftrightarrow f \circ d_{\Omega C|s^{-1}\bar{C}} = d_A \circ f|_{s^{-1}\bar{C}}$$

ce qui peut se réécrire :

$$\begin{aligned} & -f(sd_C(x)) + \sum_i (-1)^{|x_1^i|} f(s^{-1}x_1^i) \cdot f(s^{-1}x_2^i) = d_A(f(s^{-1}x)) \\ & \Leftrightarrow d_A \circ \alpha_f(x) + \alpha_f \circ d_C(x) + \sum_i (-1)^{|x_1^i|} \alpha_f(x_1^i) \cdot \alpha_f(x_2^i) = 0 \\ & \Leftrightarrow \partial(\alpha_f) + \alpha_f * \alpha_f = 0 \end{aligned}$$

Où $\alpha_f := \psi_{C,A}(f)$. L'équation

$$\partial(\alpha_f) + \alpha_f * \alpha_f = 0 \quad (\text{M.C})$$

est appelée *équation de Maurer-Cartan*.

Définition 1.8. une solution de l'équation de Maurer-Cartan $\alpha \in \mathrm{Hom}_{\mathbb{K}}(C, A)$, est appelée *morphisme tordant* si de plus elle vérifie :

- $\alpha(\bar{C}) \subset \bar{A}$
- $\alpha(\mathbb{K}) = 0$

L'espace des morphismes tordants⁵ de C dans A est noté $Tw(C, A)$.

On vient donc de montrer :

5. qui est un sous-foncteur du foncteur $\mathrm{Hom}(-, -)$

Proposition 1.2. *L'isomorphisme naturel $\text{Hom}_{g\text{-Aug-}alg}(T(s^{-1}\bar{C}), A) \simeq \text{Hom}_{g\mathbb{K}}(\bar{C}, \bar{A})_{-1}$ se restreint en un isomorphisme ϕ , donc naturel, faisant commuter le carré :*

$$\begin{array}{ccc} \text{Hom}_{g\text{-Aug-}alg}(T(s^{-1}\bar{C}), A) & \xrightarrow{\psi_{C,A}} & \text{Hom}_{g\mathbb{K}}(\bar{C}, \bar{A})_{-1} \\ \uparrow & & \uparrow \\ \text{Hom}_{dg\text{-Aug-}alg}(\Omega C, A) & \xrightarrow{\phi_{C,A}} & Tw(C, A) \end{array}$$

Si A est une dg -algèbre augmentée, la coalgèbre colibre $T^c(s\bar{A})$, munie de la différentielle $d_{BA} := d_{int} + d_{ext}$, où

- $d_{int} := \sum_{1 \leq i \leq n} Id^{\otimes i-1} \otimes d_{s\bar{A}} \otimes Id^{\otimes n-i}$.
- $d_{ext}(sx_1 \otimes \dots \otimes sx_n) := \sum_{1 \leq i \leq n-1} (-1)^{i-1+|x_1|+\dots+|x_i|} sx_1 \otimes \dots \otimes sx_i \cdot x_{i+1} \otimes \dots \otimes sx_n$.

est appelé *construction bar*⁶ de A :

$$BA := (T^c(s\bar{A}), d_{BA})$$

En faisant le même travail que précédemment, on obtient le théorème suivant :

Théorème 1.5. *Pour toute dg -coalgèbre conilpotente C et toute dg -algèbre augmentée A , on a les bijections naturelles :*

$$\text{Hom}_{dg\text{-Aug-}alg}(\Omega C, A) \simeq Tw(C, A) \simeq \text{Hom}_{dg\text{-conil-}alg}(C, BA)$$

1.2.1 Résolution cobar-bar

Le lemme de Yoneda [Mac] indique que les transformations naturelles d'un foncteur, à valeurs dans la catégorie des ensembles Ens , représentable ($\simeq \text{Hom}(X, -)$) vers un foncteur quelconque F , sont paramétrées par $F(X)$, plus précisément, on

6. fonctorielle en A

a la bijection :

$$\begin{aligned} \text{Nat}(\text{Hom}(X, -), F) &\rightarrow F(X) \\ \eta &\mapsto \eta_X(\text{Id}_X) \end{aligned}$$

La preuve de l'injectivité fournit une formule pour η , qui est la suivante : $\forall Y \in \text{Ob}(C)$,

$$\eta_Y(f) = F(f)(\eta_X(\text{Id}_X))$$

En particulier $\text{Nat}(\text{Hom}(\Omega C, -), \text{Tw}(C, -)) \simeq \text{Tw}(C, \Omega C)$, dont la source est $dg - Aug - alg$. Soit $\Phi_{C,\cdot} \in \text{Nat}(\text{Hom}(\Omega C, -), \text{Tw}(C, -))$ l'isomorphisme naturel vu à la section précédente. Par le lemme de Yoneda, celui-ci est exactement déterminé par $\iota := \Phi_{\Omega C}(\text{Id}_{\Omega C}) \in \text{Tw}(C, \Omega C)$, d'où :

$$\forall f : \Omega C \rightarrow A, \Phi_A(f) = \text{Tw}(f)(\Psi_{\Omega C}(\text{Id}_{\Omega C})) = \text{Tw}(f)(\iota) = f \circ \iota \in \text{Tw}(C, \Omega C).$$

Autrement dit, la bijection $\text{Hom}(\Omega C, A) \xrightarrow{\sim} \text{Tw}(C, A)$ est donné par la précomposition par ι , qui est appelé *morphisme tordant universel*.

On a, de façon analogue, le *morphisme tordant universel* $\pi : BA \rightarrow A$.

Théorème 1.6. *Tout morphisme tordant $\alpha : C \rightarrow A$ se factorise d'une unique façon par π et ι :*

$$\begin{array}{ccc} & \Omega C & \\ \iota \nearrow & & \searrow \exists! g_\alpha \\ C & \xrightarrow{\alpha} & A \\ \searrow \exists! f_\alpha & & \nearrow \pi \\ & BA & \end{array}$$

$$\text{Où } g_\alpha = \Phi_A^{-1}(\alpha) \text{ et } f_\alpha = \Psi_C^{-1}(\alpha) \text{ }^7$$

Théorème 1.7. *Les complexes de chaînes $BA \otimes_\pi A$, $A \otimes_\pi BA$, $C \otimes_\iota \Omega C$ et $\Omega C \otimes_\iota C$ sont acycliques.*

7. L'isomorphisme $\text{Hom}(-, BA) \simeq \text{Tw}(-, A)$ est noté Ψ .

Démonstration 1.1. Réorganisons la différentielle d_π sur $BA \otimes A$ en $d_\pi = d_{int} + d_{ext}$ où :

$$\begin{aligned} - d_{int}([sa_1 \mid \dots \mid sa_n]a_{n+1}) &= \sum_{1 \leq i \leq n} (-1)^{i+|a_1|+\dots+|a_{i-1}|} [sa_1 \mid \dots \mid sd(a_i) \mid \dots \mid \\ &\quad sa_n]a_{n+1} + (-1)^{n+|a_1|+\dots+|a_n|} [sa_1 \mid \dots \mid sa_n]d(a_{n+1}) . \\ - d_{ext}([sa_1 \mid \dots \mid sa_n]a_{n+1}) &= \sum_{1 \leq i \leq n-1} (-1)^{i-1+|a_1|+\dots+|a_i|} [sa_1 \mid \dots \mid sa_i a_{i+1} \mid \dots \mid \\ &\quad sa_n]a_{n+1} + (-1)^{n-1+|a_1|+\dots+|a_{n-1}|} [sa_1 \mid \dots \mid sa_{n-1}]a_n a_{n+1} \end{aligned}$$

Soit $\epsilon \otimes \epsilon : BA \otimes_\pi A \rightarrow \mathbb{K} \otimes \mathbb{K} \simeq \mathbb{K}$ l'augmentation, et K son noyau. On définit l'application $h : K \rightarrow K$ par

$$h([sa_1 \mid \dots \mid sa_n]a_{n+1}) := (-1)^{n+|a_1|+\dots+|a_n|} [sa_1 \mid \dots \mid sa_n \mid s(a_{n+1} - \epsilon(a_{n+1}))]$$

On a alors

$$\begin{aligned} (h \circ d_{int} + d_{int} \circ h)([sa_1 \mid \dots \mid sa_n]a_{n+1}) &= \sum_{1 \leq i \leq n} (-1)^{i+n-1+|a_1|+\dots+|a_n|} [sa_1 \mid \dots \mid sd(a_i) \mid \dots \\ &\quad \mid sa_n \mid s(a_{n+1} - \epsilon(a_{n+1}))] \\ &\quad + [sa_1 \mid \dots \mid sa_n \mid s(d(a_{n+1}) - \epsilon(d(a_{n+1})))] \\ &\quad + \sum_{1 \leq i \leq n+1} (-1)^{i+n+|a_1|+\dots+|a_n|} [sa_1 \mid \dots \\ &\quad \mid sd(a_i) \mid \dots \mid s(a_{n+1} - \epsilon(a_{n+1}))] \\ &= 0 \end{aligned}$$

$$\begin{aligned}
(d_{ext} \circ h + h \circ d_{ext})([sa_1 \mid \dots \mid sa_n]a_{n+1}) &= \sum_{1 \leq i \leq n} (-1)^{i-1+n+|a_{i+1}|+|a_n|} [sa_1 \mid \dots \mid sa_i.a_{i+1} \mid \dots \\
&\quad \mid s(a_{n+1} - \epsilon(a_{n+1}))] \\
&\quad + [sa_1 \mid \dots \mid sa_n](a_{n+1} - \epsilon(a_{n+1})) \\
&\quad + \sum_{1 \leq i \leq n-1} (-1)^{i+n+|a_{i+1}|+|a_n|} [sa_1 \mid \dots \mid sa_i.a_{i+1} \mid \dots \\
&\quad \mid s(a_{n+1} - \epsilon(a_{n+1}))] + [sa_1 \mid \dots \mid s(a_n.a_{n+1} - \epsilon(a_n.a_{n+1}))] \\
&= -[sa_1 \mid \dots \mid sa_n a_{n+1} - \epsilon(a_{n+1})] \\
&\quad + [sa_1 \mid \dots \mid s(a_n a_{n+1} - \epsilon(a_n a_{n+1}))] \\
&\quad + [sa_1 \mid \dots \mid sa_n](a_{n+1} - \epsilon(a_{n+1})) \\
&= [sa_1 \mid \dots \mid sa_n]a_{n+1}
\end{aligned}$$

Ce qui donne $h \circ d_\pi + d_\pi \circ h = Id$ et donc $BA \otimes_\pi A \simeq (Ker(\epsilon \otimes \epsilon), d_\pi) \oplus (\mathbb{K}, 0)$ est acyclique.

Définition 1.9. Un morphisme tordant $\alpha : C \rightarrow A$ est dit de Koszul, noté $\alpha \in Kos(C, A)$, si et seulement si $C \otimes_\alpha A$ est acyclique.

Remarque 1.1. Les morphismes universels π et ι sont donc de Koszul.

Théorème 1.8. Soit A une dg-algèbre connexe et C une dg-coalgèbre connexe. Pour tout morphisme tordant $\alpha : C \rightarrow A$, on a les équivalences :

1. $C \otimes_\alpha A$ est acyclique.
2. $A \otimes_\alpha C$ est acyclique.
3. $f_\alpha : C \rightarrow BA$ est un quasi-isomorphisme.
4. $g_\alpha : \Omega C \rightarrow A$ est un quasi-isomorphisme.

Pour la démonstration, on utilisera la boîte noire suivante :

Théorème 1.9. Soient $g : A \rightarrow A'$ un morphisme de wdg-algèbres connexes ($A^{(0)} \simeq A'^{(0)} \simeq \mathbb{K}$), $f : C \rightarrow C'$ un morphisme de wdg-coalgèbres connexes, $\alpha : C \rightarrow A$ et $\alpha' : C' \rightarrow A'$ des morphismes tordants tel que $\alpha' \circ f = g \circ \alpha$. Alors si deux parmi les

trois morphismes f, g et $f \otimes g : C \otimes_\alpha A \rightarrow C' \otimes_{\alpha'} A'$ sont des quasi-isomorphismes, le troisième l'est aussi.

La preuve fait appel aux suites spectrales.

Prouvons maintenant le théorème.

Démonstration 1.2. *On ne va montrer l'équivalence qu'entre les propositions 1. et 4., les autres étant similaires, et on fait l'hypothèse que A et C sont gradués par le poids et connexes, cependant le théorème est vrai sans ces hypothèses [HMS74].*

La construction bar de A (resp. cobar de C) est une wdg-coalgèbre connexe (resp. dg-algèbre connexe), et le morphisme $g_\alpha : \Omega C \rightarrow A$ est un morphisme de wdg-algèbres. De plus $g_\alpha \circ \iota = \alpha \circ Id_C$, donc par le théorème précédent, comme Id_C est un quasi-isomorphisme, g_α est un quasi-isomorphisme si et seulement si $Id_C \otimes g_\alpha$ est un quasi-isomorphisme. Or $C \otimes_\iota \Omega C$ est acyclique, donc g_α est un quasi-isomorphisme, si, et seulement si $C \otimes_\alpha A$ est acyclique.

Corollaire 1.10. *Soient A une dg-algèbre augmentée et C une dg-coalgèbre coaugmentée. L'unité $\epsilon : \Omega BA \rightarrow A$ et la co-unité $\nu : C \hookrightarrow B\Omega C$ sont des quasi-isomorphismes.*

Démonstration 1.3. *On suppose que A et C sont gradués par le poids et connexes afin de faciliter la démonstration.*

$f_\pi = \epsilon_A : \Omega BA \rightarrow A$, or $BA \otimes_\pi A$ est acyclique par le théorème 1.10, donc par le théorème précédent, ϵ_A est un quasi-isomorphisme. La preuve pour ν est semblable.

Remarque 1.2. *La construction cobar de la construction bar d'une dg-algèbre augmentée A , fournit un quasi-isomorphisme surjectif d'une algèbre quasi-libre ΩBA vers A . Cependant $\Omega BA = T(s^{-1}\overline{T(s\bar{A})})$ contient deux types de tenseurs : les premiers ceux de la construction bar et les seconds, ceux de la construction cobar. Pour certaines algèbres usuelles, dites quadratiques, dont la différentielle interne est nulle : $T(V), S(V), \Lambda(V), \dots$ Il est possible de faire mieux, en remplaçant BA par A^i qui sera (lorsque A sera de Koszul) une sous-coalgèbre quasi-isomorphe à $H^0(B^\bullet A) \subset BA$.*

On aura de plus le diagramme commutatif suivant :

$$\begin{array}{ccc} \Omega A^i & \hookrightarrow & \Omega B A \\ & \searrow & \downarrow \sim \\ & & A \end{array}$$

1.3 Une résolution plus économe

Définition 1.10. Soit V un espace vectoriel gradué et $R \subset V^{\otimes 2}$ un sous-espace vectoriel gradué.

Une algèbre graduée est dite *quadratique* si elle est isomorphe en tant qu'algèbre graduée à

$$A(V, R) := T(V)/(R)$$

Une coalgèbre graduée est dite *quadratique*⁸ si elle isomorphe en tant que coalgèbre graduée à la sous-coalgèbre de $T^c(V)$ donnée par

$$C(V, R) := \mathbb{K} \oplus V \oplus \dots \oplus \left(\bigcap_{i+j+2=n} V^{\otimes i} \otimes R \otimes V^{\otimes j} \right) \oplus \dots \subset T^c(V)$$

A partir de maintenant et jusqu'à la fin de cette section, V sera de dimension finie.

Définition 1.11. Pour $A = A(V, R)$, la *coalgèbre duale de Koszul* de A est définie par

$$A^i := C(sV, s^2 R)$$

On définit aussi *l'algèbre duale de Koszul*

$$A^! := A(V^*, R^\perp)$$

où $R^\perp \subset V^* \otimes V^* \simeq (V \otimes V)^*$ est l'orthogonal de R .

8. Une *donnée quadratique* est un couple $(V, R \subset V^{\otimes 2})$ et un morphisme de données quadratiques $(V, R) \rightarrow (V', R')$ est une application linéaire $f : V \rightarrow V'$ telle que $T(f)(R) \subset R'$. La collection des données quadratiques est ainsi organisée en une catégorie et $A(-, -)$, $C(-, -)$ sont des foncteurs.

Pour $C = C(V, R)$, l'algèbre duale de Koszul est donnée par

$$C^i := A(s^{-1}V, s^{-2}R)$$

Lemme 1.11. $(A^i)^i = A$, $(C^i)^i = C$ et $(A^!)^! = A$

Exemple 1.1. — $T(V)$ est quadratique et $T(V)^! = \mathbb{K} \otimes V^*$, muni de la multiplication nulle (si $V \simeq \mathbb{K}$, $T(V) \simeq \mathbb{K}[X]/(X^2)$).

— $S(V) := T(V)/(R := \langle x \otimes y - y \otimes x \mid x \otimes y \in V^{\otimes 2} \rangle)$ est quadratique. V est de dimension finie, donc le morphisme naturel $V^* \otimes V^* \rightarrow (V \otimes V)^*$ est un isomorphisme, et

$$\begin{aligned} f \otimes g &\in R^\perp \\ \Leftrightarrow \forall x \otimes y, f(y)g(x) &= f(x)g(y) \\ \Leftrightarrow 2.f \otimes g &= f \otimes g + g \otimes f \in \langle l \otimes l, l \in V^* \rangle \subset V^{*\otimes 2} \end{aligned}$$

(par polarisation). i.e. $R^\perp = \langle l \otimes l, l \in V^* \rangle$. Ainsi $S(V)^! = \Lambda(V^*)$.

— $S(V)^i = \Lambda^c(sV)$, où

$$\Lambda^c(sV) = \mathbb{K} \oplus sV \oplus \dots \oplus \left\langle \sum_{\sigma \in \mathfrak{S}_n} \epsilon(\sigma) s x_{\sigma(1)} \otimes \dots \otimes s x_{\sigma(n)} \mid x_1, \dots, x_n \in V \right\rangle \oplus \dots \subset T^c(sV)$$

1.3.1 Complexe de Koszul associé à une algèbre quadratique

La machinerie des morphismes tordants et le théorème fondamental (théorème 1.11) sur les morphismes tordants va permettre de fabriquer un morphisme $\Omega A^i \twoheadrightarrow A$. Soit $A = A(V, R)$ une algèbre quadratique, munie de la différentielle nulle. On définit le morphisme de complexes de chaînes⁹ de degré -1 et poids 0 comme la composée :

$$\kappa : A^i \rightarrow A := A^i \twoheadrightarrow sV \simeq V \hookrightarrow A$$

Lemme 1.12. $\kappa * \kappa = 0$, et donc $\kappa \in Tw(A^i, A)$.

9. κ est en fait naturel en la donnée quadratique (V, R)

Définition 1.12. Le complexe de chaînes $(A^i \otimes_{\kappa} A, d_{\kappa})$ est appelé le *complexe de Koszul* de l'algèbre quadratique A .

Lemme 1.13. Le complexe $(A^i \otimes_{\kappa} A, d_{\kappa})$ se scinde en

$$A^i \otimes_{\kappa} A = \bigoplus_n ((A^i \otimes_{\kappa} A)^{(n)}, d_{\kappa|})$$

Où $((A^i \otimes_{\kappa} A)^{(n)}, d_{\kappa|})$ est le complexe :

$$0 \rightarrow A^{i(n)} \rightarrow A^{i(n-1)} \otimes A^{(1)} \rightarrow \dots \rightarrow A^{i(1)} \otimes A^{(n-1)} \rightarrow A^{(n)} \rightarrow 0$$

Et $(A^i \otimes_{\kappa} A)^{(n)} = \bigoplus_{i+j=n} A^{i(j)} \otimes A^{(i)}$, et $A^{i(d)}$ est le sous espace des éléments de poids d , induit par la graduation de V .

Démonstration 1.4. d_A et d_{A^i} sont nulles donc $d_{\kappa} = d_{\kappa}^*$.

De plus $d_{\kappa}(sx_1 \otimes \dots \otimes sx_p \otimes x_{p+1}) = (-1)^{p-1+|x_1|+\dots+|x_{p-1}|} sx_1 \otimes \dots \otimes sx_{p-1} \otimes [x_p \otimes x_{p+1}]$ donc $d_{\kappa}(A^{i(p)} \otimes A^{(q)}) \subset A^{i(p-1)} \otimes A^{(q+1)}$ et ainsi $d_{\kappa}((A^i \otimes A)^{(n)}) \subset (A^i \otimes A)^{(n)}$.

Théorème 1.14 (critère de Koszul). Soit A une algèbre quadratique. Les propositions suivantes sont équivalentes :

1. Le complexe de Koszul $A^i \otimes_{\kappa} A$ est acyclique.
2. Le complexe de Koszul $A \otimes_{\kappa} A^i$ est acyclique.
3. $f_{\kappa} : A^i \hookrightarrow BA$ est un quasi-isomorphisme.
4. $g_{\kappa} : \Omega A^i \twoheadrightarrow A$ est un quasi-isomorphisme.

Définition 1.13. Si l'une de ces conditions est vérifiée, alors A est dite de *Koszul*

Proposition 1.3. Une algèbre quadratique A est de Koszul si, et seulement si $A^!$ est de Koszul.

Esquisse de démonstration 1.2. Le dual linéaire de $(A^! \otimes_{\kappa_{A^!}} (A^!)^i)$ est $A^i \otimes_{\kappa} A$ à une suspension près.

Exemple 1.2. $T(V)$ et $S(V)$ sont de Koszul [LV12].

1.3.2 Construction de $A^i \hookrightarrow B^0 A \subset BA$

Si A est quadratique, alors

$$\begin{aligned} BA &= \mathbb{K}1 \oplus s\bar{A} \oplus (s\bar{A})^{\otimes 2} \oplus (s\bar{A})^{\otimes 3} \oplus \dots \\ &= \mathbb{K}1 \oplus sV \oplus (sV^{\otimes 2}/R \oplus sV^{\otimes 2}) \\ &\quad \oplus (sV^{\otimes 3}/(VR + RV) \oplus sV \otimes sV^{\otimes 2}/R \oplus sV^{\otimes 2}/R \otimes sV \oplus sV^{\otimes 3}) \oplus \dots \end{aligned}$$

La graduation par le poids sur BA est donnée par $\omega(sa_1 \otimes \dots \otimes sa_n) := \omega(a_1) + \dots + \omega(a_n)$, et le degré de syzygie est défini par

$$\tilde{\omega}(sa_1 \otimes \dots \otimes sa_n) := \omega(sa_1 \otimes \dots \otimes sa_n) - n$$

La composante de syzygie de degré d est notée $B^d A$ et BA_n est la composante de poids n . A a une différentielle interne nulle, donc $d_{BA} = d_{ext}$ et :

- $\omega(d_{BA}(sa_1 \otimes \dots \otimes sa_n)) = \omega(sa_1 \otimes \dots \otimes sa_n)$
- $\tilde{\omega}(d_{BA}(sa_1 \otimes \dots \otimes sa_n)) = \tilde{\omega}(sa_1 \otimes \dots \otimes sa_n) - 1$

Ainsi BA se scinde en $(BA, d) = \bigoplus_n (BA_n, d|_n)$ et $BA_n = \bigoplus_d B^d A_n$. La table qui suit (on ôte les "s" des notations pour plus de lisibilité) décrit BA :

$$\dots \qquad \dots \qquad \dots \qquad \dots \qquad (4)$$

$$0 \longleftarrow V^{\otimes 3}/(VR + RV) \xleftarrow{d} (V^{\otimes 2}/R \otimes V) \oplus (V \otimes V^{\otimes 2}/R) \xleftarrow{d} V^{\otimes 3} \qquad (3)$$

$$0 \xleftarrow{d} V^{\otimes 2}/R \xleftarrow{d} V^{\otimes 2} \qquad (2)$$

$$0 \xleftarrow{d} V \qquad (1)$$

$$\mathbb{K} \qquad (0)$$

$$\begin{array}{cccc} 3 & 2 & 1 & 0 \end{array}$$

Où sont indiqués en ordonnée le poids et en abscisse le degré de syzygie. BA est la somme directe des cases du tableau et donc il y'a l'inclusion de coalgèbres :

$$A^i = C(sV, s^2R) \xhookrightarrow{i} B^0 A \subset BA$$

Lemme 1.15. $\text{Ker}(d|_{B^0 A}) = A^i$, donc i induit un isomorphisme de coalgèbres graduées :

$$i : A^i \xrightarrow{\simeq} H^0(B^\bullet A)$$

Le même travail avec une coalgèbre quadratique C , fournit un morphisme d'algèbres $p : \Omega C \rightarrow C^i$ induisant un isomorphisme

$$p : H_0(\Omega_\bullet C) \rightarrow C^i$$

L'inclusion $i : A^i \hookrightarrow BA$ et la projection $p : \Omega A^i \rightarrow A$ sont respectivement les appli-

cations f_κ et g_κ données au théorème 1.6, on en déduit le diagramme commutatif :

$$\begin{array}{ccc} \Omega A^i & \xrightarrow{\Omega i} & \Omega B A \\ & \searrow p & \downarrow \sim \epsilon_A \\ & & A \end{array}$$

Lorsque A est de Koszul, le fait que $i : A^i \hookrightarrow B A$ soit un quasi-isomorphisme nous renseigne sur l'endroit où il y'a des groupes d'homologie non nulle dans $B^\bullet A : B^0 A$. A posteriori, si on veut un objet $X \xrightarrow{\sim} A$ avec X quasi-libre, il n'est donc pas nécessaire de regarder $\Omega B A$ en entier, mais seulement $\Omega B^0 A$, voir mieux¹⁰ :

$$\Omega H^0(B^\bullet A)$$

Intuitivement, ΩA^i est comparable en taille à $T(T(V))$ alors que $\Omega B A$ est comparable à $T(T(T(V)))$.

10. Plus généralement, une wdga-algèbre connexe A est dite de Koszul si l'homologie de sa construction bar $H(B^\bullet A)$ est concentrée en degré 0.

Chapitre 2

Cobar-bar pour les opérades algébriques

La référence principale pour ce chapitre est [LV12], complétée par [Mil12].

2.1 Introduction aux opérades algébriques

2.1.1 \mathbb{S} -modules

Définition 2.1. Un \mathbb{S} -module est la donnée d'une collection d'espaces vectoriels $\mathcal{M} = \{M(n), n \in \mathbb{N}\}$ telle que pour tout n , $M(n)$ est muni d'une action à droite de \mathbb{S}_n et ses éléments sont dit *d'arité n* . Un \mathbb{S} -module \mathcal{M} est de plus dit réduit si $M(0) = 0$. Un morphisme de \mathbb{S} -modules $f : \mathcal{M} \rightarrow \mathcal{N}$ est une collection de morphismes de $\mathbb{K}[\mathbb{S}_n]$ -modules $f_n : M(n) \rightarrow N(n)$ pour tout n . La catégorie obtenue est notée $\mathbb{S}\text{-mod}$.

À tout \mathbb{S} -module \mathcal{M} , est associé un foncteur $\tilde{\mathcal{M}} : \text{Vect} \rightarrow \text{Vect}$, appelé *foncteur de Schur* associé à \mathcal{M} , défini sur les objets par :

$$\tilde{\mathcal{M}}(V) := \bigoplus_{n \in \mathbb{N}} M(n) \otimes_{\mathbb{K}[\mathbb{S}_n]} V^{\otimes n}$$

où l'action à gauche de \mathbb{S}_n sur $V^{\otimes n}$ est donnée sur les $v_1 \otimes \dots \otimes v_n$ par la permutation

des indices. On obtient ainsi un foncteur de schurisation $(\tilde{\cdot}) : \mathbb{S}\text{-mod} \rightarrow \text{Vect}^{\text{Vect}}$. Le lemme suivant permet d'induire des constructions sur la catégorie but $\text{Vect}^{\text{Vect}}$, par exemple les opérations \oplus, \otimes, \circ , en des constructions sur la catégorie $\mathbb{S}\text{-mod}$.

Proposition 2.1. *Le foncteur de schurisation $\mathcal{M} \mapsto \tilde{\mathcal{M}}$ est pleinement fidèle.*

Démonstration 2.1. *On procède en deux temps en montrant d'abord qu'une transformation naturelle $\eta_{\bullet} : \tilde{\mathcal{M}} := \bigoplus_n M(n) \otimes_{\mathbb{S}_n} (-)^{\otimes n} \rightarrow \tilde{\mathcal{N}} := \bigoplus_n N(n) \otimes_{\mathbb{S}_n} (-)^{\otimes n}$ est, avec quelques hypothèses sur \mathbb{K} , nécessairement de la forme*

$$\eta_{\bullet} = \bigoplus_n (\eta_{\bullet}^n : M(n) \otimes_{\mathbb{S}_n} (-)^{\otimes n} \rightarrow N(n) \otimes_{\mathbb{S}_n} (-)^{\otimes n})$$

Puis on montrera que la composition $M(n) \hookrightarrow \mathcal{M} \rightarrow \tilde{\mathcal{M}}$ est pleinement fidèle. Commençons par montrer le premier point : pour $(k, n) \in \mathbb{N}^2$, on considère la k -ième inclusion $i_{\bullet}^k : M(k) \otimes_{\mathbb{S}_k} (-)^{\otimes k} \hookrightarrow \tilde{\mathcal{M}}$, la n -ième projection $p_{\bullet}^n : \tilde{\mathcal{N}} \rightarrow N(n) \otimes_{\mathbb{S}_n} (-)^{\otimes n}$ et la composée $\eta_{\bullet}^{k,n} := p_{\bullet}^n \circ \eta_{\bullet} \circ i_{\bullet}^k : M(k) \otimes_{\mathbb{S}_k} (-)^{\otimes k} \rightarrow N(n) \otimes_{\mathbb{S}_n} (-)^{\otimes n}$. Pour tout $\lambda \in \mathbb{K}$, en considérant λId_V , par naturalité, on obtient $\lambda^n \eta_V^{k,n} = \lambda^k \eta_V^{k,n}$. En particulier, si \mathbb{K}^ n'est pas recouvert par les racines de l'unité et que $k \neq n$, nécessairement $\eta_{\bullet}^{k,n} = 0_{\bullet}$.*

Pour montrer la fidélité, la structure additive de $\text{Vect}^{\text{Vect}}$ induite par celle de Vect (celui du but), et l'additivité du foncteur de schurisation impliquent qu'il suffit de montrer que si $f : A \rightarrow B$ est un morphisme de \mathbb{S}_n -modules non nul alors $f_{\bullet} \neq 0_{\bullet}$. La suite exacte courte

$$0 \rightarrow \ker(f) \hookrightarrow A \twoheadrightarrow \text{Im}(f) \rightarrow 0$$

induit la suite exacte courte :

$$0 \rightarrow \ker(f) \otimes (\mathbb{K}^n)^{\otimes n} \hookrightarrow A \otimes (\mathbb{K}^n)^{\otimes n} \twoheadrightarrow \text{Im}(f) \otimes (\mathbb{K}^n)^{\otimes n} \rightarrow 0$$

Ainsi $\text{Im}(f \otimes \text{Id}) = \text{Im}(f) \otimes_{\mathbb{S}_n} (\mathbb{K}^n)^{\otimes n}$ et par le résultat rappelé en annexe, comme $\text{Im}(f) \neq 0$, on a bien $\text{Im}(f \otimes \text{Id}) = \text{Im}(f_{\mathbb{K}^{\times}}) \neq 0$ et donc $f_{\bullet} \neq 0_{\bullet}$.

La surjectivité de $\text{Hom}_{\mathbb{S}_n}(A, B) \rightarrow \text{Hom}_{\text{Vect}^{\text{Vect}}}(\tilde{A}, \tilde{B})$ est d'abord montrée dans le cas

particulier où $B = \mathbb{K}[\mathbb{S}_n]$ est la représentation régulière. En considérant l'application linéaire $\mathbb{K}^n \rightarrow V$ qui envoie e_i sur x_i , et en utilisant la naturalité de $\eta_\bullet : \tilde{A} \rightarrow \tilde{B}$, on remarque que η_\bullet est déterminée par les $\eta_{\mathbb{K}^n}(a \otimes e_1 \otimes \dots \otimes e_n)$, où a parcourt A . De plus $\eta(a \otimes e_1 \otimes \dots \otimes e_n) = \sum_{(i_1, \dots, i_n) \in \llbracket 1, n \rrbracket^n} \lambda_{i_1, \dots, i_n} e_{i_1} \otimes \dots \otimes e_{i_n}$ avec $\lambda_{i_1, \dots, i_n} \in \mathbb{K}$ et les projections de \mathbb{K}^n qui envoient e_i sur $(1 - \delta_{i,k})e_i$, ainsi que la naturalité et la \mathbb{K} -indépendance des $e_{i_1} \otimes \dots \otimes e_{i_n}$, nous apprennent que $\eta(a \otimes e_1 \otimes \dots \otimes e_n)$ s'écrit sous la forme

$$\eta(a \otimes e_1 \otimes \dots \otimes e_n) = \left(\sum_{\sigma \in \mathbb{S}_n} \lambda_\sigma \cdot \sigma \right) \cdot e_1 \otimes \dots \otimes e_n$$

L'application définie par $f(a) := \sum_{\sigma \in \mathbb{S}_n} \lambda_\sigma \cdot \sigma$, vérifie $\tilde{f} = \eta_\bullet$.

Soit B quelconque : d'après les théorèmes rappelés en annexe, on peut écrire

$$B = \bigoplus_{\lambda \in \Lambda} B_\lambda$$

$B_\lambda \in \mathbb{K}[\mathbb{S}_n]$ et on a des morphismes $i_\lambda : B_\lambda \hookrightarrow \mathbb{K}[\mathbb{S}_n]$ et $p_\lambda : \mathbb{K}[\mathbb{S}_n] \twoheadrightarrow B_\lambda$ tels que $p_\lambda \circ i_\lambda = \text{Id}_{B_\lambda}$. D'après le cas particulier, pour une transformation naturelle $\eta_\bullet^\lambda : \tilde{A} \rightarrow \tilde{B}_\lambda$, il existe $f_\lambda : A \rightarrow \mathbb{K}[\mathbb{S}_n]$ telle que $\tilde{f}_\lambda = \tilde{i}_\lambda \circ \eta_\bullet^\lambda$, et donc $\eta_\bullet^\lambda = \tilde{p}_\lambda \circ \tilde{f}_\lambda$. f définie par

$$f := \sum_{\lambda \in \Lambda} f_\lambda$$

vérifie bien $\eta_\bullet = \tilde{f}$. En effet, $\eta_\bullet = \sum_\lambda \eta_\bullet^\lambda$, ainsi $\eta_{\mathbb{K}^n}(a \otimes e_1 \otimes \dots \otimes e_n) = (\sum_\lambda f_\lambda(a)) \cdot e_1 \otimes \dots \otimes e_n \in B \otimes_{\mathbb{S}_n} \langle e_1 \otimes \dots \otimes e_n \rangle \simeq B$, donc pour tout a , $\sum_\lambda f_\lambda(a)$ est finie. La schurisation commutant¹ avec la somme directe, on peut alors conclure.

Corollaire 2.1. *Les coefficients dans le développement en série formelle d'un foncteur de Schur sont donc unique à isomorphisme près.*

La catégorie des \mathbb{S} -modules est abélienne et toute suite exacte courte s'y scinde, le foncteur de schurisation est donc exact, et l'adjonction décrite dans l'exercice 5.11.19 [LV12] implique qu'il préserve les petites colimites.

1. La schurisation admet un adjoint à droite d'après l'exercice 5.11.19 [LV12]

Lemme 2.2. *L'image essentielle de $\mathcal{M} \mapsto \tilde{\mathcal{M}}$ est fermée sous \oplus , \otimes et \circ . De plus, on a les formules suivantes :*

$$\begin{aligned} &— (\tilde{\mathcal{M}} \oplus \tilde{\mathcal{N}})(V) \simeq \bigoplus_{n \in \mathbb{N}} (M(n) \oplus N(n)) \otimes_{\mathbb{K}[\mathbb{S}_n]} V^{\otimes n} \\ &— (\tilde{\mathcal{M}} \otimes \tilde{\mathcal{N}})(V) \simeq \bigoplus_{n \in \mathbb{N}} \left(\bigoplus_{i+j=n} \text{Ind}_{\mathbb{S}_i \times \mathbb{S}_j}^{\mathbb{S}_n} M(i) \otimes N(j) \right) \otimes_{\mathbb{K}[\mathbb{S}_n]} V^{\otimes n} \\ &— (\tilde{\mathcal{M}} \circ \tilde{\mathcal{N}})(V) \simeq \bigoplus_{n \in \mathbb{N}} \left(\bigoplus_{k \in \mathbb{N}} M(k) \otimes_{\mathbb{S}_k} \left(\bigoplus_{i_1 + \dots + i_k = n} \text{Ind}_{\mathbb{S}_{i_1} \times \dots \times \mathbb{S}_{i_k}}^{\mathbb{S}_n} N(i_1) \otimes \dots \otimes N(i_k) \right) \right) \otimes_{\mathbb{S}_n} V^{\otimes n} \end{aligned}$$

Où l'action de \mathbb{S}_k sur $\bigoplus_{i_1 + \dots + i_k = n} \text{Ind}_{\mathbb{S}_{i_1} \times \dots \times \mathbb{S}_{i_k}}^{\mathbb{S}_n} N(i_1) \otimes \dots \otimes N(i_k)$ est induite² de l'action de \mathbb{S}_k sur $\llbracket i_1, \dots, i_k \rrbracket$.

Les définitions de \oplus , \otimes et \circ sur $\mathbb{S}\text{-mod}$ sont alors immédiates dès lors qu'on demande que :

$$(\tilde{\cdot}) : (\mathbb{S}\text{-mod}, \oplus, \otimes, \circ) \rightarrow (\text{Vect}^{\text{Vect}}, \oplus, \otimes, \circ)$$

commute (aux isomorphismes naturels près) canoniquement avec ces bifoncteurs. La composition de foncteurs étant associative, on en déduit que $(\mathbb{S}\text{-mod}, \circ, I)$, où $I := (0, \mathbb{K}, 0, \dots) \simeq (\tilde{\cdot})^{-1}(\mathbb{K})$ est monoïdale (à priori non strictement).

2.1.2 Opérades

Une *opérade symétrique*, dans la catégorie Vect , $\mathcal{P} = (\mathcal{P}, \gamma, \eta)$ est un monoïde associatif unitaire dans la catégorie monoïdale $(\mathbb{S}\text{-mod}, \circ, I)$: pour tout n , des applications linéaires

$$\gamma_n : \bigoplus_k P(k) \otimes_{\mathbb{S}_k} \left(\bigoplus_{i_1 + \dots + i_k = n} \text{Ind}_{\mathbb{S}_{i_1} \times \dots \times \mathbb{S}_{i_k}}^{\mathbb{S}_n} P(i_1) \otimes \dots \otimes P(i_k) \right) \rightarrow P(n)$$

et une application linéaire

$$\eta : I \rightarrow \mathcal{P}$$

où $Id := \eta(1_{\mathbb{K}}) \in P(1)$, qui satisfont les relations d'unité et associativité. De façon équivalente, c'est un foncteur de Schur muni d'une structure de monade :

$$\gamma_{\bullet} : \tilde{\mathcal{P}} \circ \tilde{\mathcal{P}} \rightarrow \tilde{\mathcal{P}}$$

2. LA notation Ind_H^G est rappelée en annexe

$$\eta_\bullet : Id \rightarrow \tilde{\mathcal{P}}$$

Un morphisme d'opéades $f : \mathcal{P} \rightarrow \mathcal{L}$ est, par définition, un morphisme de monoïdes unitaires : la donnée d'un morphisme de \mathbb{S} -modules $f : \mathcal{P} \rightarrow \mathcal{L}$ tel que les diagrammes :

$$\begin{array}{ccc} \mathcal{P} \circ \mathcal{P} & \xrightarrow{f \circ f} & \mathcal{L} \circ \mathcal{L} \\ \gamma \downarrow & & \downarrow \gamma \\ \mathcal{P} & \xrightarrow{f} & \mathcal{L} \end{array} \quad \begin{array}{ccc} & I & \\ \eta \swarrow & & \searrow \eta \\ \mathcal{P} & \xrightarrow{f} & \mathcal{L} \end{array}$$

commutent. La catégorie des opéades symétriques dans $Vect$, appelées opéades algébriques, ou juste opéades, est notée Op .

Afin de rendre plus compréhensible la notion de \mathcal{P} -algèbre, qui en sera un cas particulier, on introduit d'abord la notion de module (à gauche) sur un monoïde \mathcal{P} qui n'est autre qu'une action de \mathcal{P} sur un \mathbb{S} -module \mathcal{M} , plus formellement :

Définition 2.2. Un *module à gauche* sur un monoïde \mathcal{P} est un \mathbb{S} -module \mathcal{M} munit d'un morphisme de \mathbb{S} -modules $\mu : \mathcal{P} \circ \mathcal{M} \rightarrow \mathcal{M}$ associatif et unitaire.

Observons le cas particulier où $\mathcal{M} = (A, 0, 0, \dots)$ est concentré en arité 0 : avec la vision monade, $\tilde{\mathcal{P}} \circ \tilde{\mathcal{M}}(V) = \tilde{\mathcal{P}}(A)$ est le foncteur constant, et les diagrammes qui suivent sont commutatifs :

$$\begin{array}{ccccc} \tilde{\mathcal{P}} \circ \tilde{\mathcal{P}}(A) & \xrightarrow{=} & \tilde{\mathcal{P}}(\tilde{\mathcal{P}}(A)) & & \\ \gamma(A) \downarrow & & \downarrow \tilde{\mathcal{P}}(\mu) & A \xrightarrow{\eta(A)} \tilde{\mathcal{P}}(A) & \\ \tilde{\mathcal{P}}(A) & & \tilde{\mathcal{P}}(A) & \searrow Id & \downarrow \gamma(A) \\ & \mu \swarrow & \swarrow \mu & & A \\ & & A & & \end{array}$$

Réciproquement, la donnée d'un espace vectoriel A et d'une application linéaire $\mu = \gamma_A : \tilde{\mathcal{P}}(A) \rightarrow A$ faisant commuter ces diagrammes définissent un \mathcal{P} -module, concentré en arité 0, appelé \mathcal{P} -algèbre.

Définition 2.3. Une structure de \mathcal{P} -algèbre sur un espace vectoriel A est une application linéaire $\gamma_A : \tilde{\mathcal{P}}(A) \rightarrow A$ telle que les deux diagrammes précédent commutent.

Un morphisme entre \mathcal{P} -algèbres A et A' est une application linéaire $f : A \rightarrow A'$ telle que $f \circ \gamma_A = \gamma_{A'} \circ \mathcal{P}(f)$, et la catégorie des \mathcal{P} -algèbres est notée $\mathcal{P}\text{-alg}$.

Pour V un espace vectoriel, $\tilde{\mathcal{P}}(V)$ est canoniquement muni d'une structure de \mathcal{P} -algèbre, par $\gamma_{\tilde{\mathcal{P}}(V)} := \gamma(\tilde{\mathcal{P}}(V)) : \tilde{\mathcal{P}} \circ \tilde{\mathcal{P}}(V) \rightarrow \tilde{\mathcal{P}}(V)$, en effet, l'associativité de $\gamma : \mathcal{P} \circ \mathcal{P} \rightarrow \mathcal{P}$ implique celle de l'action $\gamma_{\tilde{\mathcal{P}}(V)}$.

Exemple 2.1. Les \mathbb{S} -modules $Ass := (0, \mathbb{K}, \mathbb{K}[\mathbb{S}_2], \dots, \mathbb{K}[\mathbb{S}_n], \dots)$ et $As := (\mathbb{K}, \mathbb{K}, \dots, \mathbb{K}[\mathbb{S}_n], \dots)$ sont munis d'une structure d'opérade induite par les inclusions $\mathbb{S}_{i_1} \times \dots \times \mathbb{S}_{i_k} \hookrightarrow \mathbb{S}_n$, $i_1 + \dots + i_k = n$. Une Ass -algèbre (resp. As -algèbre) est alors une algèbre associative (resp. unitaire) au sens classique.

Pour toute opérade \mathcal{P} il y a un foncteur d'oubli $U : \mathcal{P}\text{-alg} \rightarrow \text{Vect}$, qui à une \mathcal{P} -algèbre lui associe l'espace vectoriel sous-jacent. Il admet un adjoint à gauche, noté \mathcal{F} , qualifié de libre et dont on explicite la construction :

Construction 2.1. Pour \mathcal{F} , prenons $\tilde{\mathcal{P}}$, et pour unité de l'adjonction ν , prenons $\eta : Id \rightarrow \tilde{\mathcal{P}}$. On vérifie alors que le diagramme suivant commute, ce qui donne l'existence et l'unicité de \tilde{f} (nécessairement, $\tilde{f} = \gamma_A \circ \tilde{\mathcal{P}}(f)$).

$$\begin{array}{ccc}
 V & \xrightarrow{f} & A \xleftarrow{Id} \\
 \eta_V \downarrow & \nearrow \tilde{f} & \downarrow \eta_A \nearrow \gamma_A \\
 \tilde{\mathcal{P}}(V) & \xrightarrow{\tilde{\mathcal{P}}(f)} & \tilde{\mathcal{P}}(A) \xleftarrow{Id} \\
 \eta_{\tilde{\mathcal{P}}(V)} \downarrow & \nearrow \tilde{\mathcal{P}}(\tilde{f}) & \downarrow \eta_{\tilde{\mathcal{P}}(A)} \nearrow \gamma_{\tilde{\mathcal{P}}(A)} \\
 \tilde{\mathcal{P}}^2(V) & \xrightarrow{\tilde{\mathcal{P}}^2(f)} & \tilde{\mathcal{P}}^2(A)
 \end{array}$$

Où $\tilde{\mathcal{P}}(A)$ muni de $\gamma(\tilde{\mathcal{P}}(A))$ est vue comme une \mathcal{P} -algèbre³

Théorème 2.3. Cette construction définit une adjonction $\mathcal{P}\text{-alg} \xleftarrow[\mathcal{F}]{U} \text{Vect}$.

Définition 2.4. Soit A un espace vectoriel, on définit l'opérade End_A comme étant le \mathbb{S} -module $End_A(n) := \text{Hom}_{\mathbb{K}}(A^{\otimes n}, A)$ muni de la structure d'opérade induite par la composition des applications linéaires.

3. Par définition, $\gamma_A : \tilde{\mathcal{P}}(A) \rightarrow A$ est un morphisme de \mathcal{P} -algèbres

Proposition 2.2. *Se donner une structure de \mathcal{P} -algèbre sur l'espace vectoriel A est équivalent à se donner un morphisme d'opérade $\mathcal{P} \rightarrow \text{End}_A$.*

Esquisse de démonstration 2.1. *C'est une conséquence de l'isomorphisme naturel*

$$\text{Hom}_{\mathbb{S}_n}(P(n), \text{Hom}(A^{\otimes n}, A)) \simeq \text{Hom}(P(n) \otimes_{\mathbb{S}_n} A^{\otimes n}, A)$$

2.1.3 Algèbre universelle

Pour une opérade $\mathcal{P} = \{P(n), n \in \mathbb{N}\}$ quelconque, les éléments de $P(n)$ sont abstraits mais il est possible de les comprendre comme des opérations n -aires : si A est une \mathcal{P} -algèbre, on considère la structure canonique de End_A -algèbre sur A , dont le morphisme structurel est noté $\pi_A : \text{End}_A(A) \rightarrow A$. Se donner une structure de \mathcal{P} -algèbre sur A est équivalent à se donner un morphisme d'opérades $\mathcal{P} \rightarrow \text{End}_A$ par la proposition 2.2. Ce qui peut se reformuler de la façon suivante. π_A est universelle parmi les structures de \mathcal{P} -algèbre γ_A sur A : il existe un unique morphisme d'opérades $\psi : \mathcal{P} \rightarrow \text{End}_A$ tel que le diagramme suivant commute :

$$\begin{array}{ccc} & & \text{End}_A(A) \\ & \nearrow \psi(A) & \downarrow \pi_A \\ \mathcal{P}(A) & \xrightarrow{\gamma_A} & A \end{array}$$

En ce sens, les éléments des $P(n)$, via $\psi(A)$, sont bien des opérations n -aires sur A .

2.1.4 Définition d'une opérade par compositions partielles

Pour (\mathcal{P}, γ) une opérade, les compositions partielles \circ_i sont définies par :

$$\begin{array}{ccc} - \circ_i - : \mathcal{P}(m) \otimes \mathcal{P}(n) & \rightarrow & \mathcal{P}(m+n-1) \\ \mu \otimes \nu & \mapsto & \mu \circ_i \nu := \gamma(\mu \otimes id \otimes \dots \otimes id \otimes \nu \otimes id \dots \otimes id) \end{array}$$

Où l'identité apparaît i fois avant ν . On a alors les relations suivantes :

$$— (\lambda \circ_i \mu) \circ_{i-1+j} \nu = \lambda \circ_i (\mu \circ_j \nu), \text{ pour } 1 \leq i \leq l, 1 \leq j \leq m$$

$$— (\lambda \circ_i \mu) \circ_{k-1+m} \nu = (\lambda \circ_k \nu) \circ_i \mu, \text{ pour } 1 \leq i \leq k \leq l$$

Où $\lambda \in \mathcal{P}(l), \mu \in \mathcal{P}(m), \nu \in \mathcal{P}(n)$. Réciproquement, des compositions partielles $— \circ_i —$ sur un \mathbb{S} -module vérifiant les deux relations induisent une unique structure d'opérade sur ce \mathbb{S} -module.

2.1.5 Opérade libre

Dans une catégorie monoïdale (C, \otimes, I) possédant les sommes directes indexées par \mathbb{N} et telle que $— \otimes —$ commute des deux cotés avec celles-ci, alors en copiant la construction de l'algèbre libre sur un espace vectoriel, on obtient la notion de monoïde libre sur un objet de C . Les foncteurs de Schur ne commutant, en général, pas avec les sommes directes, on ne peut pas appliquer ces constructions pour la catégorie monoïdale $(\mathbb{S}\text{-mod}, \circ, I)$. On renvoie à [LV12] pour une construction "à la main" de l'opérade libre $\mathcal{T}(\mathcal{M})$ sur le \mathbb{S} -module \mathcal{M} ou à [Val09] pour une construction catégorique, permettant en plus d'expliquer la propérade libre du chapitre 3.

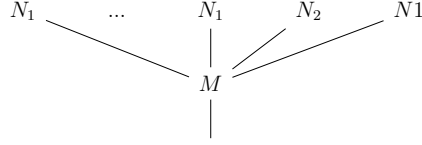
2.2 Constructions d'algèbre homologique pour les opérades

Afin que cette partie soit relativement concise, on omet volontairement les détails et renvoie à [LV12] pour un exposé précis.

2.2.1 Linéarisation

Définition 2.5. Soient M, N_1, N_2 trois \mathbb{S} -modules, le \mathbb{S} -module $M \circ (N_1; N_2)$ est par définition le sous \mathbb{S} -module de $M \circ (N_1 \oplus N_2)$ tel que $M \circ (N_1 \oplus N_2)(n)$ est le sous- \mathbb{S}_n -module de $M(n) \otimes_{\mathbb{S}_n} (N_1 \oplus N_2)^{\otimes n}$ où N_2 n'apparaît qu'une seule fois dans

chaque facteur direct. les éléments :



engendrent $M \circ (N_1; N_2)$. Cette définition est fonctorielle en M et N_2 . Le cas particulier où $N_1 = I$ est appelée *composition infinitésimale* de M et N :

$$M \circ_{(1)} N := M \circ (I; N)$$

dont les éléments typiques sont notés $(\mu; id, id, \dots, id, \nu, id, \dots)$ et pour $f : M \rightarrow N, g : M' \rightarrow N'$ des morphismes de \mathbb{S} -modules, $f \circ_{(1)} g \in \text{Hom}(M \circ_{(1)} N, M' \circ_{(1)} N')$ est définie par :

$$f \circ_{(1)} g(\mu; id, \dots, \nu, id, \dots) := (f(\mu), id, \dots, g(\nu), id, \dots)$$

Lemme 2.4. *Pour des \mathbb{S} -modules M, M', N, N' , on a :*

$$\begin{aligned} (M \oplus M') \circ_{(1)} N &= M \circ_{(1)} N \oplus M' \circ_{(1)} N \\ M \circ_{(1)} (N \oplus N') &= M \circ_{(1)} N \oplus M \circ_{(1)} N' \\ I \circ_{(1)} N &= N \\ M \circ_{(1)} I &= M \end{aligned}$$

La composition des \mathbb{S} -modules est alors linéarisée, celle des morphismes aussi. On présente maintenant une linéarisation des morphismes, sans changer la composition des \mathbb{S} -modules.

Définition 2.6. Soient $f, g : M_i \rightarrow N_i$ deux morphismes de \mathbb{S} -modules, la *composition infinitésimale* de f et g , $f \circ' g \in \text{Hom}(M_1 \circ N_1, M_2 \circ (N_1; N_2))$ est définie par

$$f \circ' g := \sum_i f \otimes (Id_{N_1} \otimes \dots \otimes Id_{N_1} \otimes g \otimes Id_{N_1} \otimes \dots \otimes Id_{N_1})$$

Où g apparaît en i -ème position. Si de plus $N_1 = N_2 = N$ alors la composée $M_2 \circ (N; N) \hookrightarrow M_2 \circ (N \oplus N) \xrightarrow{Id \circ (Id \oplus Id)} M \circ N$ est encore notée $f \circ' g$.

Lemme 2.5. *La composition infinitésimale est linéaire à droite :*

$$f \circ' (g + h) = f \circ' g + f \circ' h$$

Lorsque $(\mathcal{P}, \gamma, \eta)$ est une opérade, le produit de composition infinitésimale $\gamma_{(1)} : \mathcal{P} \circ_{(1)} \mathcal{P} \rightarrow \mathcal{P}$ est définie par la composée :

$$\gamma_{(1)} : \mathcal{P} \circ_{(1)} \mathcal{P} = \mathcal{P} \circ (I; \mathcal{P}) \hookrightarrow \mathcal{P} \circ (I \oplus \mathcal{P}) \xrightarrow{Id_{\mathcal{P}} \circ (\eta + Id_{\mathcal{P}})} \mathcal{P} \circ \mathcal{P} \xrightarrow{\gamma} \mathcal{P}$$

De même, pour une coopérade $(\mathcal{C}, \Delta, \eta)$, il y'a la décomposition infinitésimale $\Delta_{(1)} : \mathcal{C} \rightarrow \mathcal{C} \circ_{(1)} \mathcal{C}$.

2.2.2 Extension au cadre différentiel gradué

Définition 2.7. Un \mathbb{S} -module différentiel gradué, abrégé dg- \mathbb{S} -module, est une collection $\mathcal{M} = \{M(n), n \in \mathbb{N}\}$ de \mathbb{S}_n -modules différentiels gradués. Un morphisme de dg- \mathbb{S} -modules $f : (M, d_M) \rightarrow (N, d_N)$ est un morphisme de \mathbb{S} -modules gradués, de degré 0, qui commute avec les différentielles. La catégories des dg- \mathbb{S} -modules et ses morphismes est notée dg- \mathbb{S} -Mod.

Le \mathbb{S} -module I est considéré comme un \mathbb{S} -module gradué, concentré en degré 0. Soit $\mathbb{K}s := (\mathbb{K}s, 0, 0, \dots)$ le \mathbb{S} -module concentré en degré 1. La *suspension* d'un \mathbb{S} -module gradué M est définie par $sM := \mathbb{K}s \otimes M$. De façon analogue, $\mathbb{K}s^{-1}$ est le même \mathbb{S} -module, concentré en degré -1 , et la *désuspension* est donnée par $s^{-1}M := \mathbb{K}s^{-1} \otimes M$.

Définition 2.8. Soient (M, d_M) et (N, d_N) deux dg- \mathbb{S} -modules, leur *composition* $M \circ N$ est un \mathbb{S} -module gradué, qui est équipé de la différentielle

$$d_{M \circ N} := d_M \circ Id_N + Id_M \circ' d_N$$

Proposition 2.3. *La catégorie $(\text{dg-}\mathbb{S}\text{-Mod}, \circ, I)$ est monoïdale.*

Définition 2.9. Une opérade différentielle graduée est un monoïde dans la catégorie $(\text{dg-}\mathbb{S}\text{-Mod}, \circ, I)$ tel que le produit de composition et l'unité soient de degré 0.

Définition 2.10. Une *dérivation* $d : \mathcal{T}(E) \rightarrow \mathcal{T}(E)$ sur l'opérade libre sur le \mathbb{S} -module gradué E est une application linéaire telle que :

$$\gamma.(d \circ Id + Id \circ' d) = d.\gamma$$

Où $\gamma : \mathcal{T}(E) \circ \mathcal{T}(E) \rightarrow \mathcal{T}(E)$ est le produit.

Proposition 2.4. *Soit E un \mathbb{S} -module gradué. Une dérivation $d : \mathcal{T}(E) \rightarrow \mathcal{T}(E)$ est exactement déterminée par sa restriction $(\mathbb{S} - \text{linéaire})$ au sous- \mathbb{S} -module E .*

2.3 Adjonction cobar-bar opéradique

2.3.1 Morphismes tordants

Définition 2.11. Soit $(\mathcal{C}, \Delta, \epsilon)$ une coopérade et $(\mathcal{P}, \gamma, \eta)$ une opérade, la collection $\text{Hom}_{\mathbb{K}}(\mathcal{C}, \mathcal{P}) := \{\text{Hom}_{\mathbb{K}}(C(n), P(n)), n \in \mathbb{N}\}$ est munie d'une structure de \mathbb{S} -module :

$$(f.\sigma)(x) := f(x.\sigma^{-1}).\sigma$$

Proposition 2.5. *Le \mathbb{S} -module $\text{Hom}(\mathcal{C}, \mathcal{P})$ est une opérade, appelée opérade de convolution. Si de plus \mathcal{C} et \mathcal{P} sont différentiels gradués, alors pour tout \mathbb{S} -morphisme $f : \mathcal{C} \rightarrow \mathcal{P}$ de degré $|f|$, la dérivation*

$$\partial(f) := d_{\mathcal{P}} \circ f - (-1)^{|f|} f \circ d_{\mathcal{C}}$$

fait de $(\text{Hom}(\mathcal{C}, \mathcal{P}), \partial)$ une dg opérade.

Définition 2.12. Pour $f, g \in \text{Hom}_{\mathbb{S}}(\mathcal{C}, \mathcal{P})$, leur produit de convolution $f * g \in$

$\text{Hom}_{\mathbb{S}}(\mathcal{C}, \mathcal{P})$ est défini par :

$$f * g := \mathcal{C} \xrightarrow{\Delta_{(1)}} \mathcal{C} \circ_{(1)} \mathcal{C} \xrightarrow{f \circ_{(1)} g} \mathcal{P} \circ_{(1)} \mathcal{P} \xrightarrow{\gamma_{(1)}} \mathcal{P}$$

L'équation de Maurer-Cartan (M.C) dans $\text{Hom}_{\mathbb{S}}(\mathcal{C}, \mathcal{P})$ est :

$$\partial(f) + f * f = 0$$

Une solution de (M.C) de degré -1 est appelée *morphisme tordant opéradique*. Si de plus \mathcal{C} est coaugmentée par c et \mathcal{P} est augmentée par u , il faut que $u \circ f = 0$ et $f \circ c = 0$.

Lemme 2.6. *Tout morphisme de dg- \mathbb{S} -modules de degré -1 , $\alpha : \mathcal{C} \rightarrow \mathcal{P}$ induit une dérivation $d_{\alpha}^r : \mathcal{C} \circ \mathcal{P} \rightarrow \mathcal{C} \circ \mathcal{P}$, de carré nul si et seulement si α est un morphisme tordant.*

Dans ce cas, le complexe obtenu en tordant la différentielle $d_{\mathcal{C} \circ \mathcal{P}}$ est noté

$$\mathcal{C} \circ_{\alpha} \mathcal{P} := (\mathcal{C} \circ \mathcal{P}, d_{\alpha} := d_{\mathcal{C} \circ \mathcal{P}} + d_{\alpha}^r)$$

et appelé *produit de composition tordu à droite* de \mathcal{C} et \mathcal{P} .

2.3.2 Constructions cobar et bar

Pour une dg-opérade augmentée $(\mathcal{P}, \gamma, \eta, \epsilon)$, la construction bar^4 de \mathcal{P} est la coopérade $\mathcal{T}^c(s\bar{\mathcal{P}})$, munie de la somme de la différentielle d_{ext} induite par le produit de composition infinitésimal $\gamma_{(1)}$ et de celle induite par la différentielle $d_{\mathcal{P}}$ de \mathcal{P} , notée d_{int} . Elle est notée

$$\text{B}\mathcal{P} := (\mathcal{T}^c(s\bar{\mathcal{P}}), d := d_{ext} + d_{int})$$

De même, pour toute dg-coopérade conilpotent \mathcal{C} , la construction cobar de \mathcal{C} est donnée par l'opérade

$$\Omega\mathcal{C} := (\mathcal{T}(s^{-1}\bar{\mathcal{C}}), d := d_{ext} + d_{int})$$

4. Qui est fonctorielle en \mathcal{P}

Théorème 2.7. *Pour toute dg-opérade augmentée \mathcal{P} et toute dg-coopérade conilpotente \mathcal{C} , on a les isomorphismes naturels*

$$\mathrm{Hom}_{dg-Op}(\Omega\mathcal{C}, \mathcal{P}) \simeq \mathrm{Tw}(\mathcal{C}, \mathcal{P}) \simeq \mathrm{Hom}_{dg-conil-Op}(\mathcal{C}, B\mathcal{P})$$

2.4 Opérades quadratiques et dualité de Koszul

Une *donnée quadratique opéradique* est un couple (E, R) où E est un \mathbb{S} -module gradué et $R \subset \mathcal{T}(E)^{(2)}$ un sous \mathbb{S} -module gradué des éléments de poids 2 de $\mathcal{T}(E)$. Les éléments de R sont appelés les *relations*. Un *morphisme de données quadratiques* est la donnée d'un morphisme de \mathbb{S} -modules $f : E \rightarrow E'$ tel que $\mathcal{T}(f)(R) \subset R'$. On peut alors, généraliser les constructions faites au premier chapitre pour obtenir l'opérade quadratique $\mathcal{P}(E, R) := \mathcal{T}(E)/(R)$ associée à la donnée quadratique (E, R) , qui est universelle parmi les opérades quotient $\mathcal{T}(E) \twoheadrightarrow \mathcal{P}$ telles que la composée $R \hookrightarrow \mathcal{T}(E) \twoheadrightarrow \mathcal{P}$ soit nulle, ainsi que la coopérade quadratique $\mathcal{C}(E, R) \subset \mathcal{T}^c(E)$, vérifiant la propriété universelle :

$$\begin{array}{ccccc} & & 0 & & \\ & \searrow & & \searrow & \\ \mathcal{C} & \longrightarrow & \mathcal{T}^c(E) & \twoheadrightarrow & \mathcal{T}^c(E)/R \\ \exists! \downarrow & \nearrow & & & \\ & \mathcal{C}(E, R) & & & \end{array}$$

Définition 2.13. Pour une opérade quadratique $\mathcal{P} = \mathcal{P}(E, R)$, la *coopérade duale de Koszul* de \mathcal{P} est définie par :

$$\mathcal{P}^i := \mathcal{C}(sE, s^2R)$$

et l'opérade duale de Koszul :

$$\mathcal{P}^! := (\mathcal{S}^c \otimes_H \mathcal{P}^i)^*$$

Où \otimes_H est le produit d'hadamard de coopéades et $*$ est la dualité graduée arité par arité et \mathcal{S}^c est la coopéade $\text{End}_{s\mathbb{K}}^c$ [LV12] .

Proposition 2.6. *Si \mathcal{P} est quadratique, engendrée par un \mathbb{S} -module réduit de dimension finie en chaque arité, alors on a*

$$\mathcal{P}^! = \mathcal{P}(s^{-1}\mathcal{S}^{-1} \otimes_H E^*, R^\perp)$$

et

$$(\mathcal{P}^!)^! \simeq \mathcal{P}$$

Où R^\perp sera décrit dans la section sur les propéades.

Pour une donnée quadratique (E, R) , on a $\mathcal{P}(E, R)^{(1)} = E$ et $\mathcal{C}(E, R)^{(1)} = E$, on définit alors le morphisme tordant $\kappa : \mathcal{P}^i \rightarrow \mathcal{P}$ comme la composée :

$$\kappa : \mathcal{C}(sE, s^2R) \twoheadrightarrow sE \xrightarrow{s^{-1}} E \hookrightarrow \mathcal{P}(E, R)$$

Le complexe

$$\mathcal{P}^i \circ_\kappa \mathcal{P} := (\mathcal{P}^i \circ \mathcal{P}, d_\kappa)$$

est appelé *complexe de Koszul* de l'opéade \mathcal{P} . Une opéade quadratique est dite de *Koszul* si son complexe de Koszul associé est acyclique.

Théorème 2.8. *les énoncés suivants sont équivalents :*

- *Le complexe de Koszul droite $\mathcal{P}^i \circ_\kappa \mathcal{P}$ est acyclique.*
- *Le complexe de Koszul gauche $\mathcal{P} \circ_\kappa \mathcal{P}^i$ est acyclique.*
- *L'inclusion $\iota : \mathcal{P}^i \hookrightarrow B\mathcal{P}$ est un quasi-isomorphisme.*
- *La projection $\pi : \Omega\mathcal{P}^i \twoheadrightarrow \mathcal{P}$ est un quasi-isomorphisme.*

Chapitre 3

Propérades

3.1 Introduction

Définition 3.1. Un \mathbb{S} -bimodule \mathcal{P} est une collection $\{\mathcal{P}(m, n), (m, n) \in \mathbb{N}^2\}$ de $\mathbb{K}[\mathbb{S}_m] \otimes_{\mathbb{K}} \mathbb{K}[\mathbb{S}_n^{op}]$ -modules à gauche. Un morphisme de \mathbb{S} -bimodules $f : \mathcal{P} \rightarrow \mathcal{Q}$ est une collection de morphismes $f_{m,n} : \mathcal{P}(m, n) \rightarrow \mathcal{Q}(m, n)$ de $\mathbb{S}_m \otimes \mathbb{S}_n^{op}$ -modules. La catégorie des \mathbb{S} -bimodules et leurs morphismes est notée $\mathbb{S}\text{-BiMod}$.

Exemple 3.1. — *A tout \mathbb{K} -espace vectoriel V est associé le \mathbb{S} -bimodule*

$$End_V := \{Hom_{\mathbb{K}}(V^{\otimes n}, V^{\otimes m}), (m, n) \in \mathbb{N}^2\}$$

Où les actions des groupes symétriques sont données par permutation des entrées et sorties.

— *Un \mathbb{S} -module $\mathcal{M} = (M(n))_{n \in \mathbb{N}}$ peut-être considéré comme un \mathbb{S} -bimodule concentré en arités $(1, n), n \in \mathbb{N}$. Ce qui fait de $\mathbb{S}\text{-Mod}$ une sous-catégorie pleine de $\mathbb{S}\text{-BiMod}$.*

Définition 3.2. Un graphe orienté est un graphe non planaire où les orientations des arrêtes sont données par un flot descendant et tel que les entrées et sorties de chaque sommet soient numérotées par les entiers. On suppose aussi que les entrées et sorties globales du graphe sont numérotées par les entiers. L'ensemble des graphes

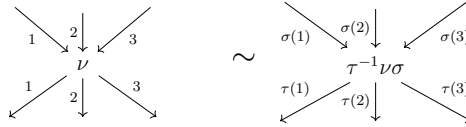
orientés est noté \mathcal{G} . Si de plus les sommets d'un graphe g peuvent être répartis en deux niveaux $\mathcal{N}_i, i \in \{1, 2\}$, g est qualifié de graphe à deux niveaux, et l'ensemble des graphes à deux niveaux est noté \mathcal{G}^2 .

La composition $- \circ -$ des \mathbb{S} -modules s'étend en une composition des \mathbb{S} -bimodules $-\boxtimes_c -$:

Définition 3.3. Pour deux \mathbb{S} -bimodules \mathcal{P} et \mathcal{Q} , leur produit de composition connexe¹ $\mathcal{Q} \boxtimes_c \mathcal{P}$ est donné par

$$\mathcal{Q} \boxtimes_c \mathcal{P} := \left(\bigoplus_{g \in \mathcal{G}_c^2} \bigotimes_{\nu \in \mathcal{N}_2} \mathcal{Q}(|\text{Out}(\nu)|, |\text{In}(\nu)|) \otimes_{\mathbb{K}} \bigotimes_{\nu \in \mathcal{N}_1} \mathcal{P}(|\text{Out}(\nu)|, |\text{In}(\nu)|) \right) / \sim$$

Où \mathcal{G}_c^2 est l'ensemble des graphes orientés à deux niveaux connexes, $\text{Out}(\nu)$ et $\text{In}(\nu)$ sont les sorties et entrées d'un sommet ν , et \sim est la relation d'équivalence engendrée par :



Autrement dit, les actions des groupes symétriques sur les sommets des graphes sont données par permutations de ses entrées et sorties.

Proposition 3.1. Soit $\mathcal{I} = \mathbb{K}$ le \mathbb{S} -bimodule concentré en arité $(1, 1)$, la catégorie $(\mathbb{S}\text{-BiMod}, \boxtimes_c, \mathcal{I})$ est monoïdale.

Démonstration 3.1. L'associativité du produit \boxtimes_c provient de celle du produit tensoriel \otimes dans $\mathbb{K}\text{-Mod}$. Pour montrer la relation d'unité $\mathcal{P} \boxtimes_c \mathcal{I} \simeq \mathcal{P}$, il suffit de remarquer que la somme est en fait prise sur les graphes connexes à deux niveaux dont le premier niveau n'est composé que de sommets à une entrée et une sortie. L'hypothèse de connexité assure que toutes les sorties sont reliées sur le même sommet au niveau 2 et la relation d'équivalence assure qu'il n'y ait qu'une seule copie de $\mathcal{P}(m, n)$ dans $\mathcal{P} \boxtimes_c \mathcal{I}$. Ainsi $\mathcal{P} \boxtimes_c \mathcal{I} \simeq \bigoplus_{(m, n)} \mathcal{P}(m, n) = \mathcal{P}$. L'isomorphisme $\mathcal{I} \boxtimes_c \mathcal{P} \simeq \mathcal{P}$ se traite de la même façon.

1. Pour une formule algébrique, on renvoie à [Val07].

Corollaire 3.1. *les inclusions de catégories monoïdales suivantes sont pleines :*

$$(\text{Vect}, \otimes, \mathbb{K}) \hookrightarrow (\mathbb{S}\text{-Mod}, \circ, I) \hookrightarrow (\mathbb{S}\text{-BiMod}, \boxtimes_c, \mathcal{I})$$

Contrairement aux \mathbb{S} -modules, il est possible de composer horizontalement les \mathbb{S} -bimodules : pour $f \in \text{Hom}(V^{\otimes n}, V^{\otimes m})$ et $g \in \text{Hom}(V^{\otimes k}, V^{\otimes l})$, la concaténation de f et g est donnée par leur produit tensoriel $f \otimes g \in \text{Hom}(V^{\otimes n+k}, V^{\otimes m+l})$, et la définition dans le cas général est :

Définition 3.4. Pour \mathcal{P} et \mathcal{Q} deux \mathbb{S} -bimodules, leur *produit de concaténation* $\mathcal{Q} \otimes \mathcal{P}$ est donné par

$$\mathcal{Q} \otimes \mathcal{P}(m, n) := \bigoplus_{\substack{m'+m''=m \\ n'+n''=n}} \mathbb{K}[\mathbb{S}_m] \otimes_{\mathbb{S}_{m'} \times \mathbb{S}_{m''}} \mathcal{Q}(m', n') \otimes \mathcal{P}(m'', n'') \otimes_{\mathbb{S}_{n'} \times \mathbb{S}_{n''}} \mathbb{K}[\mathbb{S}_n]$$

Le \mathbb{S} -bimodule \mathbb{K} concentré en arité $(0, 0)$ induit une structure monoïdale symétrique sur la catégorie des \mathbb{S} -bimodules munie du produit de concaténation.

Définition 3.5. Une *Propérade* est un monoïde dans la catégorie monoïdale $(\mathbb{S}\text{-BiMod}, \boxtimes_c, I)$ et un morphisme de propérades est un morphisme de monoïdes. La catégorie des propérades est notée *Properads*. De même, une *copropérade* est un comonoïde dans $(\mathbb{S}\text{-BiMod}, \boxtimes_c, I)$.

Définition 3.6. Soient (\mathcal{P}, μ) et (\mathcal{Q}, ν) deux propérades, leur *produit de Hadamard* $\mathcal{P} \otimes_H \mathcal{Q}$ est la propérade définie par

$$(\{\mathcal{P} \otimes_H \mathcal{Q}(m, n) := \mathcal{P}(m, n) \otimes \mathcal{Q}(m, n)\}, \mu \otimes \nu)$$

Remarquons que $\mu \otimes \nu$ est un abus d'écriture qui utilise l'identification

$$(\mathcal{P} \otimes_H \mathcal{Q} \boxtimes_c \mathcal{P} \otimes_H \mathcal{Q})(m, n) \simeq (\mathcal{P} \boxtimes_c \mathcal{P})(m, n) \otimes (\mathcal{Q} \boxtimes_c \mathcal{Q})(m, n)$$

Dans le cas gradué, les conventions de Koszul doivent être prises en compte.

Exemple 3.2. End_V est une propérade. Le cas particulier où $V = \mathbb{K}$ est concentré en degré 1 est noté s et appelé propérade signature, ou propérade suspension [Gan02]. le cas où \mathbb{K} est concentré en degré -1 est noté s^{-1} . Les collections $\Lambda := \{\Lambda(m, n) := s(m, n)[2 - 2m]\}$ et $\Lambda^{-1} := \{\Lambda^{-1}(m, n) := s^{-1}(m, n)[2m - 2]\}$ sont aussi des propérades.

Un autre exemple est donné par ce qui suit : le foncteur d'oubli $U : \text{Properads} \rightarrow \mathbb{S}\text{-BiMod}$ admet un adjoint à gauche \mathcal{F} et pour un \mathbb{S} -bimodule V , $(\mathcal{F}(V), i_V : V \rightarrow \mathcal{F}(V))$ est appelé propérade libre sur V .

Construction 3.1. Soit V un \mathbb{S} -bimodule, la propérade libre sur V est donnée par la somme sur les graphes connexes \mathcal{G}_c :

$$\mathcal{F}(V) := \left(\bigoplus_{g \in \mathcal{G}_c} \bigotimes_{\nu \in \mathcal{N}} V(|\text{Out}(\nu)|, |\text{In}(\nu)|) \right) / \sim$$

La composition μ est induite de la composition des graphes orientés, et le morphisme $V \rightarrow \mathcal{F}(V)$ est induit par l'inclusion $\mathcal{G}_{c,(1)} \hookrightarrow \mathcal{G}_c$.

L'application qui à un graphe lui associe son nombre de sommet(s) induit une partition de l'ensemble des graphes orientés connexes : $\mathcal{G}_c = \bigsqcup_{n \in \mathbb{N}} \mathcal{G}_{c,(n)}$, et ainsi une décomposition $\mathcal{F}(V) = \bigoplus_{n \in \mathbb{N}} \mathcal{F}_{(n)}(V)$.

Remarque 3.1. Pour la définition de copropérade colibre conilpotente (le coproduit itéré suffisamment de fois de tout élément est nul) sur un \mathbb{S} -bimodule V on renvoie à [Val07].

Définition 3.7. Une donnée quadratique (V, R) dans la catégorie $\mathbb{S}\text{-biMod}$ est la donnée d'un \mathbb{S} -bimodule V et d'un sous- \mathbb{S} -bimodule R de $\mathcal{F}_{(2)}(V)$. A une telle donnée quadratique est associée une propérade, qualifiée de quadratique :

$$\mathcal{F}(V, R) := \mathcal{F}(V)/(R)$$

Par extension, une propérade est dite *quadratique* si elle est isomorphe à une certaine propérade $\mathcal{F}(V, R)$, et *binnaire quadratique* si V est concentré en arité $(1, 2)$ et $(2, 1)$.

