



Génération de modèle géométrique de maille de stent

RAPPORT DE TP DE SYNTHÈSE

MASTER TRAITEMENT DE SIGNAL ET DES IMAGES EN MÉDECINE

MERYEM SIKOUKY, HADI YEHYA

09/2023 - 01/2024

Membres du jury

Laurent TRASSOUDAINE, Professeur, Université Clermont Auvergne

Émilie PÉRY, Maître de conférence, Université Clermont Auvergne

Encadrés par :

LAURENT SARRY
ÉMILIE PÉRY

Remerciements

Nous tenons à exprimer notre sincère gratitude à l'égard du Professeur Laurent SARRY pour son accompagnement dévoué et ses précieux conseils qui ont jalonné notre parcours de master. Sa présence bienveillante et motivante a été une véritable source d'inspiration et a joué un rôle essentiel dans l'atteinte de nos objectifs. Nous lui adressons nos vœux les plus chaleureux pour un prompt rétablissement.

Nous adressons également nos remerciements appuyés au Professeur Laurent TRAS-SOUDAINE, directeur de l'EUPI, et à la Professeure Émilie PERY pour leur suivi rigoureux et leur soutien indéfectible. Leur engagement et leur assistance, particulièrement lors de ces dernières semaines marquées par l'imprévu, ont été pour nous d'une valeur inestimable. Nous leur sommes profondément reconnaissants.

Notre gratitude s'étend à ces mentors qui ont été des piliers déterminants de notre développement académique. Leur influence positive et leur soutien constant nous ont été précieux et c'est avec une grande chaleur que nous leur exprimons notre reconnaissance.

Résumé

Notre travail, s'inscrivant dans le cadre du projet VSTENT, a pour but de réaliser des modélisations géométriques de différentes mailles des stents coronaires. Dans le premier chapitre, nous explorons en profondeur l'anatomie des artères coronaires, les maladies qui les affectent, et les diverses options de traitement, avec un accent particulier sur les approches chirurgicales et interventionnelles. Le deuxième chapitre se consacre à l'état de l'art sur les stents coronaires, abordant leur rôle, leur fonctionnement, leurs composants, les différents types disponibles sur le marché, ainsi que leurs propriétés spécifiques. Nous définissons également les objectifs spécifiques de notre recherche dans ce contexte. Le troisième chapitre expose les méthodes et outils employés, mettant en avant les courbes paramétrées, les courbes de Bézier, et l'utilisation d'outils tels qu'Inkscape et SVGPathTools. L'organisation du programme en orientation objet et l'intégration de bases de données sont également discutées. Le quatrième chapitre détaille l'implémentation concrète de notre programme, depuis la réalisation des schémas sur Inkscape jusqu'à la lecture et manipulation des chemins SVG, l'échantillonnage équidistant et basé sur la courbure, la correction des connexions, et la création d'une interface utilisateur. Les schémas générés et les fichiers CSV constituent les livrables visuels de cette phase. En conclusion, le dernier chapitre offre un bilan global de la recherche, soulignant nos principales contributions, et ouvre la voie à des perspectives futures dans ce domaine crucial de la médecine.

Summary :

The aim of our work, part of the VSTENT project, is to produce geometric models of various coronary stent meshes. In the first chapter, we explore in depth the anatomy of coronary arteries, the diseases that affect them, and the various treatment options, with particular emphasis on surgical and interventional approaches. The second chapter is devoted to the state of the art in coronary stents, covering their role, function, components, the different types available on the market, and their specific properties. We also define the specific objectives of our research in this context. The third chapter outlines the methods and tools employed, highlighting parametric curves, Bézier curves, and the use of tools such as Inkscape and SVGPathTools. Object-oriented program organization and database integration are also discussed. The fourth chapter details the concrete implementation of our program, from the creation of schematics on Inkscape to the reading and manipulation of SVG paths, equidistant and curvature-based sampling, connection correction, and the creation of a user interface. The generated schematics and CSV files are the visual deliverables of this phase. In conclusion, the final chapter offers an overall assessment of the research, highlighting our main contributions, and paves the way for future perspectives in this crucial area of medicine.