Exemple 3.3. La propétrade $BiLie$ encodant les bialgèbres de Lie est quadratique binaire, engendrée par le \mathbb{S} -bimodule $V \oplus W$, avec $V = V(1, 2) = m.\mathbb{K}$ où le produit m est anticommutatif et $W = W(2, 1) = \Delta.\mathbb{K}$ où le coproduit Δ est anticommutatif. Et pour espace des relations R , l'espace engendré par

$$\begin{aligned}
- R &:= \begin{array}{c} 1 \quad 2 \quad 3 \\ \diagdown \quad \diagup \\ \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \end{array} + \begin{array}{c} 2 \quad 3 \quad 1 \\ \diagdown \quad \diagup \\ \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \end{array} + \begin{array}{c} 3 \quad 1 \quad 2 \\ \diagdown \quad \diagup \\ \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \end{array} \\
- S &:= \begin{array}{c} \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \quad 2 \quad 3 \end{array} + \begin{array}{c} \quad \quad \quad \\ \diagup \quad \diagdown \\ 2 \quad 3 \quad 1 \end{array} + \begin{array}{c} \quad \quad \quad \\ \diagup \quad \diagdown \\ 3 \quad 1 \quad 2 \end{array} \\
- D &:= \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \quad 2 \end{array} - \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \quad 2 \end{array} + \begin{array}{c} 2 \quad 1 \\ \diagdown \quad \diagup \\ \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \quad 2 \end{array} - \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \quad 2 \end{array} + \begin{array}{c} 2 \quad 1 \\ \diagdown \quad \diagup \\ \quad \quad \quad \\ \diagup \quad \diagdown \\ 1 \quad 2 \end{array}
\end{aligned}$$

3.2 Calculs de duaux

Dans toute cette section, les \mathbb{S} -bimodules sont de dimensions finies en chaque arité. Pour un \mathbb{S} -bimodule \mathcal{P} , son dual de Czech \mathcal{P}^\vee est défini par

$$\mathcal{P}^\vee(m, n) := sgn_{\mathbb{S}_m} \otimes_{\mathbb{K}} \mathcal{P}(m, n)^* \otimes_{\mathbb{K}} sgn_{\mathbb{S}_n}$$

Et sa suspension $s\mathcal{P}$ est définie par

$$s \otimes \mathcal{P}$$

où s est le \mathbb{S} -bimodule concentré en arité $(0, 0)$ et degré 1 : $s(0, 0) := \mathbb{K}$.

Définition 3.8. Soit $\mathcal{P} = \mathcal{P}(E, R)$ une propétrade quadratique, sa copropétrade duale est définie par

$$\mathcal{P}^i := \mathcal{F}^c(sE, s^2R)$$

et sa propétrade duale est définie par

$$\mathcal{P}^! := \Lambda^{-1} \bigotimes_H (\mathcal{P}^i)^*$$

Proposition 3.2. Soit E un \mathbb{S} -bimodule (concentré en les arités $(1, 2)$ et $(2, 1)$).

L'inclusion $E^\vee = \Lambda^{-1}sE^* \hookrightarrow \Lambda^{-1}(\mathcal{F}^c(sE))^*$ induit un unique isomorphisme de propérades au dessus de E^\vee :

$$\Phi : \mathcal{F}(E^\vee) \rightarrow \Lambda^{-1}(\mathcal{F}^c(sE))^* \stackrel{[GK94]}{=} \mathcal{F}(E)^\vee$$

qui induit un isomorphisme encore noté Φ :

$$\Phi : \mathcal{F}(E^\vee)/(\Phi^{-1}(\Lambda^{-1}s^2R^\perp)) \rightarrow \mathcal{P}^!$$

Exemple 3.4. Soit $E = V \oplus W$ un \mathbb{S} -bimodule, où V est concentré en arité $(1, 2)$ et engendré par des produits $\begin{smallmatrix} & \diagdown & \\ & \cdot & \\ & \diagup & \end{smallmatrix}$, et W est concentré en arité $(2, 1)$, engendré par des coproduits $\begin{smallmatrix} & \diagup & \\ & \cdot & \\ & \diagdown & \end{smallmatrix}$. L'isomorphisme Φ est alors donné sur le sous-module de poids 2 en arité $(2, 2)$, $\mathcal{F}_{(2)}(E^\vee)(2, 2)$, par :

$$\begin{aligned} - & \begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} \mapsto - \begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} = \left(\begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} \right)^* \\ - & \begin{smallmatrix} & \diagup & & \diagdown & \\ & \cdot & & \cdot & \\ & \diagdown & & \diagup & \end{smallmatrix} \mapsto + \begin{smallmatrix} & \diagup & & \diagdown & \\ & \cdot & & \cdot & \\ & \diagdown & & \diagup & \end{smallmatrix} = \left(\begin{smallmatrix} & \diagup & & \diagdown & \\ & \cdot & & \cdot & \\ & \diagdown & & \diagup & \end{smallmatrix} \right)^* \\ - & \begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} \mapsto - \begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} = \left(\begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} \right)^* \\ - & \begin{smallmatrix} & \diagup & & \diagdown & \\ & \cdot & & \cdot & \\ & \diagdown & & \diagup & \end{smallmatrix} \mapsto - \begin{smallmatrix} & \diagup & & \diagdown & \\ & \cdot & & \cdot & \\ & \diagdown & & \diagup & \end{smallmatrix} \\ - & \begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} \mapsto \begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} \end{aligned}$$

Et en arité $(1, 3)$ par :

$$\begin{aligned} - & \begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} \mapsto \begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} = \left(\begin{smallmatrix} & \diagdown & & \diagup & \\ & \cdot & & \cdot & \\ & \diagup & & \diagdown & \end{smallmatrix} \right)^* \\ - & \forall \sigma \in \mathbb{S}_3 : \begin{smallmatrix} & \sigma(1) & & \sigma(2) & & \sigma(3) \\ & \cdot & & \cdot & & \cdot \\ & \diagup & & \diagdown & & \diagup \end{smallmatrix} \mapsto \epsilon(\sigma) \begin{smallmatrix} & \sigma(1) & & \sigma(2) & & \sigma(3) \\ & \cdot & & \cdot & & \cdot \\ & \diagup & & \diagdown & & \diagup \end{smallmatrix} \end{aligned}$$

Pour V et W deux \mathbb{S} -bimodules, on notera $(I; W) \boxtimes_c (I; V)$ le sous- \mathbb{S} -bimodule de $(I \oplus W) \boxtimes_c (I \oplus V)$ composé des graphes ayant exactement un sommet indexé par une opération de V au niveau 1 et un sommet indexé par une opération de W au niveau 2.

Définition 3.9. Soient V et W deux \mathbb{S} -bimodules, une *règle de réécriture* λ est un morphisme de \mathbb{S} -bimodules :

$$\lambda : (I; W) \boxtimes_c (I; V) \rightarrow (I; V) \boxtimes_c (I; W)$$

telle que si on note $D_\lambda := \text{Im}((Id, -\lambda))$ et si $\mathcal{P} := \mathcal{F}(V \oplus W)/(R \oplus D_\lambda \oplus S)$ est une propérade avec $R \subset \mathcal{F}_{(2)}(V)$, $S \subset \mathcal{F}_{(2)}(W)$, l'application

$$\mathcal{F}(V)/(R) \boxtimes_c \mathcal{F}(W)/(S) \rightarrow \mathcal{P}$$

est injective.

Lemme 3.2. Soit $\lambda : (I; W) \boxtimes_c (I; V) \rightarrow (I; V) \boxtimes_c (I; W)$ une règle de réécriture et D_λ le graphe de $-\lambda$. La règle de réécriture dans $\mathcal{F}(E^\vee)$:

$$\tilde{\lambda} := \Phi_{\downarrow}^{-1} \circ -(\Lambda^{-1}(s^2\lambda)^*) \circ \Phi_{\downarrow} : (I; V^\vee) \boxtimes_c (I; W^\vee) \rightarrow (I; W^\vee) \boxtimes_c (I; V^\vee)$$

vérifie :

$$D_{\tilde{\lambda}} = \Phi^{-1}(D_{\Lambda^{-1}s^2\lambda}^\perp)$$

Démonstration 3.2. On commence par rappeler les deux faits classiques suivants :

— Si $f : A \rightarrow A$ est une application et $\phi : A \xrightarrow{\sim} B$ est une bijection, alors

$$\phi_*(Gr(f)) = Gr(\phi \circ f \circ \phi^{-1})$$

— Si $f : A \rightarrow B$ est une application linéaire, en notant $f^* : B^* \rightarrow A^*$ l'application transposée, on a

$$Gr(f)^\perp = Gr(-f^*)$$

où $A \oplus B$ et $B \oplus A$ sont identifiés canoniquement et $Gr(f)$ est le graphe de f .

On en déduit alors

$$\begin{aligned} D_{\tilde{\lambda}} &= \Phi^{-1}(D_{-\Lambda^{-1}(s^2\lambda)^*}) \\ &= \Phi^{-1}(\Lambda^{-1}D_{s^2\lambda}^\perp) \\ &= \Phi^{-1}(D_{\Lambda^{-1}s^2\lambda}^\perp) \end{aligned}$$

Notons que la dernière égalité est obtenue par finitude des dimensions et exactitude de $\Lambda^{-1} \otimes -$.

Théorème 3.3 (proposition 8.2 [Val07]). *Soit \mathcal{P} une propétrade de la forme $F(V \oplus W)/(R \oplus D_\lambda \oplus S)$, $R \in \mathcal{F}_{(2)}(V)$, $S \in \mathcal{F}_{(2)}(W)$ et $D_\lambda \subset (\mathcal{I} \oplus W) \boxtimes_c (\mathcal{I} \oplus V) \oplus (\mathcal{I} \oplus V) \boxtimes_c (\mathcal{I} \oplus W)$, définie par une règle de réécriture λ , telle que la somme des dimensions de $V \oplus W$ soit finie. Alors la propétrade duale de \mathcal{P} est donnée par*

$$\mathcal{P}^\dagger \simeq \mathcal{F}(W^\vee \oplus V^\vee)/(\tilde{S} \oplus D_{\tilde{\lambda}} \oplus \tilde{R})$$

avec pour règle de réécriture $\tilde{\lambda}$ définie au lemme 3.2 et $\tilde{S} := \Phi^{-1}(\Lambda^{-1}s^2S^\perp)$, $\tilde{R} := \Phi^{-1}(\Lambda^{-1}s^2R^\perp)$.

Esquisse de démonstration 3.1. *Il reste à vérifier que $\tilde{D}_\lambda = D_{\tilde{\lambda}}$, c'est exactement l'objet du lemme 3.2.*

Exemple 3.5. *La règle de réécriture pour la propétrade $BiLie = \mathcal{F}(V \oplus W)/(R \oplus D \oplus S)$, avec $V = \langle \mu \rangle_{\mathbb{K}} = \begin{array}{c} \diagup \quad \diagdown \\ | \end{array} \cdot \mathbb{K}$ et $W = \langle \Delta \rangle_{\mathbb{K}} = \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \\ | \end{array} \cdot \mathbb{K}$ munis des actions par signature, est donnée par*

$$\lambda : (I; W) \boxtimes_c (I; V) = W \otimes_{\mathbb{K}} V \rightarrow (I; V) \boxtimes_c (I; W)$$

$$\begin{array}{c} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array} \end{array} \mapsto \begin{array}{c} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array} - \begin{array}{c} 2 \quad 1 \\ \diagdown \quad \diagup \\ | \end{array} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array} + \begin{array}{c} 1 \quad 2 \\ \diagup \quad \diagdown \\ | \end{array} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array} - \begin{array}{c} 2 \quad 1 \\ \diagup \quad \diagdown \\ | \end{array} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array} \end{array}$$

Calculons $\tilde{\lambda}$:

$$- \tilde{\lambda} \left(\begin{array}{c} 2 \quad 1 \\ \diagdown \quad \diagup \\ | \end{array} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array} \right) \left(\begin{array}{c} \diagup \quad \diagdown \\ | \end{array} \right) = \begin{array}{c} 2 \quad 1 \\ \diagdown \quad \diagup \\ | \end{array} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array} (- \begin{array}{c} 2 \quad 1 \\ \diagdown \quad \diagup \\ | \end{array} \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ | \end{array}) = +1$$

i.e :

$$\tilde{\lambda}(\text{diagram}) = (\text{diagram})^* = + \text{diagram}$$

$$- \tilde{\lambda}(\text{diagram})(\text{diagram}) = \text{diagram}(-\text{diagram}) = +1$$

i.e :

$$\tilde{\lambda}(\text{diagram}) = + \text{diagram}$$

$$- \tilde{\lambda}(\text{diagram}) = + \text{diagram}$$

$$- \tilde{\lambda}(\text{diagram})(\text{diagram}) = + \text{diagram}$$

Ce qui donne

$$D_{\tilde{\lambda}} = \langle \text{diagram} - \text{diagram} \rangle \oplus \langle \text{diagram} - \text{diagram} \rangle \\ \oplus \langle \text{diagram} - \text{diagram} \rangle \oplus \langle \text{diagram} - \text{diagram} \rangle$$

Corollaire 3.4. La propétrade duale de Koszul $\text{BiLie}^!$ de la propétrade des bialgèbres de Lie, BiLie , est donnée par $\text{BiLie}^! = F(V \oplus W)/(R \oplus D \oplus S)$ où V est engendré sur \mathbb{K} par un produit commutatif m en arité $(1, 2)$ et W par un coproduit cocommutatif Δ en arité $(2, 1)$. L'espace des relations, $R \oplus D \oplus S$, est engendré par

$$- \text{diagram} - \text{diagram} \quad \text{l'associateur.}$$

$$- \text{diagram} - \text{diagram} \quad \text{le coassociateur.}$$

$$- D_{\tilde{\lambda}}$$

$$- \text{diagram}$$

est $(e_{Id}, e_{(123)}, e_{(132)})$ et la relation de Jacobi s'écrit dans cette base $e_{Id} + e_{(123)} + e_{(132)}$, sont orthogonal est donc engendré par $\langle e_{Id}^* - e_{(123)}^*; e_{(123)}^* - e_{(132)}^* \rangle = \langle e_{Id}^* - e_{(123)}^* \rangle$, c'est à dire l'associateur. Le cas de l'arité $(3, 1)$ est dual à celui-ci. En arité $(1, 1)$

l'orthogonal de 0 est donné par $\langle \begin{array}{c} | \\ \diagdown \quad \diagup \\ | \end{array} \rangle$. En arité $(2, 2)$, c'est $D_{\tilde{\lambda}}$ calculé à l'exemple 3.5.

Annexe A

Rappels

On commence par rappeler quelques résultats fondamentaux sur les représentations des groupes finis, sous l'hypothèse que la caractéristique du corps \mathbb{K} ne divise par l'ordre du groupe.

Définition A.1. Soit G un groupe et $H \leq G$ un sous groupe de G . Pour un H -module à droite M , la *représentation induite* de G sur M est notée $Ind_H^G M$, définie par :

$$Ind_H^G M := M \otimes_H \mathbb{K}[G]$$

où la multiplication à gauche $H \times G \rightarrow G$ induit la structure de H -module à gauche de $\mathbb{K}[G]$.

Théorème A.1 (Maschke). *Toute représentation de G est somme directe de sous représentations irréductibles.*

Théorème A.2. *Toute représentation irréductible de G est facteur direct dans la représentation régulière.*

Une conséquence de ces deux résultats est que tout $\mathbb{K}[G]$ -module est facteur direct d'un module libre, i.e est projectif. On en déduit un lemme de non-annulation :

Lemme A.3. *Soit B un \mathbb{S}_n -module non réduit à 0, alors*

$$B \otimes_{\mathbb{S}_n} (\mathbb{K}^n)^{\otimes n} \neq 0$$

Démonstration A.1. *L'inclusion*
$$\begin{array}{ccc} i : \mathbb{K}[\mathbb{S}_n] & \hookrightarrow & (\mathbb{K}^n)^{\otimes n} \\ 1 & \mapsto & e_1 \otimes \dots \otimes e_n \end{array}$$
 induit la suite exacte courte :

$$0 \rightarrow \mathbb{K}[\mathbb{S}_n] \xrightarrow{i} (\mathbb{K}^n)^{\otimes n} \twoheadrightarrow \text{Coker}(i) \rightarrow 0$$

Par projectivité de B , on en déduit la suite exacte courte :

$$0 \rightarrow B \otimes_{\mathbb{S}_n} \mathbb{K}[\mathbb{S}_n] \simeq B \xrightarrow{i} B \otimes_{\mathbb{S}_n} (\mathbb{K}^n)^{\otimes n} \twoheadrightarrow B \otimes_{\mathbb{S}_n} \text{Coker}(i) \rightarrow 0$$

En particulier, $B \otimes_{\mathbb{S}_n} (\mathbb{K}^n)^{\otimes n}$ est non nul.

Remerciements

Je tenais, dans ces lignes, à remercier les quelques personnes sans qui ce mémoire n'existerait pas.

Je remercie Salim Rivière pour avoir encadré ce mémoire avec dévouement et patience.

Je remercie également Friedrich Wagemann pour son dévouement, son humanité et pour avoir toujours su répondre avec clarté à mes nombreuses questions.

Enfin, une pensée toute particulière pour Guillaume Roux et ces nombreuses heures passées à discuter de topologie algébrique après les cours de géométrie affine. Ces discussions, passionnées, ont très nettement influencées mes goûts mathématiques, et je tenais à l'en remercier chaleureusement.

Bibliographie

- [Gan02] Wee Liang Gan. Koszul duality for dioperads. *arXiv preprint math/0201074*, 2002.
- [GK94] Victor Ginzburg and Mikhail Kapranov. Koszul duality for operads. *Duke Math. J.*, 76(1) :203–272, 10 1994.
- [Gro57] Alexander Grothendieck. Sur quelques points d’algèbre homologique, i. *Tohoku Math. J. (2)*, 9(2) :119–221, 1957.
- [Hin97] Vladimir Hinich. Homological algebra of homotopy algebras. *Communications in algebra*, 25(10) :3291–3323, 1997.
- [HMS74] Dale Husemoller, John C. Moore, and James Stasheff. Differential homological algebra and homogeneous spaces. *Journal of Pure and Applied Algebra*, 5(2) :113 – 185, 1974.
- [LV12] Jean-Louis Loday and Bruno Vallette. *Algebraic operads*, volume 346. Springer Science & Business Media, 2012.
- [Mac] S MacLane. Categories for the working mathematician. 1972.
- [Mil12] Joan Millès. The koszul complex is the cotangent complex. *International Mathematics Research Notices*, 2012(3) :607–650, 2012.
- [Val07] Bruno Vallette. A koszul duality for props. *Transactions of the American Mathematical Society*, 359(10) :4865–4943, 2007.

- [Val09] Bruno Vallette. Free monoid in monoidal abelian categories. *Applied Categorical Structures*, 17(1) :43–61, 2009.

Cohomologie de Hochschild des schémas

Lucas Darbas

Mémoire de M2

Mathématiques fondamentales et appliquées
Algèbre et Géométrie

Université de Nantes - 2020

Sous la direction de

Hossein Abbaspour
Friedrich Wagemann

La cohomologie de Hochschild est une théorie d'algèbre homologique concernant les algèbres associatives. Elle apparait en 1945 dans les travaux de Gerhard Hochschild et sera étudié durant le XX^{eme} siècle par de nombreux mathématiciens. Gerstenhaber, Schack, Loday, on encore Grothendieck contribueront à étendre cette théorie aux schémas à travers plusieurs définitions. Le but de ce mémoire est d'étudier les travaux de Swan pour comprendre ces différentes définitions.

Le cadre général est une algèbre associative A (non nécessairement unitaire) sur un anneau commutatif k , et un bimodule M sur A , c'est à dire un A -module à droite et à gauche satisfaisant les relations suivantes

$$\lambda m = m \lambda$$

$$a(mb) = (am)b$$

pour tout $m \in M$, $\lambda \in k$ et $a, b \in A$. Lorsque A est unitaire, cela revient à considérer un module sur l'anneau $A^e = A \otimes A^{op}$, où $\otimes = \otimes_k$, à travers la formule

$$(a \otimes b)m = amb$$

pour tout $m \in M$ et $a, b \in A$. Étant donné notre motivation à étudier des schémas, A sera toujours unitaire et même commutative. On introduit ensuite le complexe de Hochschild

$$C_n(A, M) = M \otimes A^{\otimes n}$$

dont la différentielle est définie par les applications k -linéaire suivantes

$$d = \sum_{i=0}^n (-1)^i d_i : M \otimes A^{\otimes n} \rightarrow M \otimes A^{\otimes n-1}$$

$$d_0(m \otimes a_1 \otimes \cdots \otimes a_n) = ma_1 \otimes a_2 \otimes \cdots \otimes a_n$$

$$d_i(m \otimes a_1 \otimes \cdots \otimes a_n) = m \otimes a_1 \otimes \cdots \otimes a_i a_{i+1} \otimes \cdots \otimes a_n \text{ pour } 0 < i < n$$

$$d_n(m \otimes a_1 \otimes \cdots \otimes a_n) = a_n m \otimes a_1 \otimes \cdots \otimes a_{n-1}$$

Lorsque A est commutative, ces applications sont A^e -linéaires. On peut alors définir l'homologie de Hochschild de A à valeurs dans M par

$$H_n(A, M) = H_n(C_\bullet(A, M))$$

Traditionnellement, on note $C_\bullet(A) = C_\bullet(A, M)$ et $HH_\bullet(A) = H_\bullet(A, M)$. Pour mieux comprendre cette homologie, il est commode d'utiliser le complexe "bar" de A^e -module

$$B_n(A) = A \otimes A^{\otimes n} \otimes A$$

dont la différentielle est donnée par les applications A^e -linéaires

$$d' = \sum_{i=0}^n (-1)^i d'_i : A \otimes A^{\otimes n} \otimes A \rightarrow A \otimes A^{\otimes n-1} \otimes A$$

$$d'_i(a_0 \otimes \cdots \otimes a_{n+1}) = a_0 \otimes \cdots \otimes a_i a_{i+1} \otimes \cdots \otimes a_{n+1}$$

La multiplication dans A permet d'obtenir une résolution de A^e -module

$$B_\bullet(A) \rightarrow A$$

Si A est projective sur k , alors $B_\bullet(A)$ est projectif sur A^e , et dans ce cas

$$H_n(A, M) = \text{Tor}_n^{A^e}(M, A) = \text{Tor}_n^{A^e}(A, M)$$

Pour définir la cohomologie de Hochschild de A à valeurs dans M , on s'inspire la situation précédente en posant

$$H^n(A, M) = H^n(\operatorname{Hom}_{A^e}(B_\bullet(A), M))$$

de sorte que si A est projective sur k alors

$$H^n(A, M) = \operatorname{Ext}_{A^e}^n(A, M)$$

Puisque les schémas que nous considérerons seront toujours basés sur un corps, les algèbres que l'ont rencontrera seront toujours projectives. Ainsi, on préférera définir l'homologie et la cohomologie de Hochschild à travers les foncteurs dérivés $\operatorname{Tor}_\bullet^{A^e}(A, -)$ et $\operatorname{Ext}_{A^e}^\bullet(A, -)$.

Dans son article "Hochschild cohomology of quasiprojective schemes", Swan introduit trois définitions différentes de la cohomologie de Hochschild d'un schéma basé sur un corps, puis il prouve que ces trois définitions coïncident si le schéma est quasi-projectif. La démonstration fait appel à différents concepts de géométrie algébrique et utilise des techniques standard d'algèbre homologique telles que les suites spectrales. Notre objectif est de comprendre cet article en détaillant les preuves de chacun des résultats intermédiaires en apportant parfois des preuves alternatives ainsi que la démonstration générale. On conservera le plan en dix parties de l'article original.

Sommaire

1	Cohomologie de Hochschild	7
2	Définition par l'hyper-ext	12
3	Définition de Gerstenhaber-Schack	18
4	Résolutions localement libres	22
5	Suites spectrales	27
6	Résolutions plates	33
7	Lemmes théoriques sur les faisceaux	39
8	Préfaisceau de faisceau	42
9	Technique de recollement de Čech	46
10	Preuve du théorème	49

1 Cohomologie de Hochschild

Rappelons la définition de la cohomologie de Hochschild d'une algèbre commutative A sur un corps k . On considère $A^e = A \otimes A$ avec $\otimes = \otimes_k$ ainsi que la multiplication

$$\varepsilon : A^e \rightarrow A$$

qui permet de regarder un A -module M comme un A^e -module que l'on notera M_ε ou simplement M si cela ne prête à aucune confusion. La cohomologie de Hochschild de A à valeur dans un A^e -module M est définie par

$$H^\bullet(A, M) = \text{Ext}_{A^e}^\bullet(A, M)$$

On peut calculer cette cohomologie en utilisant la résolution projective de A^e -module

$$B_\bullet(A) \xrightarrow{\varepsilon} A$$

où $B_\bullet(A)$ est le complexe "bar" de A [2, 1.1.12]. Très souvent, on souhaite calculer la cohomologie de Hochschild de A à valeur dans un A^e -module M_ε . Dans cette configuration, on a l'identification

$$\text{Hom}_{A^e}(B_\bullet(A), M_\varepsilon) = \text{Hom}_A(A \otimes_{A^e} B_\bullet(A), M)$$

Combinée avec l'isomorphisme de complexe de A -module

$$A \otimes_{A^e} B_\bullet(A) \simeq C_\bullet(A)$$

où $C_\bullet(A)$ est le complexe de Hochschild de A [2, 1.1.3], ce calcul donne

$$H^n(A, M) = H^n(\text{Hom}_A(C_\bullet(A), M))$$

pour tout A -module M .

Exemple (H^0 & H^1) : La différentielle $C_1(A) \rightarrow C_0(A)$ est nulle car A est commutative

$$\begin{aligned} A^e &\longrightarrow A \\ s \otimes t &\longmapsto st - ts \end{aligned}$$

Ceci montre que $H^0(A, M) = \text{Hom}_A(A, M) = M$ et $H^1(A, M) = Z^1(\text{Hom}_A(C_\bullet(A), M))$. La différentielle $C_2(A) \rightarrow C_1(A)$ donnée par

$$\begin{aligned} A \otimes A^e &\longrightarrow A^e \\ s \otimes t \otimes r &\longmapsto st \otimes r - s \otimes tr + rs \otimes t \end{aligned}$$

et l'adjonction $\text{Hom}_A(A^e, M) = \text{Hom}_k(A, M)$ permette d'interpréter les 1-cocycles de $\text{Hom}_A(C_\bullet(A), M)$ comme les dérivations de A dans M : $H^1(A, M) = \text{Der}_k(A, M)$.

La première manière d'adapter cette définition à un schéma X sur un corps k est de considérer avec $\times = \times_k$ l'application diagonale

$$\delta : X \rightarrow X \times X$$

pour regarder chaque faisceau \mathcal{F} de \mathcal{O}_X -module comme un faisceau de $\mathcal{O}_{X \times X}$ -module en prenant $\delta_* \mathcal{F}$ mais en notant simplement \mathcal{F} à la place de $\delta_* \mathcal{F}$. On peut alors définir la cohomologie de Hochschild de X à valeur dans un faisceau \mathcal{F} de \mathcal{O}_X -module par

$$H^\bullet(\mathcal{O}_X, \mathcal{F}) = \text{Ext}_{\mathcal{O}_{X \times X}}^\bullet(\mathcal{O}_X, \mathcal{F})$$

Cette définition généralise la cohomologie de Hochschild d'une algèbre, au sens où si $X = \text{Spec } A$ est affine et $\mathcal{F} = M^\sim$ est quasi-cohérent alors

$$H^\bullet(\mathcal{O}_X, \mathcal{F}) = H^\bullet(A, M)$$

Ceci provient du fait que $X \times X = \text{Spec } A^e$, $\delta_* \mathcal{O}_X = A_e^\sim$ et $\delta_* \mathcal{F} = M_e^\sim$, ce qui implique

$$\text{Ext}_{\mathcal{O}_{X \times X}}^\bullet(\mathcal{O}_X, \mathcal{F}) = \text{Ext}_{A^e}^\bullet(A, M)$$

Cette définition arrive avec une suite spectrale de Grothendieck. De fait, le foncteur $\mathcal{H}om_{\mathcal{O}_{X \times X}}(\mathcal{O}_X, -)$ envoie les faisceaux de $\mathcal{O}_{X \times X}$ -module injectifs vers les faisceaux de $\mathcal{O}_{X \times X}$ -module Γ -acyclique. Plus généralement, si \mathcal{O} est un faisceau d'anneau sur un espace topologique, \mathcal{F} un faisceau de \mathcal{O} -module et \mathcal{I} un faisceau injectif de \mathcal{O} -module alors le faisceau $\mathcal{H}om_{\mathcal{O}}(\mathcal{F}, \mathcal{I})$ est flasque : pour toute inclusion d'ouvert $V \subset U$ et tout morphisme de faisceau de $\mathcal{O}|_V$ -module

$$\mathcal{F}|_V \rightarrow \mathcal{I}|_V$$

correspond un morphisme de faisceau de $\mathcal{O}|_U$ -module

$$(\mathcal{F}|_V)_U \rightarrow \mathcal{I}|_U$$

où $(\mathcal{F}|_V)_U$ désigne le faisceau $\mathcal{F}|_V$ étendu par 0 sur U [1, Ch.II, Ex.1.19]. On obtient alors un triangle commutatif

$$\begin{array}{ccc} & & \mathcal{I}|_U \\ & \nearrow & \uparrow \\ 0 \longrightarrow & (\mathcal{F}|_V)_U \longrightarrow & \mathcal{F}|_U \end{array}$$

En ajoutant à cela que les faisceaux flasques sont Γ -acycliques [1, Ch.III, Prop.2.5], on obtient la propriété recherchée. Ainsi, la composition de foncteur

$$\mathcal{H}om_{\mathcal{O}_{X \times X}}(\mathcal{O}_X, -) = \Gamma \circ \mathcal{H}om_{\mathcal{O}_{X \times X}}(\mathcal{O}_X, -)$$

induit pour tout faisceau \mathcal{F} de \mathcal{O}_X -module une suite spectrale de Grothendieck

$$E_2^{pq} = H^p(X \times X, \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})) \Rightarrow \text{Ext}_{\mathcal{O}_{X \times X}}^{p+q}(\mathcal{O}_X, \mathcal{F})$$

Supposons X de type fini et séparé sur k . Les fibres de $\delta_* \mathcal{F}$ sont alors données par

$$(\delta_* \mathcal{F})_{\delta(x)} = \mathcal{F}_x \text{ et } (\delta_* \mathcal{F})_y = 0 \text{ si } y \notin \delta(X)$$

pour tout faisceau \mathcal{F} de \mathcal{O}_X -module. Puisque $X \times X$ est noethérien et $\delta_* \mathcal{O}_X$ cohérent, ceci montre que le faisceau $\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$ est à support dans la diagonale $\delta(X)$:

$$(\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F}))_y = \text{Ext}_{\mathcal{O}_{X \times X, y}}^q((\delta_* \mathcal{O}_X)_y, (\delta_* \mathcal{F})_y)$$

[1, Ch.III, Prop.6.8]. En conséquence, l'unité

$$\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F}) \rightarrow \delta_* \delta^{-1} \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$$

est un isomorphisme. Puisque $\Gamma \circ \delta_* = \Gamma$, la cohomologie du faisceau $\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$ sur $X \times X$ coïncide avec la cohomologie du faisceau $\delta^{-1} \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$ sur X . On préférera donc écrire la suite spectrale de Grothendieck sous la forme

$$H^p(X, \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})) \Rightarrow H^{p+q}(\mathcal{O}_X, \mathcal{F})$$

en regardant $\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$ comme un faisceau sur X via δ^{-1} . Ce dernier hérite d'une structure de \mathcal{O}_X -module étant donnée que le morphisme de faisceau de $\delta^{-1}\mathcal{O}_{X \times X}$ -module

$$\delta^{-1}\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F}) \rightarrow \delta^*\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$$

est un isomorphisme. En effet, sur les fibres, on est ramené au morphisme

$$\mathcal{E}xt_{\mathcal{O}_{X \times X, \delta(x)}}^q(\mathcal{O}_{X, x}, \mathcal{F}_x) \rightarrow \mathcal{O}_{X, x} \otimes_{\mathcal{O}_{X \times X, \delta(x)}} \mathcal{E}xt_{\mathcal{O}_{X \times X, \delta(x)}}^q(\mathcal{O}_{X, x}, \mathcal{F}_x)$$

Pour construire la réciproque, on part de la multiplication

$$\mathcal{O}_{X, x} \times \mathcal{E}xt_{\mathcal{O}_{X \times X, \delta(x)}}^q(\mathcal{O}_{X, x}, \mathcal{F}_x) \rightarrow \mathcal{E}xt_{\mathcal{O}_{X \times X, \delta(x)}}^q(\mathcal{O}_{X, x}, \mathcal{F}_x)$$

induite par la structure de $\mathcal{O}_{X, x}$ -module de \mathcal{F}_x , puis on vérifie la bilinéarité sur $\mathcal{O}_{X \times X, \delta(x)}$ grâce au triangle commutatif donné par la co-unité

$$\begin{array}{ccc} & \xrightarrow{\delta_{\delta(x)}^\flat} & (\delta_*\mathcal{O}_X)_{\delta(x)} \\ \mathcal{O}_{X \times X, \delta(x)} & & \parallel \\ & \xrightarrow{\delta_x^\sharp} & \mathcal{O}_{X, x} \end{array}$$

où $\delta^\flat : \mathcal{O}_{X \times X} \rightarrow \delta_*\mathcal{O}_X$ et $\delta^\sharp : \delta^{-1}\mathcal{O}_{X \times X} \rightarrow \mathcal{O}_X$. Par abus de notation, on parlera donc du faisceau $\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$ sur X au lieu d'écrire $\delta^*\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\delta_*\mathcal{O}_X, \delta_*\mathcal{F})$.

On peut préciser cette suite spectrale pour les schémas lisses. Commençons par rappeler une propriété élémentaire d'algèbre homologique qui nous servira également au paragraphe 5 : si on se donne un complexe de R -module projectif

$$\cdots \longrightarrow P_2 \longrightarrow P_1 \longrightarrow P_0 \longrightarrow 0$$

tel que les homologies sont des R -modules projectifs, alors on a des isomorphismes

$$H^q(\text{Hom}_R(P_\bullet, N)) \simeq \text{Hom}_R(H_q(P_\bullet), N)$$

pour tout R -module N et tout q . Pour le démontrer, on note $Z_q = Z_q(P_\bullet)$ les cycles et $B_q = B_q(P_\bullet)$ les bords de P_\bullet puis on utilise les suites exactes courtes

$$0 \longrightarrow Z_q \longrightarrow P_q \longrightarrow B_q \longrightarrow 0$$

$$0 \longrightarrow B_{q+1} \longrightarrow Z_q \longrightarrow H_q(P_\bullet) \longrightarrow 0$$

et la projectivité de l'homologie de P_\bullet pour obtenir les isomorphismes

$$Z_q \simeq H_q(P_\bullet) \oplus B_{q+1}$$

ce qui permet de prouver par récurrence sur $q \in \mathbb{N}$

$$P_q \simeq B_q \oplus Z_q$$

de sorte que le complexe P_\bullet s'identifie au complexe

$$\begin{array}{ccccccc} \cdots & \longrightarrow & B_q \oplus Z_q & \longrightarrow & B_{q-1} \oplus Z_{q-1} & \longrightarrow & \cdots \\ & & (b, c) & \longmapsto & (0, b) & & \end{array}$$

Dans cette configuration, le complexe $\text{Hom}_R(P_\bullet, N)$ s'identifie au complexe

$$\begin{array}{ccccccc} \cdots & \longrightarrow & \text{Hom}_R(B_q, N) \oplus \text{Hom}_R(Z_q, N) & \longrightarrow & \text{Hom}_R(B_{q+1}, N) \oplus \text{Hom}_R(Z_{q+1}, N) & \twoheadrightarrow & \cdots \\ & & (f, g) & \longmapsto & (g|_{B_{q+1}}, 0) & & \end{array}$$

Les cocycles s'identifient alors à $Hom_R(B_q, N) \oplus Hom_R(H_q(P_\bullet), N)$ car les morphismes $Z_p \rightarrow N$ nuls sur B_{q+1} correspondent aux morphismes $H_q(P_\bullet) \rightarrow N$; tandis que les cobords s'identifient à l'image de $Hom_R(Z_q, N)$ par la restriction sur B_{q+1} , c'est à dire $Hom_R(B_{q+1}, N)$. En effet, la seconde suite exacte courte fournit une flèche $Z_q \rightarrow B_{q+1}$ qui est l'identité sur B_{q+1} de sorte que la restriction sur B_{q+1} est surjective. Ceci prouve l'isomorphisme annoncé. Ce résultat peut être utilisé dans la situation suivante. Soient

$$S \rightarrow R$$

un morphisme d'anneau, M un S -module et N un R -module. Les groupes

$$Ext_S^q(M, N) \text{ et } Tor_q^S(R, M)$$

sont naturellement munis d'une structure de R -module. Si on suppose pour tout q que $Tor_q^S(R, M)$ est projectif sur R alors on a un isomorphisme de R -module

$$Ext_S^q(M, N) \simeq Hom_R(Tor_q^S(R, M), N)$$

Pour le voir, on prend une résolution projective de S -module

$$P_\bullet \rightarrow M$$

Le complexe $R \otimes_S P_\bullet$ est alors projectif sur R et son homologie

$$H_q(R \otimes_S P_\bullet) = Tor_q^S(R, M)$$

est par hypothèse projective sur R . On a alors un isomorphisme

$$H^q(Hom_R(R \otimes_S P_\bullet, N)) \simeq Hom_R(H_q(R \otimes_S P_\bullet), N)$$

et le résultat se déduit alors de l'adjonction

$$Hom_R(R \otimes_S P_\bullet, N) \simeq Hom_S(P_\bullet, N)$$

En particulier, lorsque A est lisse sur k , on peut appliquer ceci au morphisme d'anneau

$$\varepsilon : A^e \rightarrow A$$

car le théorème de HKR fournit des isomorphismes naturels de A -module

$$Tor_q^{A^e}(A, A) \simeq \Omega_A^q$$

de sorte que $Tor_q^{A^e}(A, A)$ est projectif sur A . On obtient un isomorphisme de A -module

$$Ext_{A^e}^q(A, N) \simeq Hom_A(\Omega_A^q, N)$$

Exemple : Si $X = Spec A$ et $\mathcal{F} = N^\sim$ alors $H^n(\mathcal{O}_X, \mathcal{F}) = Ext_{A^e}^n(A, N) \simeq Hom_A(\Omega_A^n, N)$.

Supposons X lisse sur k . Pour tout ouvert affine $U = Spec A$ de X , on dispose d'un isomorphisme de faisceau de \mathcal{O}_U -module

$$\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{O}_X)|_U = Ext_{A^e}^q(A, A)^\sim \simeq (\Omega_U^q)^\vee$$

[1, Ch.III, Prop.6.2 & Ex.6.7]. Ces isomorphismes naturels se recollent pour former un isomorphisme de faisceau de \mathcal{O}_X -module

$$\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{O}_X) \simeq (\Omega_X^q)^\vee \simeq \bigwedge^q \mathcal{T}_X$$

où $\mathcal{T}_X = (\Omega_X^1)^\vee$ est le faisceau tangent de X , le deuxième isomorphisme étant donné par le fait que Ω_X^1 est localement libre. Avant de conclure, énonçons une propriété générale pour un faisceau d'anneau \mathcal{O} arbitraire. Pour tout faisceau \mathcal{A} , \mathcal{B} et \mathcal{G} de \mathcal{O} -module et tout ouvert U , l'application $\Gamma(U, \mathcal{O})$ -bilinéaire

$$\begin{aligned} \Gamma(U, \mathcal{H}om_{\mathcal{O}}(\mathcal{A}, \mathcal{B})) \times \Gamma(U, \mathcal{G}) &\longrightarrow \Gamma(U, \mathcal{H}om_{\mathcal{O}}(\mathcal{A}, \mathcal{B} \otimes \mathcal{G})) \\ (\theta, s) &\longmapsto (- \otimes s) \circ \theta \end{aligned}$$

induit un morphisme de faisceau de \mathcal{O} -module

$$\mathcal{H}om_{\mathcal{O}}(\mathcal{A}, \mathcal{B}) \otimes \mathcal{G} \rightarrow \mathcal{H}om_{\mathcal{O}}(\mathcal{A}, \mathcal{B} \otimes \mathcal{G})$$

Lorsque \mathcal{A} est projectif sur \mathcal{O} et \mathcal{G} est localement libre, c'est un isomorphisme :

$$\begin{aligned} \mathcal{H}om_{\mathcal{O}_x}(\mathcal{A}_x, \mathcal{B}_x) \otimes_{\mathcal{O}_x} \mathcal{G}_x &\simeq \bigoplus_{i \in I} \mathcal{H}om_{\mathcal{O}_x}(\mathcal{A}_x, \mathcal{B}_x) \\ &\simeq \mathcal{H}om_{\mathcal{O}_x}(\mathcal{A}_x, \bigoplus_{i \in I} \mathcal{B}_x) \simeq \mathcal{H}om_{\mathcal{O}_x}(\mathcal{A}_x, \mathcal{B}_x \otimes_{\mathcal{O}_x} \mathcal{G}_x) \end{aligned}$$

avec $\mathcal{G}_x \simeq \bigoplus_{i \in I} \mathcal{O}_x$. Puisque les faisceaux localement libres sont plats, on obtient en particulier pour tout faisceau localement libre \mathcal{F} de \mathcal{O}_X -module des isomorphismes

$$\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{O}_X) \otimes \mathcal{F} \simeq \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$$

On peut aussi appliquer cette propriété au faisceau canonique $\omega_X = \Omega_X^d$ de X

$$(\bigwedge^q \mathcal{T}_X) \otimes \omega_X \simeq \mathcal{H}om_{\mathcal{O}_X}(\Omega_X^q, \mathcal{O}_X) \otimes \omega_X \simeq \mathcal{H}om_{\mathcal{O}_X}(\Omega_X^q, \omega_X)$$

Et la multiplication des formes différentielles

$$\Omega_X^q \otimes \Omega_X^{d-q} \rightarrow \omega_X$$

induit par adjonction un isomorphisme

$$\Omega_X^{d-q} \simeq \mathcal{H}om_{\mathcal{O}_X}(\Omega_X^q, \omega_X)$$

ce qui peut se vérifier en utilisant la liberté sur $\mathcal{O}_{X,x}$ des fibres $\Omega_{X,x}^q = \Omega_{\mathcal{O}_{X,x}}^q$. Le faisceau $\omega_X \otimes \mathcal{F}$ étant localement libre, on peut résumer ce qui précède par l'isomorphisme

$$\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \omega_X \otimes \mathcal{F}) \simeq \Omega_X^{d-q} \otimes \mathcal{F}$$

où $d = \dim X$. Ainsi la suite spectrale donnée par la cohomologie de Hochschild de X à valeur dans $\omega_X \otimes \mathcal{F}$ s'écrit

$$H^p(X, \Omega_X^{d-q} \otimes \mathcal{F}) \Rightarrow H^{p+q}(\mathcal{O}_X, \omega_X \otimes \mathcal{F})$$

On verra au paragraphe 2 que si X est quasi-projectif sur un corps de caractéristique nulle alors cette suite spectrale dégénère et induit une décomposition

$$H^n(\mathcal{O}_X, \omega_X \otimes \mathcal{F}) \simeq \bigoplus_{p+q=n} H^p(X, \Omega_X^{d-q} \otimes \mathcal{F})$$

2 Définition par l'hyper-ext

La deuxième définition de la cohomologie de Hochschild d'un schéma X sur un corps s'inspire d'avantage du calcul

$$H^n(A, M) = H^n(\text{Hom}_A(C_\bullet(A), M))$$

Soit \mathcal{C}_\bullet le complexe de faisceau de \mathcal{O}_X -module associé au préfaisceau

$$U \mapsto C_\bullet(\Gamma(U, \mathcal{O}_X))$$

La cohomologie de Hochschild d'un faisceau \mathcal{F} de \mathcal{O}_X -module sur X peut être définie par

$$HH^\bullet(X, \mathcal{F}) = \mathbb{E}xt_{\mathcal{O}_X}^\bullet(\mathcal{C}_\bullet, \mathcal{F})$$

c'est à dire pour toute résolution injective $\mathcal{F} \rightarrow \mathcal{I}^\bullet$ de faisceau de \mathcal{O}_X -module

$$HH^n(X, \mathcal{F}) = H^n(\text{Hom}_{\mathcal{O}_X}(\mathcal{C}_\bullet, \mathcal{I}^\bullet))$$

L'objectif principal de l'article de Swan est de démontrer le théorème suivant :

Théorème 2.1 : Soit X un schéma quasi-projectif sur un corps. Il existe un isomorphisme de δ -foncteur en \mathcal{F}

$$H^\bullet(\mathcal{O}_X, \mathcal{F}) \simeq HH^\bullet(X, \mathcal{F})$$

On peut donner par exemple l'application suivante.

Corollaire 2.2 : Soit X un schéma projectif sur un corps. Pour tout faisceau cohérent \mathcal{F} de \mathcal{O}_X -module, $HH^n(X, \mathcal{F})$ est un espace vectoriel de dimension finie.

Preuve : Les faisceaux $\mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})$ sont cohérents donc les $H^p(X, \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F}))$ sont des espaces vectoriels de dimension finie [1, Ch.III, Th.5.5]. De plus, X est un espace topologique noethérien de dimension N finie, donc $H^p(X, \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})) = 0$ pour $p > N$ [1, Ch.III, Th.2.7] et ainsi $H^n(\mathcal{O}_X, \mathcal{F})$ est une somme directe finie d'espaces vectoriels de dimension finie. ■

Dans l'article original [3], Swan remarque qu'on peut également définir la cohomologie cyclique de X en considérant le complexe de faisceau \mathcal{D}_\bullet associé au préfaisceau

$$U \mapsto D_\bullet(\Gamma(U, \mathcal{O}_X))$$

où $D_\bullet(A)$ désigne le complexe total du double complexe de Connes d'une algèbre A sur un corps [2, 2.1.7] puis en posant

$$HC^\bullet(X, \mathcal{F}) = \mathbb{E}xt_{\mathcal{O}_X}^\bullet(\mathcal{D}_\bullet, \mathcal{F})$$

La suite exacte courte usuelle [2, 2.2.2]

$$0 \longrightarrow C_\bullet(A) \longrightarrow D_\bullet(A) \longrightarrow D_\bullet(A)[-2] \longrightarrow 0$$

induit une suite exacte courte de complexe de faisceau de \mathcal{O}_X -module

$$0 \longrightarrow \mathcal{C}_\bullet \longrightarrow \mathcal{D}_\bullet \longrightarrow \mathcal{D}_\bullet[-2] \longrightarrow 0$$

et par suite une longue suite exacte cohomologique de Connes

$$\cdots \longrightarrow HC^{n-2}(X, \mathcal{F}) \longrightarrow HC^n(X, \mathcal{F}) \longrightarrow HH^n(X, \mathcal{F}) \longrightarrow HC^{n-1}(X, \mathcal{F}) \longrightarrow \cdots$$

qui permet de généraliser par récurrence le corollaire 2.2 à la cohomologie cyclique de X .

La principale difficulté autour de cette définition réside dans le fait que \mathcal{C}_\bullet n'est pas quasi-cohérent en général. Néanmoins, tout comme la précédente, elle arrive avec une suite spectrale.

Lemme 2.3 : Soit \mathcal{O} un faisceau d'anneau sur un espace topologique. Pour tout complexe de faisceau \mathcal{A}_\bullet de \mathcal{O} -module borné en bas et tout faisceau \mathcal{G} de \mathcal{O} -module, il existe une suite spectrale

$$E_2^{pq} = Ext_{\mathcal{O}}^p(H_q(\mathcal{A}_\bullet), \mathcal{G}) \Rightarrow Ext_{\mathcal{O}}^{p+q}(\mathcal{A}_\bullet, \mathcal{G})$$

Preuve : Soit $\mathcal{G} \rightarrow \mathcal{I}^\bullet$ une résolution injective de faisceau de \mathcal{O} -module. Puisque \mathcal{A}_\bullet est borné en bas, la suite spectrale donnée en filtrant selon les colonnes le double complexe

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & Hom_{\mathcal{O}}(\mathcal{A}_{q+1}, \mathcal{I}^p) & \longrightarrow & Hom_{\mathcal{O}}(\mathcal{A}_{q+1}, \mathcal{I}^{p+1}) & \longrightarrow & \cdots \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & Hom_{\mathcal{O}}(\mathcal{A}_q, \mathcal{I}^p) & \longrightarrow & Hom_{\mathcal{O}}(\mathcal{A}_q, \mathcal{I}^{p+1}) & \longrightarrow & \cdots \\ & & \uparrow & & \uparrow & & \\ & & \vdots & & \vdots & & \end{array}$$

converge vers la cohomologie totale $Ext_{\mathcal{O}}^{p+q}(\mathcal{A}_\bullet, \mathcal{G})$. On peut calculer la deuxième page :

$$\begin{aligned} E_0^{pq} &= Hom_{\mathcal{O}}(\mathcal{A}_q, \mathcal{I}^p) \\ E_1^{pq} &= H^q(Hom_{\mathcal{O}}(\mathcal{A}_\bullet, \mathcal{I}^p)) = Hom_{\mathcal{O}}(H_q(\mathcal{A}_\bullet), \mathcal{I}^p) \\ E_2^{pq} &= H_h^p(H_v^q(Hom_{\mathcal{O}}(\mathcal{A}_\bullet, \mathcal{I}^\bullet))) = Ext_{\mathcal{O}}^p(H_q(\mathcal{A}_\bullet), \mathcal{G}) \blacksquare \end{aligned}$$

Lorsque l'on prend $\mathcal{O} = \mathcal{O}_X$, $\mathcal{A}_\bullet = \mathcal{C}_\bullet$ et que l'on pose $\mathcal{H}_q = H_q(\mathcal{C}_\bullet)$, on obtient

$$Ext_{\mathcal{O}_X}^p(\mathcal{H}_q, \mathcal{F}) \Rightarrow HH^{p+q}(X, \mathcal{F})$$

Pour comprendre le faisceau \mathcal{H}_q , on a besoin d'une propriété importante concernant l'homologie de Hochschild des algèbres commutatives. Pour tout morphisme plat d'algèbre

$$A \rightarrow B$$

le morphisme d'algèbre induit

$$A^e \rightarrow B^e$$

est également plat et on dispose alors d'une formule de changement de base

$$Tor_n^{A^e}(M, N) \simeq Tor_n^{B^e}(B^e \otimes_{A^e} M, N)$$

pour tout A^e -module M et tout B^e -module N . Puisque $B \otimes_A A \simeq B$, on obtient

$$\begin{aligned} B \otimes_A Tor_n^{A^e}(M, A) &\simeq Tor_n^{A^e}(M, B) \simeq Tor_n^{B^e}(B^e \otimes_{A^e} M, B) \\ &\Rightarrow B \otimes_A H_n(A, M) \simeq H_n(B, B^e \otimes_{A^e} M) \end{aligned}$$

En particulier, l'homologie de Hochschild commute avec la localisation :

$$\begin{aligned} (S^{-1}A)^e \otimes_{A^e} A &\simeq (S^{-1}A) \otimes_A A \otimes_A (S^{-1}A) \simeq S^{-1}A \\ \Rightarrow (S^{-1}A) \otimes_A HH_n(A) &\simeq HH_n(S^{-1}A) \end{aligned}$$

où $HH_\bullet(A) = H_\bullet(A, A)$. On rencontrera les morphismes plats dans d'autres situations. Par exemple, une immersion ouverte de schéma affine

$$Spec B \hookrightarrow Spec A$$

induit sur les sections globales un morphisme plat

$$\varphi : A \rightarrow B$$

Pour s'en convaincre, il suffit de prouver que tout monomorphisme de A -module

$$0 \longrightarrow M \xrightarrow{f} N$$

est envoyé par le foncteur $B \otimes_A -$ vers un monomorphisme

$$0 \longrightarrow B \otimes_A M \xrightarrow{1 \otimes f} B \otimes_A N$$

Par hypothèse, φ induit pour tout $\mathfrak{q} \in Spec B$ un isomorphisme d'anneau

$$A_{\mathfrak{p}} \simeq B_{\mathfrak{q}}$$

où $\mathfrak{p} = \varphi^{-1}(\mathfrak{q})$. On a alors un carré commutatif

$$\begin{array}{ccc} B_{\mathfrak{q}} \otimes_B B \otimes_A M & \xrightarrow{1 \otimes 1 \otimes f} & B_{\mathfrak{q}} \otimes_B B \otimes_A N \\ \Big\downarrow & & \Big\downarrow \\ A_{\mathfrak{p}} \otimes_A M & \xrightarrow{1 \otimes f} & A_{\mathfrak{p}} \otimes_A N \end{array}$$

Puisque $A_{\mathfrak{p}}$ est plat sur A , le morphisme du bas est un monomorphisme. Par conséquent,

$$\begin{aligned} \forall \mathfrak{q} \in Spec B, B_{\mathfrak{q}} \otimes_B \ker(1 \otimes f) &= 0 \\ \Rightarrow \ker(1 \otimes f) &= 0 \end{aligned}$$

d'où le résultat. Donnons à présent un lemme important qui nous accompagnera jusqu'à la démonstration du théorème 2.1.

Lemme 2.4 : Soit X un schéma de type fini sur un corps.

- (1) \mathcal{H}_q est un faisceau cohérent pour tout q .
- (2) $\Gamma(U, \mathcal{H}_q) = HH_q(\Gamma(U, \mathcal{O}_X))$ pour tout ouvert affine U de X .
- (3) Si X est lisse alors $\mathcal{H}_q \simeq \Omega_X^q$.
- (4) Si $X = Spec A$ est affine alors on a un quasi-isomorphisme

$$\delta^* \mathcal{B}_\bullet \xrightarrow{q.is.} \mathcal{C}_\bullet$$

où \mathcal{B}_\bullet est le complexe de faisceau sur $X \times X$ associé au complexe de A^e -module $B_\bullet(A)$.

Preuve : La faisceautification est un foncteur exact, donc \mathcal{H}_q est associé au préfaisceau

$$U \mapsto HH_q(\Gamma(U, \mathcal{O}_X))$$

Or, l'homologie de Hochschild commute avec la localisation. Donc si U est un ouvert affine de X alors le préfaisceau

$$U \supset V \mapsto HH_q(\Gamma(V, \mathcal{O}_X))$$

correspond à la localisation du $\Gamma(U, \mathcal{O}_X)$ -module $HH_q(\Gamma(U, \mathcal{O}_X))$. En particulier,

$$\Gamma(U, \mathcal{H}_q) = HH_q(\Gamma(U, \mathcal{O}_X))$$

Puisque X est de type fini sur un corps, $HH_q(\Gamma(U, \mathcal{O}_X))$ est de type fini sur $\Gamma(U, \mathcal{O}_X)$: si A est une algèbre de type fini sur un corps k alors on a un morphisme d'algèbre surjectif

$$k[x_1, \dots, x_n] \rightarrow A$$

qui induit un morphisme d'anneau surjectif

$$A[x_1, \dots, x_n] \rightarrow A \otimes A$$

ce qui montre que A^e est Noethérien. Ainsi, A admet une résolution projective de A^e -module de type fini et par conséquent $Tor_q^{A^e}(A, A)$ est un A^e -module de type fini. Puisque $HH_q(A) = Tor_q^{A^e}(A, A)$ est un A -bimodule symétrique, $HH_q(A)$ est un A -module de type fini. Tout ceci prouve (1) et (2). Le point (3) découle directement du théorème de HKR. Supposons $X = Spec A$ affine. Sur les ouverts principaux de $X \times X$, on a une composition

$$(A^e)_f \otimes_{A^e} B_\bullet(A) \rightarrow A_{\varepsilon(f)} \otimes_{A^e} B_\bullet(A) \simeq A_{\varepsilon(f)} \otimes_A C_\bullet(A) \rightarrow C_\bullet(A_{\varepsilon(f)})$$

qui définit un morphisme de faisceau de $\mathcal{O}_{X \times X}$ -module

$$\mathcal{B}_\bullet \rightarrow \delta_* \mathcal{C}_\bullet$$

On obtient par adjonction un morphisme de complexe de \mathcal{O}_X -module

$$\delta^* \mathcal{B}_\bullet \rightarrow \mathcal{C}_\bullet$$

C'est un quasi-isomorphisme. Pour le voir, il suffit de calculer l'homologie sur les fibres en chaque idéal premier $\mathfrak{p} \in X$:

$$(\delta^* \mathcal{B}_\bullet)_\mathfrak{p} = A_\mathfrak{p} \otimes_{(A^e)_\mathfrak{q}} ((A^e)_\mathfrak{q} \otimes_{A^e} B_\bullet(A)) = A_\mathfrak{p} \otimes_{A^e} B_\bullet(A) = A_\mathfrak{p} \otimes_A C_\bullet(A)$$

où $\mathfrak{q} \in X \times X$ désigne l'image réciproque de \mathfrak{p} par la multiplication. Le morphisme de complexe de $A_\mathfrak{p}$ -module

$$A_\mathfrak{p} \otimes_A C_\bullet(A) \rightarrow C_\bullet(A_\mathfrak{p})$$

est un quasi-isomorphisme :

$$H_q(A_\mathfrak{p} \otimes_A C_\bullet(A)) \simeq A_\mathfrak{p} \otimes_A H_q(C_\bullet(A)) = HH_q(A)_\mathfrak{p} \simeq HH_q(A_\mathfrak{p}) = H_q(C_\bullet(A_\mathfrak{p})) \blacksquare$$

Ce résultat permet de réécrire la suite spectrale précédente lorsque X est lisse :

$$Ext_{\mathcal{O}_X}^p(\Omega_X^q, \mathcal{F}) \Rightarrow HH^{p+q}(X, \mathcal{F})$$

On peut calculer $Ext_{\mathcal{O}_X}^p(\Omega_X^q, \mathcal{F})$ en utilisant la suite spectrale de Grothendieck

$$E_2^{ij} = H^i(X, Ext_{\mathcal{O}_X}^j(\Omega_X^q, \mathcal{F})) \Rightarrow Ext_{\mathcal{O}_X}^{i+j}(\Omega_X^q, \mathcal{F})$$

donnée par la composition de foncteur

$$Hom_{\mathcal{O}_X}(\Omega_X^q, -) = \Gamma \circ \mathcal{H}om_{\mathcal{O}_X}(\Omega_X^q, -)$$

L'idée est que Ω_X^q est localement libre de rang fini ce qui implique d'une part que $\mathcal{E}xt_{\mathcal{O}_X}^j(\Omega_X^q, \mathcal{F}) = 0$ pour tout $j > 0$ et par conséquent la page E_2 de la suite de Grothendieck ne comporte qu'une seule ligne ce qui induit un isomorphisme

$$\mathcal{E}xt_{\mathcal{O}_X}^p(\Omega_X^q, \mathcal{F}) \simeq H^p(X, \mathcal{H}om_{\mathcal{O}_X}(\Omega_X^q, \mathcal{F}))$$

et d'autre part que le morphisme de faisceau de \mathcal{O}_X -module

$$(\Omega_X^q)^\vee \otimes \mathcal{F} \rightarrow \mathcal{H}om_{\mathcal{O}_X}(\Omega_X^q, \mathcal{F})$$

introduit au premier paragraphe est un isomorphisme. En particulier,

$$\mathcal{H}om_{\mathcal{O}_X}(\Omega_X^q, \omega_X \otimes \mathcal{F}) \simeq \Omega_X^{d-q} \otimes \mathcal{F}$$

Ainsi la suite spectrale donnée par la cohomologie de Hochschild de $\omega_X \otimes \mathcal{F}$ sur X s'écrit

$$H^p(X, \Omega_X^{d-q} \otimes \mathcal{F}) \Rightarrow HH^{p+q}(X, \omega_X \otimes \mathcal{F})$$

Ceci nous amène au deuxième théorème principal de l'article de Swan.

Théorème 2.5 : Soit X un schéma quasi-projectif sur un corps. Si \mathcal{H}_q est localement libre pour tout q alors les suites spectrales

$$H^p(X, \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})) \Rightarrow H^{p+q}(\mathcal{O}_X, \mathcal{F})$$

$$\mathcal{E}xt_{\mathcal{O}_X}^p(\mathcal{H}_q, \mathcal{F}) \Rightarrow HH^{p+q}(X, \mathcal{F})$$

sont isomorphes. En particulier, si X est lisse alors les suites spectrales

$$H^p(X, \Omega_X^{d-q} \otimes \mathcal{F}) \Rightarrow H^{p+q}(\mathcal{O}_X, \omega_X \otimes \mathcal{F})$$

$$H^p(X, \Omega_X^{d-q} \otimes \mathcal{F}) \Rightarrow HH^{p+q}(X, \omega_X \otimes \mathcal{F})$$

sont isomorphes.

Ce théorème nous permet de démontrer la décomposition de la cohomologie de Hochschild annoncée à la fin du paragraphe 1.

Corollaire 2.6 : Soit X un schéma lisse et quasi-projectif sur un corps de caractéristique nulle. La suite spectrale

$$H^p(X, \Omega_X^{d-q} \otimes \mathcal{F}) \Rightarrow H^{p+q}(\mathcal{O}_X, \omega_X \otimes \mathcal{F})$$

dégénère et induit un isomorphisme

$$H^n(\mathcal{O}_X, \omega_X \otimes \mathcal{F}) \simeq \bigoplus_{p+q=n} H^p(X, \Omega_X^{d-q} \otimes \mathcal{F})$$

Preuve : Le théorème 2.5 nous ramène à la suite spectrale

$$H^p(X, \Omega_X^{d-q} \otimes \mathcal{F}) \Rightarrow HH^{p+q}(X, \omega_X \otimes \mathcal{F})$$

L'idée est d'utiliser la λ -décomposition de l'homologie de Hochschild d'une algèbre commutative A sur un anneaux contenant \mathbb{Q} [2, 4.5.10] :

$$C_\bullet(A) = \bigoplus_{i \geq 0} C_\bullet^{(i)}(A)$$

où $C_\bullet^{(i)}(A)$ est un sous-complexe de $C_\bullet(A)$ dont l'homologie $HH_n^{(i)}(A)$ satisfait

$$HH_0(A) = HH_0^{(0)}(A)$$

$$HH_n(A) = \bigoplus_{0 \leq i \leq n} HH_n^{(i)}(A), \quad n \geq 1$$

Lorsque A est lisse, on a $HH_n^{(i)}(A) = 0$ pour tout $i \neq n$ [2, 3.4.4 & 4.5.12]. Ceci induit une décomposition

$$\mathcal{C}_\bullet = \bigoplus_{i \geq 0} \mathcal{C}_\bullet^{(i)}$$

de sorte que la suite spectrale

$$H^p(X, \Omega_X^{d-q} \otimes \mathcal{F}) \Rightarrow HH^{p+q}(X, \omega_X \otimes \mathcal{F})$$

se décompose comme somme directe des suites spectrales associées aux doubles complexes $Hom_{\mathcal{O}_X}(\mathcal{C}_\bullet^{(i)}, \mathcal{I}^\bullet)$ où $\omega_X \otimes \mathcal{F} \rightarrow \mathcal{I}^\bullet$ est une résolution injective. La deuxième page

$$E_2^{pq} = Ext_{\mathcal{O}_X}^p(H_q(\mathcal{C}_\bullet^{(i)}), \omega_X \otimes \mathcal{F})$$

consiste alors en une seule colonne $q = i$, ce qui montre la dégénérescence à la deuxième page et donne ainsi le résultat. ■

Exemple : Si A est un anneau Noethérien alors tous les faisceaux quasi-cohérents sur $X = Spec A$ sont Γ -acycliques [1, Ch.III, Th.3.5]. Ainsi, si A est une algèbre lisse sur un corps de caractéristique nulle et si \mathcal{F} est quasi-cohérent sur X alors le corollaire 2.6 donne

$$H^n(\mathcal{O}_X, \omega_X \otimes \mathcal{F}) \simeq \Gamma(X, \Omega_X^{d-n} \otimes \mathcal{F})$$

Or, le théorème de HKR nous avait permis de démontrer au paragraphe 1 l'identité

$$H^n(\mathcal{O}_X, \omega_X \otimes \mathcal{F}) \simeq Hom_A(\Omega_A^n, \Omega_A^d \otimes_A M)$$

où $\mathcal{F} = M^\sim$. L'isomorphisme de A -module

$$Hom_A(\Omega_A^n, \Omega_A^d \otimes_A M) \simeq \Omega_A^{d-n} \otimes_A M$$

nous permet alors d'interpréter le corollaire 2.6 comme une généralisation du théorème de HKR aux schémas lisses et quasi-projectifs sur un corps de caractéristique nulle.

3 Définition de Gerstenhaber-Schack

Introduisons à présent la troisième définition de la cohomologie de Hochschild d'un schéma X sur un corps. Dans ce paragraphe, tous les préfaisceaux sont définis sur la sous-catégorie $\mathbb{A} \subset Top(X)$ des ouverts affines de X . Pour distinguer les faisceaux des préfaisceaux, on notera \mathcal{O} le faisceau d'anneau \mathcal{O}_X vu comme un préfaisceau d'anneau et

$$\mathcal{D}_\bullet : U \mapsto C_\bullet(\Gamma(U, \mathcal{O}_X))$$

le complexe de préfaisceau de \mathcal{O} -module auquel \mathcal{C}_\bullet est associé.

Considérons le préfaisceau d'anneau

$$\mathcal{O}^e : U \mapsto \Gamma(U, \mathcal{O}_X) \otimes \Gamma(U, \mathcal{O}_X)$$

La multiplication induit un morphisme de préfaisceau d'anneau

$$\mathcal{O}^e \rightarrow \mathcal{O}$$

qui permet de regarder tous les faisceaux de \mathcal{O}_X -module comme des préfaisceaux de \mathcal{O}^e -module. Gerstenhaber et Schack définissent la cohomologie de Hochschild de X à valeur dans un faisceau \mathcal{F} de \mathcal{O}_X -module par

$$Ext_{\mathcal{O}^e}^\bullet(\mathcal{O}, \mathcal{F})$$

Naïvement, on voudrait considérer le complexe de préfaisceau de \mathcal{O}^e -module

$$\mathcal{B}_\bullet : U \mapsto B_\bullet(\Gamma(U, \mathcal{O}_X))$$

comme une résolution de préfaisceau de \mathcal{O}^e -module

$$\mathcal{B}_\bullet \rightarrow \mathcal{O}$$

et utiliser l'identification suivante

$$Hom_{\mathcal{O}_X}(\mathcal{C}_\bullet, \mathcal{F}) = Hom_{\mathcal{O}}(\mathcal{D}_\bullet, \mathcal{F}) = Hom_{\mathcal{O}}(\mathcal{O} \otimes_{\mathcal{O}^e} \mathcal{B}_\bullet, \mathcal{F}) = Hom_{\mathcal{O}^e}(\mathcal{B}_\bullet, \mathcal{F})$$

Le problème est que \mathcal{B}_\bullet n'est en général pas projectif sur \mathcal{O}^e . Par contre, chaque $\mathcal{B}_\bullet(U)$ est projectif sur $\mathcal{O}^e(U)$, et cette propriété va nous permettre de construire une résolution projective de préfaisceau de \mathcal{O}^e -module convenable et de démontrer le théorème suivant.

Théorème 3.1 : $Ext_{\mathcal{O}^e}^\bullet(\mathcal{O}, \mathcal{F}) \simeq HH^\bullet(X, \mathcal{F})$.

Le but est de construire une résolution projective de préfaisceau de \mathcal{O}^e -module à partir de \mathcal{B}_\bullet , mais on peut la définir dans un cadre plus général. Soient \mathcal{A} un préfaisceau d'anneau et $\mathcal{A} - mod$ la catégorie des préfaisceaux de \mathcal{A} -module. On dispose d'une adjonction

$$\begin{aligned} R : \mathcal{A} - mod &\longrightarrow \prod_{U \in \mathbb{A}} \mathcal{A}(U) - mod \\ \mathcal{M} &\longmapsto (\mathcal{M}(U))_{U \in \mathbb{A}} \\ L : \prod_{U \in \mathbb{A}} \mathcal{A}(U) - mod &\longrightarrow \mathcal{A} - mod \\ M &\longmapsto (U \mapsto \bigoplus_{V \supset U} \mathcal{A}(U) \otimes_{\mathcal{A}(V)} M_V) \end{aligned}$$

dont on peut expliciter l'unité η et la co-unité ε :

$$(\eta_M)_U : M_U \longrightarrow \bigoplus_{V \supset U} \mathcal{A}(U) \otimes_{\mathcal{A}(V)} M_V$$

$$m \longmapsto \begin{cases} 1 \otimes m & \text{si } V = U \\ 0 & \text{si } V \neq U \end{cases}$$

$$\varepsilon_{\mathcal{M}}(U) : \bigoplus_{V \supset U} \mathcal{A}(U) \otimes_{\mathcal{A}(V)} \mathcal{M}(V) \longrightarrow \mathcal{M}(U)$$

$$(a_V \otimes m_V)_{V \supset U} \longmapsto \sum_{V \supset U} a_V \cdot m_V|_U$$

On note $P = LR$ et on définit un foncteur

$$Q : \mathcal{A} - \text{mod} \longrightarrow \mathcal{A} - \text{mod}$$

$$\mathcal{M} \longmapsto \ker(\varepsilon_{\mathcal{M}})$$

Ceci nous fournit pour tout préfaisceau \mathcal{M} de \mathcal{A} -module et tout ouvert $U \in \mathbb{A}$ une suite exacte scindée de $\mathcal{A}(U)$ -module

$$0 \longrightarrow Q\mathcal{M}(U) \xrightarrow{\subset} P\mathcal{M}(U) \xrightarrow{\varepsilon_{\mathcal{M}}(U)} \mathcal{M}(U) \longrightarrow 0$$

\xleftarrow{s}

où $s : m \mapsto 1 \otimes m \in \mathcal{A}(U) \otimes_{\mathcal{A}(V)} \mathcal{M}(U) \subset P\mathcal{M}(U)$. En particulier, pour tout $n \in \mathbb{N}$, on pose $P_n\mathcal{M} = PQ^n\mathcal{M}$ et on a une suite exacte scindée de $\mathcal{A}(U)$ -module

$$0 \longrightarrow Q^{n+1}\mathcal{M}(U) \longrightarrow P_n\mathcal{M}(U) \xrightarrow{\quad} Q^n\mathcal{M}(U) \longrightarrow 0$$

$\xleftarrow{\quad}$

On a ainsi construit une résolution de préfaisceau de \mathcal{A} -module

$$\cdots \longrightarrow P_2\mathcal{M} \longrightarrow P_1\mathcal{M} \longrightarrow P_0\mathcal{M} \longrightarrow \mathcal{M} \longrightarrow 0$$

Supposons que $\mathcal{M}(U)$ est projectif sur $\mathcal{A}(U)$ pour tout $U \in \mathbb{A}$. Dans ce cas l'objet $R\mathcal{M} = (\mathcal{M}(U))_{U \in \mathbb{A}}$ est projectif dans $\prod_{U \in \mathbb{A}} \mathcal{A}(U) - \text{mod}$. Puisque R est exact, L préserve les projectifs et donc $P\mathcal{M}$ est projectif sur \mathcal{A} . Les suites exactes scindées précédentes nous permettent de démontrer par récurrence que chaque $Q^n\mathcal{M}(U)$ est projectif sur $\mathcal{A}(U)$, de sorte que $P_n\mathcal{M}$ est également projectif sur \mathcal{A} . On a ainsi construit une résolution projective de préfaisceau de \mathcal{A} -module

$$P_{\bullet}\mathcal{M} \rightarrow \mathcal{M}$$

Au besoin, on notera plutôt $P_n^{\mathcal{A}}\mathcal{M}$ si l'on doit préciser le préfaisceau d'anneau \mathcal{A} . Avant de démontrer le théorème 3.1, on a besoin du lemme suivant.

Lemme 3.2 : Si $\mathcal{A} \rightarrow \mathcal{B}$ est un morphisme de préfaisceau d'anneau alors pour tout $n \in \mathbb{N}$

$$\mathcal{B} \otimes_{\mathcal{A}} P_n^{\mathcal{A}}\mathcal{M} \simeq P_n^{\mathcal{B}}(\mathcal{B} \otimes_{\mathcal{A}} \mathcal{M})$$

Preuve : Pour tout $U \in \mathbb{A}$, on a

$$(\mathcal{B} \otimes_{\mathcal{A}} P^{\mathcal{A}}\mathcal{M})(U) = \mathcal{B}(U) \otimes_{\mathcal{A}(U)} \left(\bigoplus_{V \supset U} \mathcal{A}(U) \otimes_{\mathcal{A}(V)} \mathcal{M}(V) \right)$$

$$\simeq \bigoplus_{V \supset U} \mathcal{B}(U) \otimes_{\mathcal{A}(U)} \mathcal{A}(U) \otimes_{\mathcal{A}(V)} \mathcal{M}(V)$$

$$\simeq \bigoplus_{V \supset U} \mathcal{B}(U) \otimes_{\mathcal{B}(V)} \mathcal{B}(V) \otimes_{\mathcal{A}(V)} \mathcal{M}(V) = P^{\mathcal{B}}(\mathcal{B} \otimes_{\mathcal{A}} \mathcal{M})(U)$$

Puis on conclue par récurrence en utilisant le diagramme à lignes exactes

$$\begin{array}{ccccccc}
0 & \longrightarrow & \mathcal{B} \otimes_{\mathcal{A}} Q^{\mathcal{A}} \mathcal{M} & \longrightarrow & \mathcal{B} \otimes_{\mathcal{A}} P^{\mathcal{A}} \mathcal{M} & \longrightarrow & \mathcal{B} \otimes_{\mathcal{A}} \mathcal{M} \longrightarrow 0 \\
& & \downarrow & & \parallel & & \downarrow = \\
0 & \longrightarrow & Q^{\mathcal{B}}(\mathcal{B} \otimes_{\mathcal{A}} \mathcal{M}) & \longrightarrow & P^{\mathcal{B}}(\mathcal{B} \otimes_{\mathcal{A}} \mathcal{M}) & \longrightarrow & \mathcal{B} \otimes_{\mathcal{A}} \mathcal{M} \longrightarrow 0
\end{array}$$

et le lemme des cinq. L'exactitude de la première ligne est donnée par la présence d'une section lorsque l'on évalue en chaque ouvert $U \in \mathbb{A}$. ■

Preuve du théorème 3.1 : On considère la résolution de préfaisceau de \mathcal{O}^e -module

$$\mathcal{B}_{\bullet} \rightarrow \mathcal{O}$$

et les résolutions projectives de préfaisceau de \mathcal{O}^e -module

$$P_{\bullet} \mathcal{B}_p \rightarrow \mathcal{B}_p$$

Si on filtre selon les colonnes le double complexe

$$\begin{array}{ccccccc}
& & \vdots & & \vdots & & \\
& & \downarrow & & \downarrow & & \\
\cdots & \longrightarrow & P_q \mathcal{B}_p & \longrightarrow & P_q \mathcal{B}_{p-1} & \longrightarrow & \cdots \\
& & \downarrow & & \downarrow & & \\
\cdots & \longrightarrow & P_{q-1} \mathcal{B}_p & \longrightarrow & P_{q-1} \mathcal{B}_{p-1} & \longrightarrow & \cdots \\
& & \downarrow & & \downarrow & & \\
& & \vdots & & \vdots & &
\end{array}$$

on obtient la suite spectrale convergente suivante :

$$E_{pq}^1 = \begin{cases} \mathcal{B}_p & \text{si } q = 0 \\ 0 & \text{si } q \neq 0 \end{cases}$$

$$E_{pq}^2 = \begin{cases} \mathcal{O} & \text{si } (p, q) = (0, 0) \\ 0 & \text{si } (p, q) \neq (0, 0) \end{cases}$$

Le complexe total fournit ainsi une résolution projective de préfaisceau de \mathcal{O}^e -module

$$P_{\bullet} \mathcal{B}_{\bullet} \rightarrow \mathcal{O}$$

ce qui donne $Ext_{\mathcal{O}^e}^n(\mathcal{O}, \mathcal{F}) = H^n(Hom_{\mathcal{O}^e}(P_{\bullet} \mathcal{B}_{\bullet}, \mathcal{F}))$. Soit $\mathcal{F} \rightarrow \mathcal{I}^{\bullet}$ une résolution injective de faisceau de \mathcal{O}_X -module. Si on regarde \mathcal{F} comme un complexe concentré en 0, on a alors un quasi-isomorphisme

$$Hom_{\mathcal{O}^e}(P_{\bullet} \mathcal{B}_{\bullet}, \mathcal{F}) \xrightarrow{q.is.} Hom_{\mathcal{O}^e}(P_{\bullet} \mathcal{B}_{\bullet}, \mathcal{I}^{\bullet})$$

Pour le voir, on filtre selon les colonnes le double complexe

$$\begin{array}{ccccccc}
& & \vdots & & \vdots & & \\
& & \uparrow & & \uparrow & & \\
\cdots & \longrightarrow & Hom_{\mathcal{O}^e}((P_{\bullet} \mathcal{B}_{\bullet})_p, \mathcal{I}^{q+1}) & \longrightarrow & Hom_{\mathcal{O}^e}((P_{\bullet} \mathcal{B}_{\bullet})_{p+1}, \mathcal{I}^{q+1}) & \longrightarrow & \cdots \\
& & \uparrow & & \uparrow & & \\
\cdots & \longrightarrow & Hom_{\mathcal{O}^e}((P_{\bullet} \mathcal{B}_{\bullet})_p, \mathcal{I}^q) & \longrightarrow & Hom_{\mathcal{O}^e}((P_{\bullet} \mathcal{B}_{\bullet})_{p+1}, \mathcal{I}^q) & \longrightarrow & \cdots \\
& & \uparrow & & \uparrow & & \\
& & \vdots & & \vdots & &
\end{array}$$

et on obtient par projectivité

$$E_1^{pq} = H^q(\text{Hom}_{\mathcal{O}^e}((P_\bullet \mathcal{B}_\bullet)_p, \mathcal{I}^\bullet)) = \text{Hom}_{\mathcal{O}^e}((P_\bullet \mathcal{B}_\bullet)_p, H^q(\mathcal{I}^\bullet))$$

d'où l'isomorphisme à la page E_1 . En utilisant l'adjonction et le lemme 3.2, on a

$$\begin{aligned} \text{Hom}_{\mathcal{O}^e}(P_\bullet^{\mathcal{O}^e} \mathcal{B}_\bullet, \mathcal{I}^\bullet) &= \text{Hom}_{\mathcal{O}}(\mathcal{O} \otimes_{\mathcal{O}^e} P_\bullet^{\mathcal{O}^e} \mathcal{B}_\bullet, \mathcal{I}^\bullet) \\ &\simeq \text{Hom}_{\mathcal{O}}(P_\bullet^{\mathcal{O}}(\mathcal{O} \otimes_{\mathcal{O}^e} \mathcal{B}_\bullet), \mathcal{I}^\bullet) \simeq \text{Hom}_{\mathcal{O}}(P_\bullet^{\mathcal{O}} \mathcal{D}_\bullet, \mathcal{I}^\bullet) \end{aligned}$$

Comme précédemment, on peut regarder \mathcal{D}_\bullet comme un double complexe centré en la ligne 0 et obtenir un quasi-isomorphisme avec le complexe total

$$P_\bullet \mathcal{D}_\bullet \xrightarrow{q.is.} \mathcal{D}_\bullet$$

car lorsque l'on filtre selon les colonnes le double complexe

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \downarrow & & \downarrow & & \\ \cdots & \longrightarrow & P_q \mathcal{D}_p & \longrightarrow & P_q \mathcal{D}_{p-1} & \longrightarrow & \cdots \\ & & \downarrow & & \downarrow & & \\ \cdots & \longrightarrow & P_{q-1} \mathcal{D}_p & \longrightarrow & P_{q-1} \mathcal{D}_{p-1} & \longrightarrow & \cdots \\ & & \downarrow & & \downarrow & & \\ & & \vdots & & \vdots & & \end{array}$$

on obtient l'isomorphisme à la page E^1 :

$$E_{pq}^1 = \begin{cases} \mathcal{D}_p & \text{si } q = 0 \\ 0 & \text{si } q \neq 0 \end{cases}$$

Puis l'injectivité donne un quasi-isomorphisme

$$\text{Hom}_{\mathcal{O}}(P_\bullet \mathcal{D}_\bullet, \mathcal{I}^\bullet) \xrightarrow{q.is.} \text{Hom}_{\mathcal{O}}(\mathcal{D}_\bullet, \mathcal{I}^\bullet)$$

ce qui se voit en filtrant selon les colonnes le double complexe

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & \text{Hom}_{\mathcal{O}}(\mathcal{D}_{q+1}, \mathcal{I}^p) & \longrightarrow & \text{Hom}_{\mathcal{O}}(\mathcal{D}_{q+1}, \mathcal{I}^{p+1}) & \longrightarrow & \cdots \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & \text{Hom}_{\mathcal{O}}(\mathcal{D}_q, \mathcal{I}^p) & \longrightarrow & \text{Hom}_{\mathcal{O}}(\mathcal{D}_q, \mathcal{I}^{p+1}) & \longrightarrow & \cdots \\ & & \uparrow & & \uparrow & & \\ & & \vdots & & \vdots & & \end{array}$$

$$E_1^{pq} = H^q(\text{Hom}_{\mathcal{O}}(\mathcal{D}_\bullet, \mathcal{I}^p)) = \text{Hom}_{\mathcal{O}}(H_q(\mathcal{D}_\bullet), \mathcal{I}^p)$$

Notons que \mathcal{I}^p est injectif comme préfaisceau de \mathcal{O} -module car l'inclusion des faisceaux de \mathcal{O}_X -module dans les préfaisceaux de \mathcal{O} -module admet comme adjoint à gauche la faisceautification, qui est exacte. Cette inclusion préserve donc les injectifs. On peut à présent terminer la démonstration :

$$\begin{aligned} \text{Ext}_{\mathcal{O}^e}^n(\mathcal{O}, \mathcal{F}) &= H^n(\text{Hom}_{\mathcal{O}^e}(P_\bullet \mathcal{B}_\bullet, \mathcal{F})) \\ &\simeq H^n(\text{Hom}_{\mathcal{O}^e}(P_\bullet \mathcal{B}_\bullet, \mathcal{I}^\bullet)) \\ &\simeq H^n(\text{Hom}_{\mathcal{O}}(P_\bullet \mathcal{D}_\bullet, \mathcal{I}^\bullet)) \\ &\simeq H^n(\text{Hom}_{\mathcal{O}}(\mathcal{D}_\bullet, \mathcal{I}^\bullet)) \\ &\simeq H^n(\text{Hom}_{\mathcal{O}_X}(\mathcal{C}_\bullet, \mathcal{I}^\bullet)) = HH^n(X, \mathcal{F}) \blacksquare \end{aligned}$$

4 Résolutions localement libres

Pour démontrer les théorèmes 2.1 et 2.5, il nous faut établir un lien entre la cohomologie de Hochschild d'un schéma X et l'hyper-ext du complexe \mathcal{C}_\bullet . Pour palier aux difficultés concernant le complexe \mathcal{C}_\bullet , on va se ramener dans un premier temps à considérer l'hyper-ext d'une résolution localement libre de $\delta_*\mathcal{O}_X$. Dans ce paragraphe, on va mettre en avant des propriétés de complexe de faisceau qui nous serviront jusqu'à la démonstration du théorème général.

On commence par énoncer un résultat fondamental pour notre étude, que l'on peut énoncer sous la généralité suivante. On considère \mathcal{O} un faisceau d'anneau sur un espace topologique X . Pour ne pas se soucier de la convergence des suites spectrales, tous les complexes de faisceau seront, dans ce paragraphe, supposés bornés en bas.

Lemme 4.1 : Un quasi-isomorphisme de complexe de faisceau de \mathcal{O} -module

$$\mathcal{A}_\bullet \xrightarrow{q.is} \mathcal{B}_\bullet$$

induit un isomorphisme entre les suites spectrales

$$\begin{aligned} Ext_{\mathcal{O}}^p(H_q(\mathcal{A}_\bullet), \mathcal{G}) &\Rightarrow \mathbb{E}xt_{\mathcal{O}}^{p+q}(\mathcal{A}_\bullet, \mathcal{G}) \\ Ext_{\mathcal{O}}^p(H_q(\mathcal{B}_\bullet), \mathcal{G}) &\Rightarrow \mathbb{E}xt_{\mathcal{O}}^{p+q}(\mathcal{B}_\bullet, \mathcal{G}) \end{aligned}$$

pour tout faisceau \mathcal{G} de \mathcal{O} -module. Deux morphismes de complexe homotopes

$$\mathcal{A}_\bullet \rightrightarrows \mathcal{B}_\bullet$$

induisent le même morphisme de suite spectrale.

Preuve : Soit $\mathcal{I}^\bullet \rightarrow \mathcal{A}_\bullet$ une résolution injective. La première suite spectrale est donnée en filtrant selon les colonnes le double complexe

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & Hom_{\mathcal{O}}(\mathcal{A}_{q+1}, \mathcal{I}^p) & \longrightarrow & Hom_{\mathcal{O}}(\mathcal{A}_{q+1}, \mathcal{I}^{p+1}) & \longrightarrow & \cdots \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & Hom_{\mathcal{O}}(\mathcal{A}_q, \mathcal{I}^p) & \longrightarrow & Hom_{\mathcal{O}}(\mathcal{A}_q, \mathcal{I}^{p+1}) & \longrightarrow & \cdots \\ & & \uparrow & & \uparrow & & \\ & & \vdots & & \vdots & & \end{array}$$

$$E_0^{pq} = Hom_{\mathcal{O}}(\mathcal{A}_q, \mathcal{I}^p)$$

$$E_1^{pq} = H^q(Hom_{\mathcal{O}}(\mathcal{A}_\bullet, \mathcal{I}^p)) = Hom_{\mathcal{O}}(H_q(\mathcal{A}_\bullet), \mathcal{I}^p)$$

$$E_2^{pq} = H_h^p(H_v^q(Hom_{\mathcal{O}}(\mathcal{A}_\bullet, \mathcal{I}^\bullet))) = Ext_{\mathcal{O}}^p(H_q(\mathcal{A}_\bullet), \mathcal{G})$$

Tout morphisme de complexe $\mathcal{A}_\bullet \rightarrow \mathcal{B}_\bullet$ induit un morphisme de double complexe

$$Hom_{\mathcal{O}}(\mathcal{B}_\bullet, \mathcal{I}^\bullet) \rightarrow Hom_{\mathcal{O}}(\mathcal{A}_\bullet, \mathcal{I}^\bullet)$$

puis un morphisme entre les suites spectrales associées. On voit qu'un quasi-isomorphisme induit un isomorphisme entre les pages E_1 et par conséquent entre les suites spectrales $(E_r)_{r \geq 2}$. De même, deux applications homotopes induisent le même morphisme sur la page E_1 et donc le même morphisme de suite spectrale $(E_r)_{r \geq 2}$. ■

Lemme 4.2 : Soient \mathcal{L}_\bullet un complexe de faisceau localement libre de \mathcal{O} -module et

$$\mathcal{A}_\bullet \xrightarrow{q.is} \mathcal{B}_\bullet$$

un quasi-isomorphisme de complexe de faisceau de \mathcal{O} -module. On a un quasi-isomorphisme

$$\mathcal{H}om_{\mathcal{O}}(\mathcal{L}_\bullet, \mathcal{A}^\bullet) \xrightarrow{q.is} \mathcal{H}om_{\mathcal{O}}(\mathcal{L}_\bullet, \mathcal{B}^\bullet)$$

Preuve : On dispose d'un morphisme de double complexe de faisceau de \mathcal{O} -module

$$\mathcal{H}om_{\mathcal{O}}(\mathcal{L}_\bullet, \mathcal{A}^\bullet) \rightarrow \mathcal{H}om_{\mathcal{O}}(\mathcal{L}_\bullet, \mathcal{B}^\bullet)$$

En filtrant selon les colonnes le double complexe de faisceau

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & \mathcal{H}om_{\mathcal{O}}(\mathcal{L}_p, \mathcal{A}^{q+1}) & \longrightarrow & \mathcal{H}om_{\mathcal{O}}(\mathcal{L}_{p+1}, \mathcal{A}^{q+1}) & \longrightarrow & \cdots \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & \mathcal{H}om_{\mathcal{O}}(\mathcal{L}_p, \mathcal{A}^q) & \longrightarrow & \mathcal{H}om_{\mathcal{O}}(\mathcal{L}_{p+1}, \mathcal{A}^q) & \longrightarrow & \cdots \\ & & \uparrow & & \uparrow & & \\ & & \vdots & & \vdots & & \end{array}$$

on obtient un isomorphisme à la page E_1 :

$$E_1^{pq} = H^q(\mathcal{H}om_{\mathcal{O}}(\mathcal{L}_p, \mathcal{A}^\bullet)) = \mathcal{H}om_{\mathcal{O}}(\mathcal{L}_p, H^q(\mathcal{A}^\bullet))$$

Cette égalité de faisceau découle d'une identification des fibres

$$\begin{aligned} (H^q(\mathcal{H}om_{\mathcal{O}}(\mathcal{L}_p, \mathcal{A}^\bullet)))_x &= H^q(\mathcal{H}om_{\mathcal{O}_x}(\mathcal{L}_{p,x}, \mathcal{A}_x^\bullet)) \\ &= \mathcal{H}om_{\mathcal{O}_x}(\mathcal{L}_{p,x}, H^q(\mathcal{A}_x^\bullet)) = (\mathcal{H}om_{\mathcal{O}}(\mathcal{L}_p, H^q(\mathcal{A}^\bullet)))_x \end{aligned}$$

en utilisant que $\mathcal{L}_{p,x}$ est un \mathcal{O}_x -module libre. Puisque les suites spectrales convergent, ceci entraîne un isomorphisme des cohomologies totales. ■

Lemme 4.3 : Si \mathcal{F} est un faisceau plat de \mathcal{O} -module et \mathcal{G} un faisceau injectif de \mathcal{O} -module alors le faisceau $\mathcal{H}om_{\mathcal{O}}(\mathcal{F}, \mathcal{G})$ est injectif.

Preuve : $\mathcal{H}om_{\mathcal{O}}(-, \mathcal{H}om_{\mathcal{O}}(\mathcal{F}, \mathcal{G})) \simeq \mathcal{H}om_{\mathcal{O}}(-, \mathcal{G}) \circ (\mathcal{F} \otimes_{\mathcal{O}} -)$ ■

Lemme 4.4 : Un quasi-isomorphisme de complexe de faisceau flasque de \mathcal{O} -module

$$\mathcal{A}^\bullet \xrightarrow{q.is} \mathcal{B}^\bullet$$

induit un quasi-isomorphisme de complexe de $\Gamma\mathcal{O}$ -module

$$\Gamma\mathcal{A}^\bullet \xrightarrow{q.is} \Gamma\mathcal{B}^\bullet$$

Preuve : Soit \mathcal{M}^\bullet le cône de $\mathcal{A}^\bullet \rightarrow \mathcal{B}^\bullet$. Par hypothèse, \mathcal{M}^\bullet est exacte. On a donc une résolution flasque du faisceau nul

$$0 \rightarrow \mathcal{M}^\bullet$$

et ainsi $0 = H^q(\Gamma\mathcal{M}^\bullet)$ pour tout q . Puisque $\Gamma\mathcal{M}^\bullet$ est le cône de $\Gamma\mathcal{A}^\bullet \rightarrow \Gamma\mathcal{B}^\bullet$, c'est le résultat. ■

Concentrons nous à présent sur le cas des schémas.

Lemme 4.5 : Si $i : Y \rightarrow X$ est un morphisme de schéma, \mathcal{A} un faisceau de \mathcal{O}_X -module et \mathcal{B} un faisceau de \mathcal{O}_Y -module alors on a un isomorphisme de faisceau de \mathcal{O}_X -module

$$\mathrm{Hom}_{\mathcal{O}_X}(\mathcal{A}, i_*\mathcal{B}) \simeq i_*\mathrm{Hom}_{\mathcal{O}_Y}(i^*\mathcal{A}, \mathcal{B})$$

Preuve : On commence par construire pour tout faisceau \mathcal{F} de \mathcal{O}_X -module un isomorphisme naturel de faisceau de \mathcal{O}_Y -module

$$i^*(\mathcal{F} \otimes_{\mathcal{O}_X} \mathcal{A}) \simeq i^*\mathcal{F} \otimes_{\mathcal{O}_Y} i^*\mathcal{A}$$

Pour ce faire, on part de l'unité $1 \rightarrow i_*i^*$ qui fournit un morphisme

$$\mathcal{F} \otimes_{\mathcal{O}_X} \mathcal{A} \rightarrow i_*i^*\mathcal{F} \otimes_{\mathcal{O}_X} i_*i^*\mathcal{A}$$

puis le morphisme $\mathcal{O}_X \rightarrow i_*\mathcal{O}_Y$ donne

$$i_*i^*\mathcal{F} \otimes_{\mathcal{O}_X} i_*i^*\mathcal{A} \rightarrow i_*i^*\mathcal{F} \otimes_{i_*\mathcal{O}_Y} i_*i^*\mathcal{A}$$

que l'on compose par le morphisme

$$i_*i^*\mathcal{F} \otimes_{i_*\mathcal{O}_Y} i_*i^*\mathcal{A} \rightarrow i_*(i^*\mathcal{F} \otimes_{\mathcal{O}_Y} i^*\mathcal{A})$$

et on obtient par adjonction le morphisme voulu. C'est un isomorphisme, comme on peut le constater sur les fibres :

$$\begin{aligned} (i^*(\mathcal{F} \otimes_{\mathcal{O}_X} \mathcal{A}))_y &= \mathcal{O}_{Y,y} \otimes_{\mathcal{O}_{X,i(y)}} (\mathcal{F}_{i(y)} \otimes_{\mathcal{O}_{X,i(y)}} \mathcal{A}_{i(y)}) \\ &\simeq (\mathcal{O}_{Y,y} \otimes_{\mathcal{O}_{X,i(y)}} \mathcal{F}_{i(y)}) \otimes_{\mathcal{O}_{X,i(y)}} (\mathcal{O}_{Y,y} \otimes_{\mathcal{O}_{X,i(y)}} \mathcal{A}_{i(y)}) = (i^*\mathcal{F} \otimes_{\mathcal{O}_Y} i^*\mathcal{A})_y \end{aligned}$$

On démontre alors le lemme grâce au plongement de Yoneda :

$$\begin{aligned} \mathrm{Hom}_{\mathcal{O}_X}(\mathcal{F}, \mathrm{Hom}_{\mathcal{O}_X}(\mathcal{A}, i_*\mathcal{B})) &\simeq \mathrm{Hom}_{\mathcal{O}_X}(\mathcal{F} \otimes_{\mathcal{O}_X} \mathcal{A}, i_*\mathcal{B}) \\ &\simeq \mathrm{Hom}_{\mathcal{O}_Y}(i^*(\mathcal{F} \otimes_{\mathcal{O}_X} \mathcal{A}), \mathcal{B}) \simeq \mathrm{Hom}_{\mathcal{O}_Y}(i^*\mathcal{F} \otimes_{\mathcal{O}_Y} i^*\mathcal{A}, \mathcal{B}) \\ &\simeq \mathrm{Hom}_{\mathcal{O}_Y}(i^*\mathcal{F}, \mathrm{Hom}_{\mathcal{O}_Y}(i^*\mathcal{A}, \mathcal{B})) \simeq \mathrm{Hom}_{\mathcal{O}_X}(\mathcal{F}, i_*\mathrm{Hom}_{\mathcal{O}_Y}(i^*\mathcal{A}, \mathcal{B})) \blacksquare \end{aligned}$$

On arrive au dernier résultat du paragraphe, dont le corollaire est le premier pas vers la démonstration du théorème général.

Proposition 4.6 : Soient $i : Y \hookrightarrow X$ une immersion fermée, \mathcal{L}_\bullet un complexe de faisceau localement libre de \mathcal{O}_X -module et \mathcal{S} un faisceau de \mathcal{O}_Y -module. On a un isomorphisme de δ -foncteur en \mathcal{S}

$$\mathbb{E}xt_{\mathcal{O}_X}^n(\mathcal{L}_\bullet, i_*\mathcal{S}) \simeq \mathbb{E}xt_{\mathcal{O}_Y}^n(i^*\mathcal{L}_\bullet, \mathcal{S})$$

Preuve : Soient $\mathcal{S} \rightarrow \mathcal{I}^\bullet$ et $i_*\mathcal{S} \rightarrow \mathcal{J}^\bullet$ deux résolutions injectives. Le fait que i soit une immersion fermée entraîne que le foncteur i_* est exact :

$$(i_*\mathcal{F})_{i(y)} = \mathcal{F}_y \text{ et } (i_*\mathcal{F})_x = 0 \text{ si } x \notin i(Y)$$

En particulier, $i_*\mathcal{S} \rightarrow i_*\mathcal{I}^\bullet$ est une résolution et on a un triangle commutatif

$$\begin{array}{ccc} & & i_*\mathcal{I}^\bullet \\ & \nearrow & \downarrow q.is. \\ i_*\mathcal{S} & & \mathcal{J}^\bullet \\ & \searrow & \end{array}$$

D'après le lemme 4.2, on obtient un quasi-isomorphisme

$$\mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, i_*\mathcal{I}^\bullet) \xrightarrow{q.is.} \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, \mathcal{J}^\bullet)$$

Or, d'après le lemme 4.5, on a un isomorphisme

$$\mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, i_*\mathcal{I}^\bullet) \simeq i_*\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet)$$

et d'après le lemme 4.3, les faisceaux $\mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, \mathcal{J}^\bullet)$ et $\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet)$ sont injectifs et donc flasques. Par suite le faisceau $i_*\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet)$ est flasque et le lemme 4.4 donne un quasi-isomorphisme

$$\Gamma(X, i_*\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet)) \xrightarrow{q.is.} \Gamma(X, \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, \mathcal{J}^\bullet))$$

c'est à dire

$$\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet) \xrightarrow{q.is.} \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, \mathcal{J}^\bullet)$$

ce qui donne l'isomorphisme recherché. Pour la naturalité, on choisit pour toute suite exacte de faisceau de \mathcal{O}_Y -module

$$0 \longrightarrow \mathcal{S}' \longrightarrow \mathcal{S} \longrightarrow \mathcal{S}'' \longrightarrow 0$$

des suites exactes de résolutions injectives

$$0 \longrightarrow \mathcal{I}'^\bullet \longrightarrow \mathcal{I}^\bullet \longrightarrow \mathcal{I}''^\bullet \longrightarrow 0$$

$$0 \longrightarrow \mathcal{J}'^\bullet \longrightarrow \mathcal{J}^\bullet \longrightarrow \mathcal{J}''^\bullet \longrightarrow 0$$

rendant le diagramme à ligne exacte suivant commutatif

$$\begin{array}{ccccccc} 0 & \longrightarrow & i_*\mathcal{I}'^\bullet & \longrightarrow & i_*\mathcal{I}^\bullet & \longrightarrow & i_*\mathcal{I}''^\bullet \longrightarrow 0 \\ & & \downarrow & & \downarrow & & \downarrow \\ 0 & \longrightarrow & \mathcal{J}'^\bullet & \longrightarrow & \mathcal{J}^\bullet & \longrightarrow & \mathcal{J}''^\bullet \longrightarrow 0 \end{array}$$

Ceci induit un diagramme à ligne exacte commutatif

$$\begin{array}{ccccccc} \dots & \longrightarrow & H^n(i_*\mathcal{I}'^\bullet) & \longrightarrow & H^n(i_*\mathcal{I}^\bullet) & \longrightarrow & H^n(i_*\mathcal{I}''^\bullet) \longrightarrow H^{n+1}(i_*\mathcal{I}'^\bullet) \longrightarrow \dots \\ & & \downarrow & & \downarrow & & \downarrow \\ \dots & \longrightarrow & H^n(\mathcal{J}'^\bullet) & \longrightarrow & H^n(\mathcal{J}^\bullet) & \longrightarrow & H^n(\mathcal{J}''^\bullet) \longrightarrow H^{n+1}(\mathcal{J}'^\bullet) \longrightarrow \dots \end{array}$$

En reprenant le quasi-isomorphisme initial

$$\mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, i_*\mathcal{I}^\bullet) \xrightarrow{q.is.} \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, \mathcal{J}^\bullet)$$

la naturalité est donnée à la page E_1

$$\begin{array}{ccccccc}
& & \vdots & & \vdots & & \\
& & \uparrow & & \uparrow & & \\
\cdots & \longrightarrow & \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_p, i_* \mathcal{I}^{q+1}) & \longrightarrow & \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_{p+1}, i_* \mathcal{I}^{q+1}) & \longrightarrow & \cdots \\
& & \uparrow & & \uparrow & & \\
\cdots & \longrightarrow & \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_p, i_* \mathcal{I}^q) & \longrightarrow & \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_{p+1}, i_* \mathcal{I}^q) & \longrightarrow & \cdots \\
& & \uparrow & & \uparrow & & \\
& & \vdots & & \vdots & &
\end{array}$$

$$E_1^{pq} = H^q(\mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_p, i_* \mathcal{I}^\bullet)) = \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_p, H^q(i_* \mathcal{I}^\bullet)) \blacksquare$$

Corollaire 4.7 : Soient X un schéma séparé et \mathcal{L}_\bullet un complexe de faisceau localement libre de $\mathcal{O}_{X \times X}$ -module tel que $H_0(\mathcal{L}_\bullet) = \delta_* \mathcal{O}_X$ et $H_p(\mathcal{L}_\bullet) = 0$ pour tout $p \neq 0$. Pour tout faisceau \mathcal{F} de \mathcal{O}_X -module, on a un isomorphisme de δ -foncteur en \mathcal{F}

$$H^n(\mathcal{O}_X, \mathcal{F}) \simeq \mathbb{E}xt_{\mathcal{O}_X}^n(\delta^* \mathcal{L}_\bullet, \mathcal{F})$$

Preuve : On part de l'isomorphisme donné par la proposition 4.6 appliquée à l'immersion fermée $\delta : X \hookrightarrow X \times X$

$$\mathbb{E}xt_{\mathcal{O}_{X \times X}}^n(\mathcal{L}_\bullet, \delta_* \mathcal{F}) \simeq \mathbb{E}xt_{\mathcal{O}_X}^n(\delta^* \mathcal{L}_\bullet, \mathcal{F})$$

puis on applique le lemme 4.1 au quasi-isomorphisme

$$\mathcal{L}_\bullet \xrightarrow{q.is.} \delta_* \mathcal{O}_X$$

ce qui fournit un isomorphisme naturel en \mathcal{F}

$$H^n(\mathcal{O}_X, \mathcal{F}) = \mathbb{E}xt_{\mathcal{O}_{X \times X}}^n(\delta_* \mathcal{O}_X, \delta_* \mathcal{F}) \simeq \mathbb{E}xt_{\mathcal{O}_{X \times X}}^n(\mathcal{L}_\bullet, \delta_* \mathcal{F}) \blacksquare$$

5 Suites spectrales

Le corollaire 4.7 établit un lien entre la cohomologie de Hochschild d'un schéma X séparé et de type fini sur un corps avec l'hyper-ext d'une résolution localement libre de $\delta_* \mathcal{O}_X$. On souhaite aller plus loin en comparant les suites spectrales associées précédemment. Pour ce faire, on va utiliser les résolutions de Cartan-Eilenberg et les techniques de convergence des suites spectrales. La référence originale de Swan est

H. Cartan and S. Eilenberg, Homological Algebra (Princeton University Press, Princeton, NJ, 1956).

Rappelons ici le vocabulaire introduit dans l'article [3]. On se place dans une catégorie abélienne ayant assez d'injectif, et on suppose tous les complexes bornés en bas.

Un CE-monomorphisme $i : A^\bullet \rightarrow B^\bullet$ est un monomorphisme de complexe tel que $i_* : H^\bullet(A^\bullet) \rightarrow H^\bullet(B^\bullet)$ est un monomorphisme. Une suite CE-exacte est une suite exacte

$$0 \longrightarrow C''^\bullet \xrightarrow{f} C^\bullet \xrightarrow{g} C'''^\bullet \longrightarrow 0$$

telle que $\text{im}(f) \rightarrow C^\bullet$ est un CE-monomorphisme. Un complexe I^\bullet est CE-injectif si pour tout CE-monomorphisme $A^\bullet \rightarrow B^\bullet$ et tout morphisme $A^\bullet \rightarrow I^\bullet$, il existe une factorisation

$$\begin{array}{ccc} A^\bullet & \longrightarrow & B^\bullet \\ & \searrow & \downarrow \\ & & I^\bullet \end{array}$$

Enfin, une CE-résolution d'un complexe A^\bullet est une suite CE-exacte

$$0 \longrightarrow A^\bullet \longrightarrow C^{0\bullet} \longrightarrow C^{1\bullet} \longrightarrow C^{2\bullet} \longrightarrow \dots$$

où $C^{p\bullet}$ est CE-injectif pour tout $p \geq 0$. Les CE-résolutions existent toujours dans les catégories ayant assez d'injectif. Une propriété importante pour la suite est que si $C^{\bullet\bullet}$ est une CE-résolution de A^\bullet et F un foncteur additif alors

$$H_v^q(F(C^{\bullet\bullet})) = F(H_v^q(C^{\bullet\bullet}))$$

pour tout $q \geq 0$. On va s'intéresser au cas des faisceaux avec $F = \Gamma$. On pourra alors calculer l'hypercohomologie d'un complexe de faisceau en utilisant les CE-résolutions.

Lemme 5.1 : Soit \mathcal{M}_\bullet un complexe de faisceau localement libre sur un schéma Y tel que $H_q(\mathcal{M}_\bullet)$ est localement libre pour tout q . Pour tout faisceau \mathcal{S} de \mathcal{O}_Y -module, on a un isomorphisme naturel

$$H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S})) \simeq \mathcal{H}om_{\mathcal{O}_Y}(H_q(\mathcal{M}_\bullet), \mathcal{S})$$

Preuve : Pour un foncteur contravariant F exact à gauche et un complexe C_\bullet arbitraires, on peut construire un morphisme naturel

$$H^q(F(C_\bullet)) \rightarrow F(H_q(C_\bullet))$$

Notons Z'_q le conoyau de la différentielle $C_{q+1} \rightarrow C_q$. Par hypothèse, $F(Z'_q)$ est le noyau de $F(C_q) \rightarrow F(C_{q+1})$ et par conséquent,

$$H^q(F(C_\bullet)) = \text{coker}(F(C_{q-1}) \rightarrow F(Z'_q))$$

On dispose d'une suite exacte

$$0 \longrightarrow H_q(C_\bullet) \longrightarrow Z'_q \longrightarrow C_{q-1}$$

qui induit une composition nulle

$$F(C_{q-1}) \rightarrow F(Z'_q) \rightarrow F(H_q(C_\bullet))$$

d'où la factorisation naturelle

$$\begin{array}{ccccc} F(C_{q-1}) & \longrightarrow & F(Z'_q) & \longrightarrow & H^q(F(C_\bullet)) \\ & & \searrow & & \downarrow \\ & & & & F(H_q(C_\bullet)) \end{array}$$

Revenons au cas où $F = \mathcal{H}om_{\mathcal{O}_Y}(-, \mathcal{S})$ et $C_\bullet = \mathcal{M}_\bullet$. Pour vérifier que l'on a un isomorphisme sur les fibres, on est ramené au cas où l'on applique $\mathcal{H}om_R(-, N)$ à un complexe de R -module projectif

$$\cdots \longrightarrow P_2 \longrightarrow P_1 \longrightarrow P_0 \longrightarrow 0$$

tel que les homologies sont des R -modules projectifs. Comme il a été remarqué au premier paragraphe, cette configuration implique l'isomorphisme voulu

$$H^q(\mathcal{H}om_R(P_\bullet, N)) \simeq \mathcal{H}om_R(H_q(P_\bullet), N) \blacksquare$$

Il faut remarquer l'utilisation de l'hypothèse "localement libre" sur le complexe \mathcal{M}_\bullet et l'homologie $H_\bullet(\mathcal{M}_\bullet)$. En fait, lemme 5.1 reste vrai si l'on suppose seulement que les fibres $\mathcal{M}_{q,y}$ et $H_q(\mathcal{M}_\bullet)_y = H_q(\mathcal{M}_{\bullet,y})$ sont des $\mathcal{O}_{Y,y}$ -modules projectifs. Cette démonstration nous montre la grande maniabilité, liée au passage aux fibres, des faisceaux pour les questions homologiques.

Corollaire 5.2 : Sous les mêmes hypothèses, si

$$0 \longrightarrow \mathcal{F}' \xrightarrow{f} \mathcal{F} \xrightarrow{g} \mathcal{F}'' \longrightarrow 0$$

est une suite exacte de faisceau de \mathcal{O}_Y -module, alors la suite

$$0 \longrightarrow \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{F}') \xrightarrow{f_*} \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{F}) \xrightarrow{g_*} \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{F}'') \longrightarrow 0$$

est CE-exacte.

Preuve : La suite est exacte sur les fibres, car $\mathcal{M}_{q,y}$ est projective sur $\mathcal{O}_{Y,y}$. Puisque f est un monomorphisme, on a $H^q(\text{im}(f_*)) \simeq H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{F}'))$ et le lemme 5.1 donne un diagramme commutatif

$$\begin{array}{ccc} H^q(\text{im}(f_*)) & \longrightarrow & H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{F})) \\ \parallel & & \parallel \\ 0 \longrightarrow \mathcal{H}om_{\mathcal{O}_Y}(H_q(\mathcal{M}_\bullet), \mathcal{F}') & \xrightarrow{f_*} & \mathcal{H}om_{\mathcal{O}_Y}(H_q(\mathcal{M}_\bullet), \mathcal{F}) \end{array}$$

Par hypothèse, les fibres de $H_q(\mathcal{M}_\bullet)$ sont projectives $\mathcal{O}_{Y,y}$, donc la ligne inférieure est exacte et notre suite est CE-exacte. \blacksquare

On arrive ici à l'étude des suites spectrales annoncée en introduction. Rappelons ici qu'une suite spectrale associée à un double complexe nul hors d'un quart de plan (par exemple une CE-résolution) converge toujours vers la cohomologie totale. Cela justifie que les complexes que l'on considère dans ce paragraphe sont supposés bornés en bas.

Lemme 5.3 : Sous les hypothèses du lemme 5.1, les suites spectrales

$$Ext_{\mathcal{O}_Y}^p(H_q(\mathcal{M}_\bullet), \mathcal{S}) \Rightarrow \mathbb{E}xt_{\mathcal{O}_Y}^{p+q}(\mathcal{M}_\bullet, \mathcal{S})$$

$$H^p(Y, H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S}))) \Rightarrow \mathbb{H}^{p+q}(Y, \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S}))$$

sont isomorphes.

Preuve : Soit $\mathcal{S} \rightarrow \mathcal{I}^\bullet$ une résolution injective. Le corollaire 5.2 montre que la suite

$$0 \longrightarrow \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S}) \longrightarrow \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{I}^0) \longrightarrow \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{I}^1) \longrightarrow \dots$$

est CE-exacte, donc si on choisit une résolution CE-injective $\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S}) \rightarrow \mathcal{J}^{\bullet\bullet}$ alors il existe un morphisme f unique à homotopie près rendant le triangle suivant commutatif

$$\begin{array}{ccc} & \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{I}^\bullet) & \\ & \uparrow & \downarrow f \\ \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S}) & \searrow & \mathcal{J}^{\bullet\bullet} \end{array}$$

En appliquant Γ , on obtient un morphisme de double complexe

$$\Gamma f : \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{I}^\bullet) \rightarrow \Gamma(\mathcal{J}^{\bullet\bullet})$$

Lorsque l'on filtre selon les colonnes le double complexe

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \uparrow & & \uparrow & & \\ \dots & \longrightarrow & \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_{q+1}, \mathcal{I}^p) & \longrightarrow & \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_{q+1}, \mathcal{I}^{p+1}) & \longrightarrow & \dots \\ & & \uparrow & & \uparrow & & \\ \dots & \longrightarrow & \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_q, \mathcal{I}^p) & \longrightarrow & \mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_q, \mathcal{I}^{p+1}) & \longrightarrow & \dots \\ & & \uparrow & & \uparrow & & \\ & & \vdots & & \vdots & & \end{array}$$

on obtient la première suite spectrale :

$$E_1^{pq} = H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{I}^p)) = \mathcal{H}om_{\mathcal{O}_Y}(H_q(\mathcal{M}_\bullet), \mathcal{I}^p)$$

$$E_2^{pq} = Ext_{\mathcal{O}_Y}^p(H_q(\mathcal{M}_\bullet), \mathcal{S})$$

Et lorsque l'on filtre selon les colonnes le double complexe

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \uparrow & & \uparrow & & \\ \dots & \longrightarrow & \Gamma(\mathcal{J}^{p,q+1}) & \longrightarrow & \Gamma(\mathcal{J}^{p+1,q+1}) & \longrightarrow & \dots \\ & & \uparrow & & \uparrow & & \\ \dots & \longrightarrow & \Gamma(\mathcal{J}^{p,q}) & \longrightarrow & \Gamma(\mathcal{J}^{p+1,q}) & \longrightarrow & \dots \\ & & \uparrow & & \uparrow & & \\ & & \vdots & & \vdots & & \end{array}$$

on obtient la deuxième suite spectrale :

$$E_1^{pq} = H^q(\Gamma(\mathcal{J}^{p\bullet})) = \Gamma(H^q(\mathcal{J}^{p\bullet}))$$

$$E_2^{pq} = H_h^p(\Gamma(H_v^q(\mathcal{J}^{\bullet\bullet}))) = H^p(Y, H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S})))$$

Cette dernière égalité découle du fait que la CE-résolution $\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S}) \rightarrow \mathcal{J}^{\bullet\bullet}$ induit des résolutions injectives $H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S})) \rightarrow H_v^q(\mathcal{J}^{\bullet\bullet})$ pour tout p . Ainsi, il nous reste à vérifier que notre morphisme Γf induit un isomorphisme sur la page E_2 . Sur la page E_1 , Γf est donné en appliquant Γ au morphisme

$$\mathcal{H}om_{\mathcal{O}_Y}(H_q(\mathcal{M}_\bullet), \mathcal{I}^\bullet) \rightarrow H_v^q(\mathcal{J}^{\bullet\bullet})$$

Or par hypothèse, le lemme 4.3 nous dit que

$$\mathcal{H}om_{\mathcal{O}_Y}(H_q(\mathcal{M}_\bullet), \mathcal{S}) \rightarrow \mathcal{H}om_{\mathcal{O}_Y}(H_q(\mathcal{M}_\bullet), \mathcal{I}^\bullet)$$

est une résolution injective, tout comme

$$H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S})) \rightarrow H_v^q(\mathcal{J}^{\bullet\bullet})$$

Puisque les deux faisceaux $\mathcal{H}om_{\mathcal{O}_Y}(H_q(\mathcal{M}_\bullet), \mathcal{S})$ et $H^q(\mathcal{H}om_{\mathcal{O}_Y}(\mathcal{M}_\bullet, \mathcal{S}))$ sont isomorphes d'après le lemme 5.1, ils ont donc la même cohomologie sur Y ce qui signifie que Γf est un isomorphisme sur la page E_2 . ■

Avant d'aboutir au résultat final de ce paragraphe, on a besoin d'un dernier lemme que l'on peut énoncer sous une forme générale.

Lemme 5.4 : Soient $i : Y \hookrightarrow X$ une immersion fermée et \mathcal{A}^\bullet un complexe de faisceau de \mathcal{O}_Y -module. Les suites spectrales

$$H^p(Y, H^q(\mathcal{A}^\bullet)) \Rightarrow \mathbb{H}^{p+q}(Y, \mathcal{A}^\bullet)$$

$$H^p(X, H^q(i_*\mathcal{A}^\bullet)) \Rightarrow \mathbb{H}^{p+q}(X, i_*\mathcal{A}^\bullet)$$

sont isomorphes.

Preuve : Soient $\mathcal{A}^\bullet \rightarrow \mathcal{I}^{\bullet\bullet}$ et $i_*\mathcal{A}^\bullet \rightarrow \mathcal{J}^{\bullet\bullet}$ deux CE-résolutions. Les double-complexes

$$\Gamma(Y, \mathcal{I}^{\bullet\bullet}) ; \Gamma(X, \mathcal{J}^{\bullet\bullet})$$

induisent alors les deux suites spectrales voulues. On construit un morphisme comme suit. i_* est exact donc $i_*\mathcal{A}^\bullet \rightarrow i_*\mathcal{I}^{\bullet\bullet}$ est une résolution. Il existe ainsi un morphisme f unique à homotopie près rendant le triangle suivant commutatif

$$\begin{array}{ccc} & & i_*\mathcal{I}^{\bullet\bullet} \\ & \nearrow & \downarrow f \\ i_*\mathcal{A}^\bullet & & \mathcal{J}^{\bullet\bullet} \end{array}$$

En appliquant $\Gamma(X, -)$, on obtient un morphisme de double complexe

$$\Gamma(X, f) : \Gamma(Y, \mathcal{I}^{\bullet\bullet}) \rightarrow \Gamma(X, \mathcal{J}^{\bullet\bullet})$$

qui correspond sur la page E_2 à l'isomorphisme

$$H^p(Y, H^q(\mathcal{A}^\bullet)) = H^p(X, i_*H^q(\mathcal{A}^\bullet)) \simeq H^p(X, H^q(i_*\mathcal{A}^\bullet)) \quad \blacksquare$$

Proposition 5.5 : Soient $i : Y \hookrightarrow X$ une immersion fermée et \mathcal{L}_\bullet un complexe de faisceau localement libre de \mathcal{O}_X -module tel que $H_0(\mathcal{L}_\bullet) = \mathcal{T}$, $H_q(\mathcal{L}_\bullet) = 0$ pour tout $q \neq 0$ et $H_q(i^*\mathcal{L}_\bullet)$ est localement libre pour tout q . Pour tout faisceau \mathcal{S} de \mathcal{O}_Y -module, les suites spectrales

$$\mathrm{Ext}_{\mathcal{O}_Y}^p(H_q(i^*\mathcal{L}_\bullet), \mathcal{S}) \Rightarrow \mathbb{E}xt_{\mathcal{O}_Y}^{p+q}(i^*\mathcal{L}_\bullet, \mathcal{S})$$

$$H^p(X, \mathcal{E}xt_{\mathcal{O}_X}^q(\mathcal{T}, i_*\mathcal{S})) \Rightarrow \mathrm{Ext}_{\mathcal{O}_X}^{p+q}(\mathcal{T}, i_*\mathcal{S})$$

sont isomorphes.

Preuve : Soit $i_*\mathcal{S} \rightarrow \mathcal{J}^\bullet$ une résolution injective. La deuxième suite spectrale est donnée par l'hypercohomologie sur X du complexe de faisceau $\mathcal{H}om_{\mathcal{O}_X}(\mathcal{T}, \mathcal{J}^\bullet)$ [4, Th.5.8.3]. Si on regarde \mathcal{T} comme un complexe de faisceau concentré en 0, on dispose par hypothèse d'un quasi-isomorphisme

$$\mathcal{L}_\bullet \xrightarrow{q.is.} \mathcal{T}$$

Par injectivité, on obtient un quasi-isomorphisme

$$\mathcal{H}om_{\mathcal{O}_X}(\mathcal{T}, \mathcal{J}^\bullet) \xrightarrow{q.is.} \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, \mathcal{J}^\bullet)$$

de sorte que la deuxième suite spectrale est donnée par l'hypercohomologie sur X du complexe total de $\mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, \mathcal{J}^\bullet)$. Soit $\mathcal{S} \rightarrow \mathcal{I}^\bullet$ une résolution injective. Le foncteur i_* est exact, donc $i_*\mathcal{S} \rightarrow i_*\mathcal{I}^\bullet$ est une résolution et on a un triangle commutatif

$$\begin{array}{ccc} & & i_*\mathcal{I}^\bullet \\ & \nearrow & \downarrow q.is. \\ i_*\mathcal{S} & & \mathcal{J}^\bullet \end{array}$$

En particulier, puisque les fibres de \mathcal{L}_\bullet sont libres, on obtient un quasi-isomorphisme

$$\mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, i_*\mathcal{I}^\bullet) \xrightarrow{q.is.} \mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, \mathcal{J}^\bullet)$$

Or le lemme 4.5 donne un isomorphisme

$$\mathcal{H}om_{\mathcal{O}_X}(\mathcal{L}_\bullet, i_*\mathcal{I}^\bullet) \simeq i_*\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet)$$

En conséquence, la deuxième suite spectrale est donnée par l'hypercohomologie sur X du complexe total de $i_*\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet)$, qui est d'après le lemme 5.4 donnée par l'hypercohomologie sur Y du complexe total de $\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet)$. Mais si on regarde \mathcal{S} comme un complexe concentré en 0, on dispose d'un quasi-isomorphisme

$$\mathcal{S} \xrightarrow{q.is.} \mathcal{I}^\bullet$$

et puisque les fibres de $i^*\mathcal{L}_\bullet$ sont libres, on obtient un quasi-isomorphisme

$$\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{S}) \xrightarrow{q.is.} \mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{I}^\bullet)$$

Ainsi la deuxième suite spectrale est isomorphe à la suite spectrale

$$H^p(Y, H^q(\mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{S}))) \Rightarrow \mathbb{H}^{p+q}(Y, \mathcal{H}om_{\mathcal{O}_Y}(i^*\mathcal{L}_\bullet, \mathcal{S}))$$

Cette dernière est, par le lemme 5.3, isomorphe à la première suite spectrale. ■

Une conséquence immédiate de cette proposition est le cas où l'immersion fermée est l'application diagonale δ d'un schéma séparé X et où $\mathcal{T} = \delta_* \mathcal{O}_X$:

Corollaire 5.6 : Soient X un schéma séparé et de type fini sur un corps et \mathcal{L}_\bullet un complexe de faisceau localement libre de $\mathcal{O}_{X \times X}$ -module tel que $H_0(\mathcal{L}_\bullet) \simeq \delta_* \mathcal{O}_X$, $H_q(\mathcal{L}_\bullet) = 0$ pour tout $q \neq 0$ et $H_q(i^* \mathcal{L}_\bullet)$ est localement libre pour tout q . Pour tout faisceau \mathcal{F} de \mathcal{O}_X -module, les suites spectrales

$$Ext_{\mathcal{O}_X}^p(H_q(\delta^* \mathcal{L}_\bullet), \mathcal{F}) \Rightarrow \mathbb{E}xt_{\mathcal{O}_X}^{p+q}(\delta^* \mathcal{L}_\bullet, \mathcal{F})$$

$$H^p(X, \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})) \Rightarrow H^{p+q}(\mathcal{O}_X, \mathcal{F})$$

sont isomorphes.

6 Résolutions plates

Après avoir établi un lien entre la cohomologie de Hochschild d'un schéma X séparé et de type fini sur un corps avec l'hyper-ext d'une résolution localement libre de $\delta_*\mathcal{O}_X$, on va se ramener à considérer des résolutions plates et quasi-cohérentes de $\delta_*\mathcal{O}_X$. Ceci nous permettra en particulier de démontrer les théorèmes 2.1 et 2.5 dans le cas où X est affine en utilisant la résolution

$$\mathcal{B}_\bullet \rightarrow \delta_*\mathcal{O}_X$$

introduite au lemme 2.4 (4).

Dans ce paragraphe, tous les complexes de chaînes sont supposés bornés en bas et X désigne un schéma quasi-projectif sur un corps. Le résultat suivant justifie la présence de cette dernière hypothèse.

Lemme 6.1 : Soit $\mathcal{F} \rightarrow \mathcal{G}$ un épimorphisme de faisceau quasi-cohérent sur X . Si \mathcal{G} est cohérent alors il existe un faisceau localement libre \mathcal{L} de \mathcal{O}_X -module ainsi qu'un morphisme de faisceau $\mathcal{L} \rightarrow \mathcal{F}$ tel que la composition

$$\mathcal{L} \rightarrow \mathcal{F} \rightarrow \mathcal{G}$$

est un épimorphisme.

Preuve : Supposons X projectif sur un anneau Noethérien et \mathcal{F} cohérent. Dans ce cas il existe un faisceau localement libre \mathcal{L} de \mathcal{O}_X -module ainsi qu'un épimorphisme

$$\mathcal{L} \rightarrow \mathcal{F}$$

[1, Ch.II, Cor.5.18]. Si X est seulement supposé quasi-projectif sur un corps, alors \mathcal{F} se prolonge sur l'adhérence de X dans \mathbb{P}^n [1, Ch.II, Ex.5.15] qui est un schéma projectif sur un anneau Noethérien. On est alors ramené à la situation précédente. Enfin, si \mathcal{F} est seulement supposé quasi-cohérent, alors \mathcal{F} est l'union de ses sous-faisceaux cohérents [1, Ch.II, Ex.5.15.e] et l'un d'entre eux est envoyé sur \mathcal{G} . En effet, la restriction de l'épimorphisme $\mathcal{F} \rightarrow \mathcal{G}$ sur un ouvert affine de X est déterminée par une application linéaire surjective $M \rightarrow N$ avec N est de type fini et se restreint donc sur un sous module de type fini $M' \subset M$ en une surjection. Puisque X est quasi-compact, on peut construire le faisceau cohérent $\mathcal{F}' \subset \mathcal{F}$ voulu. D'après ce qui précède, il existe un faisceau localement libre \mathcal{L} de \mathcal{O}_X -module ainsi qu'un épimorphisme

$$\mathcal{L} \rightarrow \mathcal{F}'$$

On obtient ainsi une composition

$$\mathcal{L} \rightarrow \mathcal{F}' \subset \mathcal{F} \rightarrow \mathcal{G}$$

qui est un épimorphisme. ■

Lemme 6.2 : Soit \mathcal{K}_\bullet un complexe de faisceau quasi-cohérent sur X tel que chaque $H_i(\mathcal{K}_\bullet)$ est cohérent. Il existe un complexe de faisceau localement libre \mathcal{L}_\bullet de \mathcal{O}_X -module et un quasi-isomorphisme

$$\mathcal{L}_\bullet \xrightarrow{q.is.} \mathcal{K}_\bullet$$

Preuve : On construit \mathcal{L}_\bullet par récurrence sur $p \in \mathbb{N}$. On initialise avec le lemme 6.1 qui fournit un faisceau localement libre \mathcal{L}_0 de \mathcal{O}_X -module et un morphisme de faisceau

$$\mathcal{L}_0 \rightarrow \mathcal{K}_0$$

tels que la composition

$$\mathcal{L}_0 \rightarrow \mathcal{K}_0 \rightarrow H_0(\mathcal{K}_\bullet)$$

est un épimorphisme. Supposons à présent que l'on dispose d'un complexe de faisceau localement libre \mathcal{L}_\bullet de \mathcal{O}_X -module

$$0 \longrightarrow \mathcal{L}_p \longrightarrow \mathcal{L}_{p-1} \longrightarrow \cdots \longrightarrow \mathcal{L}_1 \longrightarrow \mathcal{L}_0 \longrightarrow 0$$

et d'un morphisme de complexe de faisceau de \mathcal{O}_X -module

$$f : \mathcal{L}_\bullet \rightarrow \mathcal{K}_\bullet$$

tel que le morphisme induit sur l'homologie

$$f_* : H_i(\mathcal{L}_\bullet) \rightarrow H_i(\mathcal{K}_\bullet)$$

est un isomorphisme pour tout $i < p$ et un épimorphisme pour $i = p$. Soit \mathcal{P} le tiré-en-arrière suivant

$$\begin{array}{ccc} \mathcal{P} & \xrightarrow{d'} & Z_p(\mathcal{L}_\bullet) \\ f' \downarrow & & \downarrow f \\ \mathcal{K}_{p+1} & \xrightarrow{d} & Z_p(\mathcal{K}_\bullet) \end{array}$$

On dispose d'un diagramme commutatif à lignes exactes

$$\begin{array}{ccccccc} \mathcal{P} & \xrightarrow{d'} & Z_p(\mathcal{L}_\bullet) & \xrightarrow{\mu} & \text{coker}(d') & \longrightarrow & 0 \\ f' \downarrow & & \downarrow f & & \downarrow \varphi & & \\ \mathcal{K}_{p+1} & \xrightarrow{d} & Z_p(\mathcal{K}_\bullet) & \xrightarrow{\nu} & H_p(\mathcal{K}_\bullet) & \longrightarrow & 0 \end{array}$$

où φ est donné par la propriété universelle du conoyau de d' :

$$(\nu \circ f) \circ d' = \nu \circ d \circ f' = 0$$

Notre hypothèse de récurrence implique que $\varphi \circ \mu = \nu \circ f = f_*$ est un épimorphisme. Donc φ est un épimorphisme. De plus, une chasse au diagramme sur les fibres permet de montrer que φ est un monomorphisme : si $\varphi(x) = 0$ alors on choisit y tel que $\mu(y) = x$

$$\begin{array}{ccc} y & \xrightarrow{\mu} & x \\ f \downarrow & & \downarrow \varphi \\ \cdot & \xrightarrow{\nu} & 0 \end{array}$$

puis on choisit z tel que $dz = f(y)$

$$\begin{array}{ccccccc} (z, y) & \xrightarrow{d'} & y & \xrightarrow{\mu} & x \\ f' \downarrow & & \downarrow f & & \downarrow \varphi \\ z & \xrightarrow{d} & dz & \xrightarrow{\nu} & 0 \end{array}$$

et on obtient $x = \mu \circ d'(z, y) = 0$. Ceci prouve que $\text{coker}(d') \simeq H_p(\mathcal{K}_\bullet)$ est un faisceau cohérent. Par conséquent, $\text{im}(d') = \ker(\mu)$ est un faisceau cohérent [1, Ch.II, Prop.5.7]. Utilisons à nouveau le lemme 6.1 : d'une part, on peut trouver un faisceau localement libre \mathcal{L}'_{p+1} sur X et un morphisme

$$\alpha : \mathcal{L}'_{p+1} \rightarrow \mathcal{P}$$

tels que $\text{im}(d' \circ \alpha) = \text{im}(d')$; et d'autre part un faisceau localement libre \mathcal{L}''_{p+1} sur X et un morphisme

$$\beta : \mathcal{L}''_{p+1} \rightarrow Z_{p+1}(\mathcal{K}_\bullet)$$

dont la composition par la projection $Z_{p+1}(\mathcal{K}_\bullet) \rightarrow H_{p+1}(\mathcal{K}_\bullet)$ est un épimorphisme. On pose alors $\mathcal{L}_{p+1} = \mathcal{L}'_{p+1} \oplus \mathcal{L}''_{p+1}$, puis on définit une différentielle

$$(d' \circ \alpha) \oplus 0 : \mathcal{L}_{p+1} \rightarrow Z_p(\mathcal{L}_\bullet)$$

et un morphisme

$$(f' \circ \alpha) \oplus \beta : \mathcal{L}_{p+1} \rightarrow \mathcal{K}_{p+1}$$

Par construction, on obtient un morphisme de complexe

$$\begin{array}{ccccccccc} 0 & \longrightarrow & \mathcal{L}_{p+1} & \xrightarrow{(d' \circ \alpha) \oplus 0} & \mathcal{L}_p & \longrightarrow & \cdots & \longrightarrow & \mathcal{L}_1 & \longrightarrow & \mathcal{L}_0 & \longrightarrow & 0 \\ & & \downarrow (f' \circ \alpha) \oplus \beta & & \downarrow f & & & & \downarrow f & & \downarrow f & & \\ \cdots & \longrightarrow & \mathcal{K}_{p+1} & \longrightarrow & \mathcal{K}_p & \longrightarrow & \cdots & \longrightarrow & \mathcal{K}_1 & \longrightarrow & \mathcal{K}_0 & \longrightarrow & 0 \end{array}$$

Puisque $\text{im}(d' \circ \alpha) = \text{im}(d') = \ker(\mu) = \ker(\varphi \circ \mu) = \ker(\nu \circ f)$, ce morphisme induit en homologie un isomorphisme

$$H_p(\mathcal{L}_\bullet) \simeq H_p(\mathcal{K}_\bullet)$$

Enfin, la construction de β implique que ce morphisme induit en homologie un épimorphisme

$$H_{p+1}(\mathcal{L}_\bullet) \rightarrow H_{p+1}(\mathcal{K}_\bullet)$$

car $\mathcal{L}''_{p+1} \subset Z_{p+1}(\mathcal{L}_\bullet) = H_{p+1}(\mathcal{L}_\bullet)$. On peut donc répéter cette construction à l'infini. ■

Lemme 6.3 : Sous les mêmes hypothèses, supposons \mathcal{L}'_\bullet et \mathcal{L}''_\bullet deux complexes de faisceau localement libre de \mathcal{O}_X -module quasi-isomorphes à \mathcal{K}_\bullet :

$$\mathcal{L}'_\bullet \xrightarrow{q.is.} \mathcal{K}_\bullet ; \mathcal{L}''_\bullet \xrightarrow{q.is.} \mathcal{K}_\bullet$$

Il existe un complexe de faisceau localement libre \mathcal{L}_\bullet de \mathcal{O}_X -module et un diagramme commutatif à homotopie près

$$\begin{array}{ccc} \mathcal{L}_\bullet & \xrightarrow{q.is.} & \mathcal{L}'_\bullet \\ q.is. \downarrow & & \downarrow q.is. \\ \mathcal{L}''_\bullet & \xrightarrow{q.is.} & \mathcal{K}_\bullet \end{array}$$

Preuve : Soit \mathcal{M}_\bullet le cône de l'identité de $\mathcal{K}_\bullet[1]$. Il arrive avec un épimorphisme

$$\mathcal{M}_\bullet \rightarrow \mathcal{K}_\bullet$$

Soit \mathcal{G}_\bullet le noyau du morphisme

$$\mathcal{L}'_\bullet \oplus \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet \rightarrow \mathcal{K}_\bullet$$

La suite exacte courte

$$0 \longrightarrow \mathcal{G}_\bullet \longrightarrow \mathcal{L}'_\bullet \oplus \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet \longrightarrow \mathcal{K}_\bullet \longrightarrow 0$$

induit une longue suite exacte homologique

$$\cdots \longrightarrow H_n(\mathcal{G}_\bullet) \longrightarrow H_n(\mathcal{L}'_\bullet) \oplus H_n(\mathcal{L}''_\bullet) \longrightarrow H_n(\mathcal{K}_\bullet) \xrightarrow{\partial} H_{n-1}(\mathcal{G}_\bullet) \longrightarrow \cdots$$

Par hypothèse, le morphisme

$$H_n(\mathcal{L}'_\bullet) \oplus H_n(\mathcal{L}''_\bullet) \longrightarrow H_n(\mathcal{K}_\bullet)$$

est surjectif et son noyau est canoniquement isomorphe à $H_n(\mathcal{L}'_\bullet)$ de telle sorte que le morphisme connectant ∂ est nul et que la composition par la projection canonique

$$\mathcal{G}_\bullet \rightarrow \mathcal{L}'_\bullet \oplus \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet \rightarrow \mathcal{L}'_\bullet$$

est un quasi-isomorphisme. Le lemme 6.2 permet de trouver un complexe de faisceau localement libre \mathcal{L}_\bullet de \mathcal{O}_X -module et un quasi-isomorphisme

$$\mathcal{L}_\bullet \xrightarrow{q.is.} \mathcal{G}_\bullet$$

En utilisant les projections canoniques

$$\mathcal{L}'_\bullet \oplus \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet \rightarrow \mathcal{L}'_\bullet ; \mathcal{L}'_\bullet \oplus \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet \rightarrow \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet$$

on construit un carré commutatif

$$\begin{array}{ccc} \mathcal{L}_\bullet & \xrightarrow{q.is.} & \mathcal{L}'_\bullet \\ \downarrow q.is. & & \downarrow q.is. \\ \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet & \xrightarrow{q.is.} & \mathcal{K}_\bullet \end{array}$$

Enfin, en utilisant l'inclusion composée à la projection

$$\mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet \rightarrow \mathcal{L}''_\bullet \rightarrow \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet$$

on obtient le carré commutatif à homotopie près recherché

$$\begin{array}{ccc} \mathcal{L}_\bullet & \xrightarrow{q.is.} & \mathcal{L}'_\bullet \\ \downarrow q.is. & & \downarrow q.is. \\ \mathcal{L}''_\bullet & \xrightarrow{q.is.} & \mathcal{K}_\bullet \end{array}$$

Pour le voir, il suffit de vérifier que le morphisme de complexe

$$\mathcal{L}_\bullet \rightarrow \mathcal{G}_\bullet \rightarrow \mathcal{L}'_\bullet \oplus \mathcal{L}''_\bullet \oplus \mathcal{M}_\bullet \rightarrow \mathcal{L}'_\bullet \oplus \mathcal{L}''_\bullet \rightarrow \mathcal{K}_\bullet$$

est homotope à l'application nulle. Pour ce faire, on utilise la contractibilité du cône \mathcal{M}_\bullet qui fournit une homotopie s

$$\begin{array}{ccccccc} \cdots & \longrightarrow & \mathcal{M}_{p+1} & \longrightarrow & \mathcal{M}_p & \longrightarrow & \mathcal{M}_{p-1} \longrightarrow \cdots \\ & \searrow s & \downarrow 1 & \searrow s & \downarrow 1 & \searrow s & \downarrow 1 \searrow s \\ \cdots & \longrightarrow & \mathcal{M}_{p+1} & \longrightarrow & \mathcal{M}_p & \longrightarrow & \mathcal{M}_{p-1} \longrightarrow \cdots \end{array}$$

et on obtient une homotopie S

$$\begin{array}{ccccccc} \cdots & \longrightarrow & \mathcal{L}_{p+1} & \longrightarrow & \mathcal{L}_p & \longrightarrow & \mathcal{L}_{p-1} \longrightarrow \cdots \\ & \searrow S & \downarrow 1 & \searrow S & \downarrow 1 & \searrow S & \downarrow 1 \searrow S \\ \cdots & \longrightarrow & \mathcal{K}_{p+1} & \longrightarrow & \mathcal{K}_p & \longrightarrow & \mathcal{K}_{p-1} \longrightarrow \cdots \end{array}$$

en prenant $S : \mathcal{L}_p \rightarrow \mathcal{M}_p \xrightarrow{s} \mathcal{M}_{p+1} \rightarrow \mathcal{K}_{p+1}$. ■

Lemme 6.4 : Soient $\mathcal{A} \rightarrow \mathcal{B}$ un morphisme de faisceau d'anneau sur un espace topologique et $f : \mathcal{F}_\bullet \rightarrow \mathcal{G}_\bullet$ un morphisme de complexe de faisceau plat de \mathcal{A} -module. Si f est un quasi-isomorphisme alors

$$1 \otimes f : \mathcal{B} \otimes_{\mathcal{A}} \mathcal{F}_\bullet \rightarrow \mathcal{B} \otimes_{\mathcal{A}} \mathcal{G}_\bullet$$

est un quasi-isomorphisme.

Preuve : Soit \mathcal{M}_\bullet le cône de f . Par hypothèse, \mathcal{M}_\bullet est exact et se décompose en suites exactes courtes

$$0 \longrightarrow \mathcal{Z}_1 \longrightarrow \mathcal{M}_1 \longrightarrow \mathcal{M}_0 \longrightarrow 0$$

$$0 \longrightarrow \mathcal{Z}_p \longrightarrow \mathcal{M}_p \longrightarrow \mathcal{Z}_{p-1} \longrightarrow 0$$

pour tout $p > 1$. Aussi, \mathcal{M}_\bullet est plat sur \mathcal{A} , donc \mathcal{Z}_p est plat sur \mathcal{A} pour tout $p > 0$ [4, Ex.3.2.2]. Ainsi, $Tor_1^{\mathcal{A}}(\mathcal{B}, \mathcal{Z}_p)$ pour tout $p > 0$ et on a des suites exactes courtes

$$0 \longrightarrow \mathcal{B} \otimes_{\mathcal{A}} \mathcal{Z}_1 \longrightarrow \mathcal{B} \otimes_{\mathcal{A}} \mathcal{M}_1 \longrightarrow \mathcal{B} \otimes_{\mathcal{A}} \mathcal{M}_0 \longrightarrow 0$$

$$0 \longrightarrow \mathcal{B} \otimes_{\mathcal{A}} \mathcal{Z}_p \longrightarrow \mathcal{B} \otimes_{\mathcal{A}} \mathcal{M}_p \longrightarrow \mathcal{B} \otimes_{\mathcal{A}} \mathcal{Z}_{p-1} \longrightarrow 0$$

pour tout $p > 1$, de sorte que $\mathcal{B} \otimes_{\mathcal{A}} \mathcal{M}_\bullet$, le cône de $(1 \otimes f)$, est exact. ■

On arrive au résultat le plus important de ce paragraphe. En corollaire, on obtiendra les théorèmes 2.1 et 2.5 dans le cas affine, mais on utilisera également ce résultat pour le cas général.

Proposition 6.5 : (1) Soit \mathcal{G}_\bullet un complexe de faisceau plat et quasi-cohérent de $\mathcal{O}_{X \times X}$ -module tel que $H_0(\mathcal{G}_\bullet) = \delta_* \mathcal{O}_X$ et $H_q(\mathcal{G}_\bullet) = 0$ pour tout $q \neq 0$. Pour tout faisceau \mathcal{F} de \mathcal{O}_X -module, on a un isomorphisme de δ -foncteur en \mathcal{F}

$$H^n(\mathcal{O}_X, \mathcal{F}) \simeq \mathbb{E}xt_{\mathcal{O}_X}^n(\delta_* \mathcal{G}_\bullet, \mathcal{F})$$

(2) Si de plus $H_q(\delta_* \mathcal{G}_\bullet)$ est localement libre pour tout q , alors les suites spectrales

$$Ext_{\mathcal{O}_X}^p(H_q(\delta_* \mathcal{G}_\bullet), \mathcal{F}) \Rightarrow \mathbb{E}xt_{\mathcal{O}_X}^{p+q}(\delta_* \mathcal{G}_\bullet, \mathcal{F})$$

$$H^p(X, \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})) \Rightarrow H^{p+q}(\mathcal{O}_X, \mathcal{F})$$

sont isomorphes.

Preuve : (1) Le lemme 6.2 donne un complexe de faisceau localement libre \mathcal{L}_\bullet de \mathcal{O}_X -module et un quasi-isomorphisme

$$\mathcal{L}_\bullet \xrightarrow{q.is.} \mathcal{G}_\bullet$$

Le foncteur δ^{-1} est exact et $\delta^* = (\mathcal{O}_X \otimes_{\delta^{-1} \mathcal{O}_{X \times X}} -) \circ \delta^{-1}$. D'après le lemme 6.4, on a un quasi-isomorphisme

$$\delta^* \mathcal{L}_\bullet \xrightarrow{q.is.} \delta^* \mathcal{G}_\bullet$$

En utilisant le corollaire 4.7 et le lemme 4.1, on obtient un isomorphisme naturel en \mathcal{F}

$$H^n(\mathcal{O}_X, \mathcal{F}) = \mathbb{E}xt_{\mathcal{O}_X}^n(\delta_* \mathcal{L}_\bullet, \mathcal{F}) \simeq \mathbb{E}xt_{\mathcal{O}_X}^n(\delta_* \mathcal{G}_\bullet, \mathcal{F})$$

qui, d'après le lemme 6.3, ne dépend pas du choix de

$$\mathcal{L}_\bullet \xrightarrow{q.is.} \mathcal{G}_\bullet$$

(2) Plus précisément, le lemme 4.1 donne un isomorphisme entre les suites spectrales

$$Ext_{\mathcal{O}_X}^p(H_q(\delta^*\mathcal{G}_\bullet), \mathcal{F}) \Rightarrow \mathbb{E}xt_{\mathcal{O}_X}^{p+q}(\delta^*\mathcal{G}_\bullet, \mathcal{F})$$

$$Ext_{\mathcal{O}_X}^p(H_q(\delta^*\mathcal{L}_\bullet), \mathcal{F}) \Rightarrow \mathbb{E}xt_{\mathcal{O}_X}^{p+q}(\delta^*\mathcal{L}_\bullet, \mathcal{F})$$

Cette dernière est, par le corollaire 5.6, isomorphe à la suite spectrale

$$H^p(X, \mathcal{E}xt_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{F})) \Rightarrow H^{p+q}(\mathcal{O}_X, \mathcal{F}) \blacksquare$$

On peut à présent démontrer les théorèmes 2.1 et 2.5 dans le cas où $X = \text{Spec } A$ est un schéma affine sur une algèbre de type fini sur un corps. La résolution plate (car projective) de A^e -module

$$B_\bullet(A) \rightarrow A$$

induit une résolution de faisceau plat quasi-cohérent de $\mathcal{O}_{X \times X}$ -module

$$\mathcal{B}_\bullet \rightarrow \delta_*\mathcal{O}_X$$

On peut lui appliquer la proposition 6.5. Pour conclure, il suffit de constater par le lemme 2.4 (4) et le lemme 4.1 que les suites spectrales

$$Ext_{\mathcal{O}_X}^p(H_q(\delta^*\mathcal{B}_\bullet), \mathcal{F}) \Rightarrow \mathbb{E}xt_{\mathcal{O}_X}^{p+q}(\delta^*\mathcal{B}_\bullet, \mathcal{F})$$

$$Ext_{\mathcal{O}_X}^p(\mathcal{H}_q, \mathcal{F}) \Rightarrow HH^{p+q}(X, \mathcal{F})$$

sont isomorphes.

7 Lemmes théoriques sur les faisceaux

Pour pouvoir aborder le théorème dans sa généralité, il nous faut aller plus loin dans les considérations géométriques. Dans ce paragraphe, on va mettre à profit la propriété de séparation des schémas, satisfaite en particulier par les schémas quasi-projectifs. L'idée principale est la suivante. Dans un schéma X séparé sur un schéma affine, l'intersection de deux ouverts affines est encore un ouvert affine [1, Ch.II, Ex.4.3]. Une conséquence importante est que pour tout ouvert affine U de X , l'inclusion $U \hookrightarrow X$ est un morphisme affine. Ce que l'on va constater, c'est que les morphismes affines ont les bonnes propriétés concernant les faisceaux quasi-cohérents et, s'ils sont plats, envoient par poussé-en-avant les faisceaux plats quasi-cohérents vers des faisceaux plats quasi-cohérents.

Commençons par une propriété des faisceaux associés aux préfaisceaux de module :

Lemme 7.1 : Soient R un préfaisceau d'anneau sur un espace topologique X , \mathcal{R} son faisceau associé, M un préfaisceau de R -module et \mathcal{M} son faisceau associé. Si pour tout ouvert U de X , $M(U)$ est plat sur $R(U)$, alors \mathcal{M} est plat sur \mathcal{R} .

Preuve : Puisque $\mathcal{M} \otimes_{\mathcal{R}} -$ est toujours exact à droite, on doit s'assurer que si $f : \mathcal{F} \rightarrow \mathcal{G}$ est un monomorphisme de faisceau de \mathcal{R} -module alors $1 \otimes f : \mathcal{M} \otimes_{\mathcal{R}} \mathcal{F} \rightarrow \mathcal{M} \otimes_{\mathcal{R}} \mathcal{G}$ est un monomorphisme de faisceau de \mathcal{R} -module. Par hypothèse, le morphisme de préfaisceau $1 \otimes f : M \otimes_R \mathcal{F} \rightarrow M \otimes_R \mathcal{G}$ est un monomorphisme. On a donc des monomorphismes sur les fibres. Le fait est que $M \otimes_R \mathcal{F}$ et $\mathcal{M} \otimes_{\mathcal{R}} \mathcal{F}$ ont les mêmes fibres quelque soit \mathcal{F} . Ainsi, le morphisme $1 \otimes f : \mathcal{M} \otimes_{\mathcal{R}} \mathcal{F} \rightarrow \mathcal{M} \otimes_{\mathcal{R}} \mathcal{G}$ est un monomorphisme sur les fibres et par conséquent un monomorphisme de faisceau de \mathcal{R} -module. ■

Intéressons-nous à présent aux morphismes affines de schéma :

Lemme 7.2 : Soit $f : X \rightarrow Y$ un morphisme plat et affine de schéma. Si \mathcal{F} est un faisceau plat et quasi-cohérent sur X alors $f_* \mathcal{F}$ est plat et quasi-cohérent sur Y .

Preuve : La question étant locale en Y , on est ramené à traiter le cas où $Y = \text{Spec } A$ est affine, et par hypothèse sur f , où $X = \text{Spec } B$ est affine. f est alors donné par un morphisme plat d'anneau $\varphi : A \rightarrow B$, et \mathcal{F} est associé à un B -module plat M . Dans ce cas, $f_* \mathcal{F}$ est associé au A -module M (dont la loi externe est donnée par φ), ce dernier étant plat sur A étant donné les isomorphismes de foncteur

$$M \otimes_A - \simeq (M \otimes_B B) \otimes_A - \simeq (M \otimes_B -) \circ (B \otimes_A -) \quad \blacksquare$$

Lemme 7.3 : Soit $f : X \rightarrow Y$ un morphisme affine de schéma. Le foncteur

$$f_* : q - \text{Coh}(X) \rightarrow q - \text{Coh}(Y)$$

est bien défini et exact.