Table des matières

1 Contexte médical	8
1.1 Anatomie des artères coronaires	8
1.2 Maladies coronariennes	9
1.3 Traitements	10
1.3.1 Traitement chirurgical	10
1.3.2 Traitement interventionnel	11
2 État de l'art sur les stents	12
2.1 Stents	12
2.2 Principe de fonctionnement et déploiement du stent	13
2.3 Composants des stents	13
2.4 Types des stents coronaires	15
2.5 Propriétés des stents coronaires	16
2.6 Objectif de notre travail	17
3 Méthodes et Outils	18
3.1 Méthodes	18
3.1.1 Courbes paramétrées	18
3.1.2 Courbes de Bézier	19
3.1.3 Courbes de Bézier cubique	20
3.1.4 Différents types des images	21
3.2 Outils	24
3.2.1 Inkscape	24
3.2.2 SVGPathTools	26
3.2.3 Organisation du programme en Orienté objet	27
3.3 Bases de données	27
4 Implémentation de notre programme	29
4.1 Réalisation des schémas sur inkscape	29
4.2 Lecture et manipulation des chemins SVG	30
4.2.1 Lecture des chemins SVG	30
4.2.2 Manipulation des chemins SVG	31
4.3 Echantillonnage	32
4.3.1 Echantillonnage équidistant	33
4.3.2 Echantillonnage basé sur la courbure	35
4.3.3 Correction des connexions des connecteurs et la maille	36
4.4 Interface d'utilisateur	38
4.5 Visualisation des schémas et Fichiers CSV comme livrables	39
4.6 Limitations	41
5 Conclusion et perspectives	42

Table des figures

1	Schéma de la localisation des artères coronaires à la surface du cœur	8
2	Représentation des trois couches concentriques de la paroi artérielle	9
3	Représentation schématique d'une sténose artérielle	10
4	Stent coronaire	12
5	Applications cliniques de stent	12
6	Méthodes d'expansion des stents	13
7	Composants du stent	14
8	Catégories des stents	15
9	Athérosclérose des vaisseaux sanguins et resténose intra-stent : (a) Athérosclérose ; (b) RIS	17
10	Exemple de représentation d'une courbe paramétrée	19
11	Exemple de courbes de bázier cubiques	21
12	Image Bitmap vs Image vectorielle	23
13	Exemple d'une image véctorielle résultante sur SVG	23
14	Interface d'utilisateur d'InkScape	24
15	Menu des options sur Inkscape	24
16	Définition des trajets sur Inkscape	25
17	Choix de l'outil "Node"	25
18	Définition des courbes de bázier sur Inkscape	25
19	Exemple d'une image véctorielle résultante sur InkScape	26
20	Schémas longitudinales de quelques modèles de stents	27
21	Schémas longitudinales de quelques modèles de stents- partie 2	28
22	Réalisation de schéma vectoriel du modèle "Synergy" sur Inkscape	29
23	Représentation sous forme d'image vectorielle du modèle "Synergy"	30
24	Échantillonnage équidistant du modèle Synergy	34
25	Échantillonnage basé sur la courbure du modèle Synergy	36
26	Erreurs d'interesection du connecteur avec la maille	37

27	Correction d'intersection de la maille avec un connecteur	38
28	Interface d'utilisateur	39
29	Format du nom du fichier CSV résultant	40
30	Représentation des points échantillonnés dans un fichier CSV	40
31	Limitations des connecteurs haut et bas	41
32	Echantillonnage équidistant du modèle Utlimaster Nagomi	43
33	Echantillonnage basé sur la courbure du modèle Utlimaster Nagomi .	43
34	Echantillonnage équidistant du modèle Utlimaster Tansei	44
35	Echantillonnage basé sur la courbure du modèle Utlimaster Nagomi .	44
36	Echantillonnage équidistant du modèle Xience Skypoint	45
37	Echantillonnage basé sur la courbure du modèle Xience Skypoint . .	45
38	Echantillonnage équidistant du modèle Xience Xpedition/Alpine . .	46
39	Echantillonnage basé sur la courbure du modèle Xience Xpedition/Al- pine	46

Liste des abréviations

CAO	Conception Assistée par Ordinateur
CB	Connecteur bas
CD	Connecteur droit
CG	Connecteur gauche
CH	Connecteur haut
CSS	Cascading Style Sheets
CSV	Comma-Separated Values
EPS	Encapsulated PostScript
NURBS	Non-Uniform Rational B-Splines
XML	eXtensible Markup Language

Introduction générale

Le système cardiovasculaire joue un rôle crucial dans le maintien de la santé humaine, et les artères coronaires sont au cœur de cette fonctionnalité. Dans le domaine des interventions cardiaques, les stents coronaires ont émergé comme des outils essentiels pour restaurer la circulation sanguine.

Dans le cadre du projet VSTENT, le but de sa contribution est de se concentrer sur la simulation du déploiement de stents dans les artères coronaires. Pour mener à bien ces simulations en temps réel, il est impératif de disposer de modèles CAO précis représentant les différents stents disponibles sur le marché, fournis par des fabricants tels qu'Abbott, Boston Scientific, Medtronic, Terumo, et bien d'autres.