Preuve : La question étant locale en Y , on est ramené une nouvelle fois au cas où $Y = \text{Spec } A$ est affine puis par hypothèse sur f , où $X = \text{Spec } B$ est affine. Dans cette configuration, les catégories $q - \text{Coh}(X)$ et $q - \text{Coh}(Y)$ sont respectivement équivalentes aux catégories $B - \text{mod}$ et $A - \text{mod}$, et f_* correspond alors au foncteur

$$B - \text{mod} \rightarrow A - \text{mod}$$

qui envoie un B -module M vers le A -module M induit, lequel est exact. ■

Lemme 7.4 : Pour tout carré cartésien de schéma

$$\begin{array}{ccc} X' & \xrightarrow{g'} & X \\ f' \downarrow & & \downarrow f \\ S' & \xrightarrow{g} & S \end{array}$$

tel que f est affine, il existe un isomorphisme de foncteur

$$g^* f_* \simeq f'_* g'^* : q - Coh(X) \rightarrow q - Coh(S')$$

Preuve : La transformation naturelle $g^* f_* \rightarrow f'_* g'^*$ existe dès que l'on a un tel carré commutatif. Pour la construire, on part de la co-unité puis on utilise l'adjonction :

$$1 \rightarrow g'_* g'^*$$

$$f_* \rightarrow f_* g'_* g'^* = g_* f'_* g'^*$$

$$g^* f_* \rightarrow f'_* g'^*$$

Pour prouver l'isomorphisme, on commence par le cas où $S = Spec A$ et $S' = Spec A'$ sont des schémas affines. Alors par hypothèse sur f , $X = Spec B$ est un schéma affine et puisque le carré est cartésien, $X' = Spec A' \otimes_A B$ est également un schéma affine. Dans cette configuration, les catégories $q - Coh(X)$ et $q - Coh(S')$ sont respectivement équivalentes aux catégories $B - mod$ et $A' - mod$ et la transformation naturelle

$$g^* f_* \rightarrow f'_* g'^*$$

correspond à l'isomorphisme de foncteur

$$A' \otimes_A - \simeq (A' \otimes_A B) \otimes_B - : B - mod \rightarrow A' - mod$$

Pour le cas général, on remarque que pour tout faisceau \mathcal{F} de \mathcal{O}_X -module, tout ouvert U de S et tout ouvert U' de S' tel que $g(U') \subset U$, on a $g'(f'^{-1}(U')) \subset f^{-1}(U)$ et les deux identifications suivantes

$$(g^* f_* \mathcal{F})|_{U'} = (g|_{U'})^* (f|_{f^{-1}(U)})_* (\mathcal{F}|_{f^{-1}(U)})$$

$$(f'_* g'^* \mathcal{F})|_{U'} = (f'|_{f'^{-1}(U')})_* (g'|_{f'^{-1}(U')})^* (\mathcal{F}|_{f^{-1}(U)})$$

La première identification découle du fait que $(g^{-1} f_* \mathcal{F})|_{U'}$ et $(g|_{U'})^{-1} (f|_{f^{-1}(U)})_* (\mathcal{F}|_{f^{-1}(U)})$ sont deux faisceaux associés au même préfaisceau

$$U' \supset V \mapsto \operatorname{colim}_{W \supset g(V)} \Gamma(f^{-1}(W), \mathcal{F}) = \operatorname{colim}_{U' \supset W \supset g(V)} \Gamma(f^{-1}(W), \mathcal{F}|_{f^{-1}(U)})$$

La deuxième identification provient du même raisonnement : pour tout ouvert V de U' ,

$$\Gamma(V, (f'_* g'^* \mathcal{F})|_{U'}) = \Gamma(f'^{-1}(V), g'^* \mathcal{F})$$

$$\Gamma(V, (f'|_{f'^{-1}(U')})_* (g'|_{f'^{-1}(U')})^* (\mathcal{F}|_{f^{-1}(U)})) = \Gamma(f'^{-1}(V), (g'|_{f'^{-1}(U')})^* (\mathcal{F}|_{f^{-1}(U)}))$$

et les deux faisceaux $g'^{-1} \mathcal{F}$ et $(g'|_{f'^{-1}(U')})^{-1} (\mathcal{F}|_{f^{-1}(U)})$ sont associés au même préfaisceau

$$f'^{-1}(U') \supset V \mapsto \operatorname{colim}_{W \supset g'(V)} \Gamma(W, \mathcal{F}) = \operatorname{colim}_{f^{-1}(U) \supset W \supset g'(V)} \Gamma(W, \mathcal{F}|_{f^{-1}(U)})$$

Ceci étant dit, si on fixe un ouvert affine U de S , alors pour tout ouvert affine U' de S' tel que $g(U') \subset U$, on a un carré cartésien de schéma

$$\begin{array}{ccc} f'^{-1}(U') & \xrightarrow{g'|_{f'^{-1}(U')}} & f^{-1}(U) \\ f'|_{f'^{-1}(U')} \downarrow & & \downarrow f|_{f^{-1}(U)} \\ U' & \xrightarrow{g|_{U'}} & U \end{array}$$

On se retrouve alors dans la première situation considérée et on obtient l'isomorphisme des faisceaux restreints sur U' :

$$(g^* f_* \mathcal{F})|_{U'} \simeq (f'_* g'^* \mathcal{F})|_{U'}$$

Pour conclure, on choisit un recouvrement \mathcal{U} de S par des ouverts affines, puis on recouvre $g^{-1}(U)$ par des ouverts affines de S' pour tout $U \in \mathcal{U}$. On obtient alors un recouvrement \mathcal{U}' de S' par des ouverts affines tels que l'image de chaque $U' \in \mathcal{U}'$ par g est contenu dans un $U \in \mathcal{U}$. On a ainsi l'isomorphisme sur un recouvrement de S' et donc sur S' . ■

On termine ce paragraphe par un résultat élémentaire qu'on pourrait aussi énoncer dans la catégorie des espaces topologiques :

Lemme 7.5 : Soient $f : X \rightarrow S$ un morphisme de schéma et U un ouvert de X . On considère le carré cartésien

$$\begin{array}{ccc} f^{-1}(U) & \xrightarrow{j} & X \\ g \downarrow & & \downarrow f \\ U & \xrightarrow{i} & S \end{array}$$

où i et j sont les inclusions et $g = f|_{f^{-1}(U)}$. Pour tout faisceau \mathcal{F} de \mathcal{O}_X -module,

$$i^* f_* \mathcal{F} = g_* j^* \mathcal{F}$$

Preuve : Puisque i et j sont des inclusions, les foncteurs i^* et j^* correspondent aux restrictions. En fait, la restriction sur U coïncide avec le foncteur i^{-1} et on a

$$i^* = (\mathcal{O}_U \otimes_{i^{-1}\mathcal{O}_X} -) \circ i^{-1}$$

$$\mathcal{O}_U = \mathcal{O}_X|_U = i^{-1}\mathcal{O}_X$$

Ainsi, pour tout ouvert V de U , on a

$$\Gamma(V, i^* f_* \mathcal{F}) = \Gamma(f^{-1}(V), \mathcal{F}) = \Gamma(g^{-1}(V), \mathcal{F}) = \Gamma(V, g_* j^* \mathcal{F}) \quad \blacksquare$$

8 Préfaisceau de faisceau

L'objectif de ce paragraphe est d'introduire le complexe de Čech associé à un préfaisceau de faisceau. Dans certaines configurations topologiques et algébriques, la cohomologie de Čech coïncide avec la cohomologie des faisceaux [1, Ch.III, Th.4.5]. Nous allons lui trouver un intérêt dans ce qu'il nous permettra de recoller, à quasi-isomorphisme près, une famille de complexe de faisceau indexée sur les ouverts affines de notre schéma en un double complexe de faisceau. Pour pouvoir utiliser les propriétés du complexe de Čech à plusieurs reprises et dans des situations différentes, on préférera dans un premier temps travailler avec un préfaisceau à valeurs dans une catégorie abélienne arbitraire.

Soient X un espace topologique, \mathcal{A} une catégorie abélienne (complète) et Q un préfaisceau sur X à valeurs dans \mathcal{A} dont les restrictions seront toutes notées par ρ . À tout recouvrement ouvert $\mathcal{U} = (U_i)_{i \in I}$ de X , on associe un complexe de cochaîne de \mathcal{A} , noté $C^\bullet(\mathcal{U}, Q)$ comme suit. On choisit un bon ordre \leq sur I , et on considère pour tout entier naturel n le produit

$$C^n(\mathcal{U}, Q) = \prod_{i_0 < \dots < i_n} Q(U_{i_0 \dots i_n})$$

où $U_{i_0 \dots i_n} = U_{i_0} \cap \dots \cap U_{i_n}$. Considérons les projections canoniques

$$p_{i_0 \dots i_n} : C^n(\mathcal{U}, Q) \rightarrow Q(U_{i_0 \dots i_n})$$

et définissons pour tout entier ν compris entre 0 et $n+1$ les applications

$$\delta_\nu : C^n(\mathcal{U}, Q) \rightarrow C^{n+1}(\mathcal{U}, Q)$$

par $p_{i_0 \dots i_{n+1}} \circ \delta_\nu = \rho \circ p_{i_0 \dots \widehat{i_\nu} \dots i_{n+1}}$. On peut alors définir les différentielles

$$d = \sum_{\nu=0}^{n+1} (-1)^\nu \delta_\nu : C^n(\mathcal{U}, Q) \rightarrow C^{n+1}(\mathcal{U}, Q)$$

On dispose aussi d'une application définie à partir des restrictions

$$\varepsilon : Q(X) \rightarrow C^0(\mathcal{U}, Q)$$

c'est à dire $p_i \circ \varepsilon = \rho$. On peut vérifier que l'on obtient bien un complexe dans \mathcal{A}

$$0 \longrightarrow Q(X) \xrightarrow{\varepsilon} C^0(\mathcal{U}, Q) \xrightarrow{d} C^1(\mathcal{U}, Q) \xrightarrow{d} C^2(\mathcal{U}, Q) \xrightarrow{d} \dots$$

On voudrait s'assurer que ce complexe est indépendant du choix de l'ordre sur I . Pour ce faire, on étend les projections en posant

$$p_{i_0 \dots i_n} = 0$$

si pour deux indices $\mu \neq \nu$ on a $i_\mu = i_\nu$ et

$$p_{\sigma(i_0) \dots \sigma(i_n)} = \text{sgn}(\sigma) p_{i_0 \dots i_n}$$

pour toute permutation σ de $\{i_0, \dots, i_n\}$. Alors si on se donne un raffinement $\mathcal{V} = (V_j)_{j \in J}$ de \mathcal{U} , on peut choisir une application $\alpha : J \rightarrow I$ telle que $V_j \subset U_{\alpha(j)}$ pour ensuite définir un morphisme de complexe qui commute avec ε

$$\alpha^* : C^\bullet(\mathcal{U}, Q) \rightarrow C^\bullet(\mathcal{V}, Q)$$

par $p_{j_0 \dots j_n} \circ \alpha = \rho \circ p_{\alpha(j_0) \dots \alpha(j_n)}$. Lorsque $\mathcal{V} = \mathcal{U}$ avec un autre ordre, on peut prendre $\alpha = \text{id}_I$ et α^* est alors un isomorphisme.

On énonce à présent un résultat qui justifiera plus tard la généralité choisie pour notre construction du complexe de Čech.

Lemme 8.1 : Si $X \in \mathcal{U}$ alors le complexe

$$0 \longrightarrow Q(X) \xrightarrow{\varepsilon} C^0(\mathcal{U}, Q) \xrightarrow{d} C^1(\mathcal{U}, Q) \xrightarrow{d} C^2(\mathcal{U}, Q) \xrightarrow{d} \dots$$

est exact.

Preuve : On choisit sur I un ordre tel que $\min I = 0$ et $X = U_0$. On peut alors expliciter une homotopie S entre l'identité et l'application nulle

$$\begin{array}{ccccccccccc} 0 & \longrightarrow & Q(X) & \xrightarrow{\varepsilon} & C^0(\mathcal{U}, Q) & \xrightarrow{d} & C^1(\mathcal{U}, Q) & \xrightarrow{d} & C^2(\mathcal{U}, Q) & \xrightarrow{d} & \dots \\ & & \downarrow 1 & \swarrow S & \downarrow 1 & \swarrow S & \downarrow 1 & \swarrow S & \downarrow 1 & \swarrow S & \\ 0 & \longrightarrow & Q(X) & \xrightarrow{\varepsilon} & C^0(\mathcal{U}, Q) & \xrightarrow{d} & C^1(\mathcal{U}, Q) & \xrightarrow{d} & C^2(\mathcal{U}, Q) & \xrightarrow{d} & \dots \end{array}$$

définie par $p_{i_1 \dots i_n} \circ S = \begin{cases} 0 & \text{si } i_1 = 0 \\ p_{0i_1 \dots i_n} & \text{si } i_1 > 0 \end{cases}$ pour tout $n \geq 1$ et $S = p_0$ pour $n = 0$. ■

Ce lemme va s'avérer très utile lorsque \mathcal{A} sera la catégorie des faisceaux ou des complexes de faisceau sur un schéma. Le fait que l'exactitude d'une suite de faisceau peut s'étudier localement va nous permettre de s'y ramener en permanence. Quittons les généralités et concentrons-nous à présent sur les faisceaux.

Soit \mathcal{F} un faisceau sur X . On construit un préfaisceau $P_X \mathcal{F}$ sur X à valeurs dans les faisceaux sur X en posant pour tout ouvert U de X , $P_X \mathcal{F}(U) = i_*(\mathcal{F}|_U)$ où $i : U \hookrightarrow X$ désigne l'inclusion. Les restrictions sont données pour toute inclusion $V \subset U$ et tout ouvert W par

$$\Gamma(W, P_X \mathcal{F}(U)) = \Gamma(W \cap U, \mathcal{F}) \rightarrow \Gamma(W \cap V, \mathcal{F}) = \Gamma(W, P_X \mathcal{F}(V))$$

Lemme 8.2 : Pour tout ouvert U et V de X , $P_X \mathcal{F}(U)|_V = P_V \mathcal{F}(U \cap V)$.

Preuve : On applique le lemme 7.5 au carré cartésien d'inclusion

$$\begin{array}{ccc} U \cap V & \xrightarrow{j} & U \\ g \downarrow & & \downarrow f \\ V & \xrightarrow{i} & X \end{array}$$

et on obtient $P_X \mathcal{F}(U)|_V = i^* f_* f^* \mathcal{F} = g_* j^* f^* \mathcal{F} = P_V \mathcal{F}(U \cap V)$. ■

Supposons à présent que \mathcal{U} est fini. Avec cette hypothèse, le complexe de Čech $C^\bullet(\mathcal{U}, Q)$ est toujours borné en haut, car $C^n(\mathcal{U}, Q)$ est un produit vide pour tout $n \geq |\mathcal{U}|$.

Corollaire 8.3 : Pour tout ouvert V de X , $C^\bullet(\mathcal{U}, P_X \mathcal{F})|_V = C^\bullet(\mathcal{U} \cap V, P_V \mathcal{F})$.

Preuve : On a vu au lemme 7.5 que la restriction sur V est donnée par le foncteur adjoint à gauche i^* où $i : V \hookrightarrow X$ désigne l'inclusion. Cette opération commute donc avec toutes les colimites et en particulier avec les produits finis d'une catégorie abélienne. Ainsi,

$$C^n(\mathcal{U}, P_X \mathcal{F})|_V = \prod_{i_0 < \dots < i_n} P_X \mathcal{F}(U_{i_0 \dots i_n})|_V = \prod_{i_0 < \dots < i_n} P_V \mathcal{F}(U_{i_0 \dots i_n} \cap V) = C^n(\mathcal{U} \cap V, P_V \mathcal{F}) \quad \blacksquare$$

Lemme 8.4 : Le complexe de faisceau

$$0 \longrightarrow \mathcal{F} \longrightarrow C^0(\mathcal{U}, P_X \mathcal{F}) \longrightarrow C^1(\mathcal{U}, P_X \mathcal{F}) \longrightarrow C^2(\mathcal{U}, P_X \mathcal{F}) \longrightarrow \dots$$

est exact.

Preuve : L'exactitude d'un complexe de faisceau est équivalente à l'exactitude locale sur un recouvrement. Puisque \mathcal{U} recouvre X , on peut vérifier l'exactitude seulement sur les ouverts $V \in \mathcal{U}$. Le fait est que $V \in \mathcal{U} \cap V$, donc le lemme 8.1 nous dit que le complexe

$$0 \longrightarrow P_V \mathcal{F} \longrightarrow C^0(\mathcal{U} \cap V, P_V \mathcal{F}) \longrightarrow C^1(\mathcal{U} \cap V, P_V \mathcal{F}) \longrightarrow C^2(\mathcal{U} \cap V, P_V \mathcal{F}) \longrightarrow \dots$$

est exact. Mais $P_V \mathcal{F} = \mathcal{F}|_V$ donc le résultat découle du corollaire 8.3. ■

Corollaire 8.5 : Si \mathcal{F}^\bullet est un complexe de faisceau sur X alors

$$\varepsilon : \mathcal{F}^\bullet \rightarrow C^\bullet(\mathcal{U}, P_X \mathcal{F}^\bullet)$$

est un quasi-isomorphisme.

Preuve : En regardant \mathcal{F}^\bullet comme un double complexe concentrée dans la ligne indicée par 0, les morphismes $\varepsilon : \mathcal{F}^p \rightarrow C^\bullet(\mathcal{U}, P_X \mathcal{F}^p)$ induisent un morphisme de double complexe

$$\varepsilon : \mathcal{F}^\bullet \rightarrow C^\bullet(\mathcal{U}, P_X \mathcal{F}^\bullet)$$

Puisque \mathcal{U} est fini, le double complexe $C^\bullet(\mathcal{U}, P_X \mathcal{F}^\bullet)$ est borné (i.e. ses diagonales n'ont qu'un certain nombre d'objet non nul) et on peut calculer sa cohomologie à partir de suite spectrale. Lorsque l'on filtre le double complexe

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \uparrow & & \uparrow & & \\ \dots & \longrightarrow & C^{q+1}(\mathcal{U}, P_X \mathcal{F}^p) & \longrightarrow & C^{q+1}(\mathcal{U}, P_X \mathcal{F}^{p+1}) & \longrightarrow & \dots \\ & & \uparrow & & \uparrow & & \\ \dots & \longrightarrow & C^q(\mathcal{U}, P_X \mathcal{F}^p) & \longrightarrow & C^q(\mathcal{U}, P_X \mathcal{F}^{p+1}) & \longrightarrow & \dots \\ & & \uparrow & & \uparrow & & \\ & & \vdots & & \vdots & & \end{array}$$

selon les colonnes, on obtient un isomorphisme à la page E^1 d'après le lemme 8.4, car la cohomologie verticale est celle du complexe de Čech $C^\bullet(\mathcal{U}, P_X \mathcal{F}^p)$. ■

On termine ce paragraphe par un résultat que l'on utilisera lors de la démonstration du théorème générale :

Lemme 8.6 : Soit $M^\bullet \rightarrow N^\bullet$ un morphisme de complexe de préfaisceau sur X . Si pour tout ouvert U de X , c'est un quasi-isomorphisme sur les sections $M^\bullet(U) \xrightarrow{q.is.} N^\bullet(U)$ alors le morphisme de double complexe induit est un quasi-isomorphisme :

$$C^\bullet(\mathcal{U}, M^\bullet) \xrightarrow{q.is.} C^\bullet(\mathcal{U}, N^\bullet)$$

Preuve : Lorsqu'on filtre selon les colonnes le double complexe

$$\begin{array}{ccccccc}
 & & \vdots & & \vdots & & \\
 & & \uparrow & & \uparrow & & \\
 \cdots & \longrightarrow & C^p(\mathcal{U}, M^{q+1}) & \longrightarrow & C^{p+1}(\mathcal{U}, M^{q+1}) & \longrightarrow & \cdots \\
 & & \uparrow & & \uparrow & & \\
 \cdots & \longrightarrow & C^p(\mathcal{U}, M^q) & \longrightarrow & C^{p+1}(\mathcal{U}, M^q) & \longrightarrow & \cdots \\
 & & \uparrow & & \uparrow & & \\
 & & \vdots & & \vdots & &
 \end{array}$$

on obtient un isomorphisme à la page E^1 : puisque \mathcal{U} est fini, les produits sont finis et commutent avec la cohomologie, et ainsi la cohomologie verticale est

$$H_v^q(C^p(\mathcal{U}, M^\bullet)) = \prod_{i_0 < \cdots < i_p} H^q(M^\bullet(U_{i_0 \cdots i_p})) = C^p(\mathcal{U}, H^q(M^\bullet)) \blacksquare$$

9 Technique de recollement de Čech

Soit X un schéma quasi-compact et séparé sur un corps. Pour chaque ouvert affine U de X , on se donne un complexe $\mathcal{S}_{\bullet,U}$ de faisceau de \mathcal{O}_U -module. Supposons que pour chaque inclusion $V \subset U$ d'ouverts affines de X , on dispose d'un morphisme de complexe de faisceau de \mathcal{O}_V -module

$$\rho_{UV} : \mathcal{S}_{\bullet,U}|_V \rightarrow \mathcal{S}_{\bullet,V}$$

satisfaisant pour tout ouvert affine $W \subset V \subset U$ de X les relations suivantes :

$$\begin{aligned} \rho_{UU} &= id_{\mathcal{S}_{\bullet,U}} \\ \rho_{UW} &= \rho_{VW} \circ (\rho_{UV})|_W \end{aligned}$$

Concrètement, c'est une donnée de recollement de faisceau, au fait près que les ρ_{UV} ne sont à priori pas des isomorphismes. Si chaque ρ_{UV} est un isomorphisme, alors il existe un complexe \mathcal{S}_{\bullet} de faisceau de \mathcal{O}_X -module et des isomorphismes de complexe

$$\sigma_U : \mathcal{S}_{\bullet}|_U \rightarrow \mathcal{S}_{\bullet,U}$$

tels que $\rho_{U,U \cap V} \circ \sigma_U = \rho_{V,U \cap V} \circ \sigma_V$ sur $U \cap V$ pour tout ouvert affine U et V de X [1, Ch.II, Ex.1.22]. Plus généralement, en supposant que chaque ρ_{UV} est un quasi-isomorphisme, on va construire des préfaisceaux $P_U \mathcal{S}_{\bullet}$ de complexe de faisceau tel que pour tout recouvrement fini \mathcal{U} de X par des ouverts affines et tout ouvert affine V de X , on a des quasi-isomorphismes

$$\mathcal{S}_{\bullet,V} \xrightarrow{q.is.} C^{\bullet}(\mathcal{U} \cap V, P_V \mathcal{S}_{\bullet}) \xleftarrow{q.is.} C^{\bullet}(\mathcal{U}, P_X \mathcal{S}_{\bullet})|_V$$

Le faisceau $C^{\bullet}(\mathcal{U}, P_X \mathcal{S}_{\bullet})$ jouera alors le rôle de recollement à quasi-isomorphisme près. Cette construction, appliquée au cas particulier $\mathcal{S}_{\bullet,U} = C_{\bullet}(\Gamma(U, \mathcal{O}_U))^{\sim}$, nous permettra de démontrer le théorème général.

Pour tout ouvert affine U de X , soit $P_X \mathcal{S}_{\bullet}(U) = j_* \mathcal{S}_{\bullet,U}$ où $j : U \hookrightarrow X$ désigne l'inclusion. Si on a deux ouverts affines $V \subset U$, on peut définir une restriction

$$P_X \mathcal{S}_{\bullet}(U) \rightarrow P_X \mathcal{S}_{\bullet}(V)$$

donnée pour tout ouvert W de X par

$$\Gamma(W, P_X \mathcal{S}_{\bullet}(U)) = \Gamma(W \cap U, \mathcal{S}_{\bullet,U}) \rightarrow \Gamma(W \cap V, \mathcal{S}_{\bullet,U}|_V) \xrightarrow{\rho_{UV}} \Gamma(W \cap V, \mathcal{S}_{\bullet,V}) = \Gamma(W, P_X \mathcal{S}_{\bullet}(V))$$

Ceci fait de $P_X \mathcal{S}_{\bullet}$ un préfaisceau sur X de complexe de faisceau de \mathcal{O}_X -module. En fait, si les complexes $\mathcal{S}_{\bullet,U}$ proviennent d'un même complexe \mathcal{S}_{\bullet} , c'est-à-dire si $\mathcal{S}_{\bullet,U} = \mathcal{S}_{\bullet}|_U$, et si les ρ_{UV} correspondent aux restrictions de \mathcal{S}_{\bullet} , alors cette construction correspond à celle du paragraphe précédent. On va pouvoir adapter les énoncés à cette situation.

Lemme 9.1 : Si chaque $\mathcal{S}_{\bullet,U}$ est quasi-cohérent et si chaque ρ_{UV} est un quasi-isomorphisme alors on a des quasi-isomorphismes naturels

$$P_X \mathcal{S}_{\bullet}(U)|_V \xrightarrow{q.is.} P_V \mathcal{S}_{\bullet}(U \cap V)$$

Preuve : Le lemme 7.5 appliqué au carré cartésien d'inclusion

$$\begin{array}{ccc} U \cap V & \xrightarrow{i'} & U \\ j' \downarrow & & \downarrow j \\ V & \xrightarrow{i} & X \end{array}$$

permet d'obtenir fonctoriellement le morphisme suivant

$$P_X \mathcal{S}_\bullet(U)|_V = i^* j_* \mathcal{S}_\bullet U = j'_* i'^* \mathcal{S}_\bullet U = j'_* (\mathcal{S}_\bullet U|_{U \cap V}) \xrightarrow{j'_* \rho_{U, U \cap V}} P_V \mathcal{S}_\bullet(U \cap V)$$

L'hypothèse de séparation sur X montre que j' est un morphisme affine de schéma. Ainsi, d'après le lemme 7.3, j'_* est exact et $j'_* \rho_{U, U \cap V}$ est un quasi-isomorphisme. ■

Soit \mathcal{U} un recouvrement fini de X par des ouverts affines. Cette propriété de finitude permet d'utiliser les suites spectrales pour calculer l'homologie du double complexe $C^\bullet(\mathcal{U}, P_X \mathcal{S}_\bullet)$:

Corollaire 9.2 : Sous les mêmes hypothèses, on a un quasi-isomorphisme naturel

$$C^\bullet(\mathcal{U}, P_X \mathcal{S}_\bullet)|_V \xrightarrow{q.is.} C^\bullet(\mathcal{U} \cap V, P_V \mathcal{S}_\bullet)$$

Preuve : Le morphisme est donné par le lemme 9.1 :

$$C^n(\mathcal{U}, P_X \mathcal{S}_\bullet)|_V = \prod_{i_0 < \dots < i_n} P_X \mathcal{S}_\bullet(U_{i_0 \dots i_n})|_V \xrightarrow{q.is.} \prod_{i_0 < \dots < i_n} P_V \mathcal{S}_\bullet(U_{i_0 \dots i_n} \cap V) = C^n(\mathcal{U} \cap V, P_V \mathcal{S}_\bullet)$$

Si on filtre le double complexe

$$\begin{array}{ccccccc} & & \vdots & & \vdots & & \\ & & \downarrow & & \downarrow & & \\ \dots & \longrightarrow & C^p(\mathcal{U}, P_X \mathcal{S}_q)|_V & \longrightarrow & C^{p+1}(\mathcal{U}, P_X \mathcal{S}_q)|_V & \longrightarrow & \dots \\ & & \downarrow & & \downarrow & & \\ \dots & \longrightarrow & C^p(\mathcal{U}, P_X \mathcal{S}_{q-1})|_V & \longrightarrow & C^{p+1}(\mathcal{U}, P_X \mathcal{S}_{q-1})|_V & \longrightarrow & \dots \\ & & \downarrow & & \downarrow & & \\ & & \vdots & & \vdots & & \end{array}$$

selon les colonnes, on obtient un isomorphisme à la page E_1 , car l'homologie verticale est

$$H_q^v(C^p(\mathcal{U}, P_X \mathcal{S}_\bullet)|_V) = \prod_{i_0 < \dots < i_n} H_q(P_X \mathcal{S}_\bullet(U_{i_0 \dots i_n})|_V) \quad \blacksquare$$

Pour chaque ouvert affine V de X , on dispose d'une augmentation

$$\varepsilon : P_V \mathcal{S}_\bullet(V) = \mathcal{S}_{\bullet V} \rightarrow C^\bullet(\mathcal{U} \cap V, P_V \mathcal{S}_\bullet)$$

introduite au paragraphe précédent. On se retrouve alors dans une situation analogue à celle du lemme 8.4, où l'on avait construit une résolution de faisceau.

Lemme 9.3 : Sous les mêmes hypothèses, on a une résolution de complexe de faisceau

$$0 \longrightarrow \mathcal{S}_{\bullet V} \longrightarrow C^0(\mathcal{U} \cap V, P_V \mathcal{S}_\bullet) \longrightarrow C^1(\mathcal{U} \cap V, P_V \mathcal{S}_\bullet) \longrightarrow \dots$$

Preuve : Puisque l'on travaille avec des faisceaux, on peut vérifier l'exactitude seulement sur un recouvrement, typiquement sur chaque $W \in \mathcal{U} \cap V$. On a alors un carré commutatif

$$\begin{array}{ccc} \mathcal{S}_{\bullet V}|_W & \longrightarrow & C^\bullet(\mathcal{U} \cap V, P_V \mathcal{S}_\bullet)|_W \\ \downarrow q.is. & & \downarrow q.is. \\ \mathcal{S}_{\bullet W} & \longrightarrow & C^\bullet(\mathcal{U} \cap W, P_W \mathcal{S}_\bullet) \end{array}$$

de sorte que l'exactitude du complexe

$$0 \longrightarrow \mathcal{S}_{\bullet V}|_W \longrightarrow C^0(\mathcal{U} \cap V, P_V \mathcal{S}_{\bullet})|_W \longrightarrow C^1(\mathcal{U} \cap V, P_V \mathcal{S}_{\bullet})|_W \longrightarrow \dots$$

est équivalente à l'exactitude du complexe

$$0 \longrightarrow \mathcal{S}_{\bullet W} \longrightarrow C^0(\mathcal{U} \cap W, P_W \mathcal{S}_{\bullet}) \longrightarrow C^1(\mathcal{U} \cap W, P_W \mathcal{S}_{\bullet}) \longrightarrow \dots$$

Ce dernier est exact d'après le lemme 8.1, car $W \in \mathcal{U} \cap W$ et $P_W \mathcal{S}_{\bullet}(W) = \mathcal{S}_{\bullet W}$. ■

Considérons à présent le cas où $\mathcal{S}_{\bullet U}$ est le faisceau quasi-cohérent sur U associé au complexe de $\Gamma(U, \mathcal{O}_U)$ -module $C_{\bullet}(\Gamma(U, \mathcal{O}_U))$. Les ρ_{UV} sont donnés par les applications

$$\Gamma(V, \mathcal{O}_V) \otimes_{\Gamma(U, \mathcal{O}_U)} C_{\bullet}(\Gamma(U, \mathcal{O}_U)) \rightarrow C_{\bullet}(\Gamma(V, \mathcal{O}_V))$$

Ce sont des quasi-isomorphismes, comme cela a pu être remarqué au paragraphe 2 : une immersion ouverte de schéma affine $V \hookrightarrow U$ induit un morphisme plat

$$\Gamma(U, \mathcal{O}_U) \rightarrow \Gamma(V, \mathcal{O}_V)$$

Grâce aux propriétés de l'homologie de Hochschild, on obtient

$$\begin{aligned} H_n(\Gamma(V, \mathcal{O}_V) \otimes_{\Gamma(U, \mathcal{O}_U)} C_{\bullet}(\Gamma(U, \mathcal{O}_U))) &\simeq \Gamma(V, \mathcal{O}_V) \otimes_{\Gamma(U, \mathcal{O}_U)} H_n(C_{\bullet}(\Gamma(U, \mathcal{O}_U))) \\ &= \Gamma(V, \mathcal{O}_V) \otimes_{\Gamma(U, \mathcal{O}_U)} HH_n(\Gamma(U, \mathcal{O}_U)) \simeq HH_n(\Gamma(V, \mathcal{O}_V)) = H_n(C_{\bullet}(\Gamma(V, \mathcal{O}_V))) \end{aligned}$$

On peut donc utiliser les résultats précédents à ce cas particulier. En fait, ces faisceaux $\mathcal{S}_{\bullet U}$ ressemblent aux faisceaux restreints $\mathcal{C}_{\bullet}|_U$. Plus précisément, on a un morphisme

$$\mathcal{S}_{\bullet U} \rightarrow \mathcal{C}_{\bullet}|_U$$

donné sur les ouverts principaux par les applications canoniques

$$\Gamma(U, \mathcal{O}_U)_s \otimes_{\Gamma(U, \mathcal{O}_U)} C_{\bullet}(\Gamma(U, \mathcal{O}_U)) \rightarrow C_{\bullet}(\Gamma(U, \mathcal{O}_U)_s)$$

pour tout $s \in \Gamma(U, \mathcal{O}_U)$. C'est un quasi-isomorphisme, comme le montre le lemme 2.4 (4) appliqué à U , en constatant que l'isomorphisme de complexe de $\Gamma(U, \mathcal{O}_U)$ -module

$$\Gamma(U, \mathcal{O}_U) \otimes_{\Gamma(U, \mathcal{O}_U)^e} B_{\bullet}(\Gamma(U, \mathcal{O}_U)) \simeq C_{\bullet}(\Gamma(U, \mathcal{O}_U))$$

induit un isomorphisme de faisceau quasi-cohérent

$$\delta_U^*(B_{\bullet}(\Gamma(U, \mathcal{O}_U))^{\sim}) \simeq \mathcal{S}_{\bullet U}$$

Tout ceci nous amène au dernier résultat de ce paragraphe.

Lemme 9.4 : $C^{\bullet}(\mathcal{U}, P_X \mathcal{S}_{\bullet}) \xrightarrow{q.is.} C^{\bullet}(\mathcal{U}, P_X \mathcal{C}_{\bullet})$

Preuve : Puisqu'il s'agit d'un morphisme de faisceau, on peut démontrer l'énoncé sur un recouvrement, à savoir sur chaque $V \in \mathcal{U}$. On a alors le diagramme commutatif suivant

$$\begin{array}{ccc} C^{\bullet}(\mathcal{U}, P_X \mathcal{S}_{\bullet})|_V & \longrightarrow & C^{\bullet}(\mathcal{U}, P_X \mathcal{C}_{\bullet})|_V \\ \downarrow q.is. & & \downarrow = \\ C^{\bullet}(\mathcal{U}, P_X \mathcal{S}_{\bullet})|_V & \longrightarrow & C^{\bullet}(\mathcal{U}, P_X \mathcal{C}_{\bullet})|_V \\ \uparrow q.is. & & \uparrow q.is. \\ \mathcal{S}_{\bullet V} & \xrightarrow{q.is.} & \mathcal{C}_{\bullet}|_V \end{array}$$

L'égalité et les quasi-isomorphismes verticaux se justifient respectivement (de gauche à droite puis de haut en bas) par le corollaire 9.2, le lemme 8.2, le lemme 9.3 et le corollaire 8.5. On obtient le résultat voulu. ■

10 Preuve du théorème

Soit X un schéma quasi-projectif sur un corps. Pour chaque ouvert affine $U = \text{Spec } A$ de X , on peut considérer $\mathcal{B}_{\bullet U}$ le faisceau quasi-cohérent sur $U \times U = \text{Spec } A^e$ associé au complexe de A^e -module $B_{\bullet}(A)$. On peut aussi considérer $\mathcal{S}_{\bullet U}$ le faisceau quasi-cohérent sur U associé au complexe de A -module $C_{\bullet}(A)$. Comme il a été remarqué précédemment, ces deux objets sont liés par un isomorphisme de complexe de faisceau de \mathcal{O}_U -module

$$\mathcal{S}_{\bullet U} \simeq \delta_U^* \mathcal{B}_{\bullet U}$$

On va définir un préfaisceau \mathcal{E}_{\bullet} sur les ouverts affines de X à valeurs dans les faisceaux quasi-cohérents de $\mathcal{O}_{X \times X}$ -module.

Pour tout ouvert affine U de X , on note $\mathcal{E}_{\bullet}(U) = i_* \mathcal{B}_{\bullet U}$ où $i : U \times U \hookrightarrow X \times X$ désigne l'inclusion. Pour tout ouvert affine $V \in U$, on dispose d'une restriction

$$\Gamma(U, \mathcal{O}_U) \rightarrow \Gamma(V, \mathcal{O}_V)$$

qui induit canoniquement un morphisme de complexe de $\Gamma(U, \mathcal{O}_U)$ -module

$$B_{\bullet}(\Gamma(U, \mathcal{O}_U)) \rightarrow B_{\bullet}(\Gamma(V, \mathcal{O}_V))$$

et qui par suite induit un morphisme de faisceau de $\mathcal{O}_{U \times U}$ -module

$$\mathcal{B}_{\bullet U} \rightarrow i'_* \mathcal{B}_{\bullet V}$$

où $i' : V \times V \hookrightarrow U \times U$ désigne l'inclusion. En appliquant i_* on obtient une restriction

$$\mathcal{E}_{\bullet}(U) = i_* \mathcal{B}_{\bullet U} \rightarrow i_* i'_* \mathcal{B}_{\bullet V} = \mathcal{E}_{\bullet}(V)$$

qui fait de \mathcal{E}_{\bullet} un préfaisceau.

Choisissons un recouvrement fini \mathcal{U} de X par des ouverts affines. Considérons ensuite $\mathcal{F}_{\bullet} = C^{\bullet}(\mathcal{U}, \mathcal{E}_{\bullet})$ et \mathcal{F}_{\bullet} son complexe total, qui est borné en bas. Chaque $\mathcal{B}_{\bullet U}$ est un complexe de faisceau quasi-cohérent et plat de $\mathcal{O}_{U \times U}$ -module. D'après le lemme 7.2, chaque $\mathcal{E}_{\bullet}(U) = i_* \mathcal{B}_{\bullet U}$, et par suite \mathcal{F}_{\bullet} , est un complexe de faisceau quasi-cohérent et plat de $\mathcal{O}_{X \times X}$ -module. Enfin, la résolution de A^e -module

$$B_{\bullet}(A) \rightarrow A$$

induit une résolution de faisceau quasi-cohérent de $\mathcal{O}_{U \times U}$ -module

$$\mathcal{B}_{\bullet U} \rightarrow \delta_{U*} \mathcal{O}_U$$

et en appliquant i_* , le lemme 7.3 montre que l'on obtient un quasi-isomorphisme

$$\mathcal{E}_{\bullet}(U) \xrightarrow{q.is.} i_* \delta_{U*} \mathcal{O}_U = \delta_* j_* \mathcal{O}_U = \delta_* P_X \mathcal{O}_X(U)$$

où $j : U \hookrightarrow X$ désigne l'inclusion et où l'on regarde $\delta_* P_X \mathcal{O}_X(U)$ comme un complexe concentré en 0. D'après le lemme 8.6,

$$\mathcal{F}_{\bullet} = C^{\bullet}(\mathcal{U}, \mathcal{E}_{\bullet}) \xrightarrow{q.is.} C^{\bullet}(\mathcal{U}, \delta_* P_X \mathcal{O}_X)$$

En appliquant le foncteur exact δ_* à la résolution fournie par le lemme 8.4, on obtient une résolution de faisceau de $\mathcal{O}_{X \times X}$ -module

$$\delta_* \mathcal{O}_X \rightarrow \delta_* C^{\bullet}(\mathcal{U}, P_X \mathcal{O}_X) = C^{\bullet}(\mathcal{U}, \delta_* P_X \mathcal{O}_X)$$

Ceci permet d'obtenir les isomorphismes

$$H_0(\mathcal{F}_\bullet) \simeq H_0(C^\bullet(\mathcal{U}, \delta_* P_X \mathcal{O}_X)) \simeq \delta_* \mathcal{O}_X$$

$$H_q(\mathcal{F}_\bullet) \simeq H_q(C^\bullet(\mathcal{U}, \delta_* P_X \mathcal{O}_X)) = 0$$

pour tout $q \neq 0$. On peut ainsi appliquer la proposition 6.5 (1) qui fournit pour tout faisceau \mathcal{M} de \mathcal{O}_X -module l'isomorphisme de δ -foncteur en \mathcal{M}

$$H^n(\mathcal{O}_X, \mathcal{M}) \simeq \mathbb{E}xt_{\mathcal{O}_X}^n(\delta^* \mathcal{F}_\bullet, \mathcal{M})$$

Pour utiliser ce résultat, on va comparer $\delta^* \mathcal{F}_\bullet$ et \mathcal{C}_\bullet . Si on applique le lemme 7.4 au carré cartésien

$$\begin{array}{ccc} U & \xrightarrow{\delta_U} & U \times U \\ j \downarrow & & \downarrow i \\ X & \xrightarrow{\delta} & X \times X \end{array}$$

alors on obtient un isomorphisme de faisceau de \mathcal{O}_X -module

$$P_X \mathcal{S}_\bullet(U) = j_* \mathcal{S}_{\bullet U} \simeq j_* \delta_U^* \mathcal{B}_{\bullet U} \simeq \delta^* i_* \mathcal{B}_{\bullet U} = \delta^* \mathcal{E}_\bullet(U)$$

En remarquant que δ^* commute avec les sommes et les produits finis, le lemme 9.4 nous fournit un quasi-isomorphisme

$$\delta^* \mathcal{F}_\bullet = C^\bullet(\mathcal{U}, \delta^* \mathcal{E}_\bullet) \simeq C^\bullet(\mathcal{U}, P_X \mathcal{S}_\bullet) \xrightarrow{q.is.} C^\bullet(\mathcal{U}, P_X \mathcal{C}_\bullet)$$

Or, d'après le corollaire 8.5, on dispose également d'un quasi-isomorphisme

$$\mathcal{C}_\bullet \xrightarrow{q.is.} \mathcal{G}_\bullet$$

où \mathcal{G}_\bullet désigne le complexe total de $C^\bullet(\mathcal{U}, P_X \mathcal{C}_\bullet)$. Ainsi, d'après le lemme 4.1,

$$\mathbb{E}xt_{\mathcal{O}_X}^n(\delta^* \mathcal{F}_\bullet, \mathcal{M}) \simeq \mathbb{E}xt_{\mathcal{O}_X}^n(\mathcal{G}_\bullet, \mathcal{M}) \simeq \mathbb{E}xt_{\mathcal{O}_X}^n(\mathcal{C}_\bullet, \mathcal{M})$$

ce qui prouve l'isomorphisme naturel en \mathcal{M}

$$H^n(\mathcal{O}_X, \mathcal{M}) \simeq HH^n(X, \mathcal{M})$$

Supposons à présent que chaque \mathcal{H}_q est localement libre. D'après ce qui précède,

$$H_q(\delta^* \mathcal{F}_\bullet) \simeq H_q(\mathcal{G}_\bullet) \simeq H_q(\mathcal{C}_\bullet) = \mathcal{H}_q$$

D'après la proposition 6.5 (2), les suites spectrales suivantes sont isomorphes

$$Ext_{\mathcal{O}_X}^p(H_q(\delta^* \mathcal{F}_\bullet), \mathcal{M}) \Rightarrow Ext_{\mathcal{O}_X}^{p+q}(\delta^* \mathcal{F}_\bullet, \mathcal{M})$$

$$H^p(X, Ext_{\mathcal{O}_{X \times X}}^q(\mathcal{O}_X, \mathcal{M})) \Rightarrow H^{p+q}(\mathcal{O}_X, \mathcal{M})$$

Par le lemme 4.1, la première suite spectrale est isomorphe à la suite spectrale

$$Ext_{\mathcal{O}_X}^p(\mathcal{H}_q, \mathcal{M}) \Rightarrow HH^{p+q}(X, \mathcal{M})$$

C'est le résultat annoncé.

On peut vérifier que l'isomorphisme ne dépend pas du recouvrement \mathcal{U} choisi. Si \mathcal{U}' est un recouvrement fini de X par des ouverts affines alors

$$\mathcal{V} = \{U \cap U' : U \in \mathcal{U}, U' \in \mathcal{U}'\}$$

est un recouvrement fini de X par des ouverts affines qui raffine \mathcal{U} . Les restrictions induisent alors un morphisme de complexe

$$C^\bullet(\mathcal{U}, -) \rightarrow C^\bullet(\mathcal{V}, -)$$

rendant le diagramme suivant commutatif

$$\begin{array}{ccccc} \delta^* C^\bullet(\mathcal{U}, \mathcal{E}_\bullet) & \longrightarrow & C^\bullet(\mathcal{U}, P_X \mathcal{C}_\bullet) & \xleftarrow{\varepsilon} & \mathcal{C}_\bullet \\ \downarrow & & \downarrow & & \downarrow = \\ \delta^* C^\bullet(\mathcal{V}, \mathcal{E}_\bullet) & \longrightarrow & C^\bullet(\mathcal{V}, P_X \mathcal{C}_\bullet) & \xleftarrow{\varepsilon} & \mathcal{C}_\bullet \end{array}$$

On obtient ainsi le diagramme commutatif suivant

$$\begin{array}{ccccc} H^n(\mathcal{O}_X, \mathcal{M}) & \xrightarrow{\simeq} & \mathbb{E}xt_{\mathcal{O}_X}^n(\delta^* \mathcal{F}_\bullet, \mathcal{M}) & \simeq & \mathbb{E}xt_{\mathcal{O}_X}^n(\mathcal{G}_\bullet, \mathcal{M}) & \xrightarrow{\simeq} & HH^n(X, \mathcal{M}) \\ & & \downarrow & & \downarrow & & \\ & & \mathbb{E}xt_{\mathcal{O}_X}^n(\delta^* \mathcal{F}'_\bullet, \mathcal{M}) & \simeq & \mathbb{E}xt_{\mathcal{O}_X}^n(\mathcal{G}'_\bullet, \mathcal{M}) & \xrightarrow{\simeq} & \end{array}$$

où \mathcal{F}'_\bullet et \mathcal{G}'_\bullet désignent respectivement les complexes totaux de $C^\bullet(\mathcal{V}, \mathcal{E}_\bullet)$ et $C^\bullet(\mathcal{V}, P_X \mathcal{C}_\bullet)$.

Lexique des notations

$X \times_S Y$	produit des schémas X et Y fibré sur un schéma S
δ_X	application diagonale d'un schéma X sur un schéma de base
$f_*\mathcal{F}$	pushforward d'un faisceau \mathcal{F} par une application continue $f : X \rightarrow Y$
$f^{-1}\mathcal{G}$	pullback d'un faisceau \mathcal{G} par une application continue $f : X \rightarrow Y$
$f^*\mathcal{G}$	faisceau $\mathcal{O}_Y \otimes_{f^{-1}\mathcal{O}_X} f^{-1}\mathcal{G}$ si $f : X \rightarrow Y$ est un morphisme de schéma
M^\sim	faisceau quasi-cohérent sur $\text{Spec } A$ associé à un A -module M
$\mathcal{H}om_{\mathcal{O}}(\mathcal{F}, \mathcal{G})$	faisceau $U \mapsto \mathcal{H}om_{\mathcal{O} _U}(\mathcal{F} _U, \mathcal{G} _U)$
$\mathcal{E}xt_{\mathcal{O}}^\bullet(\mathcal{F}, -)$	foncteur dérivé du foncteur $\mathcal{H}om_{\mathcal{O}}(\mathcal{F}, -)$
\mathcal{F}_x	fibres d'un faisceau \mathcal{F} en un point x
Ω_A^q	module des q -formes différentielles de Kähler sur une algèbre A
$\Omega_{X/S}^q$	faisceau des q -formes différentielles relatives sur un S -schéma X
M^\vee	module dual $\mathcal{H}om_A(M, A)$ d'un A -module M
\mathcal{F}^\vee	faisceau dual $\mathcal{H}om_{\mathcal{O}}(\mathcal{F}, \mathcal{O})$ d'un faisceau de \mathcal{O} -module \mathcal{F}
$\Gamma(U, \mathcal{F})$	sections $\mathcal{F}(U)$ d'un faisceau \mathcal{F} sur un ouvert U
$H_\bullet^v(C_{\bullet\bullet})$	homologie en la première variable du double complexe $C_{\bullet\bullet}$
$H_\bullet^h(C_{\bullet\bullet})$	homologie en la seconde variable du double complexe $C_{\bullet\bullet}$
$S^{-1}A$	anneaux des fractions à numérateur dans A et à dénominateur dans S
A_s	anneaux des fractions pour $s \in A$ et $S = \{s^n : n \in \mathbb{N}\}$
$A_{\mathfrak{p}}$	anneaux des fractions pour $\mathfrak{p} \in \text{Spec } A$ et $S = A \setminus \mathfrak{p}$
$\xrightarrow{q.is.}$	quasi-isomorphisme

Bibliographie

- [1] R. Hartshorne, Algebraic Geometry (Springer, Berlin, 1977)
- [2] J.-L. Loday, Cyclic Homology, Second Edition (Springer, Berlin, 1998)
- [3] R.G. Swan, Hochschild cohomology of quasiprojective schemes, Journal of Pure and Applied Algebra 110 (1996) 57-80
- [4] C.A. Weibel, An Introduction to Homological Algebra (Cambridge University Press, Cambridge, 1994)

Année Universitaire 2019-2020

Master 2 Mathématiques Fondamentales et Appliquées
Parcours Algèbre et Géométrie

Décomposition des transformations birationnelles du plan projectif complexe

Par : Thibault Chailleux

Sous la direction de : Susanna Zimmermann

Date de soutenance : 07/07/2020

Introduction

L'étude du groupe de Cremona consiste à regarder les applications birationnelles de $\mathbb{P}^2(\mathbb{C})$ dans lui-même, appelées transformations birationnelles de $\mathbb{P}^2(\mathbb{C})$.

Une telle transformation peut s'écrire via des polynômes homogènes de même degré f_0, f_1, f_2 sous la forme :

$$f : [x : y : z] \dashrightarrow [f_0(x, y, z) : f_1(x, y, z) : f_2(x, y, z)]$$

Elle est alors bien définie sauf sur un nombre fini de points¹, correspondant aux racines communes des polynômes f_0, f_1, f_2 . A l'inverse, elle contracte des courbes sur les points-base de la transformation f^{-1} .

Un autre point de vue de la géométrie birationnelle est encore de regarder les points-base de f et leur multiplicité pour caractériser une telle transformation (à automorphisme près).

On s'intéresse ici au cas particuliers des transformations quadratiques, c'est à dire les transformations birationnelles de degré 2. Ces transformations possèdent trois points-base, chacun avec multiplicité 1, et le théorème de Noether-Castelnuovo² affirme qu'avec les automorphismes, ces transformations engendrent le groupe de Cremona :

Théorème 0.1 (Noether-Castelnuovo [3])

Les transformations birationnelles de $\mathbb{P}^2(\mathbb{C})$ peuvent être décomposées en transformations quadratiques et en automorphismes.

En 1915, J.W. Alexander rédigea une preuve courte de ce théorème dans un article "On the factorization of the Cremona plane transformations" [1], sous forme d'un algorithme permettant de décomposer progressivement la transformation. Il consiste à regarder les points-base de la transformation et leur multiplicité, et à précomposer par des inverses de transformations quadratiques bien choisies.

Ces transformations quadratiques modifient les points-base de f , qui voient alors leur multiplicité baisser, jusqu'à ce que la composition soit elle-même une transformation de degré 2. On obtient ainsi une décomposition de f en transformations quadratiques.

Il arrive cependant que l'application f ne soit pas aussi simple, et possède des points-base qui ne sont pas directement dans $\mathbb{P}^2(\mathbb{C})$, et qui n'apparaissent qu'après avoir éclaté $\mathbb{P}^2(\mathbb{C})$ en certain points. Ce phénomène complexifie l'algorithme, mais une telle décomposition est encore possible, et sous réserve de quelques pré-compositions quadratiques supplémentaires, on peut se ramener au premier cas.

L'objectif de ce mémoire est d'étudier cet algorithme et d'en donner une formulation moderne afin de pouvoir l'appliquer et, à terme, l'implémenter informatiquement.

Le premier chapitre fournit les définitions permettant d'aborder la démonstration du théorème : les notions d'applications régulières et rationnelles, les diviseurs et leurs propriétés, le produit d'intersection et les éclatements y seront rappelés, et on donnera également un théorème dû à Zariski sur la factorisation des applications birationnelles comme suite d'éclatements.

Dans le second chapitre, on démontrera les équations de conditions qui fournissent des contraintes qui permettent entre autre de compter les multiplicités possibles pour une transformation birationnelle de degré

1. Ce sont des cas particuliers de *points-base* de f que l'on définira de manière plus générale.

2. Énoncé et démontré partiellement par Max Noether ; démontré complètement par Guido Castelnuovo en 1901 [3].

donné. On exprime ensuite le degré de la composition d'une transformation birationnelle avec une transformation quadratique, ainsi que les points-base et leur multiplicité afin de caractériser de telles compositions.

Le troisième chapitre consiste en la démonstration du théorème de Noether-Castelnuovo donnée par J.W. Alexander par induction sur la complexité d'une transformation birationnelle, définie comme $2j = \deg(f) - m_f(O_0)$ où O_0 est le point-base de plus grande multiplicité de f .

On effectue ensuite quelques calculs de décompositions en appliquant l'algorithme à des transformations plus ou moins complexe, et on donnera également une preuve du test de Hudson [2, p.225] qui utilise cette décomposition algorithmique pour déterminer si des nombres qui vérifient les équations de conditions pour le degré d sont des données de multiplicités valides pour une transformation birationnelle de degré d .

Enfin, on donnera une explication de l'implémentation informatique en Python de l'algorithme de décomposition.



Vassily Kandinsky (1925) - Free Curve to the Point.

Je tiens à exprimer toute ma reconnaissance à mon encadrante, Susanna Zimmermann, pour m'avoir accompagné chaque semaine pendant ces trois mois de stage – malgré les conditions sanitaires contraignantes – avec une rigueur et une pédagogie remarquable, ainsi que pour m'avoir fait découvrir le domaine de la géométrie birationnelle.

Je remercie également les enseignants du Master de Mathématiques Fondamentales et Appliquées qui m'ont permis de développer mes connaissances mathématiques dans les domaines de l'algèbre et de la géométrie.

Je tiens à remercier mes parents et Agnès pour m'avoir soutenu et aidé avec persévérance durant toute cette année, ainsi que pendant mes études. Leur aide a grandement contribué à l'élaboration de ce mémoire.

Je souhaite aussi remercier mes amis du Master de Mathématiques Fondamentales et Appliquées, Thomas, Jean, et Clovis, ainsi que mes camarades du CRDM pour leur soutien et leurs encouragements. Conséquemment, je remercie Claude et Anh pour leur sympathie et pour avoir patiemment supporté nos incessantes élucubrations – entre autres mathématiques.

Enfin, je tiens à remercier tous mes amis non mathématiciens qui, à travers leurs connaissances philosophique, artistique ou scientifique, ont pu apporter d'autres points de vue à ma vision des mathématiques.

Table des matières

Introduction	2
1 Rappels	5
1.1 Applications rationnelles et régulières	5
1.1.1 Définitions	5
1.2 Diviseurs	6
1.2.1 Définitions	6
1.2.2 Propriétés des diviseurs	7
1.2.3 La classe de diviseurs canonique	7
1.3 Produit d'intersection	9
1.4 Éclatements	10
1.5 Théorème de Zariski	12
1.5.1 Systèmes linéaires	13
2 Propriétés des transformations de \mathbb{P}^2	14
2.1 Équations de condition	15
2.2 Composition des transformations	16
3 Théorème de Noether-Castelnuovo	20
3.1 Introduction de la preuve	20
3.2 Preuve du théorème de Noether-Castelnuovo	21
4 Applications du théorème de Noether-Castelnuovo	27
4.1 Transformations quadratiques	27
4.1.1 Deux points-base dans \mathbb{P}^2 et un point-base infiniment proche	27
4.1.2 Trois points-base infiniment proches	28
4.2 Transformation de degré 3	30
4.3 Transformation de degré 5	30
4.4 Test de Hudson	31
5 Annexe : Implémentation de l'algorithme	33
5.1 Classes d'objets : points-base et transformation birationnelle	33
5.2 La fonction <code>transfo_deg</code>	36
5.3 La fonction <code>compose_quad</code>	39
5.4 La fonction <code>make_transfo_disjoint</code>	40
5.5 La fonction <code>decomp_transfo</code>	40
5.6 Exemples et applications	43
Bibliographie	46

Chapitre 1

Rappels

Dans tout le mémoire, on se place sur le corps des nombres complexes \mathbb{C} . En particulier, \mathbb{P}^2 désignera le plan projectif complexe $\mathbb{P}^2(\mathbb{C})$.

1.1 Applications rationnelles et régulières

1.1.1 Définitions

On rappelle ici les définitions et propriétés de base en rapport avec les applications régulières et rationnelles.

Définition 1.1 (Variété quasiprojective ([6] p.46))

On appelle **variété quasiprojective** un ouvert d'un fermé de \mathbb{P}^n pour un certain n (i.e. l'intersection d'un ouvert et d'un fermé de \mathbb{P}^n).

Une **sous-variété** d'une variété quasiprojective X est un sous-ensemble $Y \subset X$ qui est une variété quasiprojective.

Définition 1.2 (Fonctions et applications régulières ([6] p.48))

Soit $X \subset \mathbb{P}^n$ une variété quasiprojective irréductible.

Soit $x \in X$. Une **fonction régulière en x** est une fonction définie au voisinage de x et à valeur dans \mathbb{C} qui est le quotient de deux polynômes homogènes de même degré $P, Q \in \mathbb{C}[T_1, \dots, T_n]$, avec $Q(x) \neq 0$. Une **fonction régulière sur X** est une fonction définie sur X et régulière au voisinage de tout $x \in X$.

On note $\mathbb{C}[X]$ l'anneau des fonctions régulières sur X .

Une **application régulière** $f : X \rightarrow \mathbb{P}^m$ est la donnée de $m + 1$ polynômes $F_0, \dots, F_m \in \mathbb{C}[T_1, \dots, T_n]$ homogènes et de même degré et tels que pour tout $x \in X$, un des F_i soit non nul en x . On note pour un $x \in X$, $f(x) = [F_0(x) : \dots : F_m(x)]$.

Soit $X \subset \mathbb{P}^n$ une variété quasiprojective irréductible. On peut considérer le sous-anneau \mathcal{O}_X des fractions rationnelles $\mathbb{C}(T_1, \dots, T_n)$ constitué des quotients P/Q tels que P et Q soient homogènes de même degrés et que Q n'est pas toujours nul sur X .

Le sous-ensemble des fractions P/Q telles que P est toujours nul sur X en forme un idéal maximal \mathcal{M}_X .

Définition 1.3 (Fonctions et applications rationnelles ([6] p.50))

Soit $X \subset \mathbb{P}^n$ une variété quasiprojective irréductible.

Le corps des **fonctions rationnelles sur X** est le corps des fractions de $\mathbb{C}[X]$, ou de manière équivalente, le corps $\mathcal{O}_X/\mathcal{M}_X$. Une telle fonction f est dite régulière en $x \in X$ s'il existe un représentant P/Q tel que $Q(x) \neq 0$. On définit alors $f(x) = P(x)/Q(x)$.

Une **application rationnelle** $f : X \dashrightarrow \mathbb{P}^m$ est la donnée de $m + 1$ polynômes $F_0, \dots, F_m \in \mathbb{C}[T_1, \dots, T_n]$ homogènes et de même degré dont l'une d'entre elle n'est pas toujours nulle sur X . On quotiente par la

relation d'équivalence :

$$f = g \iff \forall i, j \in \{0, \dots, m\}, F_i G_j = F_j G_i$$

On notera alors $f = [F_0 : \dots : F_m]$.

Remarque

Une application rationnelle $f : X \rightarrow \mathbb{P}^m$ définit la donnée de $m + 1$ fonctions rationnelles en quotientant par un des polynôme non toujours nul sur X . On dit alors qu'une application rationnelle est régulière en $x \in X$ si elle définit des fonctions rationnelles régulières en x et non toutes nulles en x . On note alors $f(x) = [F_0(x) : \dots : F_m(x)]$.

Étant donnée $Y \subset \mathbb{P}^m$ une variété quasiprojective, on dit que f envoie X vers Y s'il existe un ouvert U de X sur lequel f est régulière et tel que $f(U) \subset Y$. L'image de X dans Y par f est alors l'image de l'union de tous les ouverts vérifiant cela. On note $f : X \rightarrow Y$ une telle application.

Une application rationnelle entre deux variétés quasiprojectives $f : X \dashrightarrow Y$ est dite **birationnelle** si l'image de X dans Y par f est dense et s'il existe une application rationnelle $f' : Y \dashrightarrow X$ telle que $f \circ f'(y) = y$ et $f' \circ f(x) = x$ lorsque les images sont bien définies.

Par la suite, on appellera **transformation birationnelle** (du plan projectif) toute application birationnelle de \mathbb{P}^2 dans \mathbb{P}^2 . On peut montrer [6, Theorem 3 p.109] qu'une telle application n'est pas régulière seulement en un nombre fini de points.

On définit le **degré de f** comme étant le degré minimal des polynômes définissant la transformation, c'est à dire le degré des polynômes quand ils n'ont pas de facteurs communs.

On fournit également la notion de multiplicité d'une courbe en un point : étant donnée S une surface quasiprojective, $p \in S$, on peut considérer l'anneau local des germes de fonctions régulières en p : $\mathcal{O}_{S,p}$, d'idéal maximal $\mathfrak{m}_{S,p}$.

Alors il existe un unique $k \in \mathbb{N}$ tel que, si f est une équation locale de C en p , $f \in \mathfrak{m}^k \setminus \mathfrak{m}^{k+1}$, que l'on définit comme la multiplicité de C en p . [4, p.16]

1.2 Diviseurs

1.2.1 Définitions

Définition 1.4 (Diviseur ([6] p.151))

Étant donné une variété quasiprojective irréductible X , on appelle **diviseur premier** sur X un sous-ensemble algébrique fermé irréductible de X de codimension 1.

On note $Div(X)$ le groupe abélien libre engendré par les diviseurs premiers sur X . Un élément de $Div(X)$ est appelé **diviseur** sur X .

On note $(C_i)_{i \in I}$ l'ensemble des diviseurs premiers sur X . Si $D = \sum_{i \in I} k_i C_i \in Div(X)$, l'union des C_i tels que $k_i \neq 0$ est appelé **support de D** . Un diviseur $\sum k_i C_i$, est dit **effectif** si $k_i \geq 0$ pour tout i .

À chaque fonction rationnelle $f \in k(X)$ non nulle, on peut associer un diviseur $div(f)$ de la manière suivante : on peut montrer qu'il n'existe qu'un nombre fini de diviseurs premiers en lesquels la multiplicité de f est non nulle [6, p.153]. On peut donc considérer le diviseur :

$$div(f) = \sum_{i \in I} \nu_{C_i}(f) C_i$$

Un diviseur obtenu comme cela est appelé **diviseur principal**. L'ensemble des diviseurs principaux forment un sous-groupe de $Div(X)$, par les relations [6, p.153] :

$$\begin{cases} div(f.g) &= div(f) + div(g) \\ div(1/f) &= -div(f) \text{ et } div(c) = 0 \end{cases}$$

pour c fonction constante non nulle.

En quotientant par ce sous-groupe, on obtient le groupe $Cl(X)$ des **classes de diviseurs** sur X . Deux diviseurs de la même classe sont dits **linéairement équivalents**.

Dans le cas de \mathbb{P}^2 , on peut classer les diviseurs par leur degré, en définissant le degré d'un diviseur linéairement par rapport au degré des diviseurs premiers qui le constituent, le degré d'un diviseur étant nul si, et seulement si, il est principal. Ainsi, $Cl(\mathbb{P}^2) \simeq \mathbb{Z}$. [6, p.154].

Soit X une variété quasiprojective irréductible régulière et $D = \sum_{i=1}^n k_i C_i$ un diviseur. Comme X est régulière, il existe un ouvert U sur lequel chaque C_i est décrit par une équation locale $\pi_i = 0$. En posant alors $f = \pi_1^{k_1} \dots \pi_n^{k_n}$, on dit alors que $D = \text{div}(f)$ sur U . Réciproquement, la donnée d'ouverts U_i recouvrant X et de fonctions rationnelles $f_i : U_i \rightarrow \mathbb{C}$ formant un **système compatible**, i.e. telles que sur $U_i \cap U_j$, les fonctions f_i, f_j ne valent jamais 0 et f_i/f_j est régulière, définit un diviseur D sur X . [6, p.155]

1.2.2 Propriétés des diviseurs

On a besoin pour travailler avec les diviseurs de donner une propriété importante : étant donné un nombre fini de points x_1, \dots, x_n de X et un diviseur D , alors D est linéairement équivalent à un diviseur D' dont le support ne contient pas x_1, \dots, x_n .

Cela permet de définir le tiré-en-arrière d'une classe de diviseur par une application régulière.

Théorème 1.5 (Déplacement des diviseurs ([6] Theorem 1 p.158))

Soit D un diviseur d'une variété quasiprojectives irréductible régulière X , et $x_1, \dots, x_n \in X$. Il existe un diviseur D' linéairement équivalent à D et dont le support ne contient aucun des x_i .

Soient X, Y des variétés irréductibles quasiprojective régulières. Étant donné un diviseur D sur Y et une application régulière $\phi : X \rightarrow Y$, on peut définir un diviseur $\phi^*(D)$ sur X à la condition que $\phi(X) \not\subset \text{Supp}(D)$ en considérant un certain système compatible de fonctions rationnelles sur des ouverts de X [6, p.155-156].

On peut, plus généralement, définir le tiré-en-arrière d'une classe de diviseurs sur Y , par le théorème précédent : soit D un diviseur tel que $\phi(X) \subset \text{Supp}(D)$, et $x \in \phi(X)$. Alors il existe un diviseur D' linéairement équivalent à D et dont le support ne contient pas x , de sorte que $\phi(X) \not\subset \text{Supp}(D')$, et on peut définir $\phi^*(D')$.

On a donc une fonction $\phi^* : Cl(Y) \rightarrow Cl(X)$ bien définie (en vérifiant que deux diviseurs linéairement équivalents sont envoyés dans la même classe) et qui est un morphisme de groupe.

1.2.3 La classe de diviseurs canonique

On va définir, sans s'étendre trop sur leur propriétés, les formes différentielles régulières et rationnelles sur une variété quasiprojective irréductible régulière X .

Quelques rappels sur la différentielle d'une fonction régulière

Soit X une variété quasiprojective. On rappelle la définition de l'espace tangent de Zariski : soit $x \in X$, l'espace tangent de Zariski en x est le \mathbb{C} -espace vectoriel $T_x X = (\mathfrak{m}_x / \mathfrak{m}_x^2)^\vee$ où $\mathfrak{m}_x = \{f \in \mathbb{C}[X] \mid f(x) = 0\}$. On note T^*X le fibré tangent de X , c'est à dire l'union disjointe des $(T_x X)^\vee = \mathfrak{m}_x / \mathfrak{m}_x^2$.

A une application régulière $f : X \rightarrow Y$ on peut associer sa différentielle en $x \in X$ [6, p.88]. En effet, f induit une application entre les ensembles de fonctions régulières des deux variétés :

$$\begin{array}{ccc} f^* & : & \mathbb{C}[Y] \longrightarrow \mathbb{C}[X] \\ g & \longmapsto & g \circ f \end{array}$$

De plus, on a bien, $f^*(\mathfrak{m}_{f(x)}) \subset \mathfrak{m}_x$ et $f^*(\mathfrak{m}_{f(x)}^2) \subset \mathfrak{m}_x^2$, donc f^* se restreint et passe au quotient. Finalement la transposée de cette nouvelle application f^* fournit la différentielle de f en x , $d_x f : T_x X \rightarrow T_{f(x)} Y$.

En prenant $Y = A_1^{\mathbb{C}}$, on définit la différentielle d'une fonction régulière f . Cette différentielle est à valeur dans \mathbb{C} ,¹ Donc $d_x f \in (T_x X)^{\vee}$. On a finalement une application appelée **différentielle de f** , $df : X \rightarrow T^*X$.

Formes différentielles régulières et rationnelles

On considère ici une variété quasiprojective irréductible régulière X de dimension n . On note $\Phi^r[X]$ le $\mathbb{C}[X]$ -module des fonctions $X \rightarrow \bigwedge^r T^*X$ qui envoient x vers un élément de $\bigwedge^r (T_x X)^*$.

Définition 1.6 (Forme différentielle régulière ([6] p.195))

Une **r -forme différentielle régulière** sur X est une fonction $\phi \in \Phi^r[X]$ qui est localement le produit extérieur de différentielles de fonctions régulières :

pour tout $x \in X$, il existe un voisinage U de x tel que $\phi|_U$ soit dans le sous- $\mathbb{C}[U]$ -module de $\Phi^r[U]$ engendré par les $df_1 \wedge \dots \wedge df_r$ pour $f_1, \dots, f_r \in \mathbb{C}[U]$.

On note $\Omega^r[X]$ le sous- $k[X]$ -module de $\Phi^r[X]$ des r -formes différentielles régulières sur X .

Définition 1.7 (Forme différentielle rationnelle ([6] p.202))

On note $\Omega^r(X)$ le quotient de l'ensemble

$$\left\{ (U, \omega) \in O(X) \times \coprod_{V \subset X} \Omega^r[V] \mid \omega \in \Omega^r[U] \right\}$$

par la relation d'équivalence :

$$(U, \omega) \sim (U', \omega') \iff \exists V \subset U \cap U' \text{ ouvert non vide, } \omega|_V = \omega'|_V$$

Un élément $\tilde{\omega}$ de $\Omega^r(X)$ est une **r -forme différentielle rationnelle**.

Si $\tilde{\omega}$ contient un couple (U, ω) , on dit que ω est **régulière sur U** . On appelle **domaine de régularité** de $\tilde{\omega}$ l'ouvert $U_{\tilde{\omega}} = \bigcup_{(U, \omega) \in \tilde{\omega}} U$.

On notera abusivement par la suite $\omega = \tilde{\omega}$.

Remarque

On observe que $\Omega^0(X) = \mathbb{C}(X)$ [5, p.176, Proposition-définition 1.1], ce qui fait de $\Omega^r(X)$ un $\mathbb{C}(X)$ -espace vectoriel, via l'opération produit extérieur.

$\Omega^r(X) = \Omega^r(U)$ pour U un ouvert de X , par définition de la relation d'équivalence : on a autant de classes dans l'un que dans l'autre, même si les classes sont plus petites dans le second.

Par cette remarque, on peut montrer que $\Omega^r(X)$ est un $\mathbb{C}(X)$ -espace vectoriel de dimension $\binom{n}{r}$, en utilisant le résultat sur le rang du $\mathbb{C}[U]$ -module $\Omega^r[U]$ pour U un ouvert assez petit de X [6, p.200 Theorem 2] et le fait qu'une fonction régulière sur un ouvert de X définit une fonction rationnelle sur X [6, p.203 Theorem 3]. Par ailleurs, on a pour un voisinage assez petit d'un point $x \in X$, une base de $\Omega^r[U]$ sur $\mathbb{C}[U]$ donnée par $(du_{i_1} \wedge \dots \wedge du_{i_r})_{i_1 \leq \dots \leq i_r}$ pour $u_1, \dots, u_n \in \mathbb{C}[U]$ des fonctions régulières données².

En particulier, $\Omega^n(X)$ est de dimension 1 sur $k\mathbb{C}(X)$.

La classe de diviseurs canonique

Soit X une variété quasiprojective régulière irréductible de dimension n .

En considérant le cas particulier des n -formes différentielles rationnelles sur X , on peut obtenir une classe de diviseurs particuliers sur X :

Soit $\omega \in \Omega^n(X)$. On peut recouvrir X par un nombre fini d'ouverts affines (quitte à les intersecter avec un recouvrement affine) U_i sur lesquels $\omega = g^{(i)} du_1^{(i)} \wedge \dots \wedge du_n^{(i)}$. Les fonctions $g^{(i)} : U_i \dashrightarrow \mathbb{C}$ forment un

1. Puisque $\mathbb{C}[A_1^{\mathbb{C}}]$ est isomorphe comme \mathbb{C} -espace vectoriel à l'anneau des polynômes à une variable $\mathbb{C}[t]$ par définition. Dès lors, $\mathfrak{m}_x/\mathfrak{m}_x^2$ est isomorphe au quotient du sous-anneau des polynômes sans coefficients constants par (t^2) , lui-même isomorphe à \mathbb{C} .

2. vérifiant que $\forall y \in U, (d_y u_1, \dots, d_y u_n)$ forme une base de $(T_y X)^{\vee}$ [6, p.200 Theorem 2].

système compatible de fonctions rationnelles [6, p.209] qui définit donc un diviseur $div(\omega)$ sur X .

Ces diviseurs vérifient la relation $div(f\omega) = div(f) + div(\omega)$. En particulier, tous ces diviseurs sont dans une même classe K_X appelée **classe canonique associée à X** , puisque $\Omega^n(X)$ est de dimension 1 sur $\mathbb{C}(X)$.

Exemple 1 (Calcul de $K_{\mathbb{P}^2}$)

Pour calculer $K_{\mathbb{P}^2}$, il suffit de déterminer une 2-forme différentielle rationnelle sur \mathbb{P}^2 . On va partir d'une forme différentielle régulière sur un des ouverts affines classiques (par exemple U_0) de \mathbb{P}^2 et la prolonger aux autres ouverts, d'abord sur l'intersection, puis en une forme différentielle rationnelle sur \mathbb{P}^2 , et calculer au passage l'expression locale du diviseur associé.

On considère U_0, U_1, U_2 les ouverts affines de \mathbb{P}^2 donnés par :

$$U_i = \{(x_0 : x_1 : x_2) \in \mathbb{P}^2 \mid x_i \neq 0\}$$

Sur U_0 , on pose $y_i = \frac{x_i}{x_0}$, avec $i = 1, 2$. Les fonctions y_i sont bien définies sur U_0 et forment bien une base de $\Omega^2[U_0]$ comme $k[U_0]$ -module, donnée par la forme $\omega = dy_1 \wedge dy_2$. Cette forme fournit un diviseur $div(\omega)$, qui est nul sur U , puisque $div(\omega) = div(1)$ sur U . Etudions alors la valeur de cette forme hors de U_0 .

Sur U_1 , on a cette fois pour coordonnées locales $z_1 = \frac{x_0}{x_1}, z_2 = \frac{x_2}{x_1}$. En particulier, $z_1 = \frac{1}{y_1}, z_2 = \frac{y_2}{y_1}$ sur $U_0 \cap U_1$ (cela correspond aux changements de cartes habituels de \mathbb{P}^2).

Donc sur $U_0 \cap U_1$, on a via la différentielle d'une composée, $dy_1 = -\frac{dz_1}{z_1^2}, dy_2 = \frac{z_1 dz_2 - z_2 dz_1}{z_1^2}$. Ainsi, sur $U_0 \cap U_1$, on a :

$$\omega = dy_1 \wedge dy_2 = \left(-\frac{dz_1}{z_1^2}\right) \wedge \left(\frac{z_1 dz_2 - z_2 dz_1}{z_1^2}\right) = -\frac{1}{z_1^3} dz_1 \wedge dz_2$$

On peut ainsi prolonger ω sur U_1 . Notons que similairement au premier cas, la forme $\omega' = dz_1 \wedge dz_2$ est une base de $\Omega^2[U_1]$. Cette forme fournit un diviseur nul sur U_1 . Ainsi, on peut calculer $div(\omega)$ sur U_1 :

$$div(\omega) = div\left(-\frac{1}{z_1^3}\omega'\right) = div\left(-\frac{1}{z_1^3}\right) + div(\omega') = -3 \cdot div(z_1)$$

De même, sur U_2 , si on note $t_1 = \frac{x_0}{x_2}, t_2 = \frac{x_1}{x_2}$, alors on a encore $div(\omega) = -3 \cdot div(t_1)$ sur U_2 .

Ces deux derniers diviseurs sont les expressions locales du diviseur $div(x_0)$. En effet, géométriquement, $div(x_0) = \{(0 : x_1 : x_2) \in \mathbb{P}^2\}$, qui est un diviseur premier. Sa restriction à U_1 (resp. U_2) correspond exactement au lieu des zéros de $z_1 = \frac{x_0}{x_1}$ (resp. $t_1 = \frac{x_0}{x_2}$), avec la même multiplicité. On a donc $div(x_0) = div(z_1)$ sur U_1 , et $div(x_0) = div(t_1)$ sur U_2 .

En remontant, on a donc sur \mathbb{P}^2 , $div(\omega) = -3 \cdot div(x_0)$ puisque cette égalité est vraie localement. Ceci fournit la classe canonique de \mathbb{P}^2 : puisque $div(x_0)$ est linéairement équivalent à une droite générale L ,

$$K_{\mathbb{P}^2} = -3 \cdot L$$

1.3 Produit d'intersection

On définit le produit d'intersection sans rappeler sa construction (ce sont en particulier ses propriétés qui seront utiles) :

Définition 1.8 ([4] p.14), ou ([6] p.223-232) pour la construction)

Soit S une surface quasiprojective irréductible régulière.

Il existe une unique opération :

$$\cdot : Div(S) \times Div(S) \longrightarrow \mathbb{Z}$$

vérifiant que :

- Si deux diviseurs C et D sont en position générale [6, p.223], alors $C \cdot D = \#(C \cap D)$.
- Si C, C' sont linéairement équivalents et D, D' aussi, alors $C \cdot D = C \cdot D' = C' \cdot D = C' \cdot D'$.

Étant donnée une droite L dans \mathbb{P}^2 , son intersection avec elle-même L^2 vaut 1 puisque L est équivalent à L', L'' , deux droites en position générale, qui ne se coupent qu'en un point. De plus, comme $Cl(\mathbb{P}^2) \simeq \mathbb{Z}$, si C, D sont des diviseurs quelconques sur \mathbb{P}^2 , ils sont linéairement équivalents à $\deg(C)L$ et $\deg(D)L$ avec L une droite de \mathbb{P}^2 . On a alors une formulation du théorème de Bézout :

Théorème 1.9 (Bézout ([6] p.236))
Soient C, D des diviseurs sur \mathbb{P}^2 , alors

$$C \cdot D = \deg(C) \deg(D)$$

1.4 Éclatements

Définition 1.10 (Éclatement d'une surface quasiprojective régulière ([4] p.15))

Soit S une surface quasiprojective régulière irréductible, et soit $p \in S$.

Une application régulière $\pi : Y \rightarrow S$ (avec Y variété quasiprojective régulière) est un **éclatement de S en \mathbf{p}** si π est un isomorphisme régulier de $Y \setminus \pi^{-1}(p)$ vers $S \setminus \{p\}$.

Cet éclatement est unique à isomorphisme près, et on notera souvent $Y = Bl_p(S)$.

Exemple 2

L'éclatement de \mathbb{P}^2 en le point $[0 : 0 : 1]$ est donné par :

$$Bl_{[0:0:1]}(\mathbb{P}^2) = \{([x : y : z], [u : v]) \in \mathbb{P}^2 \times \mathbb{P}^1 \mid xv = yu\}$$

qui forme une sous-variété quasiprojective régulière irréductible de $\mathbb{P}^2 \times \mathbb{P}^1$.

Pour une surface quasiprojective irréductible régulière S , on peut considérer un ouvert affine $U \subset S$ avec ses coordonnées locales $u_1, u_2 : U \rightarrow \mathbb{C}$ pour lesquelles $p = (0, 0)$ (i.e. $u_1(p) = u_2(p) = 0$). L'image réciproque de U par l'éclatement de S en p est alors isomorphe à l'ouvert : [6, p.116]

$$\{(x, [t_1 : t_2]) \in U \times \mathbb{P}^1 \mid u_1(x)t_2 = u_2(x)t_1\}$$

On peut décomposer cet ouvert en deux ouverts affines $\{t_1 \neq 0\}$ et $\{t_2 \neq 0\}$, affines via les cartes $(x, [t_1 : t_2]) \rightarrow (u_1(x), t_2/t_1)$ et $(x, [t_1 : t_2]) \rightarrow (u_2(x), t_1/t_2)$. Dans la première carte par exemple, que l'on note via les coordonnées locales v_1, v_2 , on a pour tout $x \in \{t_1 \neq 0\}$:

$$\pi(v_1(x), v_2(x)) = (v_1(x), v_1(x)v_2(x))$$

On peut ainsi exprimer le tiré-en-arrière des fonctions coordonnées sur un de ces ouverts affines : $\pi^*u_1(x) = v_1(x)$ et $\pi^*u_2(x) = v_1(x)v_2(x)$.

Comme vu plus tôt, étant donnée une courbe irréductible sur S , on peut considérer son image réciproque par π , $\pi^*C = \pi^{-1}(C)$. On définit sa **transformée stricte** $\tilde{C} = \overline{\pi^{-1}(C \setminus \{p\})}$.

L'image réciproque de p par π , dans $Bl_p(S)$ correspond à une droite E appelée **diviseur exceptionnel associé à \mathbf{p}** . On notera souvent cette droite E_p ou F_p . On peut alors exprimer l'image réciproque par π d'une courbe irréductible de S :

Lemme 1.11 ([4] p.17)

Soit C une courbe irréductible de S , alors :

$$\pi^*(C) = \tilde{C} + m_C(p)E$$

Démonstration. Soit C une courbe irréductible de S . On considère un ouvert affine U de S , avec pour coordonnées les fonctions régulières $u_1, u_2 : U \rightarrow \mathbb{C}$ pour lesquelles $u_1(p) = u_2(p) = 0$.

Dès lors, C a une expression locale dans U en fonction des coordonnées : il existe $P_k, P_{k+1}, \dots, P_r \in \mathbb{C}[T_1, T_2]$ des polynômes homogènes de degrés respectifs $k, k+1, \dots, r$, avec $k = m_C(p)$, tels que :

$$\forall x \in U, x \in C \iff \sum_{i=k}^r P_i(u_1(x), u_2(x)) = 0$$

Le tiré-en-arrière de C par π est alors donné localement via les coordonnées π^*u_i :

$$\forall x \in \pi^{-1}(U), \pi(x) \in C \iff \sum_{i=k}^r P_i(\pi^*u_1(x), \pi^*u_2(x))$$

Comme vu plus tôt, on a alors pour v_1, v_2 un système de coordonnées locales dans un ouvert affine $V \subset \pi^{-1}(U)$:

$$\forall x \in V, \pi(x) \in C \iff \sum_{i=k}^r P_i(v_1(x), v_1(x)v_2(x)) = 0 \iff v^k \sum_{i=k}^r v_1(x)^{i-k} P_i(1, v_2(x))$$

Finalement, cela fournit une expression du tiré-en-arrière de C en tant que diviseur : $\pi^*C = k \cdot E + \tilde{C}$. \square

On a aussi les résultats d'intersection suivants :

Proposition 1.12 ([4] p.17)

Soit C, D deux diviseurs sur S . Alors :

- $\pi^*C \cdot \pi^*D = 0$.
- $\pi^*C \cdot E = 0$.
- $E^2 = -1$.

De plus, si C est une courbe irréductible, on a :

- $\tilde{C}^2 = C^2 - m_C(p)^2$.
- $\tilde{C} \cdot E = m_C(p)$.

On exprime enfin la classe de diviseur canonique dans l'éclatement $K_{Bl_p(S)}$:

Lemme 1.13

$$K_{Bl_p(S)} = \pi^*K_S + E$$

Démonstration. On considère un ouvert affine $U \subset S$ contenant p , et $u_1, u_2 : U \rightarrow \mathbb{C}$ des fonctions régulières sur U valant 0 en p et formant un système de coordonnées locales sur U .

Soit ω une 2-forme différentielle rationnelle sur U : $\omega = f du_1 \wedge du_2$. On peut considérer son tiré en arrière par $\pi : Bl_p(S) \rightarrow S$:

$$\pi^*\omega = \pi^*f d(\pi^*u_1) \wedge d(\pi^*u_2)$$

On peut déterminer ces deux nouvelles coordonnées locales en utilisant l'expression de l'éclatement sur U : $\{(x, [t_1 : t_2]) \in U \times \mathbb{P}^1 \mid u_1(x)t_2 = u_2(x)t_1\}$. En effet, les cartes locales de cet éclatement sont par exemple données par $\{t_1 \neq 0\}$:

$$(x, [t_1 : t_2]) \rightarrow (u_1(x), t_2/t_1)$$

Et on peut alors exprimer π sur cette carte locale : $\pi(x, y) = (x, xy)$. Finalement, cela donne l'expression des fonctions rationnelles π^*u_1, π^*u_2 sur l'ouvert $\{t_1 \neq 0\}$ de U : $\pi^*u_1(x, y) = x$ et $\pi^*u_2(x, y) = xy$. En remplaçant dans le tiré-en-arrière :

$$\pi^*\omega = \pi^*f dx \wedge d(xy) = \pi^*f dx \wedge (x \cdot dy + y \cdot dx) = \pi^*f \cdot x \cdot dx \wedge dy$$

Ainsi, en terme de diviseurs, $div(\pi^*\omega) = div(\pi^*f \cdot x) = div(\pi^*f) + div(x) = \pi^*div(\omega) + E$, et donc :

$$K_{Bl_p(S)} = \pi^*K_S + E$$

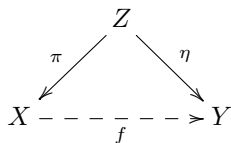
\square

1.5 Théorème de Zariski

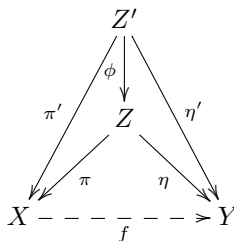
Dans cette dernière section de rappels, on donne l'énoncé du théorème de Zariski, qui permet de décomposer une transformation birationnelle entre deux surfaces projectives régulières en une suite d'éclatements et d'effondrements, ainsi que la définition du système linéaire associé à une transformation birationnelle.

Théorème 1.14 (de Zariski ([6] p.256))

Soient X, Y deux surfaces projectives régulières, et $f : X \dashrightarrow Y$. Alors il existe deux suites d'éclatements $\pi : Z \rightarrow X$ et $\eta : Z \rightarrow Y$ tels que $f = \eta \circ \pi^{-1}$:



On appelle cette donnée de deux suites d'éclatements une **résolution de f par le théorème de Zariski**. Une résolution Z est dite **minimale** si pour tout autre résolution Z' , on a une application régulière $\phi : Z' \rightarrow Z$ telle que le diagramme commute :



On peut vérifier que cette résolution minimale est unique à isomorphisme et ordre près.

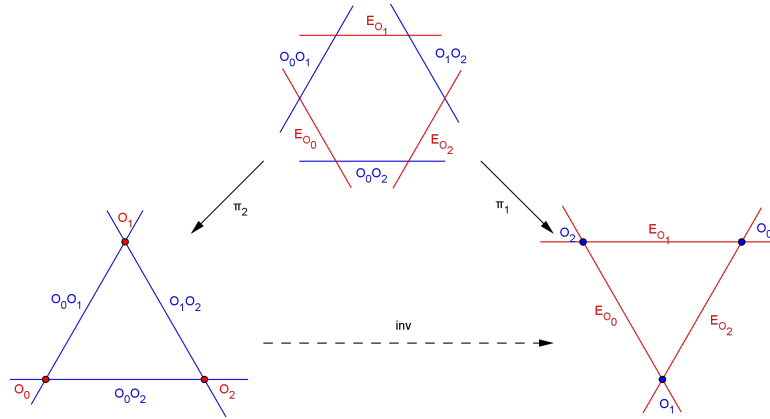
On définit alors un **point-base** d'une transformation birationnelle f comme étant un des points éclaté par π dans une résolution minimale de f par le théorème de Zariski. Les points éclatés par η sont alors les points-base de f^{-1} . Ainsi, dans une résolution minimale, π éclate les points-base de f et η contracte des droites sur les points-base de f^{-1} , et ces droites correspondent à celles qui sont contractées par f , à éclatement près.

Soient A et B deux points-base de $f : X \dashrightarrow Y$. On considère une résolution minimale Z de f par le théorème de Zariski avec les mêmes notations que précédemment, et on décompose $\pi : Z \rightarrow X$ en éclatements :

$$Z = X_n \xrightarrow{\pi_n} X_{n-1} \xrightarrow{\pi_{n-1}} \dots \xrightarrow{\pi_2} X_2 \xrightarrow{\pi_1} X_1 = X$$

Soit alors $i \in \{1, \dots, n\}$ tel que B est éclaté par π_i . On dit alors que **A est infiniment proche de B** si $A \in \pi_i^{-1}(B)$, autrement dit, si A est sur le diviseur exceptionnel associé à B .

Exemple 3



On considère la transformation birationnelle appelée **involution quadratique standard**, et donnée par :

$$inv : [x : y : z] \dashrightarrow [yz : xz : xy]$$

Elle n'est pas définie en les points $O_0 = [1 : 0 : 0]$, $O_1 = [0 : 1 : 0]$ et $O_2 = [0 : 0 : 1]$. Ces points correspondent aux points-base de inv . En effet, on obtient inv en éclatant \mathbb{P}^2 en ces trois points, puis en contractant les droites passant par deux des points sur le troisième, comme dans la figure ci-dessus.

On verra dans le chapitre 4 des exemples de transformations birationnelles dont les points-base ne sont pas tous dans \mathbb{P}^2 .

1.5.1 Systèmes linéaires

Définition 1.15 (Système linéaire d'une transformation birationnelle ([4] p.19))

Soit $f : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ une transformation birationnelle.

Le **système linéaire** associé à f , noté Λ_f est l'ensemble des préimages des droites de \mathbb{P}^2 :

Si $f = [f_0 : f_1 : f_2]$, avec les f_i sans facteurs communs, un élément du système linéaire est donc de la forme $\{[x : y : z] \mid a_0 f_0(x, y, z) + a_1 f_1(x, y, z) + a_2 f_2(x, y, z) = 0\}$ pour $[a_0 : a_1 : a_2]$.

Remarque

On définit un **élément général du système linéaire associé à f** comme la préimage d'une droite générale dans le contexte qui nous intéresse, c'est à dire d'une droite ne passant pas par les points-base de f et de f^{-1} qui sont en nombre fini. Tout élément général du système linéaire de f est une courbe irréductible passant par les points-base de f avec même multiplicité [4, p.19]. On peut alors définir la multiplicité d'une transformation birationnelle en un point P comme étant la multiplicité d'un élément général du système linéaire associé à ce point dans le cas où P est un point-base de f , et 0 sinon.

Chapitre 2

Propriétés des transformations de \mathbb{P}^2

Dans ce chapitre, on établit plusieurs propriétés des transformations de \mathbb{P}^2 qui seront utiles dans la démonstration par J.W. Alexander du théorème 0.1, de Noether-Castelnuovo.

On montre en premier lieu les équations de condition qui permettent entre autres de classer les transformations d'un degré donné, puisqu'elles fournissent une contrainte sur les points-base de celles-ci.

L'objet de la seconde partie est la caractérisation de la composition d'une transformation birationnelle avec une transformation quadratique ayant tous ses points-base dans \mathbb{P}^2 . On en calcule le degré et la multiplicité aux points-base. On utilisera ces formules dans la preuve du théorème, puisque la méthode employée consiste à précomposer la transformation considérée par une transformation quadratique afin de la simplifier.

On commence par établir un lemme qui sera utile dans les preuves suivantes :

Lemme 2.1

Soit $f : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ une transformation birationnelle de points-bases O_1, \dots, O_k , f^{-1} ayant pour points-bases Q_1, \dots, Q_l . On considère une résolution minimale de f par le théorème de Zariski : il existe deux suites d'éclatements $\pi, \eta : S \rightarrow \mathbb{P}^2$ tels que le diagramme suivant commute :

$$\begin{array}{ccc} & S & \\ \pi \swarrow & & \searrow \eta \\ \mathbb{P}^2 & \dashrightarrow & \mathbb{P}^2 \\ & f & \end{array}$$

Soit maintenant D une courbe irréductible fermée de \mathbb{P}^2 qui n'est pas contractée par f^{-1} et C l'image de D par f^{-1} . Alors la transformée stricte $\tilde{D}^{(\eta)} = \eta^{-1}(D \setminus \{Q_1, \dots, Q_l\})$ de D par η est égale à la transformée stricte $\tilde{C}^{(\pi)}$ de C par π .

Démonstration. On considère les diviseurs exceptionnels $E_1 = \pi^{-1}(O_1), \dots, E_k = \pi^{-1}(O_k)$ (resp. $F_1 = \eta^{-1}(Q_1), \dots, F_l = \eta^{-1}(Q_l)$) de π (resp. η) associés respectivement aux points O_1, \dots, O_k (resp. Q_1, \dots, Q_l).

En tant que suites d'éclatements, π et η induisent des isomorphismes $\tilde{\pi} : S \setminus (E_1 \cup \dots \cup E_k) \rightarrow \mathbb{P}^2 \setminus \{O_1, \dots, O_k\}$ et $\tilde{\eta} : S \setminus (F_1 \cup \dots \cup F_l) \rightarrow \mathbb{P}^2 \setminus \{Q_1, \dots, Q_l\}$.

Montrons l'égalité des transformées strictes par double inclusion :

Inclusion \supseteq : Soit $x \in \tilde{C}^{(\pi)}$. On suppose que $x \notin E_1 \cup \dots \cup E_k \cup F_1 \cup \dots \cup F_l$. Dès lors, l'image de x par $\tilde{\pi}$ n'est pas un point-base de f et on peut considérer $f \circ \tilde{\pi}(x) \in D$. Comme D est une courbe générale, $f \circ \tilde{\pi}(x)$ n'est pas un point-base de f^{-1} et on peut donc prendre son image par $\tilde{\eta}^{-1}$, qui vaut x par commutativité du diagramme.

Ainsi, $\pi^{-1}(D \setminus \{O_1, \dots, O_k\}) \subset \tilde{D}^{(\eta)}$, et donc c'est encore vrai pour la fermeture.

Inclusion \subseteq : Par symétrie, en appliquant l'inclusion précédente à f^{-1}

□

2.1 Équations de condition

Théorème 2.2 (Équations de condition)

Soit $f : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ une transformation birationnelle. On note O_1, \dots, O_k les points-bases de f . Alors :

$$\sum_{i=1}^k m_f(O_i)^2 = \deg(f)^2 - 1$$

$$\sum_{i=1}^k m_f(O_i) = 3(\deg(f) - 1)$$

Démonstration. On note Q_1, \dots, Q_l les points-base de f^{-1} .

On considère une résolution minimale de f par le théorème de Zariski : il existe deux suites d'éclatements $\pi, \eta : S \rightarrow \mathbb{P}^2$ tels que π éclate les points-bases de f et η ceux de f^{-1} , et que le diagramme suivant commute :

$$\begin{array}{ccc} & S & \\ \pi \swarrow & & \searrow \eta \\ \mathbb{P}^2 & \dashrightarrow_f & \mathbb{P}^2 \end{array}$$

On note E_1, \dots, E_k les diviseurs exceptionnels associés à π et F_1, \dots, F_l ceux associés à η . Soient maintenant L, L' deux droites générales de \mathbb{P}^2 . On considère les éléments généraux de Λ_f associés $C = (f^{-1})(L), C' = (f^{-1})(L')$ par f . On a alors $m_f(O_i) = m_C(O_i) = m_{C'}(O_i)$ par définition, et :

$$1 = L \cdot L' = \eta^* L \cdot \eta^* L' = \tilde{L}^{(\eta)} \cdot \tilde{L}'^{(\eta)} \quad (L, L' \text{ générales})$$

$$1 = \tilde{C}^{(\pi)} \cdot \tilde{C}'^{(\pi)} \quad (\text{lemme 2.1})$$

$$1 = \left(\pi^* C - \sum_{i=1}^k m_C(O_i) E_i \right) \left(\pi^* C' - \sum_{i=1}^k m_{C'}(O_i) E_i \right) \quad (\text{lemme 1.11})$$

$$1 = C \cdot C' - \sum_{i=1}^k m_C(O_i) m_{C'}(O_i) \quad (\text{proposition 1.12})$$

$$1 = \deg(f)^2 - \sum_{i=1}^k m_f(O_i)^2 \quad (C \cdot C' = \deg(f)^2 \text{ par Bézout})$$

Démontrons maintenant la seconde formule : si on considère à nouveau une droite générale L et $C \in \Lambda_f$ l'élément général associé à L , on peut alors calculer le produit d'intersection $L \cdot K_{\mathbb{P}^2} = L \cdot (-3H) = -3$. D'autre part :

$$L \cdot K_{\mathbb{P}^2} = \eta^* L \cdot \eta^* K_{\mathbb{P}^2} = \tilde{L}^{(\eta)} \cdot \left(K_S - \sum_{j=1}^l F_j \right) \quad (\text{par itération du lemme 1.13})$$

$$-3 = \tilde{L}^{(\eta)} \cdot K_S = \tilde{L}^{(\eta)} \cdot \left(\pi^* K_{\mathbb{P}^2} + \sum_{i=1}^k E_i \right) \quad (m_L(Q_i) = 0 \text{ car } L \text{ générale})$$

$$-3 = \tilde{C}^{(\pi)} \cdot \left(\pi^* K_{\mathbb{P}^2} + \sum_{i=1}^k E_i \right) \quad (\tilde{L}^{(\eta)} = \tilde{C}^{(\pi)} \text{ par le lemme 2.1})$$

$$-3 = \left(\pi^* C - \sum_{i=1}^k m_C(O_i) E_i \right) \left(\pi^* K_{\mathbb{P}^2} + \sum_{i=1}^k E_i \right) \quad (\text{lemme 1.11})$$

$$-3 = C \cdot K_{\mathbb{P}^2} + \sum_{i=1}^k m_C(O_i) \quad (\text{proposition 1.12})$$

$$-3 = -3 \deg(f) + \sum_{i=1}^k m_f(O_i) \quad (\text{Bézout})$$

□

2.2 Composition des transformations

Proposition 2.3

Soient f, g des transformations birationnelles de degré ν et μ . On a la formule :

$$\deg(f \circ g^{-1}) = \nu\mu - \sum_{i=1}^k m_f(O_i) m_g(O_i)$$

avec O_1, \dots, O_k les points-base de g .

Démonstration. On cherche à calculer $\deg(f \circ g^{-1})$. Soit $L \subset \mathbb{P}^2$ une droite générale et $C = (g \circ f^{-1})(L) \in \Lambda_{f \circ g^{-1}}$.

On considère alors une résolution minimale de g par le théorème de Zariski :

$$\begin{array}{ccccc} & & S & & \\ \pi_1 \swarrow & & & \searrow \pi_2 & \\ \mathbb{P}^2 & \xleftarrow{\quad g \quad} & \mathbb{P}^2 & \xrightarrow{\quad f \quad} & \mathbb{P}^2 \end{array}$$

Soit H une droite générale de \mathbb{P}^2 . Par le théorème de Bézout, $\deg(f \circ g^{-1}) = C \cdot H$ et en tirant en arrière les diviseurs, on a $\deg(f \circ g^{-1}) = \pi_1^* C \cdot \pi_1^* H = \pi_1^* C \cdot \tilde{H}^{(\pi_1)}$ puisque H est une droite générale, qui ne passe par aucun des points-bases Q_1, \dots, Q_l de g^{-1} , éclatés par π_1 .

Comme π_2 (resp. π_1) éclate les O_i (resp. Q_j). On note également E_i (resp. F_j) le diviseur exceptionnel dans S associé à O_i (resp. Q_j). En exprimant le tiré-en-arrière de C , on obtient :

$$\deg(f \circ g^{-1}) = \left(\tilde{C}^{(\pi_1)} + \sum_{j=1}^l m_C(Q_j) F_j \right) \cdot \tilde{H}^{(\pi_1)} = \tilde{C}^{(\pi_1)} \cdot \tilde{H}^{(\pi_1)} + \sum_{j=1}^l m_C(Q_j) F_j \cdot \tilde{H}^{(\pi_1)}$$

Or, H ne passe pas par Q_1, \dots, Q_l , donc son produit d'intersection avec les diviseurs exceptionnels de π_1 est nul.

De plus, si on note D (resp. G) la préimage de L (resp. H) par f^{-1} (resp. g^{-1}), on a en adaptant le lemme 2.1 au cas de C :

$$\deg(f \circ g^{-1}) = \tilde{C}^{(\pi_1)} \cdot \tilde{H}^{(\pi_1)} = \tilde{D}^{(\pi_2)} \cdot \tilde{G}^{(\pi_2)}$$

On développe alors l'expression :

$$\deg(f \circ g^{-1}) = \left(\pi_2^* D - \sum_{i=1}^k m_D(O_i) E_i \right) \cdot \left(\pi_2^* G - \sum_{i=1}^k m_G(O_i) E_i \right) = \pi_2^* D \cdot \pi_2^* G - \sum_{i=1}^k m_D(O_i) m_G(O_i)$$

Or, $\pi_2^* D \cdot \pi_2^* G = D \cdot G = \deg(g) \deg(f)$ par le lemme de Bézout. Finalement :

$$\deg(f \circ g^{-1}) = \deg(g) \deg(f) - \sum_{i=1}^k m_g(O_i) m_f(O_i)$$

□

Proposition 2.4

Soit $\text{inv} : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ l'involution standard donnée par $\text{inv} : [x : y : z] \dashrightarrow [yz : xz : xy]$, et $\alpha, \beta : \mathbb{P}^2 \rightarrow \mathbb{P}^2$ deux automorphismes de \mathbb{P}^2 . On pose $h = \beta \circ \text{inv} \circ \alpha$. La transformation h a donc pour points-base $O_1 = \alpha^{-1}([1 : 0 : 0])$, $O_2 = \alpha^{-1}([0 : 1 : 0])$, $O_3 = \alpha^{-1}([0 : 0 : 1])$, et si on considère les droites $L_{i,j}$ passant par les points O_i, O_j (i.e. l'ensemble des points dont la k -ème coordonnée est nulle, avec $k \in \{1, 2, 3\} \setminus \{i, j\}$ la coordonnée restante), alors elles sont contractées sur $Q_k = \beta(O_k)$:

- $L_{1,2}$ est contractée par h sur $Q_3 = \beta([0 : 0 : 1])$.
- $L_{1,3}$ est contractée par h sur $Q_2 = \beta([0 : 1 : 0])$.
- $L_{2,3}$ est contractée par h sur $Q_1 = \beta([1 : 0 : 0])$.

Soit $f : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ une transformation birationnelle de degré au moins 2. On note S_1, \dots, S_r les points-base de f .

Alors les points-base de $f \circ h^{-1}$ sont d'une part les Q_i , avec pour multiplicité :

$$m_{f \circ h^{-1}}(Q_i) = \deg(g) - m_g(O_j) - m_g(O_k)$$

pour $\{i, j, k\} = \{1, 2, 3\}$, et d'autre part les $h(S_j)$ avec $1 \leq j \leq r$ tel que S_j n'est pas un point-base de h avec multiplicité $m_{g \circ h^{-1}}(h(S_j)) = m_g(S_j)$.¹

Démonstration. On considère tout d'abord la résolution de h par le théorème de Zariski : il existe $\pi_1, \pi_2 : S \dashrightarrow \mathbb{P}^2$ deux suites d'éclatements du plan projectif tels que $h = \pi_2 \circ \pi_1^{-1}$. Les points éclatés par π_2 sont ceux de h , c'est à dire O_1, O_2, O_3 et ceux de π_1 sont ceux de h^{-1} , c'est à dire Q_1, Q_2, Q_3 . Pour $i \in \{1, 2, 3\}$, on note F_i (resp. E_i) le diviseur exceptionnel de π_1 (resp. π_2) associé à Q_i (resp. O_i).

$$\begin{array}{ccc} & S & \\ \pi_1 \swarrow & & \searrow \pi_2 \\ \mathbb{P}^2 & \xleftarrow{\quad h \quad} & \mathbb{P}^2 \xrightarrow{\quad f \quad} \mathbb{P}^2 \end{array}$$

Soit L une droite générale de \mathbb{P}^2 . On considère $C = (h \circ f^{-1})(L) \in \Lambda_{f \circ h^{-1}}$. On peut donner via le lemme 1.12 une expression de la multiplicité de $f \circ h^{-1}$ en les Q_i comme produit d'intersection.

Faisons le cas $i = 1$ par exemple :

$$\tilde{C}^{(\pi_1)} \cdot F_1 = \left(\pi_1^* C - \sum_{i=1}^3 m_C(Q_i) \right) \cdot F_1 = m_{f \circ h^{-1}}(Q_1)$$

En appliquant le lemme 2.1 à C , et en notant $G = (h^{-1})(C) = (f^{-1})(L)$, on a $\tilde{G}^{(\pi_2)} = \tilde{C}^{(\pi_1)}$. Donc on peut calculer le produit d'intersection :

$$m_{f \circ h^{-1}}(Q_1) = \tilde{G}^{(\pi_2)} = (\pi_2^* G - m_f(O_1)E_1 - m_f(O_2)E_2 - m_f(O_3)E_3) \cdot F_1 \quad (\text{lemme 1.11})$$

$$= (\pi_2^* G - m_f(O_1)E_1 - m_f(O_2)E_2 - m_f(O_3)E_3) \cdot \widetilde{L_{2,3}}^{(\pi_2)} \quad (F_1 = \widetilde{L_{2,3}}^{(\pi_2)})$$

1. Cela fonctionne si S_j est dans \mathbb{P}^2 , mais dans le cas où S_j est sur le diviseur exceptionnel associé à S_k , il faut éclater $h(S_j)$ pour retrouver un nouveau point-base noté abusivement $h(S_k)$.

$$= (\pi_2^* G - m_g(O_2)E_2 - m_g(O_3)E_3) \cdot (\pi_2^* L_{2,3} - E_2 - E_3) \quad (\text{multiplicités de } L_{2,3})$$

$$= G \cdot L - m_g(O_2) - m_g(O_3) = \deg(g) - m_g(O_2) - m_g(O_3) \quad (\text{proposition 1.12 et Bézout})$$

On a le résultat similairement pour Q_2, Q_3 .

Maintenant, un point-base de $f \circ h^{-1}$ est nécessairement :

1. Soit un Q_i .
2. Soit l'image $h(S_j)$ d'un point-base S_j de f qui est dans \mathbb{P}^2 et tel que $S_j \notin \{O_1, O_2, O_3\}$.
3. Soit un point situé sur le diviseur exceptionnel associé à un des points-base précédents.

Le premier cas vient d'être traité.

Pour le second cas, il faut remarquer que la transformée stricte d'un élément général du système linéaire de f passe par le diviseur exceptionnel associé à S_j exactement $m_f(S_j)$ fois, et que h^{-1} ne change pas cette propriété. Pour le dernier cas, on peut montrer qu'il n'y a un tel point-base qu'au dessus des images des $h(S_j)$, et uniquement dans le cas où il y a déjà un point-base S_k au dessus de S_j , et que les multiplicités sont conservés :

- La première assertion revient à montrer que $f \circ h^{-1}$ est toujours bien définie au dessus des Q_i qui ne sont pas des $h(S_j)$. Pour simplifier les notations, on le montre pour $i = 1$: le diviseur exceptionnel associé à Q_1 est envoyé sur la droite $L_{2,3}$.² On suppose alors par l'absurde qu'il existe un x sur le diviseur exceptionnel associé à Q_1 tel que $f \circ h^{-1}$ ne soit pas définie en x . Ainsi, l'image de x sur $L_{2,3}$ est un point-base de f puisque sinon $f \circ h^{-1}$ serait définie en x , donc ce serait un S_j . Or, c'est absurde car on a supposé que Q_1 n'était pas un $h(S_j)$, et h contracte $L_{2,3}$ sur Q_1 . On a donc démontré la première assertion.
- Dans le cas où le point considéré est un $h(S_j)$ avec S_j qui n'est pas un point-base de h , il faut remarquer qu'en éclatant S_j via $\pi : S \rightarrow \mathbb{P}^2$ d'un côté et en contractant E_{S_j} sur $h(S_j)$ via $\eta : S \rightarrow \mathbb{P}^2$ de l'autre, on obtient un diagramme commutatif :

$$\begin{array}{ccc} & S & \\ \eta \swarrow & & \searrow \pi \\ \mathbb{P}^2 & \xleftarrow{\quad h \quad} & \mathbb{P}^2 \end{array}$$

Ainsi, un point-base de f qui était au dessus de S_j se retrouve bien au dessus de $h(S_j)$, et par commutativité du diagramme c'est encore un point-base de $f \circ h^{-1}$. □

On peut en fait montrer que les transformations quadratiques ayant leurs points-base dans \mathbb{P}^2 sont toutes des transformations quadratiques standard, c'est à dire de la forme $\alpha \circ \text{inv} \circ \beta$ comme ci-dessus :

Lemme 2.5

Les transformations quadratiques ayant tous leur points-base dans \mathbb{P}^2 sont toutes les mêmes à automorphisme près.

Démonstration. Soit $g : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ une transformation quadratique dont les trois points-base A, B, C sont dans \mathbb{P}^2 .

En particulier, le théorème de Bézout contraint A, B, C à ne pas être colinéaires, puisque sinon, on aurait pour une courbe D du système linéaire de \mathbb{P}^2 :

$$2 = \deg(D) \deg(L_{AB}) = m_D(A) + m_D(B) + m_D(C) + \sum_G m_D(G) m_{L_{AB}}(G) \geq 3$$

De même pour les points-base A', B', C' de f^{-1} , qui sont également tous dans \mathbb{P}^2 . Donc on peut composer g par les automorphismes $\alpha, \beta : \mathbb{P}^2 \rightarrow \mathbb{P}^2$ qui vérifient :

2. Similairement à ce qu'on a vu dans la résolution de Zariski de l'involution standard, et c'est ainsi que l'on calcule $f \circ h^{-1}$ pour un élément sur le diviseur exceptionnel de Q_1 : en prenant l'image par f de son image dans $L_{2,3}$.

- $\alpha([1 : 0 : 0]) = A$ et $\beta([1 : 0 : 0]) = B'$.
- $\alpha([0 : 1 : 0]) = B$ et $\beta([0 : 1 : 0]) = C'$.
- $\alpha([0 : 0 : 1]) = C$ et $\beta([0 : 0 : 1]) = A'$.

La transformation $f = \beta^{-1} \circ g \circ \alpha$ a alors les mêmes points-base avec même multiplicité que l'involution standard :

$$\begin{array}{ccc} \mathbb{P}^2 & \dashrightarrow & \mathbb{P}^2 \\ [x : y : z] & \mapsto & [yz : xz : xy] \end{array}$$

Par la formule sur les composée d'une transformation birationnelle avec une transformation quadratique, on calcule alors que $f \circ \text{inv}$ a pour degré $4 - 1 - 1 - 1 = 1$ et aucun point-base. Il s'agit donc d'un automorphisme $\gamma : \mathbb{P}^2 \rightarrow \mathbb{P}^2$.

Ainsi, on a $f \circ \text{inv} = \gamma$, c'est à dire $(\gamma^{-1} \circ f) \circ \text{inv} = \text{id}$, et donc $(\beta \circ \gamma)^{-1} \circ g \circ \alpha = \gamma^{-1} \circ f = \text{inv}$. \square

En particulier, par la proposition 2.4, on peut connaître la multiplicité aux points-base la composée d'une transformation birationnelle de \mathbb{P}^2 avec n'importe quelle transformation quadratique ayant ses points-base dans \mathbb{P}^2 .

Dans la suite, on supposera par défaut que "l'involution quadratique standard de points-base $A, B, C \in \mathbb{P}^2$ " désigne une transformation quadratique ayant pour points-base A, B, C (unique à automorphisme près) et vérifiant :

- L_{AB} est contractée sur C
- L_{AC} est contractée sur B
- L_{BC} est contractée sur A

En particulier, on peut regrouper toutes les informations des propositions 2.3 et 2.4 en les appliquant à ces transformations que l'on utilisera dans la preuve :

Corrolaire 2.6

Soit $f : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ une transformation birationnelle de degré n et de points-base Q_1, \dots, Q_k .

Soit $g : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ l'involution quadratique standard de points-base A, B, C .

Alors $h = f \circ g^{-1}$ a pour degré :

$$2n - m_f(A) - m_f(B) - m_f(C)$$

et ses points-base sont :

- A, B, C avec pour multiplicité :

$$\begin{cases} m_h(A) = n - m_f(B) - m_f(C) \\ m_h(B) = n - m_f(A) - m_f(C) \\ m_h(C) = n - m_f(A) - m_f(B) \end{cases}$$

- Les $g(Q_i)$ tels que $Q_i \neq A, B, C$, avec pour multiplicité :

$$m_h(g(Q_i)) = m_f(Q_i)$$

Chapitre 3

Théorème de Noether-Castelnuovo

3.1 Introduction de la preuve

On veut démontrer dans ce chapitre le théorème 0.1.

Comme la démonstration est claire pour les transformations de degré 1, on le montre pour les transformations de degré supérieur ou égal à 2.

Commençons par introduire quelques définitions qui serviront par la suite :

Définition 3.1

Soit $f : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ une transformation birationnelle du plan projectif de degré $n \geq 2$. On note O_0, \dots, O_k ses points-bases, classés par multiplicité décroissante $\alpha_0 \geq \alpha_1 \geq \dots \geq \alpha_k$.

— On note $2j_f$ la **complexité** de f , définie par :

$$j_f = \frac{n - \alpha_0}{2} \iff 2j = n - \alpha_0$$

— On note B_f l'ensemble des points-base différents de O_0 et ayant une multiplicité supérieure à j_f . On définit par ailleurs $h_f = |B_f|$, de sorte qu'ici,

$$B_f = \{O_1, \dots, O_{h_f}\}$$

On notera $j_f = j$ et $h_f = h$ lorsqu'il n'y a pas de confusion possible. En combinant les deux équations de condition du théorème 2.2, on obtient l'équation suivante dont on se servira par la suite : $\forall j \in \mathbb{Q}$,

$$\sum_{i=0}^k \alpha_i(\alpha_i - j) = (n - 1)(n - 3j + 1) \quad (3.1)$$

On peut alors donner quelques propriétés sur la complexité $2j$ et l'ensemble B_f d'une transformation f de degré supérieur à 1 :

— On note d'abord que $2j \geq 1$, puisque $\deg(f) \geq 2$. En effet, par le théorème de Bézout, une droite passant par O_0 coupe forcément une courbe générale du système linéaire de C en au moins deux points, et donc :

$$n = \alpha_0 + \sum_p m_p(C) > \alpha_0$$

— De plus, on peut borner les multiplicité des points de B_f : par le théorème de Bézout appliqué à la droite $L_{0,1} = L_{O_0 O_1}$ et un élément C du système linéaire de f , on a :

$$\deg(L_{0,1}) \deg(C) = n = C \cdot L_{0,1}$$

Or, $C \cdot L_{0,1} = \alpha_0 m_{L_{0,1}}(O_0) + \alpha_1 m_{L_{0,1}}(O_1) + \sum_p m_C(p) m_{L_{0,1}}(p) = \alpha_0 + \alpha_1 + \sum_p m_C(p)$. Ainsi,

$$2j = n - \alpha_0 = \alpha_1 + \sum_p m_C(p) \geq \alpha_1$$

Finalement, on obtient les bornes voulues :

$$2j \geq \alpha_1 \geq \dots \geq \alpha_h > j \quad (3.2)$$

On a de plus une condition sur h_f , qui montre que B_f n'est jamais vide :

Lemme 3.2

Soit $f : \mathbb{P}^2 \dashrightarrow \mathbb{P}^2$ une transformation birationnelle de degré $n \geq 2$. Alors h_f est toujours strictement plus grand que 1.

Démonstration. On soustrait à l'équation (3.1) $3j - 1 > 0$ à droite et $\sum_{i=h_f+1}^k \alpha_i(\alpha_i - j) \leq 0$ à gauche, de sorte qu'on a l'inégalité :

$$\sum_{i=0}^{h_f} \alpha_i(\alpha_i - j) > (n-1)(n-3j+1) - (3j-1) = n(n-3j) - (n-3j) + n-1-3j+1 = n(n-3j)$$

Or, $n-3j = n-2j-j = \alpha_0-j$, donc :

$$\begin{aligned} \sum_{i=0}^{h_f} \alpha_i(\alpha_i - j) &> n(\alpha_0 - j) \\ &> n(\alpha_0 - j) - \alpha_0(\alpha_0 - j) \\ &> (n - \alpha_0)(\alpha_0 - j) \\ &> 2j(\alpha_0 - j) \end{aligned}$$

Par (3.2), on obtient en majorant les α_i à gauche puis en simplifiant :

$$\begin{aligned} 2j \sum_{i=1}^{h_f} (\alpha_i - j) &> 2j(\alpha_0 - j) \\ \sum_{i=1}^{h_f} (\alpha_i - j) &> \alpha_0 - j \end{aligned} \quad (3.3)$$

Ceci n'est pas possible si l'on a pas $h_f \geq 2$ puisque $\alpha_1 \leq \alpha_0$. □

Exemple 4 (Transformations de De Jonquières)

On peut étudier le cas spécial où f est une transformation birationnelle de complexité $2j = 1$. Une transformation de ce type s'appelle *transformation de De Jonquières*.

O_0 a alors pour multiplicité $\deg(f) - 1$ par définition de la complexité. Et les autres points sont simples puisque leur multiplicité est bornée par $\frac{1}{2}$ et 1 par les inégalités 3.2. La première équation de condition fournit alors :

$$\deg(f) - 1 + |\{\text{Points-base simples de } f\}| = 3(\deg(f) - 1)$$

Et ainsi, f a un point-base de multiplicité $\deg(f) - 1$ et $2(\deg(f) - 1)$ points-base simples.

3.2 Preuve du théorème de Noether-Castelnuovo

Soit f une transformation birationnelle de \mathbb{P}^2 de degré $n \geq 2$. Montrons dans un premier temps que si les points O_0, \dots, O_h de B_f ne sont pas tous dans \mathbb{P}^2 , on peut précomposer f par des transformations quadratiques pour les rendre ramener dans \mathbb{P}^2 :

Proposition 3.3

Soit f une transformation birationnelle de \mathbb{P}^2 de degré $n \geq 2$. On reprend pour les points-base de f les notations précédentes.

Alors il existe des involutions quadratiques standards p_1, \dots, p_m telles que $f' = f \circ p_1^{-1} \circ \dots \circ p_m^{-1}$ vérifie :

- $2j_{f'} = 2j_f$
- $h_{f'} = h_f + 2$
- O_0 est le point-base de plus grande multiplicité de f' et tous les points de $B_{f'}$ sont dans \mathbb{P}^2 .

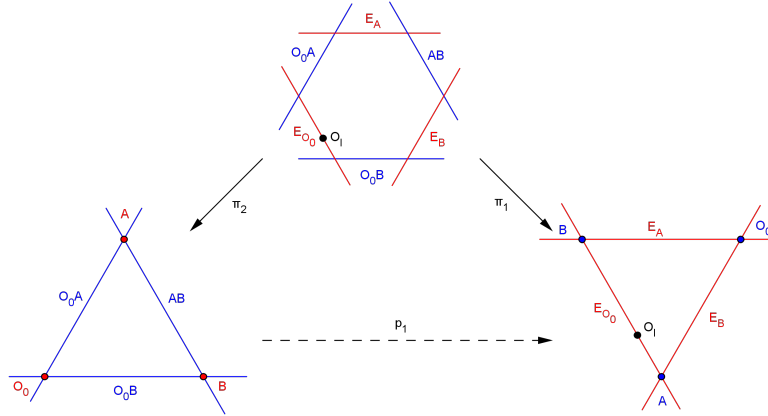
Démonstration. La preuve s'effectue en deux étapes :

Séparation des points-base en O_0 : Soient A, B deux points de \mathbb{P}^2 distincts des points-base de f , tels que O_0, A, B soient en position générale. On considère alors l'involution quadratique standard p_1 de points-base O_0, A, B . Par le théorème 2.6, on a alors le degré et les nouvelles multiplicité de $f' = f \circ p_1^{-1}$ aux points-base O_0, A, B :

$$\begin{cases} \deg(f') &= 2n - \alpha_0 = n + 2j > n \\ m_{f'}(O_0) &= n = 2j + \alpha_0 \\ m_{f'}(A) &= n - \alpha_0 = 2j \\ m_{f'}(B) &= n - \alpha_0 = 2j \end{cases}$$

Ainsi, la multiplicité de O_0 augmente et reste la plus grande, et deux nouveaux points-base sont introduit, de multiplicité $2j_f > j_f$. f' a encore $2j_f$ pour complexité, puisque $2j_{f'} = \deg(f') - m_{f'}(O_0) = 2n - \alpha_0 - n = n - \alpha_0 = 2j_f$. Donc par le calcul des nouvelles multiplicités, $h_{f'} = h_f + 2$.

De plus, si $1 \leq l \leq h_f$ est tel que O_l est infiniment proche de O_0 , son image par p_1^{-1} notée encore O_l est alors sur la droite AB , par la résolution de p_1 par le théorème de Zariski :



Donc tous les points qui étaient infiniment proches de O_0 sont désormais dans \mathbb{P}^2 .

Séparation des autres points-base : On note à nouveau $n = \deg(f')$. Soient $O' = O_i \in B_f$ qui est tel qu'un autre point-base O'' est infiniment proche de O' . On considère l'involution quadratique standard p_2 de points-base O_0, O', C , avec C en position générale. Encore par le théorème 2.6, la transformation $f'' = f' \circ p_2^{-1}$ a alors pour degré et pour multiplicité aux points-base O_0, O', C :

$$\begin{cases} \deg(f'') &= 2n - \alpha_0 - \alpha_i = n + 2j - \alpha_i > n \\ m_{f''}(O_0) &= n - \alpha_i \\ m_{f''}(O') &= n - \alpha_0 = 2j \\ m_{f''}(C) &= n - \alpha_0 - \alpha_i \end{cases}$$

O_0 est encore le point de multiplicité la plus grande, puisque par (3.2), $n - \alpha_i \geq n - 2j = \alpha_0$, et donc comme précédemment, $2j_{f''} = 2j_f$. De plus, $C \notin B_{f''}$, puisque $n - \alpha_0 - \alpha_i = 2j_f - \alpha_i < 2j_f - j_f < j_f$. Donc

$$h_{f''} = h_{f'}.$$

D'autre part, soit $O'' = O_j \in B_{f'}$, puisque la droite $L_{0,j}$ est contractée sur C , il faut vérifier deux choses afin d'éviter que d'autres points-bases ne soient infiniment proches à la suite de cette transformation :

- Que O'' ne peut pas être sur la droite $L_{0,j}$
- Que, si O'' est infiniment proche de O' , O'' ne soit pas sur la droite $L_{0,j}$ dans la variété S de la résolution de p_2 donnée par le théorème de Zariski (i.e. que O'' ne soit pas l'intersection de du diviseur exceptionnel $E_{O'}$ associé à O' et de $\widetilde{L_{0,j}}^{(\pi_2)} = \pi_2^{-1}(L_{0,j} \setminus \{O_0, O'\})$ dans S) :

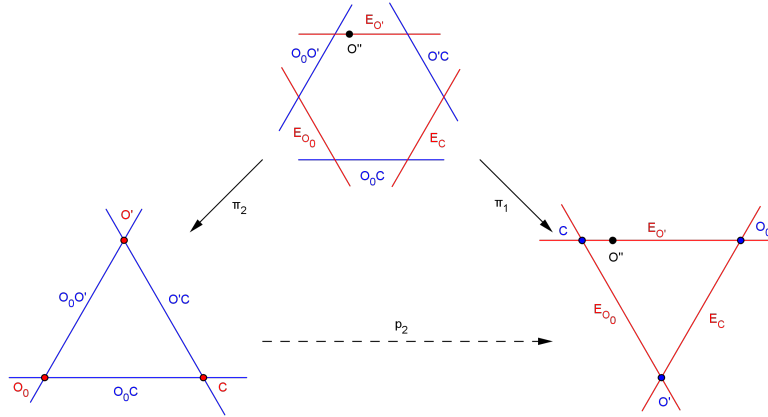
$$\begin{array}{ccc} & S & \\ \pi_2 \swarrow & & \searrow \pi_1 \\ \mathbb{P}^2 & \xrightarrow{p_2} & \mathbb{P}^2 \end{array}$$

Le théorème de Bézout empêche ces deux cas d'arriver, puisque dans les deux cas, le produit d'intersection de la droite $L_{0,j}$ et d'un membre général C' du système linéaire de f' donnerait :

$$n = L_{0,j} \cdot C' = \alpha_0 + \alpha_i + \alpha_j + \sum_p m_{f'}(p)$$

Donc $2j > \alpha_i + \alpha_j$, et par (3.2), on obtiendrait $2j > j + j$.

Ainsi, ces situations particulières ne peuvent pas arriver, et donc les points-bases qui ne sont pas concernés par p_2 restent inchangés pendant que O' et O'' sont séparés par p_2 comme le montre le diagramme suivant :



En itérant cette seconde étape pour tous les points-base de multiplicité supérieure à j qui sont infiniment proches, on peut les séparer un par un, et puisque C est en position générale, les autres points-base ne changent pas.

Finalement, la transformation obtenue a tous ses points-base de multiplicité supérieure à j dans \mathbb{P}^2 . \square

Lemme 3.4

Soit f une transformation birationnelle de \mathbb{P}^2 de degré $n \geq 2$ telle que tous les points de B_f sont dans \mathbb{P}^2 . On note $h = h_f$ et $j = j_f$, et on reprend les notations précédentes pour les points-base de f .

Alors, il existe une involution quadratique standard p de \mathbb{P}^2 telle que $f' = f \circ p^{-1}$ vérifie la disjonction suivante :

1. Si O_0 est encore le point-base de f' de plus grande multiplicité, alors tous les points de $B_{f'}$ sont dans \mathbb{P}^2 , $2j_{f'} = 2j$ et $h_{f'} = h - 2$.

2. Sinon, $j_{f'} < j$.

Démonstration. Soient $O_i, O_j \in B_f$. Alors, O_0, O_i, O_j ne sont pas alignés dans \mathbb{P}^2 :

En effet si ce n'était pas le cas, on aurait par le théorème de Bézout appliqué à la droite O_0O_i et à une courbe C du système linéaire de f :

$$n = \alpha_0 + \alpha_i + \alpha_j + \sum_p m_C(p) \geq \alpha_0 + \alpha_i + \alpha_j$$

Et donc $2j \geq \alpha_i + \alpha_j > j + j = 2j$, ce qui est absurde.

On peut donc considérer l'involution quadratique standard g de \mathbb{P}^2 de points-base O_0, O_i, O_j , et la transformation $f' = f \circ g^{-1}$. Le théorème 2.6 fournit le degré et les nouvelles multiplicités aux points concernés :

$$\begin{cases} \deg(f') &= 2n - \alpha_0 - \alpha_i - \alpha_j = n + 2j - \alpha_i - \alpha_j < n \\ m_{f'}(O_0) &= n - \alpha_i - \alpha_j \\ m_{f'}(O_i) &= j - \alpha_j \\ m_{f'}(O_j) &= j - \alpha_i \end{cases}$$

la multiplicité des autres points-base restant inchangée. On a alors la disjonction suivante :

— Si O_0 est le point-base de plus grande multiplicité de f' , alors la complexité de f' est donnée par :

$$2j_{f'} = \deg(f') - m_{f'}(O_0) = (2n - \alpha_0 - \alpha_i - \alpha_j) - (n - \alpha_i - \alpha_j) = n - \alpha_0 = 2j$$

Donc f' a encore $2j$ pour complexité, et donc O_i et O_j ne sont plus des points de multiplicité supérieure à $j_{f'} = j$. On en déduit que $B_{f'}$ a deux points en moins par rapport à f .

— Sinon, alors soit O_l le point-base de multiplicité la plus grande, on a alors $m_{f'}(O_l) > m_{f'}(O_0)$, et donc :

$$2j_{f'} = \deg(f') - m_{f'}(O_l) < \deg(f') - m_{f'}(O_0) = 2j$$

La complexité de f' est donc inférieure à celle de f .

□

On peut alors combiner la proposition 3.3 et le lemme 3.4 :

Proposition 3.5

Soit f une transformation birationnelle de \mathbb{P}^2 de degré $n \geq 2$. On note $h = h_f$ et $j = j_f$, et on reprend pour les points-base de f les notations précédentes.

Alors, il existe une composée g d'involutions quadratiques standards de \mathbb{P}^2 telles que la transformation $f' = f \circ g^{-1}$ vérifie la disjonction :

1. Si O_0 est encore le point-base de f' de plus grande multiplicité, alors $j_{f'} = j$ et $h_{f'} = h - 2$.
2. Sinon, $2j_{f'} < 2j$.

Démonstration. On applique tout d'abord la proposition 3.3 à f : il existe une composée de transformations quadratique q_1 telle que $f_1 = f \circ q_1^{-1}$ possède encore O_0 comme point-base de plus grande multiplicité, et vérifie $2j_{f_1} = 2j$ et $h_{f_1} = h + 2$.

On applique alors une première fois le lemme 3.4 à f_1 : on obtient la transformation $f_2 = f_1 \circ q_2^{-1}$. On a alors la disjonction du lemme :

- Dans le deuxième cas du lemme 3.4, f_2 a une complexité inférieure à $2j$, et donc la transformation $f_2 = f \circ (q_1 \circ q_2)^{-1}$ vérifie le deuxième cas de la proposition.
- Dans le premier cas du lemme 3.4, $2j_{f_2} = 2j$ et $h_{f_2} = h_{f_1} - 2 = h$, tous les points de B_{f_2} étant dans \mathbb{P}^2 . On applique alors à nouveau le lemme 3.4 à f_2 , et la transformation obtenue $f_3 = f_2 \circ q_3^{-1} = f \circ (q_3 \circ q_2 \circ q_1)^{-1}$ vérifie bien la proposition.

□

On peut maintenant rassembler ces résultats pour obtenir la démonstration du théorème 0.1 :

Démonstration du théorème de Noether-Castelnuovo. On va procéder par double récurrence, sur la complexité et sur le nombre de points-base de multiplicité supérieure à cette complexité pour démontrer le théorème.

On pose l'hypothèse de récurrence :

$H_{2j,h}$ = "Toute transformation birationnelle du plan projectif vérifiant $2j_f = 2j$ et $h_f = h$ se décompose en une composée de transformée quadratiques et d'automorphismes".

- **Initialisation en j** : Montrons que pour tout $h \geq 2$, on a bien $H_{1,h}$ (i.e. que les transformations de De Jonquières, vues dans l'exemple 4, se décomposent en transformations quadratiques. On effectue pour cela une récurrence sur h :
 - *Initialisation en h* : Il s'agit de montrer vérifier $H_{1,2}$. Soit f une transformation de De Jonquières avec $h_f = 2$. Comme une telle transformation a $2(\deg(f) - 1)$ points-base simples, on en déduit que $\deg(f) = 2$, et donc f est une transformation quadratique.
 - *Hérédité en h* : Soit $h > 2$. On suppose $H_{1,h-1}$ et on veut montrer $H_{1,h}$. Soit f une transformation de De Jonquières avec h points-base simples. On applique la proposition 3.5 à f et on obtient une transformation $f' = f \circ g^{-1}$ qui est encore de De Jonquières, et a $h - 2$ points-base simples. Par l'hypothèse $H_{1,h-1}$, f' se décompose alors en une composée de transformations quadratiques, et donc f également.
- **Hérédité en j** : Soit $2j < 1$ fixé. On suppose qu'on a pour tout $h \geq 2$, $H_{2(j-1),h}$. Montrons qu'on a encore pour tout $h \geq 2$, $H_{2j,h}$. Pour cela, on démontre encore la récurrence en h :
 - *Initialisation en h* : Montrons qu'on a bien $H_{2j,2}$.
 Soit f une transformation birationnelle de \mathbb{P}^2 de complexité $2j$ telle que $h_f = 2$. On reprend pour les points-base de f les notations précédentes.
 En appliquant la proposition 3.5 à f , on obtient une transformation $f' = f \circ g^{-1}$ qui a une complexité inférieure à $2j$, puisque $h < 3$.¹ Par l'hypothèse $H_{2(j-1),h'}$ pour h' assez grand, f' est bien une composée de transformation quadratiques, et donc f également puisque $f = f' \circ g^{-1}$.
 - *Hérédité en h* : Soit $h > 2$. On suppose $H_{2j,h-1}$. Montrons alors $H_{2j,h}$.
 Soit f une transformation birationnelle de \mathbb{P}^2 de complexité $2j$ telle que $h_f = h$. On reprend pour les points-base de f les notations précédentes.
 On applique encore la proposition 3.5 à f et on obtient la transformation $f' = f \circ g^{-1}$ vérifiant la disjonction de la proposition :
 1. Soit f' a la même complexité que f et $h_{f'} = h - 2$. Par l'hypothèse $H_{2j,h-1}$, f' est une composée de transformations quadratiques, et donc $f = f' \circ g^{-1}$ l'est également.
 2. Soit $2j_{f'} < 2j$, et donc par l'hypothèse $H_{2(j-1),h'}$ pour h' assez grand, f' est une composée de transformations quadratiques et c'est encore le cas pour f .

Finalement, on a montré que pour tout $h \geq 2$, $H_{2j,h}$, ce qui démontre l'hérédité en j .

Ainsi, on a démontré la double récurrence, et donc toute transformation birationnelle se décompose en transformations quadratiques. \square

Remarque

On a en fait montré un résultat un peu plus fort que celui donné par le théorème 0.1, puisque l'on utilise ici que des involutions quadratiques standardes² ayant tous leur points-base dans \mathbb{P}^2 , qui sont égales à l'involution

1. En effet, sinon cette transformation vérifierait $h_{f'} = 1$, ce qui contredit le lemme 3.2.
 2. Dans l'initialisation de h , si jamais la transformation quadratique obtenue à la fin n'est pas une involution quadratique standard, on peut encore la décomposer, comme on le fera au chapitre suivant.

standard à automorphisme près par le lemme 2.5 :

$$[x : y : z] \dashrightarrow [yz : xz : xy]$$

Le résultat que l'on a finalement montré est que l'involution standard et les automorphismes engendrent les transformations birationnelles de \mathbb{P}^2 .

Chapitre 4

Applications du théorème de Noether-Castelnuovo

Le caractère algorithmique de la preuve de J.W. Alexander permet de décomposer facilement une transformations de \mathbb{P}^2 en connaissant ses points-base avec multiplicité.

Dans ce chapitre, on applique l'algorithme à certaines transformations, d'abord de manière calculatoire, puis en décrivant ces transformations uniquement avec leurs points-base et multiplicités, les calculs devenant rapidement compliqués.

On appliquera donc d'abord l'algorithme de manière numérique pour les transformations quadratiques, puis de manière abstraite pour les transformations de degrés 3 et 5.

Enfin, on parlera du test de Hudson, qui permet de connaître exactement, étant donné un entier positif, le nombre de points-base avec multiplicité possible pour une transformation birationnelle de ce degré. On utilisera ce test dans l'implémentation informatique de l'algorithme de J.W. Alexander, donnée en annexe.

4.1 Transformations quadratiques

On a étudié précédemment les transformations quadratiques ayant tous leurs points-base dans \mathbb{P}^2 , mais ces derniers peuvent également être infiniment proches comme on l'a vu dans l'algorithme, et accessibles seulement par éclatement d'un autre point-base.

Les équations de condition appliquée à une transformation quadratique permet de dire que toute transformation quadratique a exactement trois points-base simples¹. Il s'agit donc de voir où se trouvent ces points-base :

4.1.1 Deux points-base dans \mathbb{P}^2 et un point-base infiniment proche

On cherche une transformation ayant pour points-base $[1 : 0 : 0]$ et $[0 : 1 : 0]$ dans \mathbb{P}^2 et contractant la droite passant par ces deux points. Considérons la transformation :

$$f : [x : y : z] \dashrightarrow [xz : y(z - x) : z(z - x)]$$

Cette transformation a bien pour points-base $[1 : 0 : 0], [0 : 1 : 0]$ dans \mathbb{P}^2 . On considère l'éclatement de \mathbb{P}^2 en $[0 : 1 : 0] : \pi : Bl_{[0:1:0]}\mathbb{P}^2 \rightarrow \mathbb{P}^2$ avec

$$Bl_{[0:1:0]}\mathbb{P}^2 = \{([x : y : z], [u : v]) \in \mathbb{P}^2 \times \mathbb{P}^1 \mid xu = zv\}$$

Le diviseur exceptionnel est $\pi^{-1}([0 : 1 : 0]) = \{([0 : 1 : 0], [u : v]) \in \mathbb{P}^2 \times \mathbb{P}^1\}$.

$f \circ \pi$ est alors définie sur le diviseur exceptionnel, excepté sur un point : on regarde l'ouvert $\{u \neq 0\} = \{([zv :$

1. $4 - 1 = \sum_{i=1}^k \alpha_i^2$, avec $\alpha_i \geq 1$, d'où le résultat.

$y : z], [1 : v]) \in \mathbb{P}^2 \times \mathbb{P}^1\}$. Sur cet ouvert, on a :

$$f \circ \pi([zv : y : z], [1 : v]) = f([zv : y : z]) = [z^2v : zy(1-v) : z^2(1-v)] = [zv : y(1-v) : z(1-v)]$$

On observe alors que f n'est pas définie sur un point de l'ouvert intersecté avec la droite $\{z = 0\}$, le point $([0 : 1 : 0], [1 : 1])$, qui est sur le diviseur exceptionnel E . C'est le troisième point-base de la transformation. Cette transformation est donc bien du type voulu.

On peut maintenant appliquer l'algorithme pour décomposer f : on choisit $O_0 = [1 : 0 : 0], O_1 = [0 : 1 : 0], O_2 = ([0 : 1 : 0], [1 : 1])$.

En précomposant avec la transformation quadratique standard, puisque $O_0, O_1, O_3 = [0 : 0 : 1]$ sont en position générale, on obtient la transformation :

$$h : [x : y : z] \dashrightarrow f([yz : xz : yx]) = [y^2zx : xz(yx - yz) : yx(yx - yz)] = [yz : z(x - z) : y(x - z)]$$

Cette transformation a pour points-base les points O_0, O_1 et un point sur la droite O_0O_3 dispersé par l'involution, que l'on reconnaît comme étant le point $[1 : 0 : 1]$.

Comme les points O_0, O_1, O_3 ne sont pas colinéaires, on a décomposé f en une composée de deux transformations quadratiques à points-base dans \mathbb{P}^2 : $f = h \circ inv$.

4.1.2 Trois points-base infiniment proches

On veut cette fois une transformation possédant un seul point-base dans \mathbb{P}^2 . La contrainte sur les multiplicités permet de dire que ces trois points-base ne sont pas alignés, et donc il faut nécessairement deux éclatements pour les révéler.

Il faut donc chercher une transformation ayant un point-base O_0 dans \mathbb{P}^2 , un point-base O_1 sur le diviseur exceptionnel E_{O_0} dans $Bl_{O_0}\mathbb{P}^2$ et un point-base O_2 sur le diviseur exceptionnel E_{O_1} dans $Bl_{O_1}(Bl_{O_0}\mathbb{P}^2)$.

On considère la transformation :

$$f : [x : y : z] \dashrightarrow [xz : yz - x^2 : z^2]$$

Elle possède un seul point-base dans \mathbb{P}^2 : $O_0 = [0 : 1 : 0]$. On vérifie que la transformation possède bien les propriétés mentionnées.

En éclatant \mathbb{P}^2 en ce point :

$$Bl_{[0:1:0]}\mathbb{P}^2 = \{([x : y : z], [u : v]) \in \mathbb{P}^2 \times \mathbb{P}^1 \mid xu = zv\}$$

$f \circ \pi_1$ est définie sur le diviseur exceptionnel E_{O_0} sauf sur un point. Dans l'ouvert $\{v \neq 0\} = \{([x : y : xu], [u : 1]) \in \mathbb{P}^2 \times \mathbb{P}^1\}$, on a :

$$f \circ \pi_1([x : y : xu], [u : 1]) = f([x : y : xu]) = [x^2u : xyu - x^2 : x^2u^2] = [xu : yu - x : xu^2]$$

On remarque que le point $O_1 = ([0 : 1 : 0], [0 : 1])$ est un point-base de $f \circ \pi_1$ présent dans cet ouvert. On peut le regarder dans la carte affine de $Bl_{[0:1:0]}\mathbb{P}^2$ définie par $\{v \neq 0, y \neq 0\}$. Cette carte est isomorphe au plan affine via :

$$\begin{aligned} ([x : 1 : z], [u : 1]) &\rightarrow (x, u) \\ ([x, 1, xu], [1, v]) &\leftarrow (x, u) \end{aligned}$$

Et donc O_1 correspond dans cette carte au point $(0, 0)$ et la droite $Vect((0, 1))$ correspond au diviseur exceptionnel E_{O_0} . On définit alors la fonction f sur le plan affine via l'isomorphisme de celui-ci avec la carte pour simplifier les calculs :

$$\tilde{f}(x, u) := f([x : 1 : xu]) = [xu : u - x : xu^2]$$

On effectue alors l'éclatement local de \mathbb{A}^2 en $(0, 0)$:

$$Bl_{(0,0)}\mathbb{A}^2 = \{((x, y), [s : t]) \in \mathbb{A}^2 \times \mathbb{P}^1 \mid xs = yt\}$$

Dans l'ouvert affine $\{t \neq 0\}$, on a alors

$$\tilde{f} \circ \pi((x, xs), [s : 1]) = \tilde{f}(x, xs) = [x^2s : xs - x : x^3s^2] = [xs : s - 1 : x^2s^2]$$

On observe alors que le point $((0, 0), [1 : 1])$ est un point-base de \tilde{f} , qui est présent sur le diviseur exceptionnel E_{O_1} . On a donc finalement trouvé trois points-base infiniment proches pour f . Cette transformation est donc bien du type voulu.

On peut alors appliquer l'algorithme à cette transformation :

Il faut commencer par disperser les points-bases qui sont infiniment proches du premier point O_0 . Comme dans la preuve vue avant, on prend deux autres points A et B tels que O_0, A, B soient en position générale, disons $A = [0 : 0 : 1]$ et $B = [1 : 0 : 1]$, de sorte que O_1 ne soit pas envoyé sur A ou B , mais bien entre les deux. Cette involution est donnée par :

$$g_0 : [x : y : z] \dashrightarrow [x : y : z + x] \circ [yz : xz : xy] \circ [x : y : z - x] = [y(z - x) : x(z - x) : yz]$$

On peut alors calculer la nouvelle transformation, qui devrait avoir quatre points-base dans \mathbb{P}^2 :

$$\begin{aligned} f' = f \circ g_0 &= [xz : yz - x^2 : z^2] \circ [y(z - x) : x(z - x) : yz] \\ &= [y^2z(z - x) : xyz(z - x) - y^2(z - x)^2 : y^2z^2] \\ &= [yz(z - x) : xz(z - x) - y(z - x)^2 : yz^2] \end{aligned}$$

Cette transformation est bien de degré 3 et a quatre points-base dans $\mathbb{P}^2 : [0 : 1 : 0], [0 : 0 : 1], [1 : 0 : 1]$ et $[1 : 0 : 0]$. Les trois points $[0 : 0 : 1], [1 : 0 : 1]$ et $[1 : 0 : 0]$ sont des points-base simples et $O_0 = [0 : 1 : 0]$ est un point-base double, puisque si L est une droite générale de \mathbb{P}^2 correspondant aux coefficients $[a : b : c]$, le tiré-en-arrière de L est donné par la cubique $ayz(z - x) + b(xz(z - x) - y(z - x)^2) + cyz^2 = 0$, qui a bien les multiplicités voulues aux points-base de f' .

De plus le dernier point correspond au point qui était sur le diviseur exceptionnel associé à O_0 , puisqu'il n'était pas présent dans les points-base de g (corollaire 2.6). Le dernier point-base est donc sur ce point, comme on l'a vu au paragraphe précédent.

On va donc à nouveau composer avec une involution quadratique standard de points-base $[0 : 1 : 0], [1 : 0 : 0]$ et un point en position générale, disons $[1 : 2 : 3]$. Une telle involution est donnée par la composition $[x + z : y + 2z : 3z] \circ [yz : xz : xy] \circ [3x - z : 3y - 2z : z]$, c'est à dire :

$$g_1 : [x : y : z] \dashrightarrow [x(3y - 2z) : (2y - z)(3x - z) : (3x - z)(3y - 2z)]$$

On calcule alors la nouvelle transformation :

$$\begin{aligned} f'' = f' \circ g_1 &= [yz(z - x) : xz(z - x) - y(z - x)^2 : yz^2] \circ [x(3y - 2z) : (2y - z)(3x - z) : (3x - z)(3y - 2z)] \\ &= [(2y - z)(3x - z)(2x - z) : x(3y - 2z)(2x - z) - (2y - z)(2x - z)^2 : (2y - z)(3x - z)^2] \end{aligned}$$

C'est une transformation de degré 3, ayant pour points-base :

- $[0 : 1 : 0], [1 : 0 : 0]$ comme points-base de g_1 ($[1 : 2 : 3]$ étant directement dispersé).
- $[1 : 3 : 3]$ comme nouveau point-base correspondant à celui qui était sur le diviseur exceptionnel associé à $[1 : 0 : 0]$.²
- $g_1([0 : 0 : 1]) = [0 : 1 : 2]$ et $g_1([1 : 0 : 1]) = [1 : 1 : 2]$

Le point $[0 : 1 : 0]$ est de multiplicité 2 et les autres sont de multiplicité 1. Comme c'est le seul type homaloïdal pour une transformation de degré 3, tous les points-base sont bien dans \mathbb{P}^2 .

On traite, dans la section suivante, de la décomposition des transformations birationnelles de degré 3.

Remarque

Ici on aurait aussi pu décomposer, à partir du deuxième stade, avec l'involution standard, et on se serait retrouvé dans le cas d'une transformation quadratique avec deux points-base dans \mathbb{P}^2 , que l'on a vu à la section précédente.

2. Pour le trouver, on le cherche sur la droite $z = 3x$ passant par $[0 : 1 : 0]$ et $[1 : 2 : 3]$, et donc de la forme $[x : y : 3x]$. On remplace alors dans l'équation de f'' .

4.2 Transformation de degré 3

Soit f une transformation de degré 3. On regarde quelles sont les possibilités de points-base pour f :

- La première équation de condition fournit que la somme des carrés des multiplicités vaut 8.
- La second fournit que la somme des multiplicités vaut 6.

La seule possibilité est donc un point-base de multiplicité 2 et quatre points-base simples.

On suppose dans un premier temps qu'ils sont tous dans \mathbb{P}^2 et on les note O_0, \dots, O_5 , classés par multiplicité décroissante.

Cette transformation a pour complexité $2j_f = 2$. En appliquant l'algorithme, on peut alors regarier l'involution quadratique standard g de points-base O_0, O_4, O_5 . Les multiplicités obtenues après composition pour $h = f \circ g^{-1}$:

- $m_h(O_0) = 3 - 1 - 1 = 1$
- $m_h(O_4) = 3 - 2 - 1 = 0$
- $m_h(O_5) = 3 - 2 - 1 = 0$

Les autres multiplicités restent les mêmes.

On observe que la nouvelle transformation h a trois points-base de multiplicité 1 dans \mathbb{P}^2 . C'est donc une transformation quadratique de la forme voulue.

Finalement, on a décomposé cette transformation de degré 3 en une composée de deux transformations quadratiques.

On peut maintenant regarder ce qu'il se passe lorsqu'un point-base est infiniment proche d'un autre. Imaginons par exemple que le point-base O_5 est infiniment proche du point-base O_0 .

Comme dans l'algorithme, on considère deux points O_6 et O_7 de \mathbb{P}^2 tels que O_0, O_6, O_7 soient en position générale.

On effectue l'involution quadratique standard g_1 de points-base O_0, O_6, O_7 et on note $f_1 = f \circ g^{-1}$.

Avec les formules, f_1 est de degré 4 et a pour nouvelle multiplicité en O_0, O_6, O_7 :

- $m_{f_1}(O_0) = 3$
- $m_{f_1}(O_6) = 3 - 2 = 1$
- $m_{f_1}(O_7) = 3 - 2 = 1$

On a à nouveau une complexité $2j_{f_1} = 2$, et deux nouveaux points-base simples. On poursuit alors l'algorithme jusqu'à décomposer la transformation :

On précompose f_1 avec l'involution quadratique standard g_2 de points-base O_0, O_6, O_7 , et les nouvelles multiplicités de $f_2 = f_1 \circ g_2^{-1}$ est une transformation de degré 3 ayant pour nouvelles multiplicités aux points O_0, O_6, O_7 :

- $m_{f_2}(O_0) = 4 - 1 - 1 = 2$
- $m_{f_2}(O_6) = 4 - 3 - 1 = 0$
- $m_{f_2}(O_7) = 4 - 3 - 1 = 0$

On a donc deux points-base en moins, et on obtient donc que f_2 est une transformation de degré 3 ayant tous ses points-base dans \mathbb{P}^2 . On peut donc appliquer la décomposition ci-dessus pour avoir que f_2 se décompose en une composée de deux transformations quadratiques ayant leurs points-base dans \mathbb{P}^2 : $f_2 = g_3 \circ g_4$. Finalement, on a décomposé f en une composée de quatre transformations quadratiques ayant leurs points-base dans \mathbb{P}^2 .

4.3 Transformation de degré 5

On considère une transformation f de degré 5 ayant tous ses points-base dans \mathbb{P}^2 . Appliquons l'algorithme pour la décomposer en transformations quadratiques.

Déterminons d'abord les multiplicités possibles pour f :

- La première équation de condition fournit que la somme des carrés des multiplicités vaut 24.
- La second fournit que la somme des multiplicités vaut 12.

On a donc les possibilités suivantes :

1. Un point-base de multiplicité 4 et huit points-base simples.
2. Deux points-base de multiplicité 3 et six points-base simples.
3. Un point-base de multiplicité 3, trois points-base de multiplicité 2 et trois points-base simples

4. Six points-base de multiplicité 2.

Décomposons la dernière transformation.

Tout d'abord, sa complexité est $2j_f = 3$. On note g_1 l'involution quadratique standard de points-base O_0, O_1, O_2 , et $f_1 = f \circ g_1^{-1}$. C'est une transformation de degré 4 ayant 1 pour nouvelles multiplicité aux points-base O_0, O_1, O_2 . O_0 n'est plus le point-base de plus grande multiplicité et la complexité a donc baissé :

$$2j_{f_1} = 2$$

On applique alors l'involution quadratique standard g_2 de points-base O_3, O_4, O_5 . On obtient une transformation $f_2 = f_1 \circ g_2^{-1}$ de degré 2, ayant 0 pour nouvelle multiplicité en ces points, qui ne sont donc plus des points-base de f_2 .

f_2 est une transformation quadratique ayant ses points-base O_0, O_1, O_2 dans \mathbb{P}^2 avec multiplicité 1.

On a donc décomposé f en une composée de trois transformations quadratiques ayant leur points-base dans \mathbb{P}^2 .

4.4 Test de Hudson

Comme le montrent les exemples précédents, pour désigner et manipuler une transformation rationnelle de grand degré, il peut-être préférable de la désigner par ses points-base et leur multiplicité, ainsi que par les courbes qu'elle contracte, ce qui les caractérise à automorphisme près.

Dans le cadre de l'algorithme de J.W. Alexander, les points-base d'une transformation birationnelle et leur multiplicité forment des éléments intéressants pour classer les transformations birationnelles du plan projectif en familles qui auront presque la même décomposition par l'algorithme puisque les multiplicités sont des quantités numériques, qui peuvent être calculées par les équations de condition.

On peut alors considérer le problème inverse : étant donnée un degré et une liste de multiplicités calculée avec les équations de condition, ces quantités définissent-elles une famille de transformations birationnelles, et comment le savoir ?

L'algorithme de J.W. Alexander fournit une réponse à cette question : il suffit de supposer qu'il existe une transformation birationnelle du plan projectif qui a le degré voulu et le nombre de points-base avec multiplicité voulu, et si on arrive à la décomposer via l'algorithme, alors c'est qu'elle existait bien au départ (il suffit de recomposer les transformations quadratiques – dont on sait bien qu'elles existent – pour l'obtenir).

Le test de Hudson, bien que sa preuve soit plus compliquée que cela, introduit un moyen de vérification proche de cette intuition. C'est un test purement numérique que l'on peut donc facilement implémenter en programmation. On peut trouver une démonstration de la proposition 4.2 qui énonce l'exactitude du test dans l'article de Jérémy Blanc et Alberto Calabri [2].

Définition 4.1 (Type homaloïdal)

On appelle **type homaloïdal** un couple $(d, (m_0, \dots, m_k)) \in \mathbb{N} \times \mathbb{Z}^{k+1}$ avec $m_0 \geq \dots \geq m_k$ vérifiant les équations de condition :

$$\begin{aligned} \sum_{i=0}^k m_i &= 3(d-1) \\ \sum_{i=0}^k m_i^2 &= d^2 - 1 \end{aligned}$$

Un type homaloïdal $(d, (m_0, \dots, m_k))$ est dit **propre** s'il existe une transformation birationnelle de \mathbb{P}^2 de degré d et ayant $k+1$ points-base de multiplicités respectives m_0, \dots, m_k . Il est dit **impropre** si ce n'est pas le cas.

Le problème consiste donc à trouver une méthode permettant de dire si un type homaloïdal est propre ou impropre. Le test de Hudson permet de simplifier les types homaloïdaux jusqu'à obtenir un type connu.

L'intuition de ce test est justement celle rencontrée lors que la preuve de l'algorithme de J.W. Alexander : si on a une transformation birationnelle (ayant tous ses points-base dans \mathbb{P}^2), on peut la composer avec des involutions quadratiques standards jusqu'à diminuer la complexité suffisamment pour reconnaître une transformation connue, et en vertu du théorème 0.1, si cette méthode échoue c'est que la transformation birationnelle de départ n'existait pas.

Le test de Hudson consiste justement à faire subir au type homaloïdal une "involutions quadratique standard" purement numérique, en modifiant le degré et les multiplicités du type homaloïdal, jusqu'à obtenir un automorphisme :

1. Étant donné un type homaloïdal $(d, (m_0, \dots, m_k))$, on vérifie qu'il ne contient pas de multiplicité négative. Si c'est le cas, le test échoue. Si $d = 1$ le test est vérifié.
2. On pose $\epsilon = m_0 + m_1 + m_2 - d$. Et on trie les valeurs de $(d - \epsilon, (m_0 - \epsilon, m_1 - \epsilon, m_2 - \epsilon, \dots, m_k))$. On applique encore l'étape 1 à ce dernier type homaloïdal.

On a de plus la caractérisation suivante (démontrée dans l'article [2]), qui sera utile pour l'implémentation informatique de l'algorithme de J.W. Alexander :

Proposition 4.2

Un type homaloïdal est propre si, et seulement si il vérifie le test de Hudson.

Chapitre 5

Annexe : Implémentation de l'algorithme

Le but de ce chapitre est de fournir une explication de l'implémentation en Python de l'algorithme de J.W. Alexander.