Un stent métallique, élément central de cette étude, se compose de couronnes reliées par des connecteurs, chaque couronne étant formée de mailles. Notre objectif dans ce projet est de fournir une description géométrique détaillée de ces mailles, avec la possibilité de les rééchantillonner sous la forme de segments de droites. Cette approche permettra une modélisation plus précise et flexible du stent, essentielle pour des simulations réalistes.

Pour atteindre cet objectif, nous explorons l'utilisation de courbes de Bézier, une technique bien établie dans les logiciels vectoriels tels qu'Inkscape. Ces courbes permettent de définir des points de contrôle et des tangentes dans un chemin au format SVG. En exploitant des outils tels que SVGPathTools en Python, nous serons en mesure de lire et de manipuler ces données, ouvrant ainsi la voie à une représentation optimisée des mailles de stents métalliques pour une simulation précise du déploiement dans les artères coronaires.

La section méthodes et outils détaille l'approche adoptée pour ce travail, y compris l'utilisation de courbes paramétrées, les outils spécifiques employés, et la gestion des bases de données nécessaires pour l'implémentation. Enfin, l'implémentation concrète de notre programme est examinée, depuis la création des schémas sur Inkscape jusqu'à la manipulation des chemins SVG, l'échantillonnage, la correction des connexions, et la mise en place d'une interface utilisateur pour une visualisation efficace des schémas générés.

1 Contexte médical

1.1 Anatomie des artères coronaires

Une maladie des artères coronaires est appelée maladie coronarienne. Ces artères correspondent à des vaisseaux qui se trouvent à la surface du muscle cardiaque (myocarde) et sont responsables de l'approvisionnement sanguin du muscle et de son bon fonctionnement. En règle générale, on distingue deux artères coronaires principales : l'artère coronaire droite et l'artère coronaire gauche, qui irriguent principalement les parties droite et gauche du cœur en l'entourant. Il est observé que de nombreuses bifurcations se trouvent le long de chacune de ces artères, créant ainsi des artères collatérales, qui forment un dense arbre vasculaire. La figure 1 montre un schéma des artères coronaires [1] :

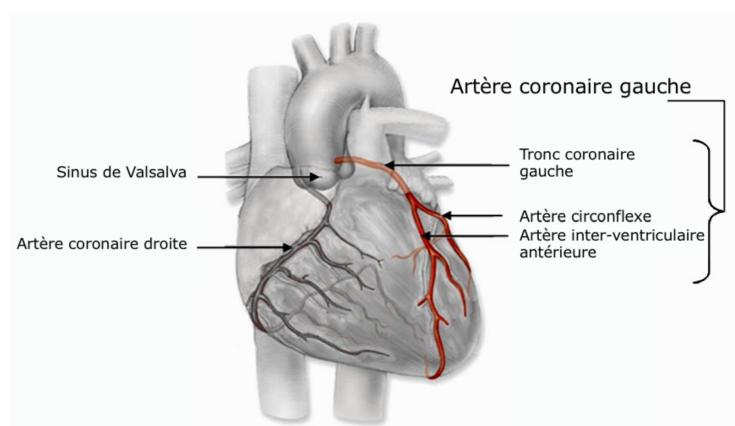


FIGURE 1 – Schéma de la localisation des artères coronaires à la surface du cœur

La particularité des artères coronaires est leur faible diamètre : dans le cas de vaisseaux sains, leur ouverture (ostium) à la base de l'artère aorte mesure 3 à 5 mm pour l'artère coronaire gauche et 2 à 3 mm pour l'artère coronaire droite [2]. Au fur et à mesure que les collatérales apparaissent, ce diamètre diminue progressivement en distal. Le diamètre initial de l'artère aorte est de 30 mm en moyenne au niveau du départ des coronaires [3].

La paroi des coronaires est constituée de trois couches (ou tuniques) distinctes en tant qu'artères : *l'intima*, qui est en contact avec le sang et qui filtre la lumière de l'artère, la couche intermédiaire *media* et la couche externe *adventice*. Les membranes élastiques interne et externe matérialisent les séparations intima/media et media/adventice. La Figure 2 montre comment les couches sont superposées. Cette configuration en trois couches est étroitement liée aux mécanismes physiologiques qui provoquent la maladie coronarienne. Les trois couches varient en composition, et leur superposition définit les caractéristiques mécaniques de l'artère [4].