On commence par introduire les convention utilisées pour les objets à manipuler (points-base et transformatione birationnelles), puis on donnera une description des fonctions utilisées, sans entrer dans le détail du code. Le but est de fournir une documentation pour expliquer comment manipuler l'algorithme.

En premier lieu, on introduit les classes d'objets utilisés par l'algorithme pour coder les informations abstraites que sont les points-base et les transformations birationnelles (vues ici comme une collection de points-base et d'une multiplicité pour chaque point-base).

Ensuite, on donne le fonctionnement de quelques fonctions auxiliaires dont l'utilité consiste à aider à la création de transformation birationnelles respectant les contraintes des équations de condition (théorème 2.2) et le test de Hudson (section 4.4), et à permettre la composition d'une transformation birationnelle avec une involution quadratique.

Enfin, on précise le fonctionnement de l'implémentation de l'algorithme de J.W. Alexander.

Il faut noter deux choses importantes quant à l'implémentation de la méthode :

- Après avoir composé une transformation birationnelle f avec une involution quadratique g , on devrait avoir comme point-base pour la composée les points-base de g et les images par g des points-base P de f : $g(P)$, en vertu du corollaire 2.6. Pour des raison de lisibilité dans l'algorithme, qui peut vite devenir compliqué, on notera ces derniers comme pour f : $g(P)$ sera noté simplement P . Il faut garder cela en tête lorsqu'on lit la décomposition de chaque fonction.
- Les points sont toujours supposés le plus général possible, même lorsque ce n'est pas censé être le cas : par exemple lorsque l'on a un point O_1 sur le diviseur associé à O_0 comme dans la preuve de la proposition 3.3, on compose f avec l'involution quadratique de points-base O_0, A, B , avec A, B en position générale (toutes ces opérations sont effectuées dans la fonction `make_transfo_disjoint`). Le point-base O_1 est alors transporté sur un point de la droite (AB) et les points-base O_1, A, B ne sont alors plus en position générale dans ce cas précis, précisément parce qu'il est sur un diviseur exceptionnel au départ.

Cependant comme tous ces points-base sont dans \mathbb{P}^2 , l'algorithme considérera qu'ils sont en position générale, et on ne pourra donc pas recomposer par la transformation quadratique pour retrouver des points-base infiniment proches.

La décomposition fonctionne donc bien dans ce cas précis, mais on ne peut pas recomposer en arrière.¹

5.1 Classes d'objets : points-base et transformation birationnelle

On commence par définir deux nouvelles classes d'objets : la classe `point` et la classe `transfo_plane`

1. Je prévois de corriger ce phénomène en essayant d'introduire une notion d'alignement dans la classe `point`.

Classe `point`

Cette classe possède 3 attributs et 2 méthodes. Elle a pour but de représenter les points abstraits que l'on trouve dans l'algorithme de J.W. Alexander : en effet, comme il est possible qu'une transformation birationnelle aie des points-base qui ne soit pas dans \mathbb{P}^2 directement, il faut définir une classe qui contienne à la fois les points dans \mathbb{P}^2 et dans l'éclatement sans différencier leur nature.

Pour cela on ajoute aux caractéristiques d'un point B la possibilité d'avoir un autre point A comme "parent", dans le cas où B est sur le diviseur exceptionnel associé à A .

Liste des attributs de la classe `point` :

- **name** (string) : le nom qui sera utilisé pour le point.
- **parent** (point ou None) : dans le cas où le point est situé sur le diviseur exceptionnel d'un autre point, on le stocke ici.
- **descr** (string) : Une brève description du point (son nom et où il se trouve).

Liste des méthodes de la classe `point` :

- **change_name** ayant pour argument :
 - name (string)
Remplace le nom du point par name.
- **parentize** ayant pour argument :
 - P (point/None)
Remplace le parent du point par le point P.

Pour déclarer un nouveau point, on écrit `point(name, parent)` (ou éventuellement `point(name)` s'il est dans \mathbb{P}^2) avec :

- name (string) : le nom du nouveau point.
- parent (point/None) : le parent du nouveau point.

Remarque

Lorsque l'on affiche un point avec `print`, la console affichera la description `descr` du point. En revanche, si l'on affiche un point indirectement, par exemple dans une liste ou dans un tuple, la console affichera seulement le nom du point. Par exemple :

```
>>> A = point("A")
>>> B = point("B", A)
>>> print(A)
A in P^2
>>> print(B)
B in E_{A}
>>> print((A, B))
(A, B)
```

Le diviseur exceptionnel associé à un point A sera toujours noté E_A dans cet algorithme.

Classe `transfo_plane`

Cette classe possède 6 attributs et 2 méthodes. Le but de cette classe est de coder les propriétés d'une transformation birationnelle. Pour cela, on utilise le fait que les points-base et les multiplicités d'une transformation birationnelle identifient cette dernière à automorphisme près. On va ainsi coder une transformation birationnelle comme la donnée de points et de multiplicités associés à ces points, à travers la classe `point` définie précédemment.

Liste des attributs de la classe `transfo_plane` :

- **name** (string) : le nom qui sera utilisé pour la transformation birationnelle.
- **degree** (int) : le degré de la transformation.
- **base_points** (list de point) : la liste des points-base de la transformation, classée par multiplicité

- décroissante.
- **multiplicities** (list de int) : la liste des multiplicités des points-base de la transformation, classée par ordre décroissant.
- **complexity** (int) : la complexité $2j$ de la transformation.
- **h** (int) : l'indice du dernier point dans la liste `base_points` qui soit de multiplicité strictement supérieure à j .
- **descr** (string) : une brève description de la transformation, qui fournit son degré et ses points-base classés par ordre de multiplicité.

Liste des méthodes de la classe `transfo_plane` :

- **change_name** ayant pour argument :
 - `name` (string)
 Remplace le nom de la transformation par `name`.
- **parentize** ayant pour argument :
 - `i` (unsigned int)
 - `j` (int)
 Met le i -ème point-base de `base_points` sur le diviseur exceptionnel du j -ème point-base de `base_points`. En particulier, on requiert que $i > j$ puisque `base_points` est classée par multiplicité croissante.
 On autorise le cas particulier $j < 0$, et la fonction replace alors le i -ème point de `base_points` dans \mathbb{P}^2 (i.e. donne la valeur `None` à son parent).

Pour déclarer une nouvelle transformation birationnelle, on écrit `transfo_plane(name, degree, l_f, mult)` avec :

- `name` (string) : le nom de la nouvelle transformation.
- `degree` (int) : le degré de la nouvelle transformation.
- `l_f` (list de point) : la liste des points-base de la transformation.
- `mult` (list de int) : la liste des multiplicité correspondant aux points-base de la transformation.

En particulier, on demande que `l_f` et `mult` soient de la même taille, et que `mult` soit classée par ordre décroissant. On supposera alors que la i -ème case de `mult` correspondra à la multiplicité du i -ème élément de `l_f`.

Remarque

Comme précédemment, lorsque l'on affiche une transformation avec `print`, la console affichera la description `descr` de la transformation. En revanche, si l'on affiche cette transformation indirectement, par exemple dans une liste ou dans un tuple, la console affichera seulement le nom `name` de la transformation. Par exemple :

```
>>> f = transfo_plane("f", 2, [point("A"), point("B"), point("C")], [1, 1, 1])
>>> print(f)
f birationnal transformation of P^2 of degree 2 with basepoints:
A in P^2 with multiplicity 1
B in P^2 with multiplicity 1
C in P^2 with multiplicity 1
>>> print([f, f.base_points[0]])
[f, A]
```

Remarque

Avec ce qui a été vu précédemment dans le mémoire, on sait qu'une transformation birationnelle d'un certain degré ne peut pas avoir n'importe quel type homaloïdal associé. Afin de résoudre ce problème, on introduira d'ici deux section une fonction permettant de calculer les types homaloïdaux propres associés à un certain degré et qui crée la transformation birationnelle pour le type choisi.

Il n'est cependant pas interdit de faire des "parentages" absurdes (par exemple de mettre les points-base d'indice 1 et 2 sur le diviseur exceptionnel de celui d'indice 0 : on a vu qu'aucune transformation birationnelle ne pouvait vérifier cela), mais nous laissons la cohérence du parentage à la discrétion de l'utilisateur.

La fonction `mult`

En plus de ces deux classes, on introduit une fonction `mult` servant à calculer la multiplicité d’une transformation birationnelle en un point.

Liste des entrées de la fonction `mult` :

- `transfo (transfo_plane)` : la transformation birationnelle considérée.
- `p (point)` : le point en lequel on veut calculer la multiplicité de `transfo`.

La fonction renvoie :

- `transfo.multiplicities[i]` si `p` et `transfo.base_points[i]` coïncident.
- 0 si `p` ne correspond à aucun des éléments de `transfo.multiplicities`.

5.2 La fonction `transfo_deg`

L’objectif de la fonction `transfo_deg` est de calculer, pour un degré donné, tous les types homaloïdaux propres pour ce degré là, de les présenter à l’utilisateur pour qu’il en choisisse un, et de renvoyer l’objet `transfo_plane` correspondant.

Elle permet en particulier d’être certain de la validité de la transformation obtenue, et évite de devoir créer les fonctions à la main.

Néanmoins, la complexité de cette fonction est élevée, et pour un degré trop grand, elle mettra beaucoup de temps à calculer les possibilités. La seule solution est alors de créer un objet `transfo_plane`, correspondant au type homaloïdal voulu, à la main.

La fonction `partition_sq`

Cette fonction permet de réduire les possibilités de types homaloïdaux à ceux vérifiant les équations de conditions démontrées dans le théorème 2.2, qui fournissent une condition nécessaires pour être un type homaloïdal propre. Il s’agit d’une fonction récursive.

Étant donnés deux nombres s_1, s_2 , le premier correspondant dans le théorème 2.2 à $3(\deg(f) - 1)$ et le second à $\deg(f)^2 - 1$, on veut trouver récursivement les solutions $(\alpha_1, \dots, \alpha_n)$ au problème :

$$\begin{cases} s_1 &= \sum_{i=1}^n \alpha_i \\ s_2 &= \sum_{i=1}^n \alpha_i^2 \end{cases}$$

Pour cela, on veut retirer peu à peu des α_i potentiels à s_1 et α_i^2 à s_2 , et refaire le problème avec les nombres obtenus jusqu’à obtenir $(s_1, s_2) = (0, 0)$ auquel cas on retient la suite de nombres soustraits. Si on obtient $s_1 < 0$ ou $s_2 < 0$, la suite de nombres soustraits n’est pas valide, et on ne la retient pas.

Le fonctionnement de `partition_sq` est décrit par la figure 5.1.

Liste des entrées de la fonction `partition_sq` :

- `s_1 (int)` : le nombre dont on veut la partition en nombres entiers.
- `s_2 (int)` : le nombre dont on veut la partition en carrés d’entiers.
- `m (int)` : une borne supérieure pour les nombres à soustraire. Par exemple $\sqrt{s_2}$.
- `l_tmp (list de int)` : une liste contenant les éléments de la partition qui est en train d’être testée.
- `l (list de list de int)` : une liste contenant les partitions qui vérifient le problème.

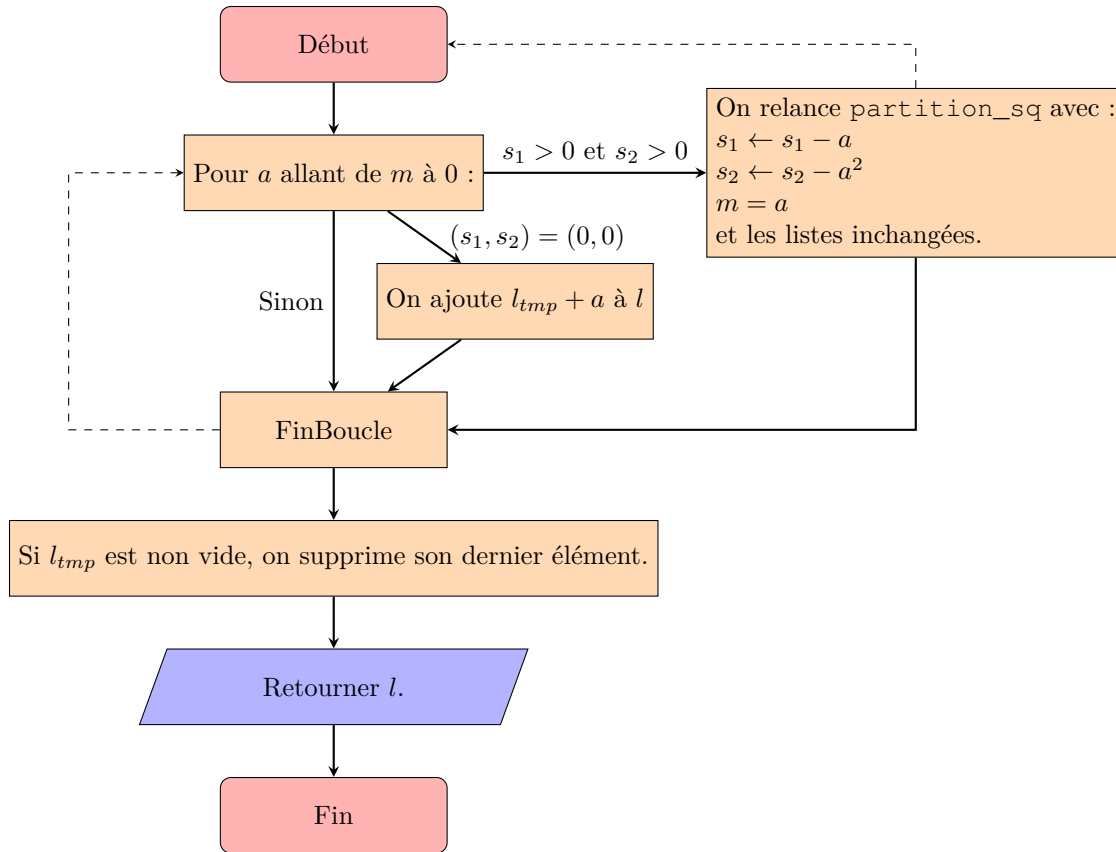


FIGURE 5.1 – Fonctionnement de `partition_sq`.

La fonction `transfo_deg`

Pour le fonctionnement de `transfo_deg`, voir figure 5.2.

Liste des entrées de la fonction `transfo_deg` :

- `n (int)` : le degré de la transformation voulue.
- `choix_input (int)` : le choix de l'utilisateur dans la liste des types homaloïdaux calculés. Cette variable est facultative et évite simplement à l'utilisateur de choisir (elle vaut -1 par défaut, et si elle a cette valeur, le choix se déclenchera).

On introduit également en figure 5.3 l'évolution du temps de calcul de `transfo_deg` afin de donner à l'utilisateur une vision de son utilisation. En effet, il devient supérieur à 1 min à partir du degré 26, et la complexité augmente exponentiellement.

À partir d'un certain degré, il convient donc de créer les transformations à la main, en connaissant déjà le type homaloïdal voulu, ou même en le vérifiant par l'algorithme de J.W. Alexander : si l'algorithme termine pour une transformation avec tous ses points-base dans \mathbb{P}^2 , c'est que le type homaloïdal est valable.

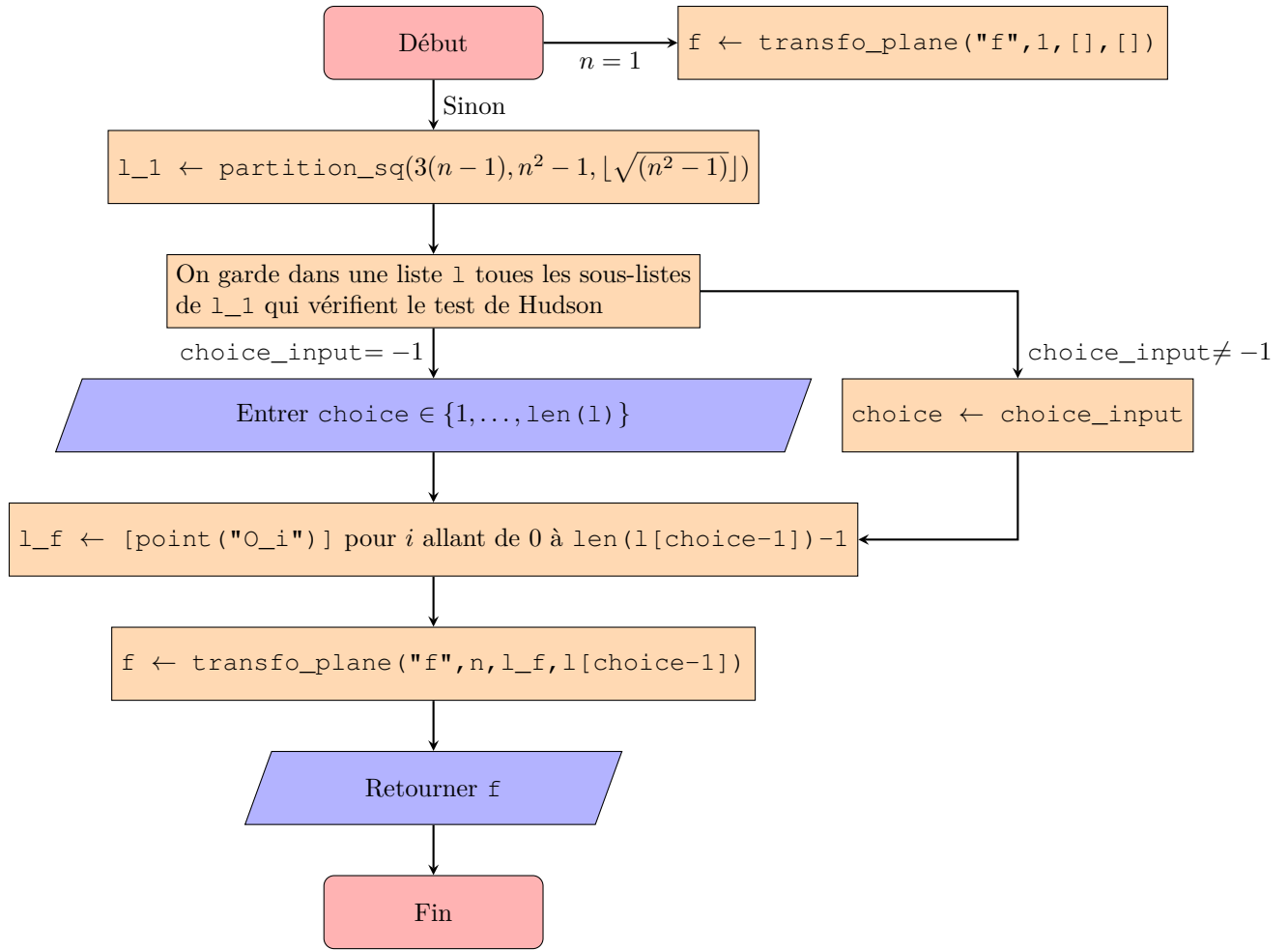


FIGURE 5.2 – Fonctionnement de transfo_deg.

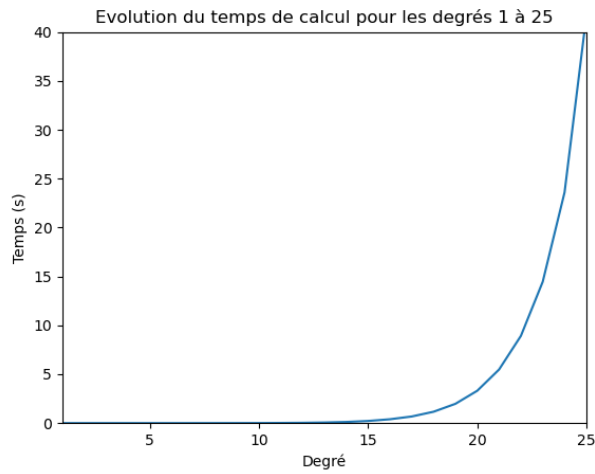


FIGURE 5.3 – Complexité de la fonction transfo_deg.

5.3 La fonction `compose_quad`

On introduit une fonction permettant de composer une transformation birationnelle de degré quelconque avec une transformation quadratique, en utilisant les résultats du chapitre 2, section 2.

Par abus de notation, et comme précisé dans l'introduction de cet annexe, afin d'éviter de complexifier les notations, on redonnera encore le même nom S_j aux points-base de la composée $f \circ h^{-1}$ qui sont de la forme $h(S_j)$ (cf. corollaire 2.6).

Cette fonction crée une nouvelle transformation à partir des points-base des deux précédentes, en respectant le résultat du corollaire 2.6.

Son fonctionnement est donné à la figure 5.4.

Liste des entrées de la fonction `compose_quad` :

- `f (transfo_plane)` : la transformation birationnelle de degré quelconque.
- `g (transfo_deg)` : la transformation quadratique avec laquelle on composera `f`. Elle est supposée de degré 2 et avec tous ses points-base dans \mathbb{P}^2 .
- `name (string)` : le nom à donner à la composée des deux transformations.

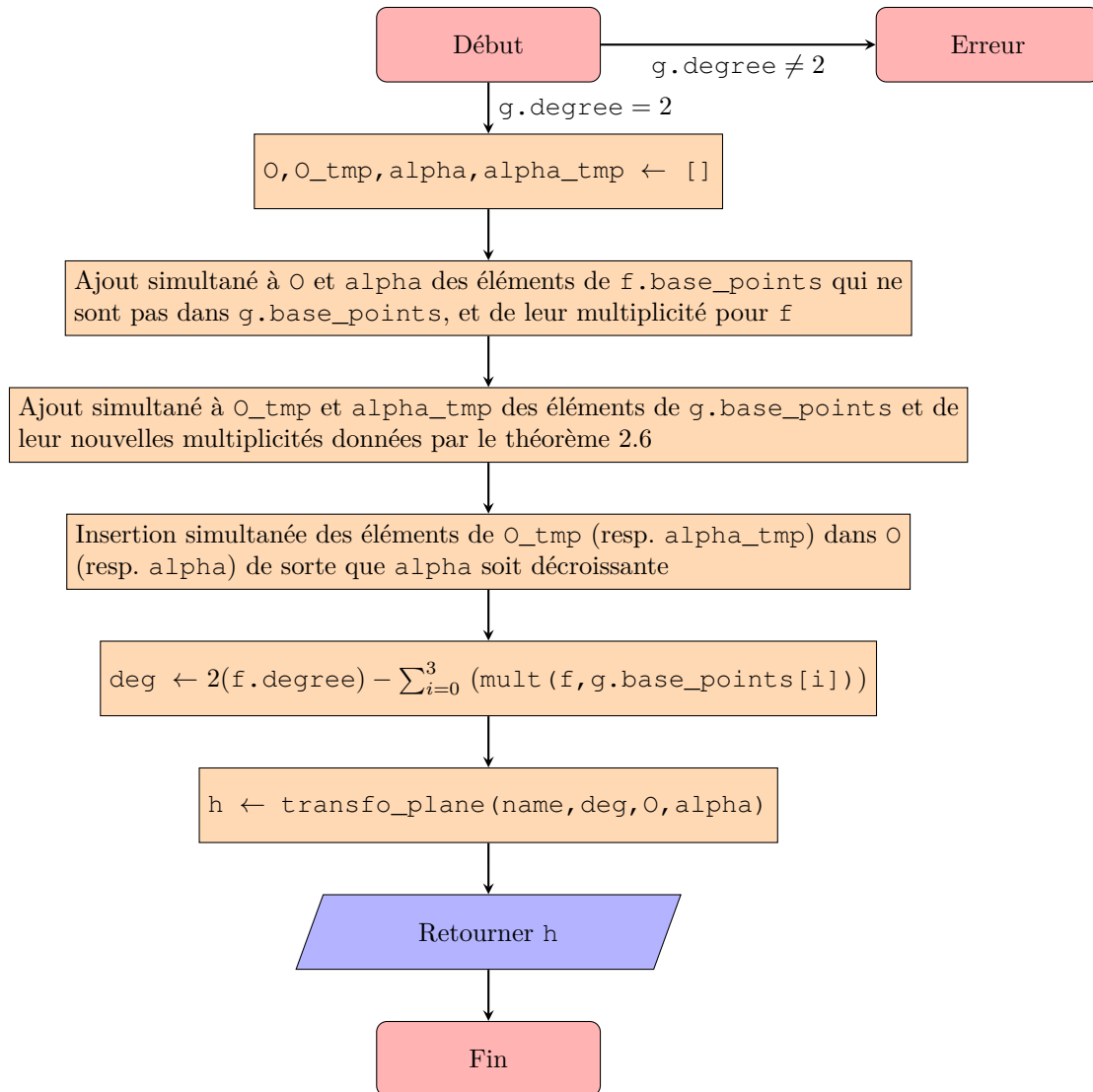


FIGURE 5.4 – Fonctionnement de `compose_quad`.

5.4 La fonction `make_transfo_disjoint`

La fonction `make_transfo_disjoint` prend en entrée une transformation birationnelle f et sépare les points-base infiniment proches de f qui sont de multiplicité supérieure à j_f , en suivant la preuve de la proposition 3.3. Pour cela, l'algorithme regarde parmi les points O_1, \dots, O_h lesquels sont infiniment proches d'un autre, ou de O_0 . S'il en existe, on les sépare comme dans la preuve, en composant avec :

- Une transformation de points-base O_0, A, B s'il y a des points-base infiniment proches de O_0 (avec A, B en position générale par rapport à tous les points-base).
- Des transformations de points-base O_0, O_i, C si le point-base O_i est infiniment proche d'un autre O_j (avec C en position générale par rapport à tous les autres points-base).

Au bout d'un certain nombre de compositions, la transformation f a la propriété voulue, et on la renvoie ainsi qu'une liste `list_comp` des transformations quadratiques appliquées (celles mentionnées ci-dessus), et un compteur `count` qui indique le nombre de fois où la fonction a été lancée (et qui sert uniquement pour numéroter les points généraux introduits, afin de ne pas les confondre entre eux d'une itération sur l'autre). Pour le fonctionnement de `make_transfo_disjoint`, voir la figure 5.6.

Liste des entrées de la fonction `make_transfo_disjoint` :

- `f_input (transfo_plane)` : la transformation birationnelle de degré quelconque à traiter.
- `count (int)` : un compteur du nombre de fois où on a lancé la fonction en séparant effectivement des points-base (reste constant s'ils sont déjà séparés).
- `l (list de transfo_plane)` : liste contenant les transformations quadratiques déjà effectuées pour simplifier la transformation. Par défaut, elle est vide.

5.5 La fonction `decomp_transfo`

Cette fonction effectue l'algorithme de J.W. Alexander en combinant les fonctions vues jusqu'à maintenant. Elle se décline en deux fonctions :

- La fonction `decomp_transfo_rec`, qui est une fonction récursive et qui applique purement l'algorithme.
- La fonction `decomp_transfo`, qui applique la fonction précédente et met en forme les résultats obtenus.

La fonction `decomp_transfo_rec`

Comme mentionné ci-dessus, cette fonction applique l'algorithme à la transformation :

- On sépare d'abord les points-base de `f_input`. Et on regarde ensuite le degré de `f`. Si elle est de degré 2 ou 1, on a fini.
- Sinon, on précompose f avec l'involution quadratique de points-base O_0, O_1, O_2 et on recommence avec la composée.

On garde en mémoire dans une liste les transformations quadratiques par lesquelles on a déjà composé, de sorte qu'à la fin, on obtient : (f, count, l) où f est la dernière transformation (quadratique) avant la sortie de l'algorithme, et l est la liste des transformations quadratiques effectuées depuis le début. Pour obtenir toutes ces transformations dans le bon ordre, il faudra donc ajouter f à la liste et inverser cette dernière, ce qui sera le rôle de la fonction `decomp_transfo`.

Liste des entrées de la fonction `decomp_transfo_rec` :

- `f (transfo_plane)` : la transformation birationnelle de degré quelconque à décomposer.
- `count (int)` : un compteur pour la fonction `make_transfo_disjoint`. Par défaut, il vaut 0.
- `l (list de transfo_plane)` : liste contenant les transformations quadratiques déjà effectuées pour simplifier la transformation. Par défaut elle est vide.

Pour le fonctionnement de `decomp_transfo_rec`, voir figure 5.5.

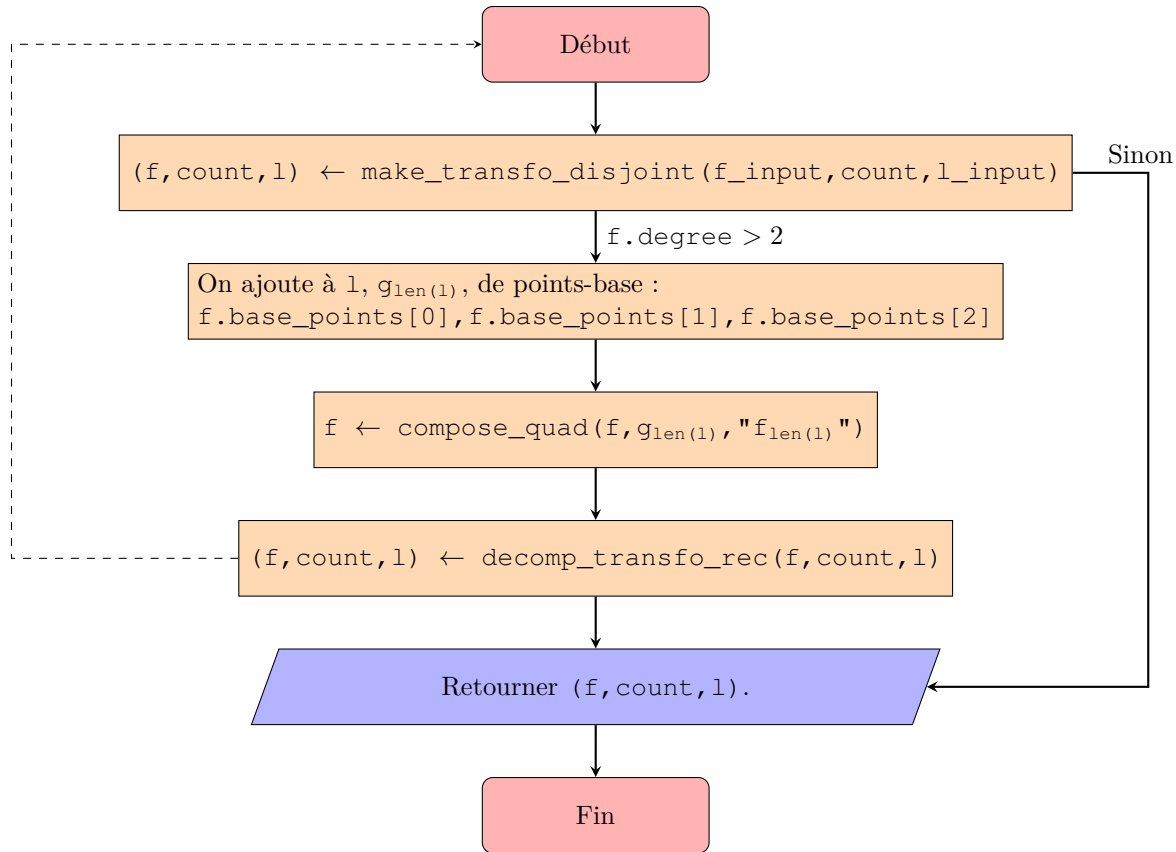


FIGURE 5.5 – Fonctionnement de `decomp_transfo_rec`.

La fonction `decomp_transfo`

La fonction `decomp_transfo` reprend la fonction récursive précédente, réorganise la liste des transformations quadratiques effectuées, et renvoie cette liste. C'est cette fonction qu'il faut utiliser pour décomposer une transformation birationnelle en transformations quadratiques.

Liste des entrées de la fonction `decomp_transfo` :

— `f_input` (`transfo_plane`) : la transformation birationnelle de degré quelconque à décomposer.

Fonctionnement de `decomp_transfo` :

La fonction effectue l'opération $(f, \text{count}, l) \leftarrow \text{decomp_transfo_rec}(f)$. Elle ajoute ensuite f à la fin de l et renvoie la liste dans l'ordre inverse. La liste correspond à la décomposition quadratique de f .

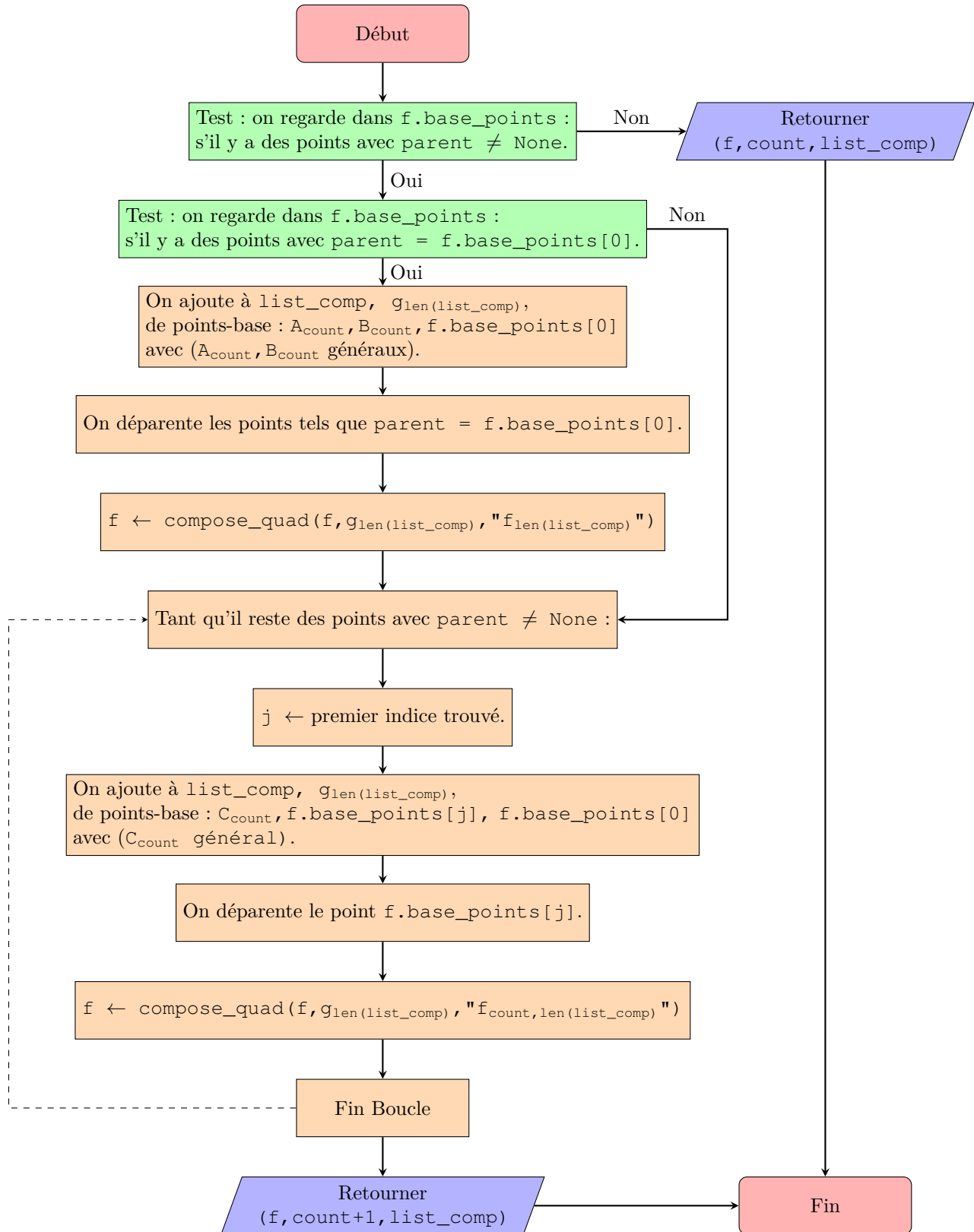


FIGURE 5.6 – Fonctionnement de make_transfo_disjoint.

5.6 Exemples et applications

Parmi les fonctions introduites plus tôt, on peut faire une distinction entre les fonctions techniques, qui ne servent que pour définir les fonctions utiles ; parmi toutes les fonctions introduites, l'utilisateur aura surtout à utiliser :

- La fonction `transfo_deg`, pour créer une nouvelle transformation de degré arbitraire sans connaître les types homaloïdaux valides pour ce degré.
- La fonction `compose_quad` pour composer une transformation avec une transformation quadratique.
- La fonction `decomp_transfo` pour avoir la décomposition en involutions quadratiques d'une transformation quadratique.

Ainsi que les initialisations des classes `point` et `transfo_plane` lorsqu'il souhaite manipuler directement ce type d'objet.

Voici un exemple d'utilisation de ces trois fonctions, qui reprend la décomposition des transformations de degré 5 que l'on a effectué à la main dans la section 4.3 :

1. On crée une transformation quadratique de degré 5 ayant 6 points-base de multiplicité 2.
2. On décompose cette transformation en involutions quadratiques.
3. On affiche un à un les éléments de la liste obtenue.
4. On recompose un à un les éléments de la liste, et on affiche la composition obtenue.

```
>>> f=transfo_deg(5)
-----
1 - [4, 1, 1, 1, 1, 1, 1, 1, 1]
2 - [3, 2, 2, 2, 1, 1, 1]
3 - [2, 2, 2, 2, 2, 2]
Choose a decomposition of multiplicities among those calculated above (type : int
between 1 and 3): 3
>>> l=decomp_transfo(f)
>>> for i in range(len(l)):
print(l[i])
print("-----")

f_2 birational transformation of P^2 of degree 2 with basepoints:
O_0 in P^2 with multiplicity 1,
O_1 in P^2 with multiplicity 1,
O_2 in P^2 with multiplicity 1.
-----
g_1 birational transformation of P^2 of degree 2 with basepoints:
O_3 in P^2 with multiplicity 1,
O_4 in P^2 with multiplicity 1,
O_5 in P^2 with multiplicity 1.
-----
g_0 birational transformation of P^2 of degree 2 with basepoints:
O_0 in P^2 with multiplicity 1,
O_1 in P^2 with multiplicity 1,
O_2 in P^2 with multiplicity 1.
-----
>>> g=l[0]
>>> for i in range(1,len(l)):
g=compose_quad(g,l[i],"g")

>>> print(g)
```

```

g birational transformation of P^2 of degree 5 with basepoints:
O_3 in P^2 with multiplicity 2,
O_4 in P^2 with multiplicity 2,
O_5 in P^2 with multiplicity 2,
O_0 in P^2 with multiplicity 2,
O_1 in P^2 with multiplicity 2,
O_2 in P^2 with multiplicity 2.

```

On effectue un autre exemple avec une transformation birationnelle qui possède des points-base infiniment proches. Comme mentionné dans l'introduction de ce chapitre, on ne peut pas recomposer les transformations une fois la décomposition effectuée.

1. On crée une transformation birationnelle f de degré 6 avec 2 points-base de multiplicité 3, 4 points-base de multiplicité 2, et un point-base de multiplicité 1.
2. On place O_1 sur le diviseur exceptionnel E_{O_0} associé à O_0 , et on place O_3 sur E_{O_1} avec la fonction `parentize`.
3. On décompose f et on affiche la décomposition.

```

>>> f=transfo_deg(6)
-----
1 - [5, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
2 - [4, 2, 2, 2, 2, 1, 1, 1]
3 - [3, 3, 3, 2, 1, 1, 1, 1]
4 - [3, 3, 2, 2, 2, 2, 1]
Choose a decomposition of multiplicities among those calculated above (type : int
between 1 and 4): 4
>>> f.parentize(1,0)
>>> f.parentize(3,1)
>>> print(f)
f birational transformation of P^2 of degree 6 with basepoints:
O_0 in P^2 with multiplicity 3,
O_1 in E_O_0 with multiplicity 3,
O_2 in P^2 with multiplicity 2,
O_3 in E_O_1 with multiplicity 2,
O_4 in P^2 with multiplicity 2,
O_5 in P^2 with multiplicity 2,
O_6 in P^2 with multiplicity 2.
>>> l=decomp_transfo(f)
>>> for i in range(len(l)):
print(l[i])
print("-----")

```

```

f_5 birational transformation of P^2 of degree 2 with basepoints:
O_6 in P^2 with multiplicity 1,
O_1 in P^2 with multiplicity 1,
O_0 in P^2 with multiplicity 1.
-----

```

```

g_4 birational transformation of P^2 of degree 2 with basepoints:
O_3 in P^2 with multiplicity 1,
O_4 in P^2 with multiplicity 1,
O_5 in P^2 with multiplicity 1.
-----

```

```

g_3 birational transformation of P^2 of degree 2 with basepoints:

```

```

O_1 in P^2 with multiplicity 1,
O_0 in P^2 with multiplicity 1,
O_2 in P^2 with multiplicity 1.
-----
g_2 birational transformation of P^2 of degree 2 with basepoints:
O_0 in P^2 with multiplicity 1,
A_0 in P^2 with multiplicity 1,
B_0 in P^2 with multiplicity 1.
-----
g_1 birational transformation of P^2 of degree 2 with basepoints:
O_0 in P^2 with multiplicity 1,
O_1 in P^2 with multiplicity 1,
C_0,1 in P^2 with multiplicity 1.
-----
g_0 birational transformation of P^2 of degree 2 with basepoints:
O_0 in P^2 with multiplicity 1,
A_0 in P^2 with multiplicity 1,
B_0 in P^2 with multiplicity 1.
-----

```

Bibliographie

- [1] James W. ALEXANDER, *On the factorization of the Cremona plane transformations*. 1915, Transaction of American Mathematical Society 17 (1916), 295-300.
- [2] Jérémy BLANC, Alberto CALABRI, *On degenerations of plane Cremona transformations*. 2015, Mathematische Zeitschrift 282 (2016), 223-245.
- [3] Guido CASTELNUOVO, *Le trasformazioni generatrici del gruppo cremoniano nel piano*. 1901, Atti della R. Accad. delle Scienze di Torino 36 (1901), 861-874.
- [4] Julie DESERTI, *Some properties of the Cremona Group*. 2012, Sociedade Brasileira de Matematica, vol. 21.
- [5] Daniel PERRIN, *Géométrie algébrique, Une introduction*. 1995, InterÉditions / CNRS Éditions.
- [6] Igor SHAFAREVICH, *Basic Algebraic Geometry 1 : Varieties in Projective Space*. 1988, Springer-Verlag, 2e édition.

Mahmoud Elimam

Lagrangian Cobordisms And Surgery

Master Thesis

Le Laboratoire de Mathématiques Jean Leray
Université de Nantes

Supervision

Prof. Baptiste Chantraine

Abstract

This document consists of three chapters. The first chapter introduce basic concepts and definitions in symplectic geometry and contact geometry. In the second chapter we define Lagrangian cobordism between two legendrian submanifolds of a contact manifold, and prove that legendrian isotopy is realized by such cobordism. We also study gf-compatible Lagrangian cobordisms and prove a gf version of such realization. In the third chapter we give two descriptions of Lagrangian surgery, the second of which applies to exact symplectic manifolds and exact Lagrangians and is realized by Lagrangian cobordism from the lift of the Lagrangian submanifold before surgery to the lifting of Lagrangian submanifold after surgery.

Contents

Abstract	1
Dedication	1
1 Symplectic and Contact Geometry	4
1.1 Symplectic Manifolds	4
1.1.1 Skew Symmetric Bilinear forms	4
1.1.2 Basic concepts for symplectic manifolds	6
1.1.3 Moser-type theorems	9
1.2 Contact Manifolds	16
1.2.1 Basic concepts for contact manifolds	17
1.2.2 Symplectization and Contactisation	25
1.2.3 Moser type theorems	28
1.2.4 Generating families	29

2	Lagrangian cobordism	31
2.0.1	Definition and compatible generating functions	31
2.0.2	Isotopy is realised by lagrangian cobordism	33
2.0.3	Some results on gf-compatible lagrangian cobordisms	36
3	Lagrangian Surgery	40
3.1	First Description	41
3.2	Second Discription via cobordism	47
	Bibliography	51

Chapter 1

Symplectic and Contact Geometry

In this chapter we introduce basic concepts of symplectic and contact geometries.

1.1 Symplectic Manifolds

1.1.1 Skew Symmetric Bilinear forms

In this section we will review symplectic linear algebra

Definition 1.1.1. Let V be an m -dimensional vector space over \mathbb{R} . A Bilinear form $\Omega : V \times V \rightarrow \mathbb{R}$ is skew symmetric if $\Omega(x, y) = -\Omega(y, x)$ for all $x, y \in V$.

We have standard form of such forms.

Proposition 1.1.1. *Let Ω be a skew-symmetric form on V . Then there is a basis $u_1, \dots, u_k, e_1, \dots, e_n, f_1, \dots, f_n$ such that*

$$\Omega(u_i, \cdot) = 0, \quad \Omega(e_i, e_j) = \Omega(f_i, f_j) = 0, \quad \Omega(e_i, f_j) = \delta_{ij}$$

Proof. Let $U = \ker \Omega = \{u \in V \mid \Omega(u, \cdot) = 0\}$. Choose a basis u_1, \dots, u_k of U , and let W be complementary space of U , that's $V = U \oplus W$. Let $e_1 \in W$, then there is f_1 such that $\Omega(e_1, f_1) = 1$. We let

$$W_1 = \text{span}\{e_1, f_1\}, \quad W_1^\Omega = \{w \in W \mid \Omega(w, v) = 0 \text{ for all } v \in W_1\}$$

We have

- $W_1 \cap W_1^\Omega = \{0\}$: For let $v = ae_1 + bf_1 \in W_1 \cap W_1^\Omega$, then $0 = \Omega(v, e_1) = -b$ and $0 = \Omega(v, f_1) = a$. Hence $v = 0$.
- $W = W_1 \oplus W_1^\Omega$: For let $v \in W$, and suppose that $\Omega(v, e_1) = c$ and $\Omega(v, f_1) = d$. Then

$$v = (-cf_1 + de_1) + (v + cf_1 - de_1),$$

where we note that $-cf_1 + de_1 \in W_1$ and $v + cf_1 - de_1 \in W_1^\Omega$

We continue the process by picking up $0 \neq e_2 \in W_1^\Omega$, then there is $f_2 \in W_1^\Omega$ such that $\Omega(e_2, f_2) = 1$. We let

$$W_2 = \text{span}\{e_2, f_2\}, \quad W_2^{\Omega|_{W_1^\Omega}} = \{w \in W_1^\Omega \mid \Omega(w, v) = 0 \text{ for all } v \in W_2\}$$

We get again $W_1^\Omega = W_2 \oplus W_2^{\Omega|_{W_1^\Omega}}$, continuing in this way we get $W_n^{\Omega|_{W_{n-1}^\Omega}} = 0$ for some n (as $\dim V < \infty$, and hence

$$V = U \oplus W_1 \oplus \dots \oplus W_n,$$

where $W_i = \text{span}\{e_i, f_i\}$ and all summands are orthogonal with respect to Ω . \square

We say Ω is non-degenerate(or symplectic) if the map $\tilde{\Omega} : V \rightarrow V^*$ defined by $\tilde{\Omega}(u) = \Omega(u, \cdot)$ is isomorphism. The kernel of this map is U constructed in the proof above, so Ω is non-degenerate if and only if $U = \{0\}$. In this case we say (V, Ω) a *symplectic vector space*. Note that by the above proposition V is

even dimensional. A basis $e_1, \dots, e_n, f_1, \dots, f_n$ as in the proposition is called a symplectic basis.

Corollary 1.1.2. *Let V be a vector space of dimension n , Ω is non-degenerate if and only if $\Omega^n \neq 0$. (Here $\Omega^n = \Omega \wedge \dots \wedge \Omega$)*

The following proposition follows easily from definitions

Proposition 1.1.3. *Let (V, Ω) be a symplectic vector space. Let $W \subset V$ be a subspace, then $\dim W + \dim W^\Omega = \dim V$, where $W^\Omega = \{v \in V \mid \Omega(v, u) = 0 \text{ for all } u \in W\}$*

We say $W \subset (V, \Omega)$ an *isotropic* subspace if $W \subset W^\Omega$. It follows from above proposition that $\dim W \leq \frac{1}{2} \dim V$. If $W = W^\Omega$ we say W is *Lagrangian*, it follows in this case that $\dim W = \dim W^\Omega = \frac{1}{2} \dim V$.

1.1.2 Basic concepts for symplectic manifolds

We begin with the definition of symplectic manifolds.

Definition 1.1.2. Let M be a manifold. A 2-form ω which is closed and $\omega_p : T_p \times T_p M \rightarrow \mathbb{R}$ is symplectic for all p is called *symplectic form*. The pair (M, ω) is then called *symplectic manifold*. If $\omega = d\theta$, then M is called exact symplectic manifold.

A submanifold $L \subset M$ is called lagrangian if $T_x L$ is lagrangian subspace of $T_x M$ for each $x \in L$, so $\dim L = \frac{1}{2} \dim M$, and immersion $i : L \rightarrow M$ is lagrangian if $di(x)(T_x L)$ is lagrangian subspace of $T_x M$ for each $x \in L$ or equivalently $i^* \omega = 0$. Now we give the definition of symplectomorphisms

Definition 1.1.3. Let (M_1, ω_1) and (M_2, ω_2) be $2n$ - dimensional symplectic manifolds and let $\varphi : M_1 \rightarrow M_2$ be a diffeomorphism. Then φ is *symplectomorphism* if $\varphi^* \omega_2 = \omega_1$.

We give the important examples

Example 1.1.1. 1. Let $M = \mathbb{R}^{2n}$ with coordinates $q_1, \dots, q_n, p_1, \dots, p_n$. Then the form

$$\omega_{std} = \sum_{i=1}^n dq_i \wedge dp_i$$

is symplectic. So $(\mathbb{R}^{2n}, \omega_{std})$ is symplectic manifold.

2. (Important example) Let X be a manifold, and $M = T^*X$ be its cotangent bundle, we have a canonical symplectic form ω_{can} defined on M defined as follows: We first define canonical 1-form λ_{can} . Denote by $\pi : M \rightarrow X$ the projection and let $p = (x, \alpha) \in T_x X$, define $\lambda_{can}|_{(x, \alpha)}(v) = \alpha(d\pi(x, \alpha)(v))$. To check smoothness, we compute λ_{can} in coordinates. Let (U, q_1, \dots, q_n) be coordinate chart of M , and $(T^*U, q_1, \dots, q_n, p_1, \dots, p_n)$ be the associated coordinate chart of T^*M (which means that for $p = (x, \alpha) \in T^*U$, we have $\alpha = \sum_i p_i(p)(dq_i)_x$). It is easily seen that

$$\lambda_{can} = \sum_{i=1}^n p_i dq_i$$

which is smooth. Define

$$\omega_{can} = -d\lambda_{can},$$

this is trivially closed, and in local coordinates

$$\omega_{can} = \sum_i dq_i \wedge dp_i.$$

So ω_{can} is symplectic, we call it the canonical symplectic form. For future reference, it can be checked that considering the natural identification $T_{(x,0)} T^*X \equiv T_x X \oplus T_x^* X$, we have

$$\omega_{can}|_{(x,0)}(v, w) = w_1^*(v_0) - v_1^*(w_0)$$

for $v = (v_0, v_1^*), w = (w_0, w_1^*) \in T_{(x,0)} T^*X$

Given a diffeomorphism $f : M_1 \rightarrow M_2$, we can lift it to a symplectomorphism $f_\# : (T^*M_1, \omega_1) \rightarrow (T^*M_2, \omega_2)$, where ω_1 and ω_2 are the corresponding canonical symplectic forms. Define

$$f_\#(x_1, \alpha_1) = (f(x), ((df_x)^*)^{-1}\alpha_1),$$

In fact $f_\#$ turns out to be exact symplectomorphism in the sense of the following claim

Claim. $f_\#^*\lambda_2 = \lambda_1$, where λ_i is the canonical 1-form of T^*M_i so that $\omega_i = -d\lambda_i$.

Proof. let $p_1 = (x_1, \alpha_1) \in T^*M_1$ and $p_2 = (x_2, \alpha_2) = f_\#(p_1)$, then we have to show that

$$(df_\#)^*_{p_1}(\lambda_2)_{p_2} = (\lambda_1)_{p_1} \quad (1.1)$$

We have the following facts

- $(df_{x_1})^*\alpha_2 = \alpha_1$ and $x_2 = f(x_1)$ (by definition of $f_\#$)
- $(\lambda_1)_{p_1} = (d\pi_1)^*\alpha_1$ and $(\lambda_2)_{p_2} = (d\pi_2)^*\alpha_2$ where π_j is the projection $T^*M_j \rightarrow M_j$.
- $\pi_2 \circ f_\# = f \circ \pi_1$

The proof of 1.1 is

$$\begin{aligned} (df_\#)^*_{p_1}(\lambda_2)_{p_2} &= (df_\#)^*_{p_1}(d\pi_2)^*\alpha_2 = (d(f_\# \circ \pi_2))^*_{p_1}\alpha_2 \\ &= (d(f \circ \pi_1))^*_{p_1}\alpha_2 = (d\pi_1)^*_{p_1}(df)_{x_1}^*\alpha_2 \\ &= (d\pi_1)^*_{p_1}\alpha_1 = (\lambda_1)_{p_1} \end{aligned}$$

□

An important class of symplectomorphisms is the class of hamiltonian differemorphisms. To define it we have to define the Hamiltonian vector field. Fix a symplectic

manifold (M, ω) . We say that a vector field $X : M \rightarrow TM$ is *Hamiltonian vector field* if there is a function (called *Hamiltonian function*) such that

$$\iota_X \omega = dH$$

We denote X by X_H . A diffeomorphism $\phi : M \rightarrow M$ is hamiltonian if there is isotopy

$$[0, 1] \times M \rightarrow M, (t, x) \mapsto \psi_t(x)$$

such that ψ_t is generated by time dependent Hamiltonian vector field X_{H_t} where H_t is smooth family of Hamiltonians and $\phi = \psi_1$. We can see that ϕ is symplectomorphism, for

$$\frac{d}{dt} \psi_t^* \omega = \psi_t^* (\mathcal{L}_{X_{H_t}} \omega) = \psi_t^* (d\iota_{X_{H_t}} \omega) + \iota_{X_{H_t}} d\omega = \psi_t^* (ddH_t) = 0,$$

so $\psi_t^* \omega = \psi_0^* \omega = \omega$.

1.1.3 Moser-type theorems

One of the fundamental techniques in symplectic geometry is Moser's argument. Given smooth family of symplectic forms ω_t on M , with the property

$$\frac{d}{dt} \omega_t = d\sigma_t \tag{1.2}$$

The goal of Moser argument is to construct smooth family of diffeomorphisms ψ_t such that

$$\psi_t^* \omega_t = \omega \tag{1.3}$$

The idea is to construct ψ_t as flows of (to be determined) time dependent vector field X_t . That's

$$\frac{d}{dt} \psi_t = X_t \circ \psi_t \tag{1.4}$$

If this is the case then taking the derivative of 1.3 with respect to t , we get

$$\phi_t^*\left(\frac{d}{dt}\omega_t + \mathcal{L}_{X_t}\omega_t\right) = 0$$

By 1.7 and Cartan formula,

$$0 = d\sigma_t + d(\iota_{X_t}\omega_t) + \iota_{X_t}(d\omega_t) = d\sigma_t + d(\iota_{X_t}\omega_t)$$

This equation is satisfied if

$$\sigma_t = \iota_{X_t}\omega_t$$

But by non-degeneracy of ω_t , we get a unique time dependent vector field X_t . It follows that ψ_t is determined by 1.4. Using this argument, we prove the following:

Theorem 1.1.4. *[Moser isotopy theorem] Let (M^{2n}, ω) be symplectic manifold, and $S \subset M$ be a submanifold (not necessarily compact). Suppose ω_0, ω_1 be symplectic forms such that for all $x \in X$, $\omega_0|_x$ and $\omega_1|_x$ are equal. Then there exist neighborhoods $\mathcal{U}_0, \mathcal{U}_1$ of S in M and diffeomorphism $\psi : \mathcal{U}_0 \rightarrow \mathcal{U}_1$ such that*

$$\psi|_S = \text{id} \quad , \quad \psi^*\omega_1 = \omega_0$$

Proof. We use Moser argument above. We find a neighborhood \mathcal{U}_0 of S such that there is $\sigma \in \Omega^1(\mathcal{U}_0)$ satisfying

$$\sigma_x = 0 \quad \text{for all } x \in S \quad , \quad d\sigma = \omega_1 - \omega_0$$

We endow M with riemannian metric. We know by (*Tubular neighborhood theorem*) that there exists function $\epsilon : S \rightarrow \mathbb{R}_+$ such that the restriction of $\exp : TS^\perp \rightarrow M$ to

$$U_\epsilon = \{(x, v) | x \in S, v \in TS^\perp, |v| < \epsilon(x)\}$$

is embedding, we denote the image by \mathcal{U}_0 . For $0 \leq t \leq 1$, define $\phi_t : \mathcal{U}_0 \rightarrow \mathcal{U}_1$ by

$$\phi_t(\exp(p, v)) = \exp(p, tv)$$

Clearly $\phi_0(\mathcal{U}_0) \subset S$ and ϕ_t is embedding whenever $t > 0$, with $\phi_t|_S = \text{id}$. Letting $\tau = \omega_1 - \omega_0$, we get

$$\phi_0^*(\tau) = 0, \quad \phi_1^*\tau = \tau$$

Define

$$Y_t = \left(\frac{d}{dt}\phi_t\right) \circ \phi_t^{-1}$$

This vector field may be singular at $t = 0$. However, we have

$$\frac{d}{dt}\phi_t^*\tau = \phi_t^*(\mathcal{L}_{Y_t}\tau) = \phi_t^*(d(\iota_{Y_t}\tau) + \iota_{Y_t}d\tau) = d(\phi_t^*(\iota_{Y_t}\tau)),$$

where σ_t is smooth family of 1-forms $\phi_t^*(\iota_{Y_t}\tau)$. Note that for $v \in T_x\mathcal{U}_0$ we have

$$\sigma_t|_x(v) = (\phi_t^*(\iota_{Y_t}\tau)(v))_x = (\iota_{Y_t}\tau)_{\phi_t(x)}(d\phi_t(x)(v)) = \tau_{\phi_t(x)}(Y_t(\phi(x)), d\phi_t(x)(v))$$

which is smooth at $t = 0$, and it vanishes on S . Putting $\sigma = \int_0^1 \sigma_t dt$ (note that $\sigma_x = \int_0^1 \sigma_t|_x dt = 0$ for $x \in X$), we get

$$\tau = \phi_1^*\tau - \phi_0^*\tau = \int_0^1 \frac{d}{dt}(\phi_t^*\tau) dt = \int_0^1 d\sigma_t dt = d\sigma$$

Now we start Moser argument, let $\omega_t = (1-t)\omega_0 + t\omega$, since $\omega_t|_x = \omega_0|_x$ for all $x \in S$, then by a compactness argument it follows that by shrinking \mathcal{U}_0 , ω_t is nondegenerate on \mathcal{U}_0 . By nondegeneracy we find time dependent vector field X_t on \mathcal{U}_0 by

$$\iota_{X_t}\omega + \sigma = 0,$$

Because $\sigma_x = 0$ for all $x \in S$, it follows that $X_t = 0$ on S . Now we have to show that by shrinking \mathcal{U}_0 if necessary, the family of maps ψ_t defined as solution to the initial value problem

$$\frac{d}{dt}\psi_t = X_t \circ \psi_t, \quad \psi_0 = \text{id},$$

is defined on \mathcal{U}_0 for all $t \in [0, 1]$. It suffices to show that for every $x \in S$, there is a neighborhood \mathcal{V}_x of x , such that for the integral curve $\theta(t)$ starting at $y \in \mathcal{V}_x$ is defined for $t \in [0, 1]$, where then we can replace \mathcal{U}_0 by $\mathcal{U}_0 \cap \bigcup_{x \in S} \mathcal{V}_x$. Translating the claim to local neighborhood we find that we need to show

Claim. Let $F : [0, 1] \times B(0, \epsilon) \rightarrow \mathbb{R}^n$ be continuous and lipschitz in the second argument uniformly with respect to the first argument. Suppose that $F(t, 0) = 0$, then there exist $\delta < \epsilon$ such that for any $x_0 \in B(0, \delta)$, the solution θ of initial value problem

$$\theta'(t) = F(t, \theta(t)) , \quad \theta(0) = x_0,$$

is defined on $[0, 1]$

To prove this claim, we find an apriori estimate on the solution. Because of lipschitz condition in the claim, there is constant C such that

$$|F(t, x)| = |F(t, x) - F(t, 0)| \leq C|x|$$

Suppose θ is solution of the initial value problem in the claim, then

$$|\theta'(t)| = |F(t, \theta(t))| \leq C|\theta(t)|,$$

hence by Gronwall inequality, we get

$$|\theta(t)| \leq |x_0|e^{Ct} \tag{1.5}$$

We let $\delta = \frac{1}{2}e^{-C}\epsilon < \epsilon$, suppose $x_0 \in B(0, \delta)$. If θ (the solution of the initial value problem) has maximal interval $[0, \eta)$, then by ODE theory $|\theta(t)| \rightarrow \epsilon$ as $t \rightarrow \eta^-$. But from 1.5 we get

$$|\theta(t)| < \delta e^{C\eta} < \frac{1}{2}e^{C\eta-C}\epsilon < \frac{1}{2}\epsilon$$

which is a contradiction. So θ is defined on $[0, 1]$ for any $x_0 \in B(0, \delta)$. This proves the claim.

Finally let $\mathcal{U}_1 = \psi_1(\mathcal{U}_0)$, then $\psi_1 : \mathcal{U}_0 \rightarrow \mathcal{U}_1$ is the desired diffeomorphism. Note that $\psi|_S = \text{id}$ because $X_t = 0$ on S . \square

We have a corollary

Corollary 1.1.5 (Darboux theorem). *Every symplectic form is locally diffeomorphic to ω_{std} on \mathbb{R}^{2n} .*

Proof. Follows from 1.1.4 and 1.1.1 by setting $L = \{pt\}$ and \square

A coordinate chart $(\mathcal{U}, (q_1, \dots, q_n, p_1, \dots, p_n))$ of symplectic manifold (M, ω) is called Darboux coordinate chart, if

$$\omega = \sum_{i=1}^n dq_i \wedge dp_i$$

The corollary states that there is Darboux coordinates around any point of M .

Theorem 1.1.6 (Weinstein-Darboux Theorem). *Let (M^{2n}, ω) be symplectic manifold, and $i : L \hookrightarrow M$ be Lagrangian submanifold, then there exist a neighborhood \mathcal{U}_0 of the zero section L_0 in T^*L and neighborhood \mathcal{U} of L and diffeomorphism $\psi : \mathcal{U}_0 \rightarrow \mathcal{U}$ such that*

$$i = \psi \circ i_0 \quad , \quad \psi^* \omega = \omega_{can}$$

Proof. Since L is Lagrangian, then for $x \in L$ the map

$$\beta : T_x M \rightarrow T_x^* L ; \quad u \mapsto \Omega(u, \cdot)$$

descends to map

$$\tilde{\beta}_x : N_x L = T_x M / T_x L \rightarrow T_x^* L ; \quad [u] \mapsto \Omega(u, \cdot),$$

in fact $\tilde{\beta}_x$'s give isomorphism $\tilde{\beta} : NL \rightarrow T^*L$. We have the following easy to prove claim

Claim. If J is a compatible almost complex structure on (M, ω) , then JTL_x is Lagrangian subspace of $T_x M$ and in fact with respect to the riemannian metric g_J , the bundle JTL is orthogonal to TL .

Fix an almost complex structure J , by the claim we can identify JTL with NL by the isomorphism $v \mapsto [v]$, so we can consider $\tilde{\beta} : JTL \rightarrow T^*L$. Fix the riemannian metric g_J and recall that there is a neighborhood \mathcal{V}_0 of 0-section of JTL and neighborhood \mathcal{U}' of L in M such that the map $\mathcal{V}_0 \rightarrow \mathcal{U}'$ defined by $(x, v) \mapsto \exp_x(-v)$ is diffeomorphism. Composing the later map with $\tilde{\beta}^{-1}$, we get a diffeomorphism

$$\phi : \mathcal{U}'_0 \rightarrow \mathcal{U}' ; (x, \alpha) \mapsto \exp_x(-\tilde{\beta}^{-1}\alpha)$$

where \mathcal{U}'_0 is a neighborhood of the zero section. We have to check that $\phi^*\omega$ and ω_{can} agree on the zero section. Indeed, for $(v_0, v_0^*) \in T_{(x,0)}T^*L = T_x L \oplus T_x^* L$, we have

$$d\phi(v_0, v_0^*) = v_0 - \tilde{\beta}^{-1}(v_0^*)$$

hence for $v = (v_0, v_0^*)$ and $w = (w_0, w_0^*)$ lying in $T_{(x,0)}^* L$

$$\begin{aligned} \phi^*\omega_{(x,0)}(v, w) &= \omega_q(d\phi_{(x,0)}(v), d\phi_{(x,0)}(w)) \\ &= \omega_q(d\phi_{(x,0)}(v), d\phi_{(x,0)}(w)) \\ &= \omega_q(v_0 - \tilde{\beta}^{-1}(v_0^*), w_0 - \tilde{\beta}^{-1}(w_0^*)) \\ &= \omega_q(\beta^{-1}(w_0^*), v_0) - \omega_q(\beta^{-1}(v_0^*), w_0) \\ &= w_0^*(v_0) - v_0^*(w_0) \\ &= \omega_{can}|_{(q,0)}(v, w) \end{aligned}$$

Now by Moser stability theorem, we can find neighborhoods $\mathcal{U}_0, \mathcal{U}_0'' \subset \mathcal{U}'_0$ of the zero section of T^*L and diffeomorphism $\varphi : \mathcal{U}_0 \rightarrow \mathcal{U}_0''$ which is identity on the zero-section and satisfies $\varphi^*(\phi^*\omega) = \omega_{can}$. It follows that $\psi = \phi \circ \varphi : \mathcal{U}_0 \rightarrow \phi(\mathcal{U}_0) =: \mathcal{U}$

is the required map. □

We have the following easy corollary

Corollary 1.1.7. *Let (M^{2n}, ω) be a symplectic manifold and let L be a Lagrangian submanifold of M . Then around each point of L , there is Darboux coordinate chart $(\mathcal{U}, q_1, \dots, q_n, p_1, \dots, p_n)$ such that*

$$L \cap \mathcal{U} = \{x | p_1(x) = \dots = p_n(x) = 0\}$$

Proof. Pick a point $x_0 \in L$. By Weinstein-Darboux theorem, we can find neighborhoods $\mathcal{U}_1, \mathcal{U}_0$ of L and the zero section of T^*L respectively, such that there is symplectomorphism $\psi : \mathcal{U}_0 \rightarrow \mathcal{U}_1$ satisfying

$$\psi^*(\omega) = \omega_{can} , \quad \psi \circ i_0 = i$$

Let $(V, q_1, q_2, \dots, q_n)$ be coordinate chart in L centered at x_0 , let $(\pi^{-1}(V), \varphi = (q_1, \dots, q_n, p_1, \dots, p_n))$ be the associated coordinates on T^*L , then by composing, $\phi = \varphi \circ \psi^{-1} : \mathcal{U} \rightarrow \mathbb{R}^{2n}$ gives coordinate chart of M centered at x_0 , we abuse notation and denote again $\phi = (q_1, \dots, q_n, p_1, \dots, p_n)$. We get clearly,

$$\omega = \sum_i dq_i \wedge dp_i , \quad L \cap \mathcal{U} = \{p | p_1(x) = \dots = p_n(x) = 0\}$$

□

Theorem 1.1.8. *Let (M^{2n}, ω) be a symplectic manifold. Let L_1, L_2 be Lagrangian submanifolds intersecting transversely at x_0 , then there is Darboux coordinates $(\mathcal{U}, \tilde{q}_1, \dots, \tilde{q}_n, \tilde{p}_1, \dots, \tilde{p}_n)$ around x , such that*

$$L_1 \cap \mathcal{U} = \{x \in M | \tilde{p}_1(x) = \dots = \tilde{p}_n(x) = 0\} , \quad L_2 \cap \mathcal{U} = \{y \in M | \tilde{q}_1(x) = \dots = \tilde{q}_n(x) = 0\}$$

Proof. (Sketch) The problem is local, so we can assume that L_1, L_2 are Lagrangian submanifolds of $(\mathbb{R}^{2n}, \omega_{std})$ intersecting transversally at the origin. By the above corollary, we can further assume that $L_1 = \mathbb{R}^n \times \{0\}$. It follows by transversality

that near the origin, L_2 coincides with a graph of function $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, that's L_2 coincides with $\{(f(p), p) | p \in \mathbb{R}^n\}$ near the origin. In other words, L_2 is defined by equations $q_i = f_i(p_1, \dots, p_n)$ near 0, where f_i is the i -th component of f . Because L_2 is Lagrangian, we should have

$$\sum_i df_i \wedge dp_i = 0 \quad (1.6)$$

Now we set

$$\tilde{q}_i = q_i - f_i(p_1, \dots, p_n), \quad \tilde{p}_i = p_i$$

and restrict to a small neighborhood around zero. It follows that

- L_2 is the set of all points satisfying $\tilde{q}_1 = \dots = \tilde{q}_n$
- $\tilde{q}_1, \dots, \tilde{q}_n, \tilde{p}_1, \dots, \tilde{p}_n$ are Darboux coordinates:

$$\begin{aligned} \omega_{std} &= \sum_i dq_i \wedge dp_i \\ &= \sum_i d(\tilde{q}_i + f_i) \wedge dp_i \\ &= \sum_i d\tilde{q}_i \wedge d\tilde{p}_i + df_i \wedge dp_i \\ &= \sum_i d\tilde{q}_i \wedge d\tilde{p}_i \end{aligned}$$

where the last equality follows by equation

□

1.2 Contact Manifolds

Throughout M is a manifold.

1.2.1 Basic concepts for contact manifolds

Definition 1.2.1. A k -dimensional Distribution on M is a choice of k -dimensional linear subspace $\xi_p \subset T_p M$ for each $p \in M$. This distribution is smooth if $\xi := \bigcup_{p \in M} \xi_p \subset TM$ is smooth sub-bundle. In this case we say ξ is a smooth distribution on M .

We have the following "local frame" criterion of smooth distributions

Proposition 1.2.1. *Let $\xi_p \subset T_p M$ constitute a k -dimensional distribution on M . Then $\xi = \bigcup_{p \in M} \xi_p \subset TM$ is smooth distribution on M if and only if every point $p \in M$ has a neighborhood U on which there are smooth vector fields $X_1, \dots, X_k \in \Gamma(U, TM)$ such that $X_1|_q, \dots, X_k|_q$ form a basis of ξ_q for every $q \in U$.*

We also have the the following 1-form criterion of smooth distributions

Proposition 1.2.2. *Let $\xi_p \subset T_p M$ constitute a k -dimensional distribution on M . Then $\xi = \bigcup_{p \in M} \xi_p \subset TM$ is smooth distribution on M if and only if every point $p \in M$ has a neighborhood U on which there are smooth 1-forms $\alpha^1, \dots, \alpha^{n-k}$ such that*

$$\xi_q = \ker \alpha_q^1 \cap \dots \cap \ker \alpha_q^{n-k} \quad (1.7)$$

Proof. Suppose $\alpha^1, \dots, \alpha^{n-k}$ are 1-forms satisfying equation (1.7) on neighborhood of p . We can extend them on possibly smaller neighborhood to smooth coframe $(\alpha^1, \dots, \alpha^n)$. Let (E_1, \dots, E_n) be the dual frame. Then by (1.7) it follows that ξ is spanned by E_{n-k+1}, \dots, E_n on neighborhood of p . It follows by 1.2.1 that ξ is smooth distribution.

Conversely, suppose ξ is smooth. Then by 1.2.1 there is a neighborhood of any $p \in M$ on which there are smooth vector field X_1, \dots, X_k spanning ξ . On possibly smaller neighborhood of p , these can be extended to smooth frame (X_1, \dots, X_n) . Let $(\epsilon^1, \dots, \epsilon^n)$ be the dual coframe, then it is easy to see

$$\xi_q = \ker \epsilon_q^{k+1} \cap \dots \cap \ker \epsilon_q^n$$

□

A particular case of the above proposition is the case of codimension 1 smooth distributions, for which at every point $p \in M$ there exist 1-form α on neighborhood U of p such that $\ker \alpha = \xi$ on U . We say α is *local defining form* of ξ near p . One observes by basic linear that if α and α' are two local defining forms of ξ on U and U' respectively, then $\alpha' = f\alpha$ on $U \cap U'$ for some smooth non-vanishing function $f : U \cap U' \rightarrow \mathbb{R}$.

Definition 1.2.2. A *contact structure* on M is a smooth distribution ξ of codimension 1 on M such that for every p there is a local defining form α near p such that $d\alpha|_{\xi}$ is non-degenerate (i.e symplectic). The pair (M, ξ) is called *contact manifold*, and any local defining form α with the above property is called *(local) contact form*. If α is defined on all of M , we say α is a global contact form.

Remark 1.2.1. In fact it follows that *any* local defining form of contact structure ξ is contact form. Indeed, let α' be local defining form on U , pick $q \in U$ and let α be contact form on neighborhood U' of q . Then on $U \cap U'$ we have $\alpha' = f\alpha$. Taking differentials,

$$d\alpha' = df \wedge \alpha + f d\alpha$$

By restricting to ξ (noting that $\alpha|_{\xi} = 0$), we get $d\alpha'|_{\xi} = f d\alpha|_{\xi}$. Hence $d\alpha'|_{\xi}$ is non-degenerate on $U \cap U'$.

Remark 1.2.2. It follows from non-degeneracy that $\text{rank} \xi = 2n$, and hence M has odd dimension $2n + 1$

From now on (M, ξ) is a contact manifold. We have the following obvious proposition of symplectic linear algebra.

Proposition 1.2.3. *Let V be vector space and $\Omega : V \times V \rightarrow \mathbb{R}$ be skew symmetric bilinear map, such that $\Omega|_{W \times W}$ is non-degenerate where W is subspace of V of codimension 1, then $V = W \oplus \ker \Omega$*

Accordingly, given local contact form α on contact manifold (M, ξ) . we get

$$T_p M = \ker \alpha_p \oplus \ker d\alpha_p$$

We have the following characterization of contact structures

Proposition 1.2.4. *ξ is contact structure on M if and only if $\alpha \wedge (d\alpha)^n \neq 0$ for any local defining 1-form.*

Proof. Take a basis $\{e_1, f_1, \dots, e_n, f_n, r\}$ of $T_p M$ such that $\text{span}\{e_1, \dots, f_n\} = \xi_p = \ker \alpha_p$ and $\text{span}\{r\} = \ker d\alpha_p$, then

$$\alpha_p \wedge (d\alpha_p)^n(r, e_1, f_1, \dots, e_n, f_n) = \alpha_p(r)(d\alpha_p)^n(e_1, f_1, \dots, e_n, f_n)$$

Here we used the formula

$$\omega \wedge \eta(X_1, \dots, X_n) = \sum_{(k,l) \text{ shuffles } \sigma} \omega(X_{\sigma(1)}, \dots, X_{\sigma(k)}) \eta(X_{\sigma(k+1)}, \dots, X_{\sigma(k+l)})$$

and the observation $(d\alpha_p)^n(\dots, r, \dots) = 0$. As $\alpha_p(r) \neq 0$, we conclude $\alpha_p \wedge (d\alpha_p)^n \neq 0$ if and only if $(d\alpha_p)^n(e_1, f_1, \dots, e_n, f_n) \neq 0$ if and only if $(d\alpha_p)^n|_{\xi_p} \neq 0$ if and only if $(d\alpha_p)|_{\xi_p}$ is non-degenerate by corollary 1.1.2 \square

Perhaps a natural question is when we can find a global contact, we have the following proposition.

Proposition 1.2.5. *A contact manifold (M, ξ) has a global contact form if and only if ξ is coorientable (that's TM/ξ is orientable).*

If a contact manifold (M, ξ) is given global contact form, we say (M, α) is contact manifold. We have the following definition.

Definition/Proposition 1.2.6. *Given a contact manifold (M, α) , there is a*

unique vector field R_α satisfying

$$\alpha(R_\alpha) = 1 \quad (1.8)$$

$$\iota_{R_\alpha} d\alpha = 0 \quad (1.9)$$

This vector field is called *Reeb vector field* determined by α . The contact form is invariant under the flow of R_α

Proof. $\ker d\alpha \simeq TM/\xi$ is a trivial bundle line bundle, hence there is a nowhere vanishing vector field \tilde{R} such that $\tilde{R}_p \in \ker d\alpha_p$, so $\alpha(\tilde{R})$. It follows that $R_\alpha = \frac{\tilde{R}}{\alpha(\tilde{R})}$ satisfies 1.8 1.9. Uniqueness is clear. The last statement follows directly from Cartan formula of lie derivatives. \square

Definition 1.2.3. A submanifold L of M is called *isotropic* if $T_x L \subset \xi_x$ for all $x \in L$

We have dimensional constraint on L

Definition/Proposition 1.2.7. Suppose $L \hookrightarrow (M^{2n+1}, \xi)$ is isotropic, then $\dim L \leq n$. If $\dim L = n$, then L is called *Legendrian submanifold*.

Proof. If L is isotropic, then $\alpha|_L = 0$. This implies $(d\alpha)|_L = d(\alpha|_L)$ if $T_x \subset (\xi_x, d\alpha_x)$ is isotropic, hence $\dim(T_x L) \leq \frac{1}{2} \dim(\xi_x) = n$ \square

Now we define *contactomorphism* of contact manifolds

Definition 1.2.4. Two contact manifolds (M_1, ξ_1) and (M_2, ξ_2) are said to be contactomorphic if there is diffeomorphism $f : M_1 \rightarrow M_2$ such tha $Tf(\xi_1) = \xi_2$. If $\xi_i = \ker \alpha_i$, then this is equivalent to $f^* \alpha_2 = \lambda \alpha$, where $\lambda : M_1 \rightarrow \mathbb{R} \setminus \{0\}$

Let $(M, \xi = \ker \alpha)$ be a contact manifold. A *contact isotopy* is a smooth family ψ_t of contactomorphisms with $\psi_0 = \text{id}$, then $\psi_t^* \alpha = \lambda_t \alpha$. Because $\lambda_0 = 1 > 0$, then

λ_t . Suppose that X_t is the generating time-dependent vector field. That's

$$\frac{d}{dt}\psi_t = X_t \circ \varphi_t$$

Then

$$\frac{d}{dt}\psi_t^*\alpha = \dot{\lambda}_t\alpha = \psi(\mu_t\alpha),$$

where $\mu_t = \dot{\lambda}_t \circ \psi_t^{-1}$. Because $\psi_t^*\mathcal{L}_{X_t}\alpha = \frac{d}{dt}\psi_t^*\alpha$, it follows that

$$\mathcal{L}_{X_t}\alpha = \mu_t\alpha$$

Conversely, given X_t satisfying this condition, we get

$$\frac{d}{dt}\psi_t^*\alpha = (\mu_t \circ \psi_t)\psi_t^*\alpha,$$

hence $\psi_t\alpha = e^{\int_0^t \mu_s \psi_s^* ds} \alpha$. A vector field X on M is called contact vector field. The discussion above illustrates that the lie algebra of group of contactomorphisms is the space of contact vector fields

Proposition 1.2.8. *Let $(M, \xi = \ker \alpha)$ be a contact manifold. Then there is a one-to-one correspondance between contact vector fields and functions $H : M \rightarrow \mathbb{R}$ given by*

- $X \rightarrow H_X = \alpha(X)$
- $X \rightarrow X_H$, defined uniquely by

$$\alpha(X_H) = H ; \iota_{X_H}d\alpha = dH(R_\alpha)\alpha - dH$$

Example 1.2.1.

(1) On \mathbb{R}^{2n+1} with coordinates $(x_1, y_1, \dots, x_n, y_n, z)$, define $\alpha_1 = dz - \sum_{i=0}^n y_i dx_i$. Let's compute compute $\alpha_1 \wedge (d\alpha_1)^n$:

$$d\alpha_1 = \sum_i dx_i \wedge dy_i \implies (d\alpha_1)^n = n! dx_1 \wedge dy_1 \wedge \dots \wedge dx_n \wedge dy_n$$

Hence

$$\alpha_1 \wedge d\alpha_1 = n! dz \wedge dx_1 \wedge dy_1 \wedge \cdots \wedge dx_n \wedge dy_n \neq 0.$$

α_1 is sometimes denoted by α_{std} . The reeb vector field is $R_{\alpha_{std}} = \frac{\partial}{\partial z}$

(2) Another contact structure on \mathbb{R}^{2n+1} is defined by the form

$$\alpha_2 = dz + \sum_{j=1}^n (x_j dy_j - y_j dx_j) = dz + \sum_{j=1}^n r_j^2 d\theta_j,$$

where (r_j, θ_j) is the polar coordinates in (x_j, y_j) -plane $j = 1, \dots, n$. It follows that $(\mathbb{R}^{2n+1}, \alpha_1)$ is strictly contactomorphic to $(\mathbb{R}^{2n+1}, \alpha_2)$. Indeed, let

$$f(x_1, y_1, \dots, x_n, y_n, z) = \left(\frac{y_1 - x_1}{2}, \frac{-x_1 - y_1}{2}, \dots, \frac{y_n - x_n}{2}, \frac{-x_n - y_n}{2}, z - \frac{1}{2} \sum_j x_j y_j \right)$$

Then f is diffeomorphism and

$$\begin{aligned} f^* \alpha_2 &= d\left(z - \frac{1}{2} \sum_j x_j y_j\right) + \frac{1}{4} \sum_j (y_j - x_j) d(-x_j - y_j) + (x_j + y_j) d(y_j - x_j) \\ &= dz - \frac{1}{2} \sum_j (x_j dx_j + y_j dy_j) + \frac{1}{4} \sum_j x_j dx_j + x_j dy_j - y_j dx_j - y_j dy_j + x_j dy_j - x_j dy_j + y_j dy_j - \\ &= dz - \sum_j y_j dx_j = \alpha_1 \end{aligned}$$

(3) Regard S^{2n+1} as the set of unit vectors

$$\{(x_1, y_1, \dots, x_{n+1}, y_{n+1}) \mid r = \sum_{i=1}^{n+1} (x_i^2 + y_i^2) = 1\}$$

We let $i : S^{2n+1} \hookrightarrow \mathbb{R}^{2n+1}$ be the inclusion. Let $\sigma = \frac{1}{2} \sum_i (x_i dy_i - y_i dx_i)$. We have

$$\begin{aligned}
(d(\sigma))^{n-1} &= \left(\sum_i dx_i \wedge dy_k - dy_k \wedge dx_k \right)^{n-1} \\
&= 2^n \left(\sum_k dx_k \wedge dy_k \right) \\
&= 2^n (n-1)! \sum_k dx_1 \wedge dy_n \wedge \cdots \wedge \widehat{dx_k \wedge dy_k} \wedge \cdots \wedge dx_k \wedge dy_k
\end{aligned}$$

and

$$\begin{aligned}
rdr \wedge \sigma &= \frac{1}{2} \left(\sum_i (x_i dx_i - y_i dy_i) \right) \wedge \left(\sum_j (x_j dy_j - y_j dx_j) \right) \\
&= \frac{1}{2} \sum_i \sum_j x_i x_j dx_i \wedge dy_j + y_i x_j dy_i \wedge dy_j - x_i y_j dx_i \wedge dx_j - y_i y_j dy_i \wedge dx_j
\end{aligned}$$

As can be easily seen, the wedge of each term on the right side of this equation with $i \neq j$ against $(d\sigma)^{n-1}$ is zero, hence $rdr \wedge \sigma \wedge (d\sigma)^{n-1}$ is

$$2^{n-1}(n-1)! \left(\sum_i (x_i^2 + y_i^2) dx_i \wedge dy_i \right) \wedge \left(\sum_k dx_1 \wedge dy_1 \wedge \cdots \wedge dx_k \wedge dy_k \wedge \cdots \wedge dx_{n+1} \wedge dy_{n+1} \right)$$

which is clearly equal to

$$2^{n-1} n! r^2 dx_1 \wedge dy_1 \wedge \cdots \wedge dx_{n+1} \wedge dy_{n+1}$$

which is not equal to 0 whenever $r \neq 0$. As S^{2n+1} is level surface of S^{2n+1} of r , then $\alpha = i^* \sigma$ defines contact structure on S^{2n+1} .

(4) Let X be a manifold of dimension n , then the jet space $J^1(X) := T^*X \times \mathbb{R}$ is contact manifold with contact form $\alpha = dz - \lambda_{can}$, where z is the real coordinate.

Indeed,

$$\alpha \wedge (d\alpha)^n = (dz - \lambda_{can}) \wedge (-d\lambda_{can})^n = (-1)^n dz \wedge d\lambda_{can}$$

The reeb vector field is $R_\alpha = \frac{\partial}{\partial z}$, and for any function $f : X \rightarrow \mathbb{R}$. The submanifold

$$\{(q, df(q), f(q)) | q \in X\} \subset J^1 M$$

is Legendrian.

Now we describe how contact manifolds arise from symplectic manifolds, but first let's give a definition

Definition 1.2.5. Let (W^{2n+2}, ω) be symplectic manifold. A vector field X on W is called *Liouville vector field* if $\mathcal{L}_X \omega = \omega$.

Remark 1.2.3. If (W, ω) has liouville vector field, then it is exact. Indeed by Cartan formula $\omega = \mathcal{L}_X \omega = d(\iota_X \omega) + \iota_X d\omega = d(\iota_X \omega)$

We have the required proposition

Proposition 1.2.9. *Let X be a liouville vector field on (W, ω) . Let $M \hookrightarrow W$ be a hypersurface (i.e submanifold of codimension 1). If X is transverse to M (i.e $X_p \notin T_p M$ for all $p \in M$), then $(M, \iota_X \omega|_M)$ is contact manifold.*

Proof. let $\tilde{\alpha} = \iota_X \omega$, we have

$$\tilde{\alpha} \wedge (d\tilde{\alpha})^n = (\iota_X \omega) \wedge (d\iota_X \omega)^n = (\iota_X \omega) \wedge \omega^n = \frac{1}{n+1} \iota_X (\omega^{n+1})$$

where in the last equality we used the antiderivation property of interior multiplication.

Because ω^{n+1} is volume form on W , it follows that $\tilde{\alpha} \wedge (d\tilde{\alpha})^n$ restricted to M is volume form, for if v_1, \dots, v_{2n+1} span $T_p M$, then $v_1, \dots, v_{2n+1}, X_p$ span $T_p W$, Hence

$$\tilde{\alpha}_p \wedge (d\tilde{\alpha}_p)^n(v_1, \dots, v_{2n+1}) = \omega_p^{n+1}(X_p, v_1, \dots, v_{2n+1}) \neq 0$$

□

In fact every contact manifold arises in this way, as we will see after defining symplectization.

1.2.2 Symplectization and Contactisation

Symplectization

Definition/Proposition 1.2.10. *Let (M^{2n+1}, ξ) and (N^{2n+1}, ξ) be contact manifolds. Then*

(1) *The following set*

$$W = \{(x, \beta) | x \in M, \beta \in T_p^*M \text{ s.t. } \ker(\beta) = \xi_p\}$$

*is symplectic submanifold of (T^*M, ω_{can}) and is called the symplectization of (M, ξ) .*

(2) *If ξ is cooriented and α is global form giving the coorientation, then the map*

$$\Phi_\alpha : \mathbb{R}^* \times M \rightarrow W, \ ; \ (t, x) \mapsto t\alpha_x$$

is symplectomorphism from $(\mathbb{R}^ \times M, d(\pi_2^*\alpha))$ to (W, ω_{can}) . In this case we call the component $(W_0 = \Phi_\alpha(\mathbb{R}_+ \times M), \omega_{can}|_{W_0})$ the intrinsic symplectization of M (It doesn't depend on choice of α). Given α , we call $(\mathbb{R} \times M, d(e^s\alpha))$ the extrinsic symplectization. It is symplectomorphic to W_0 via composition of $(s, x) \mapsto (e^s, x)$ and Ψ_α .*

(3) *let $f : M \rightarrow N$ be a diffeomorphism. Then, denoting by W and X the symplectizations of M and N respectively, f is contactmorphism if and only if $f_\#(W) = X$ in which case $f_\#|_W : W \rightarrow X$ is symplectomorphism. If M, N are cooriented by contact forms α, β , then the same assertion holds with W replaced with W_0 given that f preserves coorientations. If f is strict contactomorphism,*

then under identification above of intrinsic symplectizations $\mathbb{R} \times M$ and $\mathbb{R} \times N$ with W_0 and X_0 , $f_\#$ is given by $(s, x) \mapsto (s, f(x))$.

Proof.

(1) We begin by showing W is submanifold of dimension $2n + 2$. We let α be local defining form of ξ near p , we extend to local coframe $\alpha_1 = \alpha, \alpha_2, \dots, \alpha_n$ defined on $V \ni p$. Let (U, x_1, \dots, x_n) be a coordinate chart around p . We let for each $\gamma \in T_x^*U$, we let $y^i(\gamma)$ be the i -th component of γ , with respect to the above frame, that's $\gamma = \sum_i y^i(\gamma) \alpha_i|_x$. Defining $\bar{x}_i = x_i \circ \pi : T^*U \rightarrow \mathbb{R}$, we get coordinate chart of $(T^*U, \bar{x}_1, \dots, \bar{x}_n, y_1, \dots, y_n)$ of T^*M . Note that $W \cap T^*U$ is the intersection of $T^*M \setminus \{0\}$ and the set $\{y_2 = \dots = y_n = 0\}$, which is submanifold of T^*U . It follows that W is submanifold of T^*M . Now we check it is symplectic, that is $d\lambda_{can}|_W$ is symplectic. Let α be local defining form defined on U , and let $\phi : \mathbb{R}^* \times U \rightarrow W$ be defined by $\phi(t, x) = t\alpha_x$. We observe that $\phi^*(\lambda_{can}|_W) = t\alpha$. But as $\dim W = 2n + 1$ and

$$(d(t\alpha))^{n+1} = (dt \wedge \alpha + t d\alpha)^{n+1} = (n+1)t^n dt \wedge \alpha \wedge (d\alpha)^n$$

But the latter is non-zero, so $\phi^*(\omega_{can}|_W) = \phi^*(d\lambda_{can}|_W)$ is non-degenerate. This proves (1)

(2) follows from last part of the previous argument.

(3) Clear from the definitions. □

Now we make an important observation. Keeping the notation of the above proposition. If $\psi_t : M \rightarrow M$ is contact isotopy, then the the lift $\tilde{\psi}_t : W \rightarrow W$ Hamiltonian isotopy. Indeed, let \tilde{X}_t be the vector field generating $\tilde{\psi}$ and observe that

$$\psi_t^*(\lambda_{can}|_W) = \lambda_{can}|_W =: \theta$$

Differentiating this equation with respect to t , we get

$$0 = \mathcal{L}_{\tilde{X}_t} \theta = d(\theta(\tilde{X}_t)) + \iota_{\tilde{X}_t} d\theta$$

which proves \tilde{X}_t is Hamiltonian.

Contactisation

Let $(M, d\theta)$ be exact symplectic manifold. We define its contactisation to be (W, β) where $W = M \times \mathbb{R}$ with coordinate z on \mathbb{R} and $\beta = dz + \theta$. It is obvious that β is indeed a contact form, for

$$\beta \wedge (d\beta)^n = (dz + \theta) \wedge (dz + \theta)^n = dz \wedge (d\theta)^n$$

which is non-degenerate. The Reeb vector field is $R = \frac{\partial}{\partial z}$.

Now we say (L, ι) is a nice exact Lagrangian immersed submanifold in $(M, d\theta)$ if it is a generic immersed submanifold (in the sense that all self intersections are double, transversal and isolated) and $\iota^* \theta = df$ such that for every self-intersection point p the values of the potential f at the two preimages of p are distinct.

Any nice exact Lagrangian submanifold (L, ι) of M with potential f defines Lagrangian embedding

$$\iota^+ : L \rightarrow W, \quad x \rightarrow (\iota(x), -f(x))$$

The image of ι^+ is called Legendrian lift of (L, ι) , it is indeed legendrian $(\iota^+)^*(dz + \theta) = -df + \iota^* \theta = 0$. On the other hand any Legendrian submanifold of W projects to Lagrangian immersed submanifold of M via the projection $(x, z) \mapsto x$. Points of self intersection of L corresponds to Reeb chords of L^+ .

1.2.3 Moser type theorems

First of all we have the following theorem of Gray, which roughly states that there is not non-trivial deformations of contact structure.

Theorem 1.2.11. *If $\{\xi_t\}_{t \in [0,1]}$ is smooth family of contact structures on compact M . Then there is isotopy $\{\phi_t : M \rightarrow M\}_{t \in [0,1]}$ such that $(\phi_t)_*(\xi_0) = \xi_t$*

Proof. Let α_t be smooth family of 1-forms such that $\ker \alpha_t = \xi_t$. We want to find ϕ_t , such that

$$\phi_t^* \alpha_t = f_t \alpha_0 \quad (1.10)$$

for some non-vanishing functions $f_t : M \rightarrow \mathbb{R} - \{0\}$. We find ψ_t as flow of time dependent vector X_t , that's

$$\frac{d}{dt} \psi_t = X_t \circ \psi_t, \quad \psi_0 = \text{id}$$

Differentiating 1.10, we get

$$\psi_t^* \left(\frac{d}{dt} \psi_t + \mathcal{L}_{X_t} \alpha_t \right) = \frac{1}{f_t} \frac{df_t}{dt} \psi_t^* \alpha_t = \psi_t^* (g_t \alpha_t),$$

where $g_t = \left(\frac{1}{f_t} \frac{df_t}{dt} \right) \circ \psi_t$. If we choose X_t to lie in $\ker \alpha_t = \xi_t$, then the equation is satisfied if

$$\psi_t^* \left(\frac{d}{dt} \alpha_t + \iota_{X_t} d\alpha_t \right) = g_t \alpha_t \quad (1.11)$$

Plugging the Reeb vector field R_t in 1.11, we get

$$\frac{d}{dt} \alpha_t(R_t) = g_t$$

We use this to define g_t . By the non-degeneracy of $d\alpha_t|_{\xi_t}$, and the fact $R_t \in \ker(g_t \alpha_t - \frac{d}{dt} \alpha_t)$, there is a unique vector field X_t satisfying 1.11.

□

Theorem 1.2.12. *Let (M^{2n+1}, α) be a contact manifold. For any $p \in M$, there is*

a coordinate $(x_1, y_1, \dots, x_n, y_n, z)$ around p in which α is standard form, that's

$$\alpha = dz - \sum_{i=1}^n y_i dx_i$$

Proof. Let $(U, (u_1, \dots, u_{2n}))$ be coordinate cube, such that $R = R_\alpha = \frac{\partial}{\partial u_1}$. Let S be the submanifold defined by $u_1 = 0$. Since S is trasversal to R , then $d\theta|_S$ is symplectic, hence after shrinking U and S if necessary, we can find Darbouc coordinates $(x_1, y_1, \dots, x_n, y_n)$ for S and extend it to U by being constant on integral curves of R . Let θ be the 1-form $\sum_i y_i dx_i$ on U , so $d\alpha|_S = -d\theta|_S$. But as $\iota_R d\alpha = \iota_R d\theta$, it follows that $d\theta + d\alpha = 0$ at points of S . Then $\mathcal{L}_R \theta = \mathcal{L}_R \alpha = 0$ which implies $d(\theta + \alpha)$ is invariant under the flow of R , hence $d(\theta + \alpha) = 0$ on U . By Poincare lemma, there is smooth function $z : U \rightarrow \mathbb{R}$ such that $dz = \theta + \alpha$, we can assume that $z(p) = 0$. Since $dz_p(R_p) = 1$, we have $\{dx_i|_p, dy_i|_p, dz_p\}$ are linearly independent, so there is neighborhood around p on which $(x_1, y_1, \dots, x_n, y_n, z)$ are coordinates. \square

Theorem 1.2.13. *Let $L \subset (M, \xi)$ be Legendrian submanifold, then there exists neighborhood \mathcal{U} of L and neighborhood \mathcal{U}_0 of the zero section of $J^1(L)$ such that there exist contactomorphism $\psi : \mathcal{U}_0 \rightarrow \mathcal{U}$ such that $\psi(L_0) = L$, where L_0 is the zero section of L in $J^1(L)$ (in otherwords $L_0 = L \times \{0\}$)*

1.2.4 Generating families

Let M be a manifold, and let $f : M \times \mathbb{R}^N \rightarrow \mathbb{R}^N$ be a smooth map, where $M \times \mathbb{R}^N$ has coordinates (x, ξ) . Suppose that 0 is regular value of $\partial_\xi f : M \times \mathbb{R}^N \rightarrow \mathbb{R}^N$. The submanifold $(\partial_\eta f)^{-1}(0)$ is called the *fiber critical set* and denote it by Σ_f . Define

$$i_f : \Sigma_f \rightarrow T^*M \quad ; \quad (x, \eta) \mapsto (x, \partial_x f(x, \eta)),$$

and

$$j_f : \Sigma_f \rightarrow J^1 M \quad ; \quad (x, \eta) \mapsto (x, \partial_x f(x, \eta), f(x, \eta)),$$

it can be checked that i_f and j_f are immersions. The image L of i_f is immersed Lagrangian submanifold; the image λ of j_f is immersed Legendrian submanifold. We say that f generates L and Λ or that f is generating family. It is important to note that the Reeb chords of Λ are in bijective correspondence with critical points of $\delta(x, \xi, \xi') = f(x, \xi) - f(x, \xi')$ with positive critical value.

We say a generating family f is linear at infinity, if there is a linear function $A : \mathbb{R}^N \rightarrow \mathbb{R}$ such that $f(x, \xi) = A(\xi)$ outside a compact set.

Chapter 2

Lagrangian cobordism

2.0.1 Definition and compatible generating functions

Definition 2.0.1. Let Λ^- and Λ^+ be two legendrian submanifolds of $(M, \xi = \ker(\alpha))$, then a lagrangian cobordism between Λ_- and Λ_+ is a lagrangian submanifold of the symplectisation $(\mathbb{R} \times M, d(e^t \alpha))$ such that there exists T with

$$\bar{L} \cap ((-\infty, -T] \times X) = (-\infty, T] \times \Lambda_-$$

$$\bar{L} \cap ([T, \infty) \times X) = [T, \infty) \times \Lambda_+.$$

That cobordism is denoted by $\Lambda_- \prec \Lambda_+$.

We will be mainly concerned with lagrangian cobordisms of legendrian submanifolds in $(J^1 M, dz - \lambda)$. To use generating families to study them, we need to identify its symplectisation with cotangent bundle.

Proposition 2.0.1. *Let M be a manifold, then the symplectisation $(\mathbb{R} \times J^1 M, d(e^t \alpha))$ of $(J^1 M, \alpha = dz - \lambda)$ is symplectomorphic to $(T^*(\mathbb{R}_+ \times M), \omega_{can})$.*

In fact the symplectomorphism is given by

$$\theta : \mathbb{R} \times J^1 M \rightarrow T^*(\mathbb{R}_+ \times M) \quad ; \quad (s, q, p, z) \mapsto (e^s, q, z, e^s p)$$

Proof. Suppose that a point in $T^*(\mathbb{R}_+ \times M)$ has coordinates (t, x, z, y) , then

$$\begin{aligned} \theta^*(\omega_{can}) &= \theta^*(-d(zdt + ydx)) \\ &= -d(zd(e^s) + e^s pdq) \\ &= -d(-e^s dz + e^s pdq) = d(e^s \alpha) \end{aligned}$$

□

So the lagrangian cobordism \bar{L} can be viewed as Lagrangian submanifold of $T^*(\mathbb{R}_+ \times M)$, and hence may be constructed by generating function. We will be interested in the case, we have a generating function compatible with those of given ones for Λ_- and Λ_+

Definition 2.0.2. Let $f_{\pm} : M \times \mathbb{R}^N$ and $F : (\mathbb{R}_+ \times M) \times \mathbb{R}^N$ be functions. We say (F, f_-, f_+) is compatible if for some $S > 1$, get

$$f(t, x, \xi) = \begin{cases} tf_-(x, \xi) & t \leq 1/S \\ tf_+(x, \xi) & t \geq S \end{cases}$$

A *gf-compatible* lagrangian cobordism consists of a lagrangian cobordism $\Lambda_- \prec_{\bar{L}} \Lambda_+$ and a compatible triple of generating functions (F, f_-, f_+) of $\theta(\bar{L}) \subset T^*(\mathbb{R}_+ \times M)$, $\Lambda_-, \Lambda_+ \subset J^1 M$ respectively. A gf-compatible lagrangian cobordism is denoted by

$$(\Lambda_-, f_-) \prec_{(\bar{L}, F)} (\Lambda_+, f_+)$$

We should note that if we are given compatible triple (F, f_-, f_+) with 0 being a regular value of $\partial_{\xi} f_-, \partial_{\xi} f_+, \partial_{\xi} F$, then F determines an *immersed* lagrangian cobordism from λ_- to Λ_+ . We call the resulting immersed cobordism together with the triple a *gf-compatible immersed lagrangian cobordism*.

2.0.2 Isotopy is realised by lagrangian cobordism

The aim of this section is to prove the following theorem which is due to Chantraine [3]

Theorem 2.0.2. *[legendrian isotopy gives "cylindrical" cobordism] Let $(M, \xi = \ker \alpha)$ be a compact contact manifold, and $j_t : \Lambda \rightarrow M$ be isotopy of legendrian embeddings into M . Then there is a lagrangian cobordism from $\Lambda_0 = j_0(\Lambda)$ to $\Lambda_1 = j_1(\Lambda)$*

We need to prove isotopy extension theorem, which says that an isotopy of isotropic submanifolds can be realized by ambient contact isotopy. First we need the following lemma

Lemma 2.0.3. *Let M be a manifold, and S be a submanifold of M . Suppose that $f : S \rightarrow \mathbb{R}$ is smooth function, and λ is one form in N along S (that's $\lambda \in \Gamma(S, T^*M|_S)$). If*

$$df_p(v) = \lambda_p(v) \text{ for } v \in T_p S,$$

then there is $F : M \rightarrow \mathbb{R}$ such that

- $F|_S = f$
- $dF_p = \lambda_p$ for all $p \in S$

If S is compact, then we can choose F to be compactly supported.

We illustrate the lemma by proving the case $M = \mathbb{R}^m \times \mathbb{R}^n$, $S = \mathbb{R}^m \hookrightarrow M$ is submanifold by inclusion of the first coordinate. So we have as given $f : \mathbb{R}^m \rightarrow \mathbb{R}$ and smooth family of linear maps $\alpha_x : \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}$ parametrized by $x \in S$, such that $\alpha_x((v, 0)) = df_x(v)$. We construct F by the formula

$$F(x, y) = f(x) + \alpha_x((0, y))$$

We see that

- $F(x, 0) = f(x)$
- $dF_{(x,0)}(v_1, v_2) = df_x(v_1) + \alpha_x((0, v_2) = \alpha_x((v_1, v_2))$

This construction can be adapted to the general case by use of tubular neighborhood and cut off function.

Theorem 2.0.4 (isotopy extension theorem). *Let $j_t : L \rightarrow (M, \xi = \ker(\alpha))$ be an isotopy of isotropic embeddings of a closed manifold L in a contact manifold. Then there exist compactly supported contact isotopy ψ_t of (M, ξ)*

Proof. Define time-dependent vector field X_t along $j_t(L)$ by

$$X_t \circ j_t = \frac{d}{dt} j_t$$

Assume L is submanifold of M and j_0 the inclusion $L \subset M$. We want to find compactly supported smooth function $\tilde{H}_t : M \rightarrow \mathbb{R}$ whose hamiltonian vector field \tilde{X}_t equals X_t along $j_t(L)$, this will prove the theorem. \tilde{X}_t is defined in terms of \tilde{H}_t by

$$\alpha(\tilde{X}_t) = \tilde{H}_t \quad , \quad \iota_{X_t} d\alpha = d\tilde{H}_t(R_\alpha)\alpha - d\tilde{H}_t$$

We need

$$\alpha(X_t) = \tilde{H}_t \quad , \quad \iota_{X_t} d\alpha = d\tilde{H}_t(R_\alpha)\alpha - d\tilde{H}_t \quad \text{along } j_t(L) \quad (2.1)$$

We use the lemma above to construct \tilde{H}_t with 2.1 satisfied. Define $H_t : j_t(L) \rightarrow \mathbb{R}$ by $H_t = \alpha(X_t)$, and let λ_t be the one form along $j_t(L)$ defined by

$$\lambda_t = -\iota_{X_t} d\alpha$$

In particular $\lambda(R_\alpha) = 0$. So, what we need is \tilde{H}_t such that

- $\tilde{H}_t = H_t$ along $j_t(L)$

- $d\tilde{H}_t = \lambda$ along $j_t(L)$

By lemma, we need only to show that

$$dH_t(v) = \lambda_t(v) \text{ for } v \in T(j_t(L))$$

That's

$$d(\iota_{X_t}\alpha)(v) = -\iota_{X_t}d\alpha(v) \text{ for } v \in T(j_t(L))$$

This is equivalent to

$$j_t^*(\iota_{X_t}d\alpha) + j_t^*d(\iota_{X_t}\alpha) \equiv 0$$

which is in turn equivalent to

$$0 = j_t^*(\mathcal{L}_{X_t}) = \frac{d}{dt}(j_t^*\alpha)$$

but this is automatically true as j_t is isotropic embedding. \square

Now we sketch the proof of theorem 2.0.2. We let j_t be such isotopy, then we extend j_t to compactly supported contact isotopy ψ_t . This in turn lift to $\tilde{\psi}_t : \tilde{M} \rightarrow \tilde{M}$. As we have already seen in subsection 1.2.2, $\tilde{\psi}_t$ is generated by Hamiltonian vector field X_{H_t} for function $H_t : \tilde{M} \rightarrow \mathbb{R}$. Pick $S > 0$ We define Hamiltonian $H'_t : \tilde{M} \rightarrow \mathbb{R}$ by

$$H'_t(s, x) = \begin{cases} H_t(s, x) & s > S \\ 0 & s < -S \end{cases},$$

and let ϕ_t be its Hamiltonian flow, that's the one generated by $X_{H'_t}$. The following properties of ϕ_t are clear

- $\phi_t(s, x) = \tilde{\psi}_t(s, x) = (s, \psi_t(x))$ for $s > S$
- $\phi_t(s, x) = (s, x)$ for $s < -S$.

Let T be large enough so that $\phi_t([-S, S] \times M) \subset (-T, T) \times M$. Denote by \bar{L} the lagrangian submanifold $\phi_1(\mathbb{R} \times \Lambda_-)$ (note that ϕ_1 is hamiltonian diffeomorphism). Then by the properties above we get

- $\bar{L} \cap ([T, \infty) \times M) = [T, \infty) \times \Lambda_+$: By choice of T and properties of ϕ_t above, we note that $\phi_1(s, x)$ lies in $[T, \infty) \times M$ if and only if $s \geq T$. So the intersection on the left hand side is $\phi_1([T, \infty) \times \Lambda_-) = (\text{id} \times \psi_1)([T, \infty) \times \Lambda_-) = [T, \infty) \times \Lambda_+$ as required.
- $\bar{L} \cap ((-\infty, -T] \times M) = (-\infty, T] \times \Lambda_-$: Similar.

2.0.3 Some results on gf-compatible lagrangian cobordisms

Theorem 2.0.5. *Let $\Lambda \subset J^1M$ be a legendrian submanifold with linear-at-infinity generating function f , then there exist an immersed gf-compatible lagrangian cobordism $\emptyset \prec_{(\bar{L}, F)} (\Lambda, f)$*

Proof. The idea is to deform the f to linear function. First let f agree with linear function A outside a compact subset of $M \times \mathbb{R}^N$. We have to construct a function $F : \mathbb{R}_+ \times M \times \mathbb{R}^N \rightarrow \mathbb{R}$ satisfying

1. for all t , $F(t, x, \xi) = B_t(\xi)$ outside a compact set of $\{t\} \times M \times \mathbb{R}^N$ for some non-zero linear function B_t
2. There exists $T > 0$ such that $F(t, x, \xi)$ equals $B_t(\xi)$ for $t < 1/T$ and $tf(x, \xi)$ for all $t > T$.
3. 0 is regular value of $\partial_\xi F$

The construction of a function $G : \mathbb{R}_+ \times M \times \mathbb{R}^N \rightarrow \mathbb{R}$ satisfying (1) and (2) is simple. Just let $\rho : \mathbb{R}_+ \rightarrow \mathbb{R}$ which is 0 on $(0, 1]$ and 1 on $[2, \infty)$. Define G by

$$G(t, x, \xi) = t(\sigma(t)f(x, \xi) + (1 - \sigma(t))A(\xi)).$$

For any $T > 2$ to see that (1) and (2) are satisfied. Moreover there for $t < 1/T$, $G(t, x, \xi)$ agrees with $tA(\xi)$ outside a compact set of $\{t\} \times M \times \mathbb{R}^N$. Now we need to modify G to F to get the final property (3) satisfied. The modification is quite standard, we perturbate by adding $\epsilon \cdot \xi$ for well chosen ϵ . Note that as f is linear-at-infinity and the set of critical p, there exist a open and oints convex ball U around 0 which consists entirely of regular values of ∂_ξ . There exists $\epsilon \in U$ which is regular value of $\partial_\xi G$ and such that $\epsilon \cdot \xi \neq tA(\xi)$ for all t . Choose a smooth path $\sigma : \mathbb{R}_+ \rightarrow U$ such that $\sigma(t) = \epsilon$ for $t < 2$ and $\sigma(t) \cdot \xi \neq tA(\xi)$ for $t \in [2, T]$ and $\sigma(t) = 0$ for $t \in [T, \infty)$. We define F to be

$$F(t, x, \xi) = G(t, x, \xi) - \epsilon(t) \cdot \xi = \begin{cases} G(x, t, \xi) - \epsilon \cdot \xi & t \leq 2 \\ tf(x, \xi) - \sigma(t) \cdot \xi & t \in [2, T] \\ tf(x, \xi) & t \geq Tx \end{cases}$$

The properties (1) and (2) are clearly satisfied for G . Now we prove (3). On one hand ϵ is regular value of $\partial_\xi G$, so 0 is regular value of $\partial_\xi F = \partial_\xi G - \epsilon$ for $t \leq 2$. On the other hand we have for $t \geq 2$, $\partial_\xi f(t, x, \xi) = 0$ if and only if $\partial f(x, \xi) = \sigma(t)/t$, but by convexity of U , $\sigma(t)/t$ is regular value of $\partial f(x, \xi)$. It follows immediately that 0 is regular value of $\partial_\xi F$ as $d(\partial_\xi F)(t, x, \xi) : \mathbb{R} \oplus (T_x M \oplus \mathbb{R}^N) \rightarrow \mathbb{R}^N$ is

$$\begin{pmatrix} \partial_\xi f(t, \xi) - \sigma'(t) & t \cdot d(\partial_\xi f)(x, \xi) \end{pmatrix}$$

□

We need a lemma

Lemma 2.0.6. *Let $(f_t : M \times \mathbb{R}^N \rightarrow \mathbb{R})_{t \in \mathbb{R}_+}$ be a smooth one parameter family of generating functions. Let $F : \mathbb{R}_+ \times M \times \mathbb{R}^N \rightarrow \mathbb{R}^N$ be defined*

$$F(t, x, \xi) = tf_t(x, \xi)$$

Suppose that

- 0 is regular value of F
- for all (x, ξ, ξ') in the fiber critical set of $\delta_t(x, \xi, \xi') = f_t(x, \xi) - f_t(x, \xi')$, we have

$$\partial_x \delta_t(x, \xi, \xi') \implies \delta_t(x, \xi, \xi') \neq -t \partial_t \delta_t(x, \xi, \xi'),$$

then F generates an embedded submanifold in $T^*(\mathbb{R}_+ \times M)$

Proof. Since 0 is regular value of $\partial_\xi F$, then F generates an immersed lagrangian

$$\bar{L} = \{(t, x, f_t(x, \xi) + t \partial_t f_t(x, \xi), t \partial_x f_t(x, \xi)) | \partial_\xi f_t(x, \xi) = 0\}.$$

We need to find double points... So, double points are in bijective correspondence with points (t, x, ξ, ξ') with $\xi \neq \xi'$ satisfying

- (x, ξ, ξ') is in the fiber critical set of λ_t
- $\partial_x \delta_t(x, \xi, \xi') = 0$
- $\delta_t f(x, \xi, \xi') = -t \delta_t f(x, \xi, \xi')$

The lemma clearly follows. □

We have the following version of 2.0.2 with gf-compatible cobordisms.

Theorem 2.0.7. *Suppose that Λ_- is legendrian submanifold of $J^1 M$ with linear-at-infinity generating function f_- , and that Λ_- is legendrian isotopic to Λ_+ with linear-at-infinity generating function f_+ . Then there is a gf-compatible cobordism $(\Lambda_-, f_-) \prec_{(\bar{L}, F)} (\Lambda_+, f_+)$*

Proof. (Sketch.) Let $(j_t : \Lambda \rightarrow M)_{t \in \mathbb{R}_+}$ be the legendrian isotopy. Denote $\Lambda_t = j_t(\Lambda)$ such that $\Lambda_t = \Lambda_-$ for $t \leq 1/T$ and $\Lambda_t = \Lambda_+$ for $t \geq T$. There is a one-parameter family of generating functions linear-at-infinity f_t that generates Λ_t .

We will reparametrize the one-parameter family f_t , so that $F(t, x, \xi) = tf_t(x, \xi)$ satisfies lemma 2.0.6. Since 0 is regular value of $\partial_\xi f_t$ for all t , then 0 is regular value of $\partial_\xi F$. Observe that $\partial_x \delta_t(x, \xi, \xi') = 0$ if (x, ξ) and (x, ξ') are endpoints of reeb chord, where then the length of the reeb chord is $|\delta_t(x, \xi, \xi')|$. Let $h > 0$ be the minimum length of reeb chords of all of the legendrians Λ_t of the legnedrian isotopy. It suffices to show that for every (x, ξ, ξ') in the fiber critical set of δ_t

$$|\partial_t \delta_t(x, \xi, \xi')| < \frac{h}{t}.$$

Since the fiber critical set of δ_t is compact for each t , and $\partial_t \delta_t$ vanishes outside a compact interval, then $\partial_t \delta_t$ is bounded as function of (t, x, ξ, ξ') where (x, ξ, ξ') lies in the fiber critical set of δ_t . Choose $\rho : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ such that

$$0 < \rho'(t) < \frac{h}{t \max |\partial_t \delta_t|},$$

then letting $\tilde{f}_t = f_{\rho(t)}$, $\tilde{\delta}_t(x, \xi, \xi') = \tilde{f}_t(x, \xi) - \tilde{f}_t(x, \xi')$ we get

$$|\partial_t \tilde{\delta}_t(x, \xi, \xi')|_{t=t_0} = \rho'(t_0) |\partial_t \delta_t(x, \xi, \xi')|_{t=\rho(t_0)} < \frac{h}{t_0 \max |\partial_t \delta_t|} \cdot |\partial_t \delta_t(x, \xi, \xi')|_{t=\rho(t_0)} < \frac{h}{t_0}$$

□

Chapter 3

Lagrangian Surgery

Recall that an immersed submanifold of M is a pair (L, ι) of manifold L and immersion $\iota : L \rightarrow M$. In this chapter all immersed submanifolds are considered to be generic. That's all points of self intersections are double, transversal and isolated. So if x is a point of self intersection, then the pre-image of ι consists of two points $\{x_1, x_2\}$ and there is open neighborhood $U \subset M$ of x containing no other self intersection such that $\iota^{-1}(U)$ is union of disjoint open neighborhoods U_1, U_2 of x_1, x_2 respectively and $\iota|_{U_i}$ is embedding, in particular $U \cap \iota(L) = \iota(U_1) \cup \iota(U_2)$ and $\iota(U_1)$ intersect $\iota(U_2)$ transversally at x . An equipment of (L, ι) at x , is an order of tangent spaces $T_x \iota(U_1)$ and $T_x(\iota(U_2))$. We say (L, ι) oriented if L is oriented.

Throughout $P^n = S^{n-1} \times S^1$ and Q^n the manifold obtained from $S^{n-1} \times I$ by identifying $(x, 1)$ with $(\tau(x), 0)$ where $\tau : S^{n-1} \rightarrow S^{n-1}$ is the standard orientation reversing involution (given by reflection).

All symplectic manifolds are oriented by its symplectic structure, that's (M, ω) is oriented by ω^n .

3.1 First Description

Lagrangian handles

Let $(\mathbb{R}^{2n}, \omega_{std})$ be the standard symplectic manifold, suppose it is oriented by the form ω^n . Denote by L_x and L_y the Lagrangian submanifolds $\mathbb{R}^n \times \{0\}$ and $\{0\} \times \mathbb{R}^n$ respectively. We have the following definition

Definition 3.1.1. Let $\epsilon > 0$ and $\sigma : S^{n-1} \times \mathbb{R} \rightarrow \mathbb{R}^{2n}$ be a Lagrangian embedding satisfying:

- $\sigma(S^{n-1} \times (-\epsilon, \epsilon)) \subset B(0, \epsilon)$
- $\sigma(z, t) = -tz$ for $(z, t) \in S^{n-1} \times (-\infty, -\epsilon]$
- $\sigma(z, t) = tz$ for $(z, t) \in S^{n-1} \times [\epsilon, \infty)$.

So in particular, denoting the two n -dimensional discs of radius ϵ containing 0 by $B_x \subset L_x$ and $B_y \subset L_y$ respectively, we get

$$\sigma(S^{n-1} \times (-\infty, -\epsilon]) = L_x \setminus B_x ; \quad \sigma(S^{n-1} \times [\epsilon, \infty)) = L_y \setminus B_y$$

σ or its image is called an ϵ -Lagrangian handle. If Γ denotes this image and L_x and L_y are oriented, then we say the Lagrangian handle is *positive* (denoted $\text{sgn } \Gamma = 1$) if $L_x \setminus B_x \hookrightarrow \Gamma$ and $L_y \setminus B_y \hookrightarrow \Gamma$ induce the same orientations on Γ and *negative* (denoted $\text{sgn } \Gamma = -1$) if these orientations are different.

Lagrangian handles exist by the following theorem

Theorem 3.1.1. Suppose that L_x and L_y are oriented and let $\epsilon > 0$. Then there exists an ϵ -Lagrangian handle Γ with $\text{sgn } \Gamma = (-1)^{\frac{n(n-1)}{2}+1} L_x \cdot L_y$

Proof. Let f and g be functions $\mathbb{R} \rightarrow \mathbb{R}_+$ such that

- $f(t) = t$ for $t \geq \epsilon$ and $f(t) = 0$ for $t \leq -\epsilon$
- $g(t) = -t$ for $t \leq -\epsilon$ and $g(t) = 0$ for $t \geq \epsilon$
- $t \mapsto (f(t), g(t))$ is embedding $\mathbb{R} \rightarrow \mathbb{R}^2$
- $\sqrt{f(t)^2 + g(t)^2} \leq \epsilon$ for $t \in (-\epsilon, \epsilon)$

Define $\sigma : S^{n-1} \times \mathbb{R} \rightarrow \mathbb{R}^{2n}$ by

$$\sigma(z, t) = (f(t)z, g(t)z).$$

We have

1. $\sigma^* \omega_{std} = 0$: Let $\iota : S^{n-1} \times \mathbb{R} \hookrightarrow \mathbb{R}^n \times \mathbb{R}$ be the inclusion, and define $\tilde{\sigma} : \mathbb{R}^n \times \mathbb{R}$ by $\tilde{\sigma}(z, t) = (f(t)z, g(t)z)$, with coordinates of first factor \mathbb{R}^n be z_1, \dots, z_n . We have $\sigma = \tilde{\sigma} \circ \iota$, and

$$\begin{aligned} \tilde{\sigma}^* \omega_{std} &= \sum_i (f'(t)z_i dt + f(t)dz_i) \wedge (g'(t)z_i dt + g(t)dz_i) \\ &= \sum_i (f(t)g'(t) - g'(t)f(t))z_i dz_i \wedge dt \end{aligned}$$

and it follows that

$$\sigma^* \omega_{std} = \sum_i (f(t)g'(t) - g'(t)f(t))(z_i \circ \iota) d(z_i \circ \iota) \wedge dt = 0,$$

because $\sum_i (z_i \circ \iota)^2 = 1 \implies \sum_i (z_i \circ \iota) d(z_i \circ \iota) = 0$.

2. σ is immersion: We compute the differential, let $(v, s) \in T_{(t,z)}(S^{n-1} \times \mathbb{R}) = T_z S^{n-1} \times T_t \mathbb{R} = T_z S^{n-1} \mathbb{R}$, we have

$$d\sigma(z, t)(v, s) = (f'(t)sz + f(t)v, g'(t)sz + g(t)v)$$

As $\{z, v\} = 0$, then z and v are linearly independent unless $v = 0$. Hence for $(v, s) \neq (0, 0)$

$$d\sigma(z, t)(v, s) = 0 \implies f'(t) = g'(t) \text{ or } f(t) = g(t)$$

But this is impossible by choice of f and g

3. σ is injective: This follows by positivity of f and g and the last property of f and g .
4. σ is proper: clear from the construction.

It follows that σ is Lagrangian embedding. It is clear that conditions in definition of handle are satisfied, by choice of f and g . So σ is ϵ -lagrangian handle. The statement about sign is straightforward. \square

Construction of Lagrangian surgery

Next we turn to the construction of surgery of an immersed Lagrangian submanifold. Let (M, ω) be a symplectic manifold and (L, ι) be an immersed Lagrangian submanifold, and let x be a point of self-intersection of L , write $\iota^{-1}(x) = \{x_1, x_2\}$. Suppose that L is equipped at x by equipment (l_1, l_2) . Pick a neighborhood $U \subset M$ of x small enough so that

- There is open neighborhood $U \subset M$ of x containing no other self intersection such that $\iota^{-1}(U)$ is union of disjoint open neighborhoods U_1, U_2 of x_1, x_2 respectively and $\iota|_{U_i}$ is embedding, in particular $U \cap \iota(L) = \iota(U_1) \cup \iota(U_2)$. Denoting $\iota(U_i) = L_i$, we get L_1 intersecting L_2 transversally at x . We assume $T_x L_1 = l_1$ and $T_x L_2 = l_2$
- There is symplectomorphism $\varphi : (U, x) \rightarrow (B(0, \epsilon), 0)$ such that $\varphi^{-1}(L_x) = L_1$ and $\varphi^{-1}(L_y) = L_2$.

This possible by theorem 1.1.8. Now we let $\sigma : S^{n-1} \times \mathbb{R} \rightarrow \mathbb{R}^{2n}$ be a Lagrangian $\epsilon/3$ -handle with image Γ . Let $V = \varphi^{-1}(\overline{B(0, 2\epsilon/3)})$

$$\tilde{L} = (L \setminus \iota^{-1}(V)) \cup (S^{n-1} \times (-\epsilon, \epsilon)) / \sim$$

where \sim is the equivalence class generated by $(z, t) \sim \iota^{-1}\varphi^{-1}\sigma(z, t)$ for $t \in (-\epsilon, -2\epsilon/3) \cup (2\epsilon/3, \epsilon)$. There is a unique manifold structure on \tilde{L} such that the inclusions

$$i_1 : L \setminus \iota^{-1}(V) \hookrightarrow \tilde{L}, \quad i_2 : S^{n-1} \times (-\epsilon, \epsilon) \hookrightarrow \tilde{L}$$

are open embeddings. We define

$$\tilde{\iota} : \tilde{L} \rightarrow M$$

by

$$\tilde{\iota}(p) = \begin{cases} \iota(x) & \text{if } p = [x], \quad x \in L \setminus \iota^{-1}(V) \\ \varphi^{-1}\sigma(z, t) & \text{if } p = [z, t], \quad (z, t) \in S^{n-1} \times (-\epsilon, \epsilon) \end{cases}$$

This is clearly well defined and is smooth Lagrangian immersion because the composition with i_1 and i_2 are smooth Lagrangian immersions. It is seen that the image of this immersion coincides with $\iota(L)$ outside V and has one self intersection removed. The image is exactly $(L \setminus U) \cup \varphi^{-1}(\Gamma)$. We call $(\tilde{L}, \tilde{\iota})$ a result of Lagrangian surgery at point x . It is a generic immersed submanifolds, because self intersections happen outside U , where the new immersion coincides with the old one. If L is oriented, then L_1 and L_2 inherits orientation and we say the surgery is positive(negative) if the handle Γ is positive(negative) with respect to orientations on L_x and L_y induced by those of L_1 and L_2 respectively. So the surgery sign depends on the equipment in fact the sign is exactly $(-1)^{n(n-1)/2+1}l_1 \cdot l_2$. By changing equipment, we can control the sign as in following proposition which is clear

Proposition 3.1.2. *Let (L, ι) be an oriented immersed Lagrangian submanifold with point of self intersection x*

1. If n is odd, then there exists a Lagrangian surgery of any sign at point x .
2. If n is even, then there exists Lagrangian surgery of sign $(-1)^{n(n-1)/2+1}\text{ind}(x)$.

Remark 3.1.1. If (L, ι) is oriented and the surgery is positive, then (\tilde{L}, ι) inherits natural orientation which agrees with orientation of L under embedding $i_1 : L \setminus \iota^{-1}(V) \hookrightarrow \tilde{L}$.

The following two propositions will follow directly from constructions:

Proposition 3.1.3. *Let (L, ι) be immersed Lagrangian with L connected and let x be a point of self intersection. Let $(\tilde{L}, \tilde{\iota})$ be a result of Lagrangian surgery at x .*

1. *If L is oriented, Then $\tilde{L} \cong L \# P$ if the surgery is positive and $\tilde{L} \cong L \# Q$ if the surgery is negative.*
2. *If L is non-orientable, then $\tilde{L} \cong L \# Q \cong L \# P$.*

On the other hand if $(L = L_1 \sqcup L_2, \iota)$ is immersed submanifold where L_1 and L_2 are connected and x is a point of self-intersection, then denoting by $(\tilde{L}, \tilde{\iota})$ a result of Lagrangian surgery at x , we have $\tilde{L} \cong L_1 \# L_2$

Proposition 3.1.4. *Let (L, ι) be a closed immersed Lagrangian submanifold of (M^{2n}, ω) , and let $(\tilde{L}, \tilde{\iota})$ be a result of Lagrangian surgery.*

1. *$\iota_*([L])$ and $\tilde{\iota}_*([\tilde{L}])$ are homologous (mod 2)*
2. *if L is oriented and the surgery is positive, then $\iota_*([L])$ and $\tilde{\iota}_*([\tilde{L}])$ are homologous, where \tilde{L} is given the natural orientation of remark.*

Now we give an application to the surgery construction on embedding problems:

Theorem 3.1.5. *1. Let L_1 and L_2 be closed connected manifolds with Lagrangian embedding into \mathbb{R}^{2n} . Then there is a Lagrangian embedding of $L_1 \# L_2 \# Q$ into \mathbb{R}^{2n} . If n is odd, then $L_1, \# L_2 \# P$ admits Lagrangian embedding into \mathbb{R}^{2n} .*

2. Let L be a closed connected manifold having Lagrangian immersion into \mathbb{R}^{2n} . Then there is a Lagrangian embedding of $L \# kQ$ into \mathbb{R}^{2n} for some k . If n is odd, then there is Lagrangian embedding of $L \# kP$.

Proof. To prove (1) Let $i_j : L_j \hookrightarrow \mathbb{R}^{2n}$ denote the Lagrangian embeddings. By linear map and perturbation we can assume that $i_1(L_1)$ and $i_2(L_2)$ intersect transversally at exactly two points x_1 and x_2 . Let $(\tilde{L} = L_1 \sqcup L_2, \tilde{\iota})$ be defined by $\tilde{\iota}(x) = i_j(x)$ if $x \in L_j$. Then $(\tilde{L}, \tilde{\iota})$ is immersed Lagrangian submanifold with two points of self intersections x_1, x_2 . We perform Lagrangian surgery at x_1 , to get an immersed Lagrangian submanifold $(\tilde{L}^{(1)}, \tilde{\iota}^{(1)})$. By proposition 3.1.3, $\tilde{L}^{(1)} \cong L_1 \# L_2$. If one of L_1 or L_2 is non-orientable, then $\tilde{L}^{(1)}$ is non-orientable, and by the same proposition after applying a surgery at x_2 we get a generic immersed submanifold $(\tilde{L}^{(2)}, \tilde{\iota}^{(2)})$ such that $\tilde{L}^{(2)} \cong \tilde{L}^{(1)} \# Q \cong L_1 \# L_2 \# Q$. But this immersed submanifold has no self intersections and generic, so we get embedding $L_1 \# L_2 \# Q \hookrightarrow \mathbb{R}^{2n}$. Now assume that both L_1 and L_2 are oriented, and fix a choice of orientations. We have two cases

- Let n be odd, then $\tilde{L}^{(2)}$ is $\tilde{L}^{(1)} \# Q$ or $\tilde{L}^{(2)} \# P$ according to the choice of sign of surgery, so we get embeddings $L_1 \# L_2 \# P \hookrightarrow \mathbb{R}^{2n}$ and $L_1 \# L_2 \# Q \hookrightarrow \mathbb{R}^{2n}$.
- Let n be even. Suppose that $\text{ind}(x_1) = (-1)^{\frac{n(n-1)}{2}+1}$. Then the first surgery is positive and we get the natural orientation on $(\tilde{L}^{(1)}, \tilde{\iota}^{(1)})$, because of compatibility of this orientation with that of $(\tilde{L}, \tilde{\iota})$, we get the new index at x_2 the same as old index at the same point which is $-\text{ind}(x_1)$, so the second surgery is negative, and we get $\tilde{L}^{(2)} = \tilde{L}^{(1)} \# Q$. So we get embedding $\tilde{\iota}^{(2)} : \tilde{L}^{(2)} \cong L_1 \# L_2 \# Q \hookrightarrow \mathbb{R}^{2n}$

Proving (2) is similar by performing surgery k times. □

Another application is

Theorem 3.1.6. Let (M^{2n}, ω) be a symplectic manifold. Suppose that $G = \mathbb{Z}$ when n is odd and $G = \mathbb{Z}_2$ when n is even. Let $\alpha \in H_n(M, G)$ be represented by

Lagrangian immersion (that's there is a G -oriented Lagrangian immersed submanifold (L, ι) such that $\iota_([L]) = \alpha$). Then α can be represented by a Lagrangian embedding.*

Proof. Let α be represented by (L, ι) , we know that ι is homotopic to ι' which is generic. ι' also represents α . We can apply successive Lagrangian surgeries to get embedded submanifold $(\tilde{L}, \tilde{\iota}')$. If L is oriented and n is odd, then we can demand the surgeries to be positive. The theorem follows by proposition 3.1.4.

□

3.2 Second Discription via cobordism

Basically I will reproduce section 8.2 of the paper [4] here. In exact symplectic manifold, we construct surgery of lagrangian submanifold obtained by first lifting the submanifold to the contactization, and then remove reeb chords, and then projecting back. There is lagrangian cobordism from the lift before surgery to that after surgery.

Local model of Lagrangian surgery

Let $\eta, \delta > 0$, and consider the open subset

$$V_{\eta, \delta} = \{(q, p, z) \mid |q| < \eta, |p| < 2\delta, z \in \mathbb{R}\}$$

let $\zeta > 0$, and denote by $\Lambda_{\eta, \delta, \zeta}^+$ the submanifold given by the two sheets

$$\{(q, \pm df_{\eta, \delta, \zeta}(|q|), \pm f_{\eta, \delta, \zeta}(|q|)) \mid |q| \leq \eta\} \text{ where } f_{\eta, \delta, \zeta}(s) = \frac{\delta}{2\eta}s^2 + \frac{\zeta}{2}$$

which is legendrian as we have seen in Chapter 1. More explicitly,

$$\Lambda_{\eta,\delta,\zeta}^+ = \{(q, \pm \frac{\delta}{\eta}q, \pm \frac{\delta}{2\eta}|q|^2 \pm \frac{\zeta}{2}) \mid |q| < \eta\}$$

It is seen that it has one Reeb chord of length ζ , namely the one joining $(0, 0, -\frac{\zeta}{2})$ to $(0, 0, \frac{\zeta}{2})$. We note that $\Lambda_{\eta,\delta,\zeta}$ can be given by generating function

$$F_{\eta,\delta,\zeta}^+ : B(0, \eta) \times \mathbb{R} \rightarrow \mathbb{R}, (q, \xi) \mapsto \frac{\xi^3}{3} - g_{\eta,\delta,\zeta}^+(|q|)\xi,$$

where

$$g^+(s) = \left(\frac{3}{2} f_{\eta,\delta,\zeta}(s) \right)^{3/2}$$

Now let $g_{\eta,\delta,\zeta}^- : \mathbb{R}_+ \rightarrow \mathbb{R}$ be a function such that

- $g_{\eta,\delta,\zeta}^-(s) = \left(\frac{3}{2} f_{\eta,\delta,\zeta}(s) \right)^{3/2}$ for $s > \frac{3\eta}{4}$
- $g_{\eta,\delta,\zeta}^-(s) < 0$ for $s < \frac{\eta}{2}$
- $0 < (g_{\eta,\delta,\zeta}^-)'(s) < \frac{2\delta\eta}{\delta\eta + \zeta}$

The last condition can be achieved if $\zeta < \frac{7\delta\eta}{16}$. Let $\Lambda_{\eta,\delta,\zeta}^- \subset J^1B(0, \eta)$ be the legendrian submanifold generated by

$$F_{\eta,\delta,\zeta}^- : B(0, \eta) \times \mathbb{R} \rightarrow \mathbb{R}, (q, \xi) \mapsto \frac{\xi^3}{3} - g_{\eta,\delta,\zeta}^-(|q|)\xi$$

By the last condition $\Lambda_{\eta,\delta,\zeta}^-$ is subset of $V_{\eta,\delta,\zeta}$. Also we should note that $\Lambda_{\eta,\delta,\zeta}^+$ agrees with $\Lambda_{\eta,\delta,\zeta}^+$ in $\{(q, p, z) \mid 3\eta/4 < |q| < \eta, |p| < 2\delta, z \in \mathbb{R}\}$.

Constuction of Surgery and cobordism

Fix an exact symplectic manifold (M, θ) . Let L be a generic nice exact Lagrangian immersed submanifold with self intersection points x_1, \dots, x_n and let L^+ be the

legendrian lift to the contactization (W, β) . The self intersection point x_i corresponds to a Reeb chord which we denote by a_i .

Definition 3.2.1. With the above notation, the set of Reeb chords $\{a_1, \dots, a_n\}$ is called contractible if there are disjoint neighborhoods U_i of a_i such that there is a strict contactomorphism $(U_i, U_i \cap \iota^+(L^+)) \cong (V_{\eta_i, \delta_i, \zeta_i}, \Lambda_{\eta_i, \delta_i, \zeta_i}^+)$ for $\eta_i, \delta_i, \zeta_i > 0$ with $\zeta_i < \frac{7\delta_i\eta_i}{16}$.

We let $\{a_1, \dots, a_n\}$ be a set of contractible Reeb chords. Denote by $\phi_i : (U_i, U_i \cap \iota^+(L^+)) \rightarrow (V_{\eta, \delta, \zeta}, \Lambda_{\eta, \delta, \zeta}^+)$ a strict contactomorphism as in definition. Let $L^+(a_1, \dots, a_n)$ be the legendrian submanifold $(L^+ \setminus \bigcup_i (U_i)) \cup \bigcup_i \Lambda_{\eta_i, \delta_i, \zeta_i}^-$. This is indeed a legendrian submanifold by the fact that $\Lambda_{\eta, \delta, \zeta}^+$ agrees with $\Lambda_{\eta, \delta, \zeta}^+$ in $\{(q, p, z) | 3\eta/4 < |q| < \eta, |p| < 2\delta, z \in \mathbb{R}\} \subset V_{\eta_i, \delta_i, \zeta_i}$, and that ϕ_i is strict contactomorphism.

Denote by $L(x_1, \dots, x_n)$ the Lagrangian projection of $L^+(x_1, \dots, x_n)$, then $L(a_1, \dots, a_n)$ agrees with L outside neighborhoods of x_i and has k self intersections removed. The latter fact follows because ζ_i can be made arbitrarily small, so no Reeb chord is created when passing from $\Lambda_{+\eta_i, \delta_i, \zeta_i}$ to $\Lambda_{\eta_i, \delta_i, \zeta_i}^-$.

Next we construct Lagrangian cobordism from $L^+(a_1, \dots, a_n)$ to L^+ , but first we construct a Lagrangian cobordism from $\Lambda_{\eta, \delta, \zeta}^-$ to $\Lambda_{\eta, \delta, \zeta}^+$ in $\mathbb{R} \times J^1B(0, \eta)$. In fact we construct a gf-compatible cobordism. Define $G_{\eta, \delta, \zeta} : \mathbb{R}_+ \rightarrow \mathbb{R}$ to be a function satisfying for $S > 1$ the following properties

- $G_{\eta, \delta, \zeta}(t, s) = g_{\eta, \delta, \zeta}^-(s)$ for $s < 1/S$
- $G_{\eta, \delta, \zeta}(t, s) = g_{\eta, \delta, \zeta}^+(s)$ for $s > S$
- $\frac{\partial}{\partial t} G(t, 0) > 0$
- $G_{\eta, \delta, \zeta}(t, s) = g_{\eta, \delta, \zeta}^+(s) = g_{\eta, \delta, \zeta}^-(s)$ for $s > 3\eta/4$

We define

$$F_{\eta,\delta,\zeta} : (\mathbb{R}_+ \times B(0, \eta)) \times \mathbb{R} \rightarrow \mathbb{R}, \quad (t, x, \xi) \mapsto t \cdot \left(\frac{\xi^3}{3} + G_{\eta,\delta,\zeta}(t, |q|) \right)$$

According to Chapter 2, this gives immersed Lagrangian cobordism $\Sigma_{\eta,\delta,\zeta}$ in $\mathbb{R} \times J^1 B(0, \eta)$ from $\Lambda_{\eta,\delta,\zeta}^-$ to $\Lambda_{\eta,\delta,\zeta}^+$ with points of self intersection possibly corresponding to critical points of

$$\delta_{F_{\eta,\delta,\zeta}}(t, q, \xi_1, \xi_2) = F_{\eta,\delta,\zeta}(t, q, \xi_1) - F_{\eta,\delta,\zeta}(t, q, \xi_2)$$

with positive values, But the third property of G eliminates this possibility. Thus the cobordism is embedded.

In the trivial cobordism $\mathbb{R} \times L^+$, we replace $\mathbb{R} \times (U_i \cap L^+)$ with the $(\text{id} \times \phi_i)^{-1}(\Sigma_{\eta_i, \delta_i, \zeta_i})$. This can be glued to yield smooth embedded submanifold $\mathbb{R} \times W$ because of the fourth property of G , and it is Lagrangian because $\text{id} \times \phi_i : \mathbb{R} \times W \rightarrow \mathbb{R} \times J^1 B(0, \eta_i)$ is symplectomorphism. So we have obtained cobordism from $L^+(a_1, \dots, a_n)$ to L^+ .

Bibliography

- [1] Frederic Bourgeois, Joshua M Sabloff, and Lisa Traynor. *Lagrangian cobordisms via generating families: Construction and geography*. Algebr. Geom. Topol., Volume 15, Number 4 (2015), 2439-247.
- [2] Robert Bryant. <https://mathoverflow.net/questions/332965/lagrangian-surgery>.
- [3] Baptiste Chantraine. Lagrangian concordance of legendrian knots. *Algebr. Geom. Topol.*, 10(1):63–85, 2010.
- [4] Baptiste Chantraine, Georgios Dimitroglou Rizell, Paolo Ghiggini, and Roman Golovko. *Geometric generation of the wrapped Fukaya category of Weinstein manifolds and sectors*. arXiv:1712.09126.
- [5] H.Geiges. *An Introduction to Contact Topology*. Cambridge University Press, 2008.
- [6] Dusa Macdoff and Dietmer Salamon. *Introduction to Symplectic Topology* . Oxford University Press, 1999.
- [7] L. Polterovich. The surgery of lagrange submanifolds. *Geometric & Functional Analysis GAFA*, 1991.

Année Universitaire 2019-2020

Master 2 Mathématiques Fondamentales et Appliquées
Parcours Algèbre et Géométrie

Existence d'un 3-filet géodésique stationnaire sur une 2-sphère

Par : **Jean Chartier**

Sous la direction de : **Laurent Hauswirth** et **Samuel Tapie**

Date de soutenance : **07/07/2020**

Université de Nantes

Faculté des Sciences et des Techniques

Introduction

Le présent mémoire vient à la fois clore un cycle d'études à l'Université de Nantes et ouvrir une perspective de recherche en mathématiques. En effet, le Master de Mathématiques Fondamentales et Appliquées (MFA) proposé à Nantes, prévoit en deuxième année l'écriture d'un mémoire, sous la direction d'un chercheur en mathématiques. L'objet de ce travail peut être une analyse d'article de recherche récent, ou la résolution d'un problème ouvert raisonnable. J'ai travaillé sous la direction de Laurent Hauswirth, Professeur à l'Université de Marne la Vallée et Samuel Tapie, Maître de Conférences à l'Université de Nantes. Je les remercie chaleureusement pour l'aide qu'ils m'ont apportée, dans des circonstances de confinement délicates. Les objets étudiés et les outils utilisés dans ce mémoire sont préalables à un futur travail de recherche en thèse, dirigé par Laurent Hauswirth et Arnaud de Mesmay. Il s'agira de montrer l'existence, dans une 3-sphère riemannienne, de doubles bulles stationnaires qui divisent la courbure totale de la sphère en trois parts égales. Pour l'heure nous tentons de résoudre un problème analogue, en plus basse dimension : l'existence d'un 3-filet géodésique stationnaire à la surface d'une 2-sphère riemannienne, qui divise la courbure totale de la sphère en trois parts égales. Nous nous appuyons sur des techniques de balayages introduites par Birkhoff et reprises dans un article de Colding et Minicozzi, *Min-Max for sweepouts by curves*, écrit en 2007. Le Chapitre 1 donne une caractérisation de l'objet étudié. Les Chapitres 2 et 3 décrivent les approches de plusieurs chercheurs qui ont franchi des étapes décisives dans ce domaine. Dans les chapitres 4 et 5 enfin, nous esquissons une preuve d'existence dans un cas particulier de sphères et donnons des résultats qui pourraient aider notre recherche à venir.



FIGURE 1 – Jacques Lipchitz - The Rescue, 1945

Chapitre 1

Filets géodésiques stationnaires.

Considérons une variété Riemannienne M et un multi-graphe G . Un *filet* \mathcal{F} sur M modelé par G est un plongement \mathcal{C}^1 par morceaux de G dans M . Les sommets et arêtes du graphe sont ainsi envoyés sur les sommets et arêtes du filet - ou net suivant l'usage anglais. Un filet est dit *géodésique* si chacune de ses arêtes sont des géodésiques. En outre, un filet est dit *stationnaire* si, pour tout flot Φ_t de difféomorphismes de M , la fonctionnelle de longueur $L(\Phi_t(\mathcal{F}))$ admet un point critique en zéro. Il ne s'agit pas nécessairement d'un extremum local ; cela complique sérieusement la recherche d'un tel filet. Le théorème suivant montre en particulier qu'un filet non géodésique ne saurait être stationnaire. C'est pourquoi un filet stationnaire sera souvent appelé *filet géodésique stationnaire*. Un *filet géodésique minimisant* peut être deux choses distinctes en général. Soit un filet géodésique stationnaire, critique en tant que minimum pour la fonctionnelle de longueur. Soit un filet géodésique stationnaire dont les arêtes réalisent la distance (cf. Ian Adelstein). Ces conditions ne sont ni nécessaires ni suffisantes l'une pour l'autre.

Exhibons maintenant une caractérisation utile - qui chez certains auteurs fait office de définition - du filet stationnaire.

Theorème 1.1

Soit \mathcal{F} un filet sur une variété Riemannienne M . Une condition nécessaire et suffisante pour que \mathcal{F} soit stationnaire est qu'en chacun de ses sommets, la somme des vecteurs unitaires tangents aux arêtes soit nulle - cela correspond, pour les sommets de valence trois, à une trisection angulaire - et qu'il soit géodésique.

Démonstration. Pour simplifier l'écriture, nous supposons que \mathcal{F} est constitué de deux sommets A et B , reliés par n arêtes, disjointes en dehors de A et B . Cette restriction n'est que combinatoire et n'affecte pas l'essentiel de la preuve. Supposons d'abord que les arêtes c_1, \dots, c_n de \mathcal{F} sont des géodésiques paramétrées par longueur d'arc comme suit :

$$\forall i \in \{1, \dots, n\}, c_i : [0, l_i] \longrightarrow M$$

Avec les rencontres suivantes : $c_i(0) = A$ et $c_i(l_i) = B$.

Supposons aussi la condition aux sommets :

$$\sum_{i=1}^n c'_i(0) = 0 \text{ et } \sum_{i=1}^n c'_i(l_i) = 0$$

Soit maintenant un groupe \mathcal{C}^1 de difféomorphismes à un paramètre Φ_t .

On pose, pour $u \in [0, l_i]$ et $i \in \{1, \dots, n\}$:

$$V_i(u) = \frac{d}{dt} \Phi_t(c_i(u))|_{t=0}$$

On veut dériver en zéro la fonctionnelle de longueur totale :

$$L(t) = \sum_{i=1}^n \int_0^{l_i} \left| \frac{\partial}{\partial u} \Phi_t(c_i(u)) \right| du$$

La première formule variationnelle (\star) donne :

$$L'(0) = \sum_{i=1}^n \left(\langle c'_i, V_i \rangle \Big|_0^{l_i} - \int_0^{l_i} \langle \nabla_{c'_i} c'_i(u), V_i(u) \rangle du \right)$$

Or $\forall i \in \{1, \dots, n\}$, $c''_i = 0$ et $V_1(0) = \dots = V_n(0)$ et $V_1(l_1) = \dots = V_n(l_n)$. Donc :

$$L'(0) = \langle \sum_{i=1}^n c'_i(l_i), V_i(l_i) \rangle - \langle \sum_{i=1}^n c'_i(0), V_i(0) \rangle$$

Ainsi, sous nos hypothèses, $L'(0) = 0$, i.e. \mathcal{F} est stationnaire.

Réciproquement, nous devons montrer une formule de la forme $P \implies (Q \wedge R)$. Une table de vérité montre qu'elle est équivalente à la formule $P \implies Q \wedge (\neg R \wedge Q) \implies \neg P$. Montrons d'abord qu'un filet stationnaire est nécessairement géodésique. Supposons que $L'(0)$ donnée par (\star) est nulle, pour tout flot Φ_t de M . En particulier, pour un champ vectoriel V sur M , restreint à \mathcal{F} , nul en 0 et en l_i , on a :

$$\sum_{i=1}^n \int_0^{l_i} \langle \nabla_{c'_i} c'_i(u), V_i(u) \rangle du = 0$$

Si de plus V_i est positivement colinéaire à $\nabla_{c'_i} c'_i$ le long de c_i et $\forall j \neq i, V_j = 0$, alors il vient :

$$\int_0^{l_i} \langle \nabla_{c'_i} c'_i(u), V_i(u) \rangle du = 0 \text{ avec } \forall u \in [0, l_i], \langle \nabla_{c'_i} c'_i(u), V_i(u) \rangle = k(u) \|\nabla_{c'_i} c'_i(u)\|^2 \geq 0$$

D'où nécessairement : $\nabla_{c'_i} c'_i = 0$.

Nous n'avons pas formulé d'hypothèses sur i , donc toutes les arêtes de \mathcal{F} sont des géodésiques.

Montrons ensuite qu'un filet géodésique qui ne satisfait pas la condition de somme nulle aux sommets ne peut être stationnaire. Supposons qu'en A , on ait $\Sigma =: \sum_{i=1}^n c'_i(0) \neq 0 \in T_A M$.

On transporte Σ parallèlement le long de chaque c_i en $\Sigma_i(u)$. En particulier, $\forall i \in \{1, \dots, n\}$, $\Sigma_i(0) = \Sigma$. On construit i champs de vecteurs le long de c_1, \dots, c_n en réduisant quadratiquement les Σ_i dans un rayon géodésique r suffisamment petit devant \mathcal{F} :

$$V_i(u) = 0 \text{ pour } u > r \text{ et } V_i(u) = \Sigma_i(u) \left(\frac{u-r}{r} \right)^2 \text{ pour } u \in [0, r[$$

On construit une variation de \mathcal{F} autour de A , raccordée à l'identité au-delà, de manière \mathcal{C}^1 :

$$\forall i \in \{1, \dots, n\}, \forall u \in [0, r[, \forall t \in [-\epsilon, \epsilon], c_i^t(u) = \exp_{c_i(u)}(tV_i(u))$$

On construit un flot Φ_t sur M qui est l'identité à l'extérieur du disque géodésique $D(A, r)$ et qui, à l'intérieur du disque, se comporte le long de $\mathcal{F} \cap D(A, r)$ comme la variation précédente. Ainsi obtient-on par définition :

$$\forall j \in \{1, \dots, n\}, \forall u \in [0, l_i], \frac{d}{dt} \Phi_t(c_i(u))|_{t=0} = V_i(u)$$

La première formule variationnelle donne :

$$L'(0) = - \sum_{i=1}^n \langle c'_i(0), V_i(0) \rangle = - \left\langle \sum_{i=1}^n c'_i(0), \Sigma_i(0) \right\rangle = - \langle \Sigma, \Sigma \rangle \neq 0$$

Donc \mathcal{F} n'est pas stationnaire. □

Enfin, la formule de Gauss-Bonnet implique qu'un filet géodésique stationnaire, dont les sommets sont tous de degré 3 et les faces bordées par le même nombre d'arêtes, découpe la sphère en portions de courbures totales égales. C'est le cas pour un filets modelé sur le Θ -graphe : 2 sommets et trois arêtes.

Chapitre 2

Preuves d'existence des filets géodésiques stationnaires : un balayage historique, de Poincaré à Hass & Morgan.

Le graphe élémentaire - consistant en un unique sommet bordant une unique arête fermée - modèle le lacet simple dans M . Il est clair qu'un tel lacet pointé, géodésique et stationnaire (i.e. sans angle au sommet) n'est autre qu'une géodésique fermée et simple. Ainsi, les premières preuves d'existence de filets géodésiques stationnaires coïncident avec les premières constructions de géodésiques périodiques sans points d'auto-intersection. Le 20e siècle est riche de développements autour de ces problèmes d'existence. Nous en parcourons un chemin chronologique.

2.1 La preuve incomplète de Poincaré

Les travaux mené par Henri Poincaré en 1892 dans l'*Analysis Situs*, autour du Groupe Fondamentale des variétés, permettent de construire des géodésiques fermées dans les surfaces riemanniennes dont le π_1 est non trivial. En effet, dans une classe d'homotopie non-triviale, on désigne une suite minimisante pour la longueur, qui converge dans l'espace des courbes \mathcal{C}^1 , vers une géodésique fermée, ayant la même régularité que la métrique. La difficulté demeure, comme dans le cas de la 2-sphère, quand $\pi_1(M) = 0$. Dans un article de 1905 intitulé *Sur les lignes géodésiques des surfaces convexes*, Poincaré propose une preuve de l'existence d'une géodésique fermée et simple dans la 2-sphère en toute courbure. Il considère d'abord un certain espace de lacets divisant la 2-sphère en deux hémisphères de courbures totales égales.

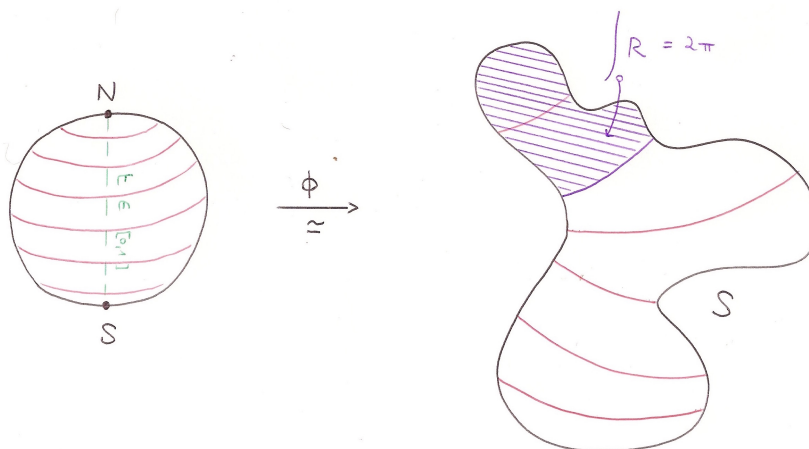


FIGURE 2.1 – Recherche d'un lacet divisant en deux parts égales la courbure totale de S .