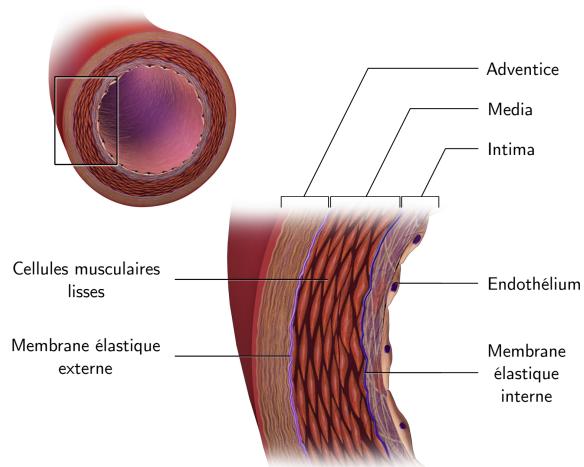


FIGURE 2 – Représentation des trois couches concentriques de la paroi artérielle

1.2 Maladies coronariennes

Le corps reçoit de l'oxygène et des substances nutritives du sang par le cœur, une pompe musculaire puissante. Le cœur est constamment irrigué par les artères coronaires, qui lui fournissent le sang nécessaire à son oxygénation, afin qu'il puisse effectuer cet énorme travail de pompe. Les maladies coronariennes sont celles qui surviennent lorsque l'approvisionnement en sang du muscle cardiaque (également connu sous le nom de myocarde) est interrompu ou bloqué. La cause la plus courante est l'obstruction des artères coronaires par une accumulation de matières grasses, le "mauvais" cholestérol. Ils forment des plaques qui rétrécissent les artères (ce qui est connu sous le nom d'artériosclérose) et peuvent ensuite se durcir. Ce dernier implique un changement dans la structure de l'intima, la couche la plus interne de l'artère, menant à la formation d'une plaque d'athérome. Cette plaque se compose généralement d'un noyau lipidique, de cellules musculaires lisses, et de cellules inflammatoires telles que les macrophages, le tout recouvert d'une fine couche fibreuse constituée de cellules endothéliales [5].

Bien que la formation de ces plaques soit un processus naturel et potentiellement réversible, elles peuvent rester asymptomatiques jusqu'à ce que leur croissance excessive réduise le diamètre interne de l'artère, phénomène connu sous le nom de sténose (Figure 3). Cette réduction du diamètre artériel diminue le flux sanguin vers les organes alimentés par ces artères, entraînant une ischémie, c'est-à-dire une insuffisance de l'apport sanguin à ces organes. L'ischémie peut se manifester de différentes manières selon la gravité de l'obstruction artérielle, allant du dysfonctionnement organique à la défaillance totale. Dans le contexte des artères coronaires, une croissance pathologique de l'athérome conduit à la maladie coronarienne [6].

L'évolution de la maladie coronarienne varie grandement en fonction de la taille et

de la nature des plaques athéromateuses. Une aggravation des symptômes peut indiquer soit une croissance de la plaque, soit une rupture soudaine de l'endothélium au niveau de la plaque, menant à la formation de thrombus (caillots sanguins). Ces caillots peuvent migrer dans le flux sanguin, bloquer totalement une artère en aval (embolie), et provoquer un infarctus.

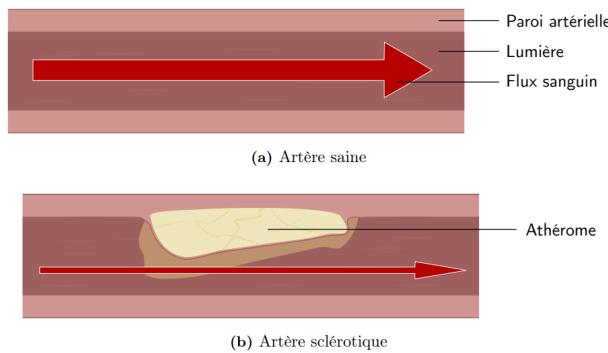


FIGURE 3 – Représentation schématique d'une sténose artérielle

1.3 Traitements

La revascularisation d'une artère sténosée est le principal objectif du traitement des maladies coronaires. Il existe de nombreuses options, chacune avec ses propres avantages et inconvénients, qui doivent être prises en compte en fonction des cas spécifiques de chaque patient. Ci-dessous sont présentées les deux méthodes principales, la chirurgie et l'intervention.