Cet espace n'est certes pas vide. Pour en construire un élément, il suffit d'envoyer la 2-sphère S considérée vers une 2-sphère ronde S° via un difféomorphisme Φ , de balayer S avec les images réciproques des parallèles de S° et d'appliquer le Théorème des valeurs intermédiaires. La deuxième idée de Poincaré est de considérer le minimum des longueurs des lacets de cet espace. Il suppose que ce minimum est réalisé par un lacet γ , dont il démontre enfin le caractère géodésique. Son ellipse réside dans l'affirmation qu'un lacet de l'espace doit réaliser le minimum de longueur ; cet argument est donné par Christopher Croke en 1982, dans *Poincaré's problem and the length of the shortest closed geodesic on a convex hypersurface*.

2.2 L'idée de Birkhoff

En 1917, George David Birkhoff expose dans un article intitulé *Dynamical systems with two degrees of freedom*, un protocole de recherche d'une géodésique fermée et simple dans une 2-sphère S de courbure quelconque. Il y décrit comment envoyer une 2-sphère de paramètres dans la 2-sphère S via une application $\mathcal{B} : S^2 \rightarrow S$, continue et de degré 1 (homotope à un homéomorphisme), appelée balayage¹. Il considère ensuite la classe d'homotopie Ω_0 d'un tel balayage \mathcal{B}_0 et définit la largeur de S , relativement à Ω_0 , comme le nombre positif :

$$W =: \min_{\mathcal{B} \in \Omega_0} \max_{t \in [0,1]} E(\mathcal{B}(\cdot, t))$$

Où E est l'énergie de la fibre $\mathcal{B}(\cdot, t)$. Le degré de \mathcal{B} empêche W d'être nul. On construit une suite de balayages dont les pics d'énergie convergent vers W . On ne sait pas encore à ce stade si la suite des fibres maximales converge vers une certaine lacet. Birkhoff introduit alors un nouvel outil : le resserrement des courbes. Cette application est homotope à l'identité, elle agit sur un espace approprié de lacets sur S et satisfait entre autres deux propriétés : elle diminue la longueur et discrimine les courbes presque géodésiques, sur lesquelles elle a un faible effet. Cela permet en particulier de resserrer continûment chacun des balayages de la suite et d'obtenir une suite $\mathcal{B}^j(\cdot, t_j)$ qui converge vers G , l'ensemble des géodésiques fermées et simples de S . G est *a priori* non vide, puisqu'il contient les points, mais notre limite ne saurait être un point : elle réalise W . C'est ainsi la première preuve d'existence d'au moins une géodésique fermée sur toute sphère.

2.3 Un Théorème optimal de Lyusternik et Schnirelmann

En 1929, Lazar Lyusternik and Lev Schnirelmann démontrent dans une note publiée au Comptes Rendus de l'Académie des Sciences de Paris, le théorème suivant :

Théorème 2.1

*Toute variété riemannienne homéomorphe à la sphère possède au moins trois géodésiques fermées non triviales, distinctes et simples.*²

2.4 Un Théorème de Hass et Morgan en courbure positive

À l'instar de Poincaré et Croke, Joël Hass et Frank Morgan ont utilisé des arguments de minimisation sur un espace de filets divisant en parts égales la courbure totale d'une sphère. Ils l'ont fait pour montrer l'existence d'un 3-filet géodésique stationnaire sur toute sphère riemannienne de courbure positive³. Un 3-filet divise la sphère en trois zones. Il peut-être modélisé sur trois topologie de graphes :

- Le Θ -graphe (2 sommets et trois arêtes).
- Le graphe en 8 (1 sommet et 2 arêtes en lacets).
- La paire de lunettes (2 sommets joints par une arête et deux arêtes en lacets supplémentaires).

1. La source peut-être transformée en $[0, 1] \times [0, 1]$, pourvu qu'on impose d'une part : $\forall t \in [0, 1], \mathcal{B}(0, t) = \mathcal{B}(1, t)$, d'autre part : $\forall s \in [0, 1], \mathcal{B}(s, 0) = A$ et $\mathcal{B}(s, 1) = B$. C'est le point de vue que nous adoptons.

2. Ce résultat est optimal dans le sens où certaines variétés sphériques, dont les ellipsoïdes, n'admettent que trois géodésiques fermées et simples.

3. Rappelons le **Théorème de plongement d'Alexandrov** : Une 2-sphère riemannienne a une courbure positive si et seulement si elle est isométrique au bord d'un convexe de R^3 muni de la métrique induite.

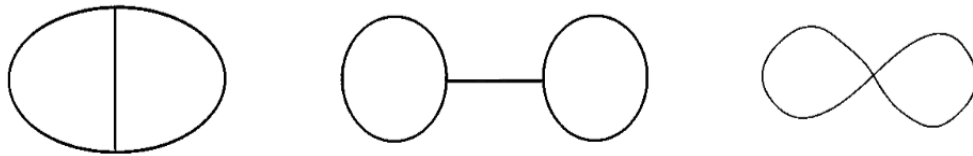
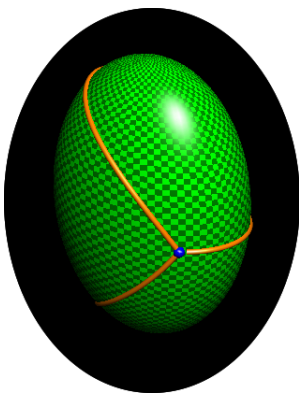


FIGURE 2.2 – Trois modèles de graphes pour les 3-filets

Leur preuve repose fondamentalement sur un théorème que Morgan démontre en 1994 dans un article intitulé *Soap bubbles in \mathbb{R}^2 and surfaces* et qui permet de déformer un graphe donné pour amener chaque zone qu'il délimite vers une zone d'aire donnée.

Théorème 2.2 (Frank Morgan - *Soap bubbles in \mathbb{R}^2 and surfaces* - Théorème 3.2)

Soit M une surface riemannienne lisse, compacte et connexe. Soit G_0 un graphe plongé dans M , avec m faces F_0, \dots, F_m . Soient $A_1, \dots, A_m > 0$, avec $\sum_i A_i < \text{aire } M$. Il y a une déformation continue f_t de M telle que $f_0 = \text{Id}$, f_t injective pour $0 \leq t < 1$, $\mathcal{A}(f_1(F_i)) = A_i$ et $G = f_1(G_0)$ minimise la longueur parmi tous les graphes dont les faces atteignent ces aires cibles. G consiste en la rencontre d'arcs de courbure constante, disjoints ou coïncidant. De plus, aux sommets de G , les vecteurs unitaires tangents sont de sommes nulles.



Morgan ajoute en fin d'article (Paragraphe 3.4 - page 360) que la mesure d'aire peut être remplacée par n'importe quelle intégrale d'une fonction réelle positive définie sur M . Aussi adaptent-ils ce théorème en prenant la fonction courbure sur une sphère convexe et en considérant les trois graphes qui modèlent les 3-filets sur la sphère. Ils parviennent ainsi à construire sur une sphère convexe un 3-filet qui partage la courbure totale suivant une prescription et qui minimisant pour cette propriété. Dans leur article intitulé *Geodesic nets on the 2 sphere (1997)*, Frank et Morgan vérifient enfin que les arêtes de ce filet sont nécessairement des géodésiques. Ainsi démontrent-ils l'existence de 3-filets géodésiques stationnaires sur les sphères de courbure positive.

On commence par définir la classe des filets admissibles sur laquelle on pourra déterminer un filet minimisant. Considérons un filet G parmi les trois modèles de 3-filets et assignons à chaque angle, selon la valence du sommet, une valeur cible :

- Pour un sommet de valence trois : $t_i = 2\pi/3$.
- Pour un sommet de valence quatre : $\pi/3 \leq t_i \leq 2\pi/3$ et $t_i = t_j$ pour les angles opposés par le sommet.

Assignons à chaque face $F_j \subset S^2 - G$ une courbure totale cible K_j , par la formule :

$$K_j = 2\pi - \sum (\pi - t_i),$$

où la somme est prise sur tous les angles cibles contenus dans F_j . Rappelons la formule de Gauss Bonnet appliquée à une face F_j :

$$\int_{F_j} R + \int_{\partial F_j} k + \sum (\pi - s_i) = 2\pi,$$

où R est la courbure de Gauss⁴, k est la courbure géodésique - c'est-à-dire la composante normale du vecteur de courbure $\nabla_{c'} c'$ - et où s_i est l'angle actuel du i -ème sommet de F_j . Si toutes les arêtes de G sont des

4. Dans une variété Riemannienne, la courbure sectionnelle en un point x , dans la direction d'un plan P de l'espace tangent, correspond à la courbure de Gauss en x du morceau de surface engendré par le disque unitaire de P , via l'application exponentielle. En dimension deux, courbure sectionnelle et courbure de Gauss sont une même chose. On rappelle que la courbure de Gauss

géodésiques et si les angles s_i de chaque face correspondent aux angles cibles t_i , alors la courbure totale des F_j est égale à la courbure totale cible :

$$\int_{F_j} R = 2\pi - \sum (\pi - s_i) = 2\pi - \sum (\pi - t_i) = K_j$$

Bilan : pour chaque région du 3-filet modelé sur le θ -graphe, la courbure totale cible est $4\pi/3$. Les monogones du 3-filet modelé sur le graphe à lunettes ont une courbure totale cible égale à $5\pi/3$, tandis que le quadrilatère a une courbure totale cible égale à $2\pi/3$. Enfin, le 3-filet en huit délimite deux monogones dont les courbures totales cibles sont égales et comprises entre $4\pi/3$ et $5\pi/3$, ainsi qu'un 2-gones dont la courbure totale cible est comprise entre $2\pi/3$ et $4\pi/3$. On appelle *filets admissibles* sur la 2-sphère les 3-filets qui atteignent effectivement ces trois répartitions de la courbure totales, avec possiblement des écrasements d'arêtes.

On note G_0 le 3-filet minimisant dans cette classe de filets admissibles, comme annoncé. Si l'une des régions F_j de G_0 est bordée par les angles s_i , alors la formule de Gauss-Bonnet implique que la courbure géodésique à la frontière de F_j satisfait :

$$(\star)^5 \int_{\partial F_j} k = \sum (s_i - t_i)$$

Considérons un chemin fermé γ traversant G_0 transversalement. La première formule de variation de la longueur d'une courbe \mathcal{C}^2 implique que la dérivée de la longueur de la courbe, lorsqu'elle est perturbée dans la direction d'un vecteur $W \in T_x M$, où $x = c(u)$ est assez loin des bords, est donnée par le produit scalaire de W et du vecteur de courbure de la courbe :

$$L'(0) = - \int_{u-\epsilon}^{u+\epsilon} \langle \nabla_{c'} c', W \rangle$$

En réalité, on considère un champ de vecteur le long de la courbe qui vaut W dans un petit voisinage de x et s'annule brusquement au-delà, via une approximation de l'identité. On néglige le régime transitoire de la variation. Par ailleurs, la dérivée de la courbure totale d'une face jouxtant une courbe soumise à une telle perturbation est donnée par le produit de R et de la partie normale de W :

$$\frac{d}{dt}|_{t=0} \int_{D(t)} R dV = \int_{\partial D} R \langle W, n_s \rangle du,$$

où n_s est la normale sortante. Montrons d'abord que le long d'une arête de $c \in G_0$ le produit kR est constant. On sait déjà par le Théorème 2.2 que k est constant le long des arêtes de G_0 . Le scalaire k est signé et dépend de la normale à c choisie. On prend soin de considérer le long de c une normale qui ne change pas d'orientation. Soient x et y deux points distincts de c . On pousse c en x dans le sens de la normale avec un coefficient α_1 et y dans le sens opposé à la normale avec un coefficient α_2 . On obtient d'une part $L'(0) = 2\epsilon k(\alpha_2 - \alpha_1)$ et d'autre part :

$$\frac{d}{dt}|_{t=0} \int_{D(t)} R dV = 2\epsilon k(R_1 \alpha_1 - R_2 \alpha_2),$$

où $D(t)$ est le domaine délimité par c pour lequel la normale à c est la normale sortante. On veut absolument préserver les courbures totales des faces adjacentes en c . C'est pourquoi on impose $R_1 \alpha_1 = R_2 \alpha_2$, c'est-à-dire $\alpha_1 = R_2 \alpha_2 / R_1$. Cette condition implique :

$$L'(0) = 2\epsilon k \left(\frac{\alpha_2(R_2 - R_1)}{R_1} \right)$$

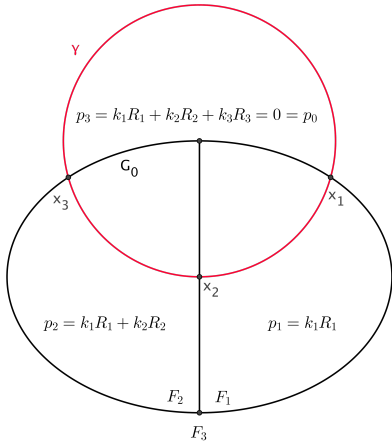
est en particulier le produit des courbures principales, c'est-à-dire les courbures planes - minimale et maximale - des sections orthogonales au plan tangent. On rappelle enfin, dans un cadre abstrait, la définition de la courbure sectionnelle, où R est le tenseur de Riemann associé à une variété (M, \langle, \rangle) , x est un point de M et u, v sont des vecteurs de $T_x M$:

$$K_x(u, v) = \frac{\langle R(u, v)v, u \rangle}{\langle u, u \rangle \langle v, v \rangle - \langle u, v \rangle^2}$$

5. Détail : $\int_{\delta F_j} k = 2\pi - \sum (\pi - s_i) - \int_{F_j} R = 2\pi - \sum (\pi - s_i) - K_j = 2\pi - \sum (\pi - s_i) - 2\pi + \sum (\pi - t_i) = \sum (s_i - t_i)$
Seuls les sommets de degré 4 donnent une contribution non nulle, car pour un sommet de degré 3, on a $s_i = t_i = \frac{2\pi}{3}$.

Si $R_1 \neq R_2$, on pourrait avoir $L'(0) < 0$ en choisissant convenablement le signe de α_2 . Cela est impossible. En effet, on ne peut obtenir par variation de G_0 aucun graphe de plus petite longueur qui conserve les courbures totales de ses faces. Conclusion : $R_1 = R_2$ et le produit kR est constant le long de c .

Un raisonnement similaire permet de montrer qu'on a nécessairement $\sum_i k_i R_i = 0$, où la somme est prise sur l'ensemble des points d'intersection de γ avec G_0 et où les normales permettant de définir k sont prises dans le sens de la marche⁶. En effet, dans le cas contraire, on devrait être en mesure de construire une poussée infinitésimale de G_0 dans la direction de γ qui occasionnerait un nouveau graphique de plus petite longueur et divisant S^2 en régions de mêmes courbures totales, ce qui est impossible. On peut donc définir une fonction de pression p à valeur réelle sur l'ensemble des régions avec la propriété que la différence des pressions de deux régions séparées par un arc g_0 de G_0 ayant une courbure k est donnée par $\Delta p = kR$. La fonction p est définie comme étant nulle sur une région arbitraire et donnée par la formule $\Delta p = kR$, successivement sur des régions adjacentes séparées par un arc de multiplicité un. Pour deux régions adjacentes séparées par un arc de multiplicité m , nous utilisons la formule $\Delta p = mkR$. Notez que si deux régions voisines ont la même pression, alors l'arête du graphique qui les sépare vérifie $k = 0$ (car $R > 0$, par hypothèse) et sont des géodésiques, du moins là où elles ne se heurtent pas à d'autres arêtes.



Nous considérons maintenant chacun des trois types combinatoires de graphes admissibles. Supposons d'abord que G_0 soit un Θ -graphe qui ne s'est pas effondré en huit. Si toutes les pressions sont égales, alors G_0 donne un filet géodésique. C'est ce que nous allons montrer. Soit F_j une région de pression maximale. On a :

$$p_j - p_{j-1} = k_j R_j \text{ et } p_j \geq p_{j-1}$$

Donc $k_j R_j \geq 0$, i.e. $k_j \geq 0$. De même on montre que $k_{j+1} \leq 0$. En signant chaque courbure géodésique des arêtes bordant F_j vers l'intérieur de la face, on arrive à $k \geq 0$ partout au bord de F_j . La formule (\star) dans le cas du theta donne :

$$\int_{\delta F_j} k = 0$$

Il vient $k \equiv 0$. Cela implique que la pression dans les deux autres régions est égale à celle de F_j de sorte que les trois régions ont une pression égale. Étant donné que les arcs de délimitation ne se heurtent pas près d'un sommet (condition au sommets, voir Théorème 2.2), ils commencent comme des géodésiques et ne se heurtent jamais. Par conséquent, tous les arcs de frontière sont des géodésiques. Finalement, si G_0 ne dégénère pas en huit, il doit former un réseau géodésique modelé sur le Θ -graphe.

Un argument similaire s'applique au graphes à lunettes. Soit encore F_j une région de pression maximale. On a $k \geq 0$ à la frontière et par (2) : $k \equiv 0$. Si F_j est le 4-gone, cela implique que toutes les pressions sont égales puisque les deux autres régions sont adjacentes à F_j . Si F_j est un monogone, alors le 4-gone a la même pression maximale et encore une fois toutes les pressions sont égales. Ainsi, si G_0 ne dégénère pas en huit, il forme alors un réseau géodésique. Le dernier cas est plus délicat à traiter. Nous renvoyons à la preuve du **Claim 1**, dans Hass et Morgan (1997), page 3847.

Dans cette approche, l'hypothèse de convexité est décisive. Nous espérons, à l'issue de ce mémoire, pouvoir démontrer l'existence de 3-filets géodésiques stationnaires sur les 2-sphères riemannienne en toute courbure. Pour cela, nous allons privilégier les stratégies de balayages et de rétrécissement des courbes propres à Birkhoff. Nous aurons recours à la formulation moderne donnée par Tobias Colding et William Minicozzi.

Remarque 2.1. Sur un ellipsoïde d'équation $ax^2 + by^2 + cz^2 = 1$, les symétries facilitent la recherche d'un Θ -filet. Considérons l'une des trois géodésiques fermées et simples. Le long de l'une des deux autres - dans un plan perpendiculaire à la première - on fait glisser un point P d'où partent deux arcs géodésiques dans les deux directions parallèles au plan de la première géodésique. Ces deux arcs rencontrent la première géodésique en deux points. La continuité du flot géodésique permet de désigner P de telle sorte que la rencontre se fasse avec un angle de $2\pi/3$. Un raisonnement symétrique dans l'hémisphère opposé achève la construction.

6. Au point $x \in G_0 \cap \gamma$, on s'assure que $\langle c', n \rangle$ soit positif.

10

Chapitre 3

Colding & Minicozzi : Une relecture de Birkhoff.

Dans cette partie, nous nous reportons à l'article *Min-Max for sweepouts by curves* de Tobias Colding et William Minicozzi écrit en 2007. Nous présentons les objets et techniques utilisées, puis détaillons les preuves de deux résultats clés. L'enjeu est de définir le bon espace de courbes sur lequel doit s'appliquer le *pull tight* introduit par Birkhoff. On introduit dans ce but l'espace Λ des applications géodésiques par morceaux de S^1 dans M avec exactement L angles (éventuellement des angles plats) telles que la longueur de chaque géodésique ne dépasse pas 2π , paramétrée par un multiple constant de longueur d'arc. De telles courbes, parcourues à vitesse 1, sont L -Lipschitz (conséquence immédiate de l'inégalité triangulaire). L'espace Λ est inclus dans l'espace de Sobolev $W^{1,2}$ des applications de S^1 dans M , qui lui confère une topologie induite. La norme dans $W^{1,2}$, donnée par $\int_{S^1} |f|^2 + |f'|^2$, dépend du plongement de M choisi. Des conditions sur ce plongement sont données dans l'article de Colding & Minicozzi¹. Attention : cette norme confère à Λ , non pas une structure d'espace normé - car la norme de Sobolev est définie dans un espace vectoriel de chemins vivant dans l'espace du plongement de M - mais une structure induite d'espace métrique. Dans tout ce chapitre, nous reprenons les notations introduites dans la section 2.2.

3.1 Raccourcissement de courbes

Le raccourcissement de courbes - ou pull tight - est une application $\Psi : \Lambda \longrightarrow \Lambda$ telle que :

- (1) $\Psi(\gamma)$ est homotope à γ et $L(\Psi(\gamma)) \leq L(\gamma)$.
- (2) $\Psi(\gamma)$ dépend continûment de γ .
- (3) Il existe une fonction continue $\phi : [0, \infty) \longrightarrow [0, \infty)$ avec $\phi(0) = 0$ et telle que :

$$\text{dist}^2(\gamma, \Psi(\gamma)) \leq \phi \left(\frac{L^2(\gamma) - L^2(\Psi(\gamma))}{L^2(\Psi(\gamma))} \right).$$

- (4) Étant donné $\epsilon > 0$, il existe $\delta > 0$ tel que, si $\gamma \in \Lambda$ avec $L(\gamma) - L(\Psi(\gamma)) < \delta$, alors $\text{dist}(\gamma, G) < \epsilon$.

Il existe une telle application : on en trouvera une construction explicite dans [Co Mi]. On notera que cette construction permet de définir Ψ sur $W^{1,2}$ tout entier. En revanche, les propriétés qui viennent d'être énoncées nécessitent la compacité de Λ . La propriété (4) met en évidence l'atout discriminatoire de Ψ , au sens où son action a d'autant moins d'effet sur une courbe $\gamma \in \Lambda$ que celle-ci se trouve proche de G au sens de la norme $W^{1,2}$ sur Λ . Elle découle à la fois de la **compacité de Λ** , de la **continuité de Ψ** et d'une

1. Notons qu'il est possible de définir une norme intrinsèque sur Λ . Le recours naturel au plongement permet de simplifier certaines preuves. Composer un plongement avec une homothétie n'affecte pas les propriétés variationnelles de la variété. C'est pourquoi Colding et Minicozzi proposent une dilatation de M qui rende sa courbure suffisamment petite et son rayon d'injectivité suffisamment grand. Cela se traduit par trois exigences :

- (M1) $\sup_M |A| \leq 1/16$ où A est la seconde forme fondamentale de M .
- (M2) $R_{inj}(M) \geq 8\pi$ et $R \leq 1/64$, de sorte que toute boule géodésique de rayon $\leq 4\pi$ dans M est strictement convexe.
- (M3) $\forall x, y \in M, |x - y| \leq 1 \implies \text{dist}_M(x, y) \leq 2|x - y|$.

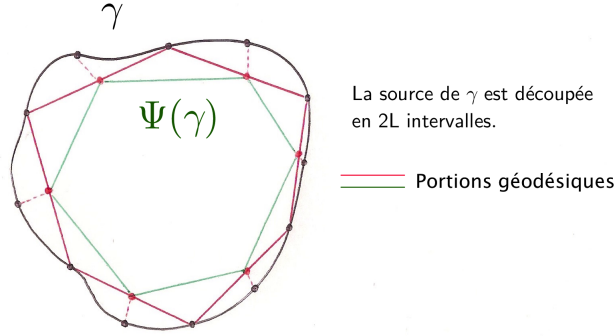


FIGURE 3.1 – Raccourcissement d’une courbe dans Λ .

propriété propre à la construction décrite dans [Co Mi] : les points fixes de Ψ sont exactement les éléments de G . Montrons cette propriété (4), via un raisonnement par l’absurde. S’il existe $\epsilon > 0$ et une suite γ_j de Λ telle que pour tout j , $L(\gamma_j) - L(\Psi(\gamma_j)) < 1/j$, avec pourtant $\text{dist}(\gamma_j, G) \geq \epsilon$. Puisque Λ est compacte, il existe une sous suite γ_{j_n} qui converge dans Λ vers γ . Par hypothèse, la sous suite $\Psi(\gamma_{j_n})$ converge aussi vers γ . Or, par continuité de Ψ , cette limite est aussi $\Psi(\gamma)$, d’où $\Psi(\gamma) = \gamma$. En tant que point fixe de Λ , on a nécessairement $\gamma \in G$. Cela qui contredit l’hypothèse selon laquelle les γ_j restent à distance non nulle de G . Le Lemme suivant permet de démontrer le Théorème de la section 3.

Lemme 3.1 (Colding & Minicozzi - *Min-Max for sweepouts by curves* - Lemme 1.4)

Soient $W \geq 0$ et $\epsilon > 0$. Il existe $\delta > 0$, qui tend vers zéro avec ϵ , tel que si $\gamma \in \Lambda$ et

$$(\star) \quad 2\pi(W - \delta) < L^2(\Psi(\gamma)) \leq L^2(\gamma) < 2\pi(W + \delta),$$

alors $\text{dist}(\Psi(\gamma), G) < \epsilon$.

Démonstration. Soient $W \geq 0$ et $\epsilon > 0$. Soit $\eta_1 > 0$, donné par (3), tel que $\forall \delta > 0$, $\delta \leq \eta_1 \implies \phi(\delta) < \epsilon^2/4$. On rappelle que ϕ est continue positive, avec $\phi(0) = 0$. Soit $\eta_2 > 0$, donné par (4), tel que $L(\gamma) - L(\Psi(\gamma)) < \eta_2 \implies \text{dist}(\gamma, G) < \epsilon/2$, où $\gamma \in \Lambda$.

On pose $\hat{\delta} = \min(\eta_1, \eta_2)$ et $\delta = \hat{\delta}/4\pi$.

Supposons que l’on ait $2\pi(W - \delta) < L^2(\Psi(\gamma)) \leq L^2(\gamma) < 2\pi(W + \delta)$.

Il vient d’une part :

$$\begin{aligned} L^2(\gamma) - L^2(\Psi(\gamma)) &< 4\pi\delta \\ L(\gamma) - L(\Psi(\gamma)) &< \frac{4\pi\delta}{L(\gamma) + L(\Psi(\gamma))} < \hat{\delta} \leq \eta_2 \end{aligned}$$

Donc $\text{dist}(\gamma, G) < \epsilon/2$.

Il vient d’autre part :

$$\frac{L^2(\gamma) - L^2(\Psi(\gamma))}{L^2(\Psi(\gamma))} < \hat{\delta} \leq \eta_1$$

Donc $\text{dist}^2(\Psi(\gamma), \gamma) = \phi\left(\frac{L^2(\gamma) - L^2(\Psi(\gamma))}{L^2(\Psi(\gamma))}\right) < \epsilon^2/4$, i.e. $\text{dist}(\Psi(\gamma), \gamma) < \epsilon/2$.

L’inégalité triangulaire donne, pour tout $g \in G$:

$$\text{dist}(\Psi(\gamma), g) \leq \text{dist}(\Psi(\gamma), \gamma) + \text{dist}(\gamma, g)$$

En minimisant sur G à droite et à gauche, on arrive à la conclusion :

$$\text{dist}(\Psi(\gamma), G) \leq \text{dist}(\Psi(\gamma), \gamma) + \text{dist}(\gamma, G) < \epsilon$$

□

2. Si aucune hypothèse n’accompagne *a priori* le réel W , il se trouve que la prémisses (\star) pourra être satisfaite si l’on prend pour W la largeur de M et pour γ , une courbe suffisamment proche de son image par Ψ , à la mesure de δ .

3.2 Une bonne suite de balayages.

Soit Ω l'ensemble des applications continues $\mathcal{B} : S^1 \times [-1, 1] \rightarrow M$ telles que pour tout t , l'application $\mathcal{B}(\cdot, t)$ est dans $W^{1,2}$, l'application $t \mapsto \mathcal{B}(\cdot, t)$ est continue de $[-1, 1]$ vers $W^{1,2}$ et enfin, \mathcal{B} envoie $S^1 \times \{-1\}$ sur des points. Étant donnée une application $\hat{\mathcal{B}} \in \Omega$, la classe d'homotopie $\Omega_{\hat{\mathcal{B}}}$ est définie comme l'ensemble des applications $\mathcal{B} \in \Omega$ qui sont homotopes à $\hat{\mathcal{B}}$ à travers des applications de Ω . Fixons désormais une classe d'homotopie non triviale de balayages et notons-la simplement Ω pour plus de lisibilité. Choisissons une suite d'applications $\mathcal{B}^j \in \Omega$ (cf. 2.2) avec :

$$\max_{t \in [-1, 1]} E(\mathcal{B}^j(\cdot, t)) < W + \frac{1}{j} \quad (\star\star)$$

On trouvera dans [Co Mi] une construction d'une nouvelle suite de balayages $\sigma^j \in \Omega$, homotopes à \mathcal{B}^j et dont les fibres, en plus de vérifier l'inégalité $(\star\star)$, appartiennent à Λ . Sur chaque fibre de ces nouveaux balayages, s'applique le rétrécissement des courbes. On obtient ainsi une troisième suite de balayages $\gamma^j \in \Omega$, homotopes à σ^j et définis par : $\gamma^j(\cdot, t) = \Psi(\sigma^j(\cdot, t))$.

3.3 Presque maximal implique presque critique

Théorème 3.1 (Colding & Minicozzi - *Min-Max for sweepouts by curves* - Théorème 1.9)
Soient $W \geq 0$ et $\epsilon > 0$; il existe $\delta > 0$ tel que, si $j > 1/\delta$ et pour un certain t_0 :

$$2\pi E(\gamma^j(\cdot, t_0)) = L^2(\gamma^j(\cdot, t_0)) > 2\pi(W - \delta),$$

alors pour ce j nous avons $\text{dist}(\gamma^j(\cdot, t_0), G) < \epsilon$.

Démonstration. Précisons que sur Λ , l'énergie n'est autre que la longueur au carré, divisée par 2π . Cela dit, soit $W \geq 0$ et $\epsilon > 0$. Soit encore $\delta > 0$, donné par le Lemme 3.1.1, tel que :

$$(\sigma \in \Lambda \text{ et } 2\pi(W - \delta) < L^2(\Psi(\sigma)) \leq L^2(\sigma) < 2\pi(W + \delta)) \implies \text{dist}(\Psi(\sigma), G) < \epsilon$$

Supposons que $j > 1/\delta$ et qu'il existe t_0 tel que :

$$L^2(\gamma^j(\cdot, t_0)) > 2\pi(W - \delta)$$

Comme $\gamma^j(\cdot, t_0) = \Psi(\sigma^j(\cdot, t_0))$ avec $L(\Psi(\sigma^j(\cdot, t_0))) < L(\sigma^j(\cdot, t_0))$, il vient :

$$2\pi(W - \delta) < L^2(\Psi(\sigma^j(\cdot, t_0))) < L^2(\sigma^j(\cdot, t_0))$$

Or, par $(\star\star)$, $L^2(\sigma^j(\cdot, t_0)) = 2\pi E(\sigma^j(\cdot, t_0)) < 2\pi(W + 1/j) < 2\pi(W + \delta)$.

En vertu des qualités conférées à δ , il vient : $\text{dist}(\gamma^j(\cdot, t_0), G) < \epsilon$.

□

Appliquons enfin le **théorème 3.3.1** à la largeur W de M (relativement à un balayage canonique non trivial) et à $\epsilon > 0$, aussi petit que l'on veut. Prenons $j > 1/\delta$ où δ est donné par le Théorème. Ainsi a-t-on :

$$W \leq \max_{t \in [0, 1]} E(\gamma^j(\cdot, t))$$

Soit t_0 le paramètre qui réalise le maximum. On a $E(\gamma^j(\cdot, t_0)) > W - \delta$. Cela implique que $\text{dist}(\gamma^j(\cdot, t_0), G) < \epsilon$. Nous pouvons donc construire une sous-suite de fibres qui converge dans Λ compact, vers une limite γ , tout en s'approchant toujours plus de G . Comme G est fermé, si γ n'appartenait pas à G , les fibres de la sous-suite finiraient par s'accumuler dans un voisinage ouvert de γ distinct de G , c'est-à-dire à une distance minimum de G : Impossible ! Donc $\gamma \in G$.

Chapitre 4

Traque d'un 3-filet géodésique stationnaire dans une 2-sphères à pantalon, par balayage tripodal.

L'article de Joël Hass *What is an almost normal surface* montre l'existence d'un découpage de la 2-sphère le long d'un ensemble saturé maximal \mathcal{G} de géodésiques fermées et simples - saturé au sens où toute autre géodésique fermée et simple doit couper un élément de l'ensemble ; maximal au sens du cardinal ¹ - découpage qui, à défaut d'être unique, dresse une liste exhaustive de constituants topologiques permettant de décrire une 2-sphère : le pantalon, la nasse, le bonnet. Le premier est une sphère privée de trois disques disjoints délimités par des éléments stables de \mathcal{G} . La nasse est un anneau bordé par deux éléments de \mathcal{G} , l'un stable, l'autre instable. Le bonnet est un disque bordé par une géodésique instable de \mathcal{G} . La présence d'un pantalon étant salutaire pour la recherche que nous faisons, nous classons les 2-sphères de la façon suivante :

- Les sphères portant au moins un pantalon.
- Les double-bonnets, dont les sphères convexes.
- Les concaténations de nasses par paires renversées, se refermant sur des bonnets.

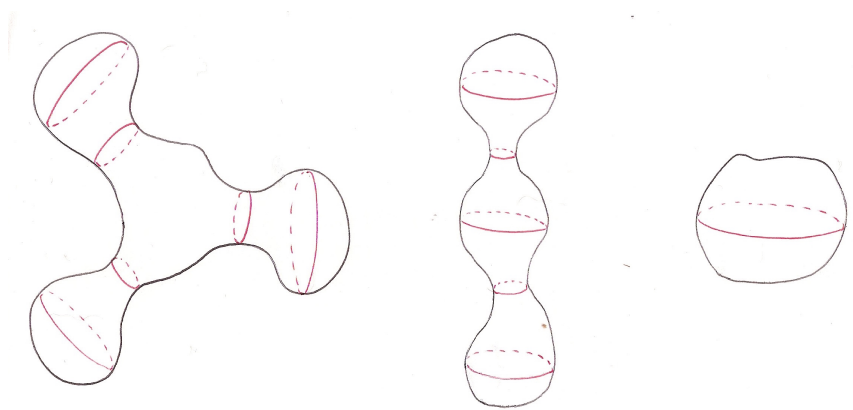


FIGURE 4.1 – Familles saturées maximales de géodésiques fermées sur des 2-sphères

Notre objectif dans cette section, est de prouver l'existence d'un 3-filet géodésique stationnaire au niveau des pantalons. Ne suivent pour le moment que des esquisses de preuves. Nous allons élaborer une procédure de min-max adaptée à la topologie du pantalon et à celle de la cible : le 3-filet. Nous nous inspirons du formalisme de Colding et Minicozzi. Il faut pourtant définir une nouvelle procédure de rétrécissement, plus délicate à manier. Le balayage standard que nous proposons consiste à accrocher trois anneaux aux bords du pantalon et de les faire se nouer au centre, le long d'un 3-filet. Détaillons maintenant cette construction.

1. Cardinal éventuellement infini si la sphère contient un tronçon annulaire plat ou torique.

4.1 Balayage tripodal d'un pantalon.

Soient $P \subset M$ un pantalon de bords $\gamma_1, \gamma_2, \gamma_3$ (géodésiques) et Ω l'ensemble des applications continues $\mathcal{B} : S^1 \times \{1, 2, 3\} \times [0, 1] \rightarrow P$, injectives sur $S^1 \times \{1, 2, 3\} \times [0, 1[$ où, $\forall i, \forall t, \mathcal{B}(\cdot, i, t)$ est dans $W^{1,2}$. On demande, pour $i = 1, 2, 3$, le raccord aux bords : $S^1 \times \{i\} \times 0 \mapsto \text{Im}(\gamma_i)$. À i et t fixés, on rappelle que es applications $s \mapsto \mathcal{B}(s, i, t)$ sont munies de la norme de Sobolev donnée par :

$$\|\mathcal{B}(\cdot, i, t)\| = \int_{S^1} (|\mathcal{B}|^2 + |\mathcal{B}'|^2)$$

De plus, on exige en $t = 1$ une rencontre des trois manches en un 3-filet \mathcal{F} :

$A =: \mathcal{B}(0, 1, 1) = \mathcal{B}(0, 2, 1) = \mathcal{B}(0, 3, 1)$ et $B =: \mathcal{B}(0.5, 1, 1) = \mathcal{B}(0.5, 2, 1) = \mathcal{B}(0.5, 3, 1)$.
 $\forall s \in [0, 0.5], \mathcal{B}(s, 1, 1) = \mathcal{B}(s, 3, 1), \mathcal{B}(s, 2, 1) = \mathcal{B}(s + 0.5, 3, 1)$ et $\forall s \in [0.5, 1], \mathcal{B}(s, 1, 1) = \mathcal{B}(s, 2, 1)$.

La réunion des courbes $t \mapsto \mathcal{B}(0, i, t), i = 1, 2, 3$ d'une part, des courbes $t \mapsto \mathcal{B}(0.5, i, t), i = 1, 2, 3$ d'autre part, forme deux tripodes T_A et T_B qui sont la colonne vertébrale de notre balayage.

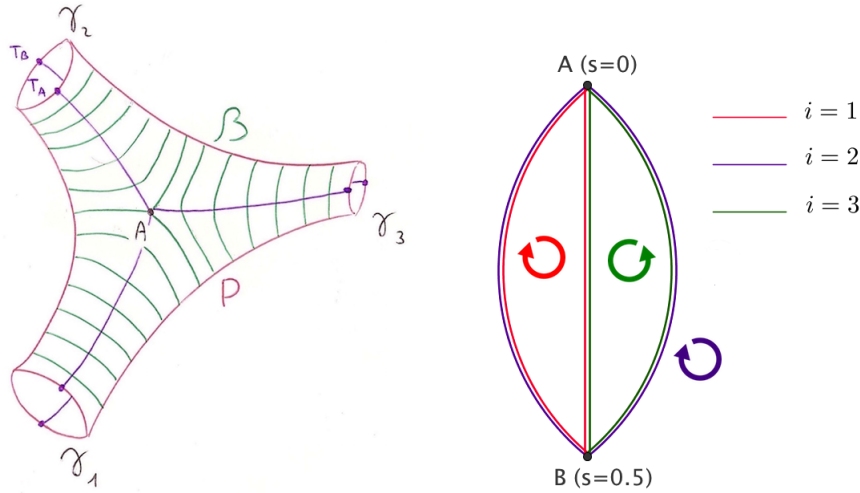


FIGURE 4.2 – Balayage tripodal d'un pantalon

4.2 Largeur tripodale d'un pantalon.

Étant donné une application $\hat{\mathcal{B}} \in \Omega$, la classe d'homotopie $\Omega(\hat{\mathcal{B}})$ est définie comme l'ensemble des applications $\mathcal{B} \in \Omega$ homotopes à $\hat{\mathcal{B}}$ à travers des applications de Ω . À toute classe d'homotopie $\Omega(\hat{\mathcal{B}})$, on associe la *largeur* $W = W(\hat{\mathcal{B}})$, définie en prenant l'infimum du maximum des énergies des tranches $\mathcal{B}(\cdot, \{1, 2, 3\}, t)$:

$$W = \inf_{\mathcal{B} \in \Omega(\hat{\mathcal{B}})} \max_{t \in [-1, 1]} E(\mathcal{B}(\cdot, \{1, 2, 3\}, t))$$

où l'énergie est donnée par $E(\mathcal{B}(\cdot, \{1, 2, 3\}, t)) = \sum_{i=1}^3 \int_{S^1} |\partial_s(s, i, t)|^2 ds$.

La topologie de P et les hypothèses sur Ω impliquent² qu'il existe un réel strictement positif qui minore l'énergie de toute fibre $\mathcal{B}(\cdot, i, t)$. C'est pourquoi la largeur W est nécessairement strictement positive.

4.3 Procédure de resserrement du balayage

Soit $\mathcal{B} \in \Omega$. À l'instar de Birkhoff, nous allons définir une procédure de rétrécissement de \mathcal{B} . Le paragraphe suivant résume une fausse piste que nous avons empruntée et dont on a tiré des leçons.

². Ce point est à détailler.

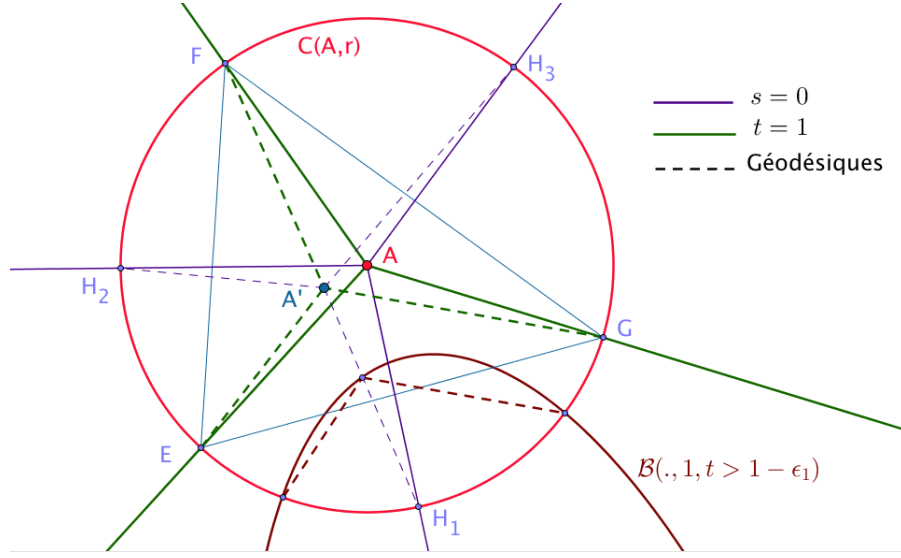


FIGURE 4.3 – Point de Fermat

Remarque 4.1 (Fausse piste). *Pour augmenter nos chances qu’une suite décroissante de balayages converge vers un filet stationnaire, nous avons eu l’idée de redresser chaque balayage en ses deux sommets de sorte que le 3-filet central satisfasse la condition au angle du Théorème 1.1. Malheureusement, l’angle formé par deux arêtes n’est pas une fonction continue sur l’espace des paires jointes d’arêtes $W^{1,2}$. En témoigne la figure 4.4 ci-dessous. Voici le détail de cette construction. Il s’agit de redresser le 3-filet \mathcal{F} ($t = 1$) au niveau de ses deux sommets, ainsi que les fibres de \mathcal{B} pour t proche de 1. Nous détaillons le protocole pour le point A. Soit $r > 0$ suffisamment petit mais maximal, pour que $D(A, r)$ et $D(B, r)$ satisfassent trois conditions : Il doivent être convexes, contenus dans les disques d’injectivité de chacun de leurs points et leurs frontières ne doivent ni se toucher, ni couper les fibres de \mathcal{B} en plus de deux points. On note E, F, G les points d’intersection de $C(A, r)$ et \mathcal{F} . On note également H_1, H_2, H_3 les points d’intersections de $C(A, r)$ et du tripode T_A ($s = 0$). Il existe trois réels $\epsilon_1, \epsilon_2, \epsilon_3$ dans $[0, 1]$ tels que $H_i = \mathcal{B}(0, i, 1 - \epsilon_i)$ pour $i = 1, 2, 3$. Soit A' le point de Fermat³ du triangle EFG . On remplace le 3-filet, entre A et E, F, G, par des arcs géodésiques issus de A' . On remplace le tripode T_A , entre A et H_1, H_2, H_3 par des arcs géodésiques issus de A' . Enfin, pour $i = 1, 2, 3$ et $t \in]1 - \epsilon_i, 1[$, la courbe $\mathcal{B}(., i, t)$ rencontre le cercle $C(A, r)$ en deux points et le nouvel arc géodésique $A'H_i$ en un point situés entre les deux premiers. On interpole alors ces trois points par deux géodésiques. On obtient un nouveau balayage tripodale $\hat{\mathcal{B}} \in \Omega$. On pourra vérifier sa continuité en t .*



FIGURE 4.4 – Non continuité de l’angle

D’abord, transformons chaque demi-fibre⁴ de $\hat{\mathcal{B}}$ en une application géodésique par morceaux. Cette opération, continue en t , consiste à scinder les intervalles $[0, 0.5]$ et $[0.5, 1]$ en intervalles conjoints sur lesquels on remplace la fibre par un tronçon de géodésique⁵. Le nombre d’intervalles doit être pair et suffisamment grand

3. Le point de Fermat d’un triangle ABC dans le plan - dont les angles n’excèdent pas 120° - est l’unique point F tel que $(FA, FB) = (FB, FC) = 2\pi/3$. De plus, la somme des distances FA, FB et FC est minimisante. L’énoncé est transposable sur une surface riemannienne, à condition que le triangle se trouve à l’intérieur du disque d’injectivité de chacun de ses sommets : il existe un point F et trois géodésiques $[FA], [FB]$ et $[FC]$ qui se rencontrent en F suivant un angle de $2\pi/3$ et telles que la somme de leurs longueurs est minimisante. En revanche, si l’un des sommets forme un angle de plus de 120° , alors le point de Fermat n’est autre que ce sommet.

4. Pour $0 < s < 1/2$ et $1/2 < s < 1$.

5. On vérifiera qu’au bord des manches du pantalon, cette procédure est bien définie. Cela tient au fait que le voisinage des bords est géodésiquement convexe.

pour que l'interpolation soit bien définie. Elle permet de regarder les nouvelles demi-fibres dans l'espace Γ défini ci-après, sur lequel s'applique la procédure de rétrécissement introduite dans la définition 4.1.

Soit $\Gamma_{[a,b]}^L$, ou simplement Γ s'il n'y a pas d'ambiguïté, l'espace des chemins $c : [a, b] \rightarrow P$ géodésiques par morceaux avec L angles, éventuellement plats, correspondant à une partition de la source, propre à chaque chemin :

$$[a, b] = \bigcup_{i=0}^L [x_i, x_{i+1}], \text{ où } a = x_0 < \dots < x_{L+1} = b$$

Sur chaque morceaux, on demande que c soit l'unique plus courte géodésique de x_i à x_{i+1} . On remarque que L doit être impair et suffisamment grand pour que $((L+1)R_{\text{inj}}(P))^2 \gg 2\pi W$. L'espace Γ est métrique pour la norme $W^{1,2}$. On vérifiera qu'il est compact. Nous définissons maintenant une application de resserrement des fibres du balayage, qui se veut un analogue dans notre situation à celle de Colding-Minicozzi que nous avons utilisée à la section 3.1.

Définition 4.1

Soit (x_0, \dots, x_{L+1}) un découpage de $[a, b]$ en $L+1$ intervalles et soit $\gamma \in \Gamma_{[a,b]}^L$. On précise que les angles de γ ne sont pas nécessairement situés en (x_0, \dots, x_{L+1}) .

- On remplace γ par une géodésique sur chaque intervalle $[x_{2k}, x_{2k+2}]_{k=0, \dots, (L-1)/2}$ pour obtenir $\tilde{\gamma}$.
- On remplace $\tilde{\gamma}$ par une géodésique sur chaque intervalle $[x_{2k+1}, x_{2k+3}]_{k=0, \dots, (L-3)/2}$, pour obtenir $\hat{\gamma}$.
- On re-paramétrise $\hat{\gamma}$ pour obtenir une courbe à vitesse constance, en tout point de dérivabilité.

On note $\Pi(\gamma)$ la nouvelle application $[a, b] \rightarrow P$.

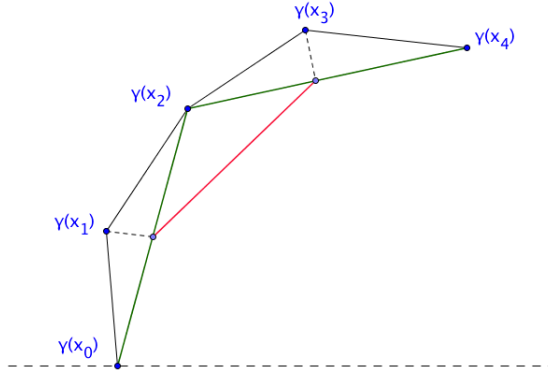


FIGURE 4.5 – Application Π

Proposition 4.1 (Non démontrée - Jugée raisonnable, à l'aune de la section 3.1)

L'opérateur $\Pi : \Gamma \rightarrow \Gamma$ de la précédente définition vérifie quatre propriétés, semblables à celles de Ψ :

- (1) $\Pi(\gamma)$ est homotope à γ et $L(\Pi(\gamma)) \leq L(\gamma)$.
- (2) $\Pi(\gamma)$ dépend continûment de γ .
- (3) Il existe une fonction continue $\phi : [0, \infty) \rightarrow [0, \infty)$ avec $\phi(0) = 0$ et telle que :

$$\text{dist}^2(\gamma, \Pi(\gamma)) \leq \phi \left(\frac{L^2(\gamma) - L^2(\Pi(\gamma))}{L^2(\Pi(\gamma))} \right).$$

- (4) Étant donné $\epsilon > 0$, il existe $\delta > 0$ tel que, si $\gamma \in \Gamma$ avec $L(\gamma) - L(\Pi(\gamma)) < \delta$, alors $\text{dist}(\gamma, \hat{G}) < \epsilon$, où \hat{G} est l'espace fermé des géodésiques à deux bords.

4.4 Vers une preuve d'existence.

Soit \mathcal{B}^j une suite de balayages qui satisfait :

$$\max_{t \in [0,1]} E(\mathcal{B}^j(., \{1, 2, 3\}, t)) < W + \frac{1}{j}$$

Soit $\Gamma^{3,2}$ l'ensemble des triplets de paires de chemins dans Γ joints en leurs extrémités. Plus précisément, c'est l'ensemble des applications : $\mathcal{T} : [0, 1] \times \{1, 2, 3, 4, 5, 6\} \rightarrow P$ où $\forall i = 1, \dots, 6 : \mathcal{T}(., i) \in \Gamma$ et $\mathcal{T}(0, 1) = \mathcal{T}(0, 2)$; $\mathcal{T}(0, 3) = \mathcal{T}(0, 4)$; $\mathcal{T}(0, 5) = \mathcal{T}(0, 6)$ (idem pour $\mathcal{T}(1, -)$). On munit $\Gamma^{3,2}$ d'une structure d'espace normé où la norme d'un élément \mathcal{T} est la somme des normes de ses 6 arêtes dans Γ (idem pour la longueur). Les arêtes jointes le demeurent à la limite, c'est pourquoi $\Gamma^{3,2}$ est fermé. Ci-après deux résultats dans $\Gamma^{3,2}$. Ce sont des analogues du Lemme et du Théorème de la partie 3. On note $\Pi(\mathcal{T})$, pour $\mathcal{T} \in \Gamma^{3,2}$, l'élément de $\Gamma^{3,2}$ tel que $\forall i = 1, \dots, 6 : \Pi(\mathcal{T})(., i) = \Pi(\mathcal{T}(., i))$. Comme Π fixe les extrémités, on a bien $\Pi(\mathcal{T}) \in \Gamma^{3,2}$. On note $G^{3,2}$ l'ensemble des triplets de paires de géodésiques dans \hat{G} jointes en leurs extrémités.

Lemme 4.1

Soient $W \geq 0$ et $\epsilon > 0$. Il existe $\delta > 0$ tel que, si $\mathcal{T} \in \Gamma^{3,2}$ et

$$(\star) \quad 2\pi(W - \delta) < L^2(\Pi(\mathcal{T})) \leq L^2(\mathcal{T}) < 2\pi(W + \delta),$$

alors $\text{dist}(\Pi(\mathcal{T}), G^{3,2}) < \epsilon$.

Démonstration. Soient $W \geq 0$ et $\epsilon > 0$. Soit $\eta_0 > 0$, donné par (3), tel que $\forall \delta > 0, \delta \leq \eta_0 \implies \phi(\delta) < \epsilon^2/144$. On rappelle que ϕ est continue positive, avec $\phi(0) = 0$. Soit, pour $i = 1, \dots, 6$, $\eta_i > 0$, donnés par (4), tels que $L(c_i) - L(\Pi(c_i)) < \eta_i \implies \text{dist}(c_i, \hat{G}) < \epsilon/12$, où les c_i sont les arêtes d'un $\mathcal{T} \in \Gamma^{3,2}$, jointes deux par deux en leurs extrémités. Si $L(\mathcal{T}) - L(\Pi(\mathcal{T})) < \min \eta_i$, alors $\text{dist}(\mathcal{T}, G^{3,2}) < \epsilon/2$.

On pose $\hat{\delta} = \min(\eta_0, \eta_i)$ et $\delta = \hat{\delta}/4\pi$.

Supposons que l'on ait $2\pi(W - \delta) < L^2(\Pi(\mathcal{T})) \leq L^2(\mathcal{T}) < 2\pi(W + \delta)$.

Il vient d'une part :

$$\begin{aligned} L^2(\mathcal{T}) - L^2(\Pi(\mathcal{T})) &< 4\pi\delta \\ L(\mathcal{T}) - L(\Pi(\mathcal{T})) &< \frac{4\pi\delta}{L(\mathcal{T}) + L(\Pi(\mathcal{T}))} < \hat{\delta} \leq \min \eta_i \end{aligned}$$

Donc $\text{dist}(\mathcal{T}, G^{3,2}) < \epsilon/2$.

De même, on montre que $\text{dist}(\Pi(\mathcal{T}), \mathcal{T}) < \epsilon$.

L'inégalité triangulaire donne :

$$\text{dist}(\Pi(\mathcal{T}), G^{3,2}) \leq \text{dist}(\Pi(\mathcal{T}), \mathcal{T}) + \text{dist}(\mathcal{T}, G^{3,2}) < \epsilon$$

□

Théorème 4.1 (Voir Théorème 3.1)

Soient $W \geq 0$ et $\epsilon > 0$; il existe $\delta > 0$ tel que, si $k > 1/\delta$ et pour un certain t_0 :

$$2\pi E(\mathcal{Z}^k(., \{1, 2, 3\}, t_0)) = L^2(\mathcal{Z}^k(., \{1, 2, 3\}, t_0)) > 2\pi(W - \delta),$$

alors pour ce k nous avons $\text{dist}(\mathcal{Z}^k(., \{1, 2, 3\}, t_0), G^{3,2}) < \epsilon$.

Dans l'état actuel de nos recherches, nous aboutissons à deux conjectures, sous la forme d'un porisme :

Conjecture 4.1.

Si 1 est valeur d'adhérence des temps de réalisation des maximums d'énergie, alors on considère une sous-suite de tranches $\mathcal{B}^k(., \{1, 2, 3\}, t_k)$ telle que :

$$\max_{t \in [0,1]} E(\mathcal{B}^k(., \{1, 2, 3\}, t)) = E(\mathcal{B}^k(., \{1, 2, 3\}, t_k)) < W + \frac{1}{k} \text{ avec } t_k \longrightarrow 1$$

On redresse chaque balayage \mathcal{B}^k pour construire un balayage $\hat{\mathcal{B}}^k$ dont les tranches $\hat{\mathcal{B}}^k(., \{1, 2, 3\}, t)$ sont dans $\Gamma^{3,2}$. On applique enfin l'opérateur Π à toutes les demi-fibres de $\hat{\mathcal{B}}^k$, pour obtenir un troisième balayage noté \mathcal{Z}^k , homotope à \mathcal{B}^k . On a : $\forall t \in [0, 1], \mathcal{Z}^k(., \{1, 2, 3\}, t) = \Pi(\hat{\mathcal{B}}^k(., \{1, 2, 3\}, t))$. Le théorème 4.1 précédent s'applique alors et une certaine suite de tranches $\mathcal{Z}^k(., \{1, 2, 3\}, t'_k)$ doit converger vers trois paires de demi-fibres géodésiques. Comme $t_k \longrightarrow 1$, et bien qu'en général $t_k \neq t'_k$, nous pensons que cette limite est un 3-filet géodésique stationnaire.

Conjecture 4.2.

Si 1 n'est pas valeur d'adhérence des temps de réalisation des maximums d'énergie, alors redresse chaque balayage \mathcal{B}^k comme précédemment. Il existe h tel que pour tout $s \in [1 - h, 1]$, s n'est pas non plus valeur d'adhérence des temps de réalisation des maximums d'énergie. On sait que Ψ est homotope à l'identité. On construit une famille continue d'opérateurs Ψ_s , pour $s \in [0, 1]$ telle que $\Psi_s = \Psi$ pour tout $s \in [0, 1 - h]$ et $\Psi_1 = \text{Id}$. Cette homotopie, dont on trouve une description dans Croke⁶, est telle que $s < t \implies (L(\gamma) - L(\Psi_t(\gamma))) \leq (L(\gamma) - L(\Psi_s(\gamma)))$. On applique, pour tout $s \in [0, 1]$, l'opérateur Ψ_s aux trois fibres $\mathcal{B}^k(., i, s)$ pour obtenir un troisième balayage noté \mathcal{Z}^k . Il s'agira de montrer que \mathcal{Z}^k est continu en t et homotope à \mathcal{B}^k . Nous pensons pouvoir conclure qu'il existe une suite de tranches de \mathcal{Z}^k qui converge vers trois géodésiques fermées, simples et disjointes.

Supposons que ces conjectures soient vraies. Dans un pantalon, trois géodésiques fermées, simples et disjointes ne peuvent être que les trois bords γ_1 , γ_2 et γ_3 . La situation de la deuxième conjecture est donc impossible, puisqu'en courbure négative, ces bords ne sauraient ni réaliser un maximum d'énergie, ni minimiser une suite de maximums d'énergie. Donc, seule la situation de la première conjecture a lieu et il existe un 3-filet géodésique stationnaire dans P .

4.5 Autre esquisse de preuve - où l'on n'utilise pas de balayage

Soit Γ l'espace des chemins $c : [0, 1] \longrightarrow P$ géodésiques par morceaux avec L angles, éventuellement plats, correspondant à une partition de la source, propre à chaque chemin :

$$[0, 1] = \bigcup_{i=0}^L [x_i, x_{i+1}], \text{ où } 0 = x_0 < \dots < x_{L+1} = 1$$

Sur chaque morceaux, on demande que c soit l'unique plus courte géodésique de x_i à x_{i+1} . On remarque que L doit être impair et suffisamment grand pour que $((L + 1)R_{\text{inj}}(P))^2 \gg 2\pi W$. L'espace Γ est compact.

Soit \mathcal{H} la classe d'homotopie - dans l'espace des Θ -filets sur P avec arêtes $c_{1,2,3}$ dans Γ - d'un Θ -filet canonique tel que $\forall i \in \mathbb{Z}/3\mathbb{Z}$, la concaténation de c_i et c_{i+1} est homotope à γ_{i+2} . Nous pensons qu'un tel représentant existe (cf. figure 4.5). Par ailleurs, étant donnée une suite de Θ -filets dans \mathcal{H} , les suites induites d'arêtes $(c_i^n)_{n \in \mathbb{N}}$ ne peuvent converger dans Γ que pour former un Θ -filet limite ou un graphe en huit. En effet, pour $i \neq j$, on montre qu'il ne peut y avoir convergence des deux suites (c_i^n) et (c_j^n) , ni vers deux points (nécessairement confondus), ni vers une même arête. Supposons que (c_i^n) et (c_j^n) convergent⁷ vers un même chemin constant. Pour tout n , il existe par hypothèse une homotopie $h_i^n : [n, n + 1] \times [0, 1] \longrightarrow P$ entre c_i^n et c_i^{n+1} , ainsi qu'une homotopie $h_j^n : [n, n + 1] \times [0, 1] \longrightarrow P$ entre c_j^n et c_j^{n+1} , toutes deux issues de l'homotopie dans \mathcal{H} entre les deux Θ -filets dont on a extrait les arêtes c_i^n et c_j^n d'une part, c_i^{n+1} et c_j^{n+1} d'autre part. On

6. *Area and the length of the shortest closed geodesic*, 1986 - Pages 4 et 5.

7. On pourra observer qu'une suite de paires accrochées d'éléments de Γ ne peut converger que vers une paire accrochée d'éléments de Γ , qui est compact.

a pour tout t et pour tout n : $h_i^n(t, 0) = h_j^n(t, 0)$ et $h_i^n(t, 1) = h_j^n(t, 1)$. On construit deux homotopies qui relient chaque terme des suites (c_i^n) et (c_j^n) :

$$h_i : [0, 1] \times [0, 1] \longrightarrow P \text{ telle que } h_i(t, s) = h_i^{\lfloor f(t) \rfloor}(f(t), s) \text{ où } f(t) = \frac{1}{1-t} - 1$$

$$h_j : [0, 1] \times [0, 1] \longrightarrow P \text{ telle que } h_j(t, s) = h_j^{\lfloor f(t) \rfloor}(f(t), s) \text{ où } f(t) = \frac{1}{1-t} - 1$$

On prolonge ces homotopies en 1 en posant $h_i(1, \cdot) = \lim_{n \rightarrow \infty} (c_i^n)$ et $h_j(1, \cdot) = \lim_{n \rightarrow \infty} (c_j^n)$.

On a pour tout t : $h_i(t, 0) = h_j(t, 0)$ et $h_i(t, 1) = h_j(t, 1)$. Ainsi la concaténation de c_i^0 et c_j^0 forme un lacet homotope à un point, à travers des lacets de P . Or, par hypothèse, elle est aussi homotope à l'une des trois composantes de bord de P . Le lemme suivant montre qu'il n'est pas possible qu'une composante de bord de P soit homotope à un point.

Lemme 4.2

Une variété topologique compacte connexe à bord, de dimension 2, dont l'une des composantes de bord est un cercle homotope à 0, est nécessairement un disque.

Ainsi, (c_i^n) et (c_j^n) ne peuvent s'écraser en un seul point. On montre de même qu'elles ne peuvent s'écraser l'une contre l'autre sur toute leur longueur. On pose :

$$W = \min_{\mathcal{H}} E(\mathcal{F})$$

En vertu du raisonnement précédent, on a nécessairement $W > 0$.

On extrait dans \mathcal{H} une suite de Θ -filets telle que : $W < E(\mathcal{F}^j) < W + 1/j$ (\star).

On note $\Pi(\mathcal{F}) \in \mathcal{H}$ le filet obtenu en appliquant Π aux trois arêtes de \mathcal{F} . On a pour tout j :

$$W \leq E(\Pi(\mathcal{F}^j)) \leq E(\mathcal{F}^j) < W + 1/j$$

Cela implique :

$$\begin{aligned} 0 &\leq \sum_{i=1}^3 E(\Pi(c_i^j)) - \sum_{i=1}^3 E(c_i^j) < \frac{1}{j} \\ 0 &\leq \sum_{i=1}^3 \left(E(\Pi(c_i^j)) - E(c_i^j) \right) < \frac{1}{j} \end{aligned}$$

Pour $i = 1, 2, 3$, il vient :

$$\begin{aligned} 0 &\leq E(\Pi(c_i^j)) - E(c_i^j) < \frac{1}{j} \\ L(\Pi(c_i^j)) - L(c_i^j) &< \frac{2\pi}{j(L(\Pi(c_i^j)) + L(c_i^j))} < \frac{2\pi}{j} \end{aligned}$$

Soit $\epsilon > 0$ et soit δ prescrit par la propriété (4) de la Proposition 4.3.1. Soit enfin $j > 2\pi/\delta$. D'après cette même propriété (4), $L(c_i^j) < \epsilon$ pour $i = 1, 2, 3$. Ainsi la suite \mathcal{F}^j converge-t-elle vers un 3-filet géodésique \mathcal{N} . Une des arêtes peut converger vers un point ; il s'agit alors d'un 3-filet géodésique modelé sur le graphe en 8. Nous savons en outre que $E(\mathcal{N}) = W$, de sorte que toute variation de \mathcal{N} augmente son énergie et sa longueur totale. C'est donc un 3-filet géodésique stationnaire.

Conclusion : Notre procédure de minimisation sur une famille de Θ -filets conduit à l'existence d'un 3-filet stationnaire sur les pantalons, modelé sur le Θ -graphe ou le graphe en 8. Néanmoins, une procédure de minimisation sur un espace convenable de 3-filets modelés sur le graphe en 8, décrits comme images de S^1 dans un pantalon géodésiquement convexe, où l'on impose un point de contact, donne déjà depuis Poincaré une géodésique fermée qui s'auto-intersecte une fois, c'est-à-dire un cas particulier de huit stationnaire. Ainsi,

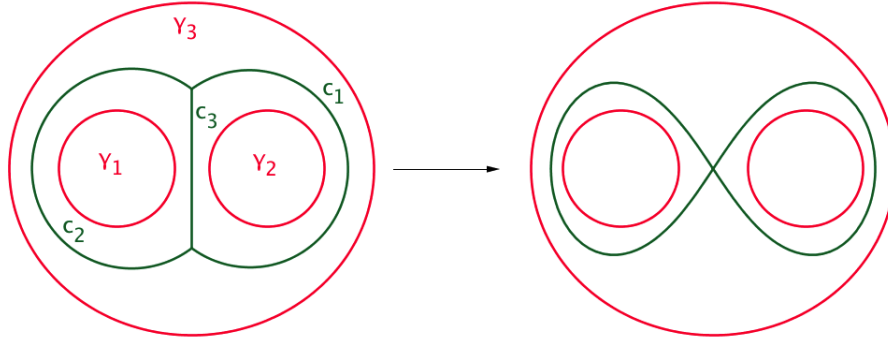


FIGURE 4.6 – Dégénérescence du Θ -filet

notre preuve n'est-elle peut-être pas satisfaisante, dans le sens où elle ne garantit pas l'existence d'un objet nouveau. Nous aimerions savoir par exemple s'il existe toujours un 3-filet stationnaire modelé sur le Θ -graphe dans un pantalon - c'est précisément l'enjeu de la partie précédente, demeurée au stade de la conjecture. Par ailleurs, si un 8 limite est minimisant parmi les 8 et les Θ , il ne l'est pas parmi tous les 3-filets. En effet, un graphe à lunettes pourrait modeler un 3-filet stationnaire optimal. La topologie particulière du pantalon nous invite à rechercher un résultat plus fort que la stationnarité, à savoir être minimisant.

Calendrier de recherche : Dans un premier temps, nous nous attacherons à prouver les deux conjectures du chapitre 4, nous aurons alors montré l'existence de Θ -filets stationnaires dans les pantalons. Si le cas du pantalon présente des avantages topologiques, les deux autres types de 2-sphères - à savoir les doubles bonnets et les concaténations de nasses - charrient de nouveaux problèmes. Nous privilégierons deux approches pour tenter de construire des 3-filets stationnaires sur n'importe quelle 2-sphère. La première sera quantitative : Le chapitre suivant nous donne des estimations pour balayer une sphère avec des cercles de longueur limité, qui doivent s'articuler autour de Θ -filets. Une traque de type algorithmique pourrait nous permettre de resserrer le filet sur notre proie. La deuxième est qualitative : nous essaierons de balayer la sphère directement avec des Θ -filets. Il s'agira de décrire un représentant de classe convenable pour qu'une procédure de min-max puisse aboutir. L'avantage de cette procédure est que la procédure de min-max ne peut qu'aboutir à un Θ -filet.

Chapitre 5

Un nouvel horizon : L'article de Yevgeny Liokumovich

Dans son article de 2014 *Slicing a 2-sphère*, Yevgeny Liokumovich tente de séparer la 2-sphère en zones d'aires maximales, avec des lacets de longueurs minimales. Par ailleurs, il décrit comment envoyer une sphère dans un arbre trivalent, de telle sorte que les pré-images des sommets trivalents soient des Θ -filets de longueur contrôlée. Nous rapportons ici quelques résultats qui nous semblent pouvoir éclairer notre recherche à venir sur les filets.

Il y a une obstruction à la division d'une 2-sphère M en deux hémisphères d'aires grandes par une courbe simple et fermée petite. En effet, si M est une pieuvre aux tentacules suffisamment minces et longues, alors pour tout $r > \frac{1}{3}$, la longueur de la courbe fermée simple la plus courte subdivisant M en deux régions d'aire $\geq r|M|$ peut être arbitrairement grande. Alexander Nabutovsky pose la question suivante : quelle est la valeur maximale de $r \in [\frac{1}{4}, \frac{1}{3}]$ telle que pour un certains $c(r)$, chaque 2-sphère riemannienne d'aire 1 peut être subdivisée en deux disques d'aire $\geq r$ par une simple courbe fermée de longueur $\leq c(r)$?¹ Un premier résultat apporte une réponse à cette question.

Theorème 5.1 (*Slicing a 2-sphère*, Theorem 1, page 2)

Il existe une simple courbe fermée γ de longueur inférieure à $26\sqrt{|M|}$ qui subdivise M en deux sous-disques d'aire supérieure à $\frac{1}{3}|M|$.

Theorème 5.2 (*Slicing a 2-sphère*, Theorem 2, page 2)

Il existe une application f de M dans un arbre trivalent T , telle que les fibres de f ont une longueur $\leq 26\sqrt{|M|}$ et une topologie contrôlée : la pré-image de chaque point intérieur est une simple courbe fermée, la pré-image de chaque sommet terminal est un point et la pré-image de chaque sommet de degré 3 est homéomorphe au Θ -graphe.

Theorème 5.3 (*Slicing a 2-sphère*, Theorem 3, page 2)

Il existe une fonction de Morse $f : M \rightarrow \mathbb{R}$ dont les fibres ont une longueur qui n'excède pas $52\sqrt{|M|}$.

Définition 5.1 (*Slicing a 2-sphère*, Definition 9, page 8)

Une application f de M_k vers un arbre trivalent T est appelée une T -application si la topologie des fibres de f est contrôlée dans le sens suivant : la pré-image de tout point à l'intérieur d'une arête de T est un cercle, il existe k sommets terminaux $x_k \in T$, tels que $f^{-1}(x_k)$ est une composante connexe de ∂M_k , la pré-image des autres points terminaux de T sont des points et la pré-image d'un sommet trivalent de T est homéomorphe au θ -graphe.

1. Pour $r = \frac{1}{4}$, Panos Papasoglu a trouvé $c(r) = 2\sqrt{3}$ (*Cheeger constants of surfaces and isoperimetric inequalities*, 2009).

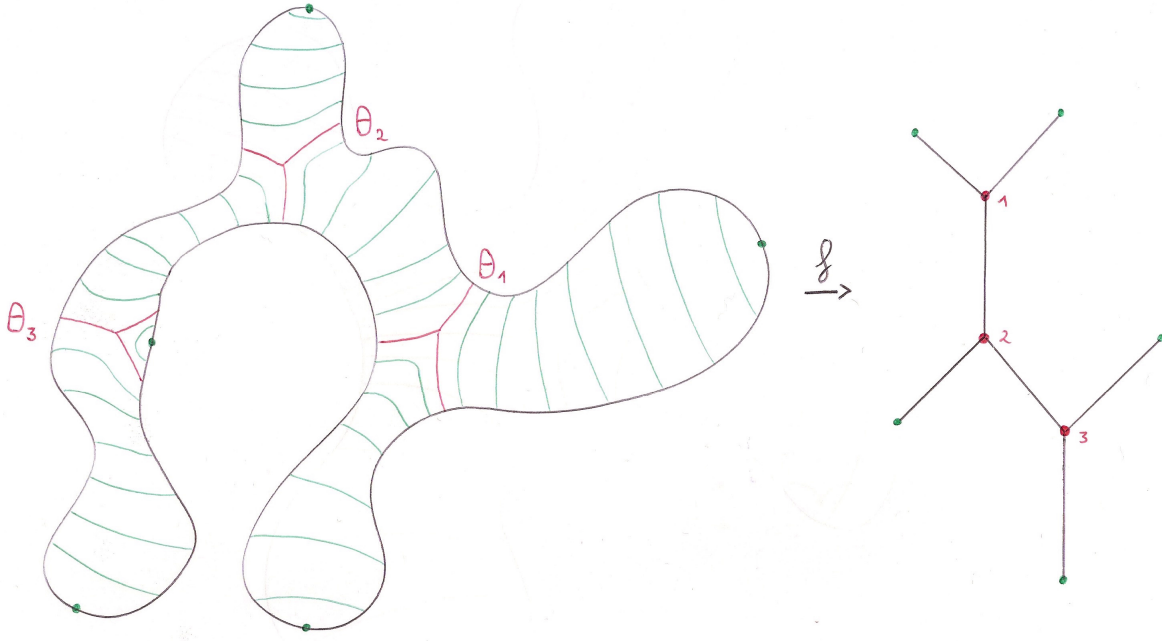


FIGURE 5.1 – T -application vers un arbre trivalent à 8 sommets, dont 5 points terminaux.

Remarque 5.1. Le balayage du pantalon que nous décrivons dans la partie 4 peut être décrit comme les niveaux d'une T -application vers un tripode, où les 3 points terminaux ont pour pré-images les 3 composantes de bord du pantalon. La pré-image du sommet trivalent est un Θ -filet.

Theorème 5.4 (*Slicing a 2-sphère*, Theorem 10, page 8)

Pour $r \in (0, \frac{1}{4}]$ et tout $\epsilon > 0$, il existe une T -application f de M_p , $p \geq 0$, telle que chaque fibre de f a une longueur inférieur à $\frac{c(r)}{1-\sqrt{1-r}} + |\partial M_p| + \epsilon$.

Lemme 5.1 (*Slicing a 2-sphère*, Lemma 11, page 9)

Soit A_1 et A_2 deux sous-variétés lisses fermées (avec frontière) de M_p , telles que $\alpha = A_1 \cap A_2$ est un arc connexe. Soit c_i la composante connexe de ∂_i qui contient α . Supposons que $|c_1 \cup c_2| < L$ et que chaque A_i admet une T -application avec des fibres de longueur $< L$, alors $A_1 \cup A_2$ admet une T -application avec des fibres de longueur $\leq L$.

Lemme 5.2 (*Slicing a 2-sphère*, Lemma 12, page 9)

Pour tout $\epsilon > 0$, il existe $l > 0$ tel que pour tout disque $D \subset M_p$ avec $|\partial D| \leq l$, il existe un difféomorphisme f de D vers le disque fermé standard $D_{st} = \{x^2 + y^2 \leq 1\}$ tel que la longueur de chaque pré-image $f^{-1}(\{x^2 + y^2 = cte\})$ n'excède pas $(1 + \epsilon)|\partial D|$.

Lemme 5.3 (*Slicing a 2-sphère*, Lemma 13, page 10)

Pour tout $\epsilon > 0$, il existe $A > 0$ tel que pour tout disque $D \subset M_p$ avec $|D| \leq A$, il existe une T -application f de D avec des fibres plus courtes que $|\partial D| + \epsilon$.

Bibliographie

- Ian Adelstein, *Minimizing geodesic nets and critical points of distance*, Differential Geometry and its Applications **2019**.
- Tobias Colding et William Minicozzi, *Min-max for sweepouts by curves*, 2007.
- Christopher Croke, *Poincaré's problem on the shortest closed geodesic on a convex hypersurface*, Journal of Differential Geometry **17**, 1982.
- Joël Hass et Frank Morgan, *Geodesic nets on the 2-sphere*, Proceedings of the American Mathematical Society **124**(12), 1996.
- Yevgeny Liokumovich, *Slicing a 2-sphere*, Journal of Topology and Analysis **6**(4), 2014.
- Lazar Lusternik et Lev Schnirelman, *Sur le probleme de trois géodesiques fermées sur les surfaces de genre 0*, Comptes Rendus de l'Académie des Sciences de Paris **189**, 1929.
- Frank Morgan, *Soap bubbles in \mathbb{R}^2 and in surfaces*, Pacific Journal of Mathematics **165**, 1994.
- Alexander Nabutovsky et Fabian Parsch, *Geodesic nets : some examples and open problems*, Experimental Mathematics, 2020.
- Panos Papasoglu, *Cheeger constants of surfaces and isoperimetric inequalities*, Transactions of the American Mathematical Society **361**(10), 2009.
- Tristan Rivière, *Minmax Methods for Geodesics and Minimal Surfaces*, Powerpoint.
- Tristan Rivière, *Méthodes de min-max et la Conjecture de Willmore*, Séminaire Bourbaki **1080**, 2014.