1.3.1 Traitement chirurgical

Historiquement, la méthode de référence est chirurgicale : elle consiste à prélever puis apposer un autre vaisseau au niveau de l'athérome pour réaliser une dérivation. On parle de la chirurgie de pontage aortocoronarien, également connue sous le nom de pontage coronarien, est une procédure chirurgicale majeure. Cette intervention consiste à créer un « pont » autour des artères coronaires obstruées pour rétablir un flux sanguin adéquat vers le cœur. Le chirurgien prélève un segment d'une veine ou d'une artère saine du patient (souvent de la jambe ou de la poitrine) et l'utilise pour contourner les zones rétrécies ou bloquées des artères coronaires [7]. Cette procédure est généralement recommandée pour les patients présentant plusieurs artères coronaires sévèrement obstruées. Bien que ce soit une opération à cœur ouvert, nécessitant une anesthésie générale et un temps de récupération significatif, elle est reconnue pour améliorer significativement les symptômes, notamment la douleur thoracique, et pour prolonger la durée de vie des patients souffrant de maladies coronaires graves [8].

1.3.2 Traitement interventionnel

L'intervention coronarienne percutanée (ICP), plus communément appelée angioplastie coronaire, est une procédure moins invasive. Elle implique l'insertion d'un cathéter fin dans une artère, généralement au niveau de l'aine ou du bras, et son guidage jusqu'à l'artère coronarienne affectée. Une fois en place, un petit ballon attaché à l'extrémité du cathéter est gonflé pour ouvrir l'artère obstruée. Cette méthode a présenté initialement un risque accru de recours à une nouvelle revascularisation dans les 3 ans suite au redéveloppement de la plaque conduisant à une resténose [9]. Pour y pallier, la solution proposée pour stabiliser la plaque était de fixer un petit dispositif métallique (une endoprothèse plus communément appelé *stent*) sur le ballonnet dégonflé afin de pouvoir maintenir l'artère ouverte. L'angioplastie est typiquement utilisée pour traiter les cas de sténose (rétrécissement de l'artère) moins complexes ou en cas de syndrome coronarien aigu. Cette procédure a l'avantage d'être moins traumatisante pour le patient, avec un temps de récupération plus court et un risque moindre par rapport à la chirurgie ouverte [10].

2 État de l'art sur les stents

2.1 Stents

Un stent est une forme d'endoprothèse. Les stents sont de petits tubes métalliques ou en polymère utilisés pour maintenir ouvertes des structures tubulaires dans le corps, comme les artères ou les vaisseaux sanguins, qui peuvent être rétrécies ou bloquées. Ils sont fréquemment utilisés dans le traitement de maladies cardiaques pour ouvrir des artères coronaires rétrécies et prévenir les crises cardiaques. Les stents peuvent également être utilisés dans d'autres parties du corps, telles que les voies urinaires, le tube digestif ou les voies biliaires. La figure 4 présente un schéma classique d'un stent coronaire.



FIGURE 4 – Stent coronaire

Le stent, a fait ses preuves dans de multiples domaines médicaux (Figure 5) :

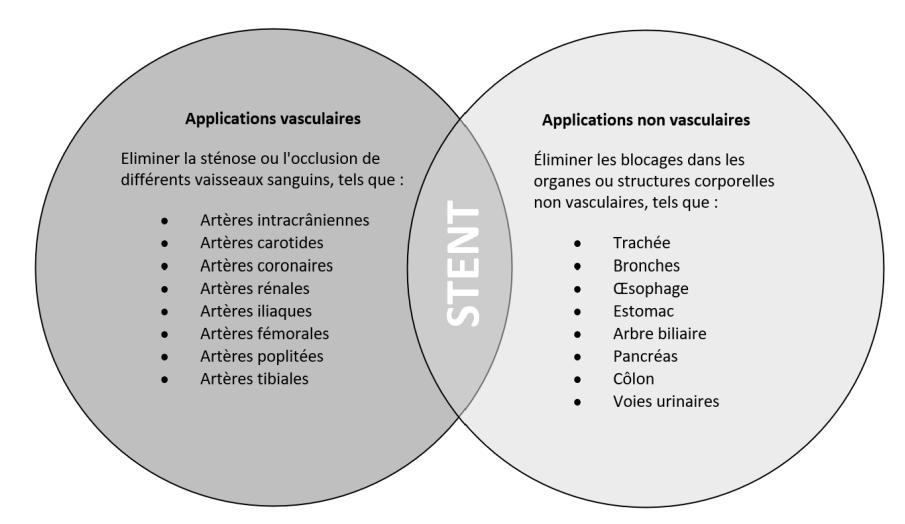


FIGURE 5 – Applications cliniques de stent

2.2 Principe de fonctionnement et déploiement du stent

Comme nous l'avons présenté dans les parties précédentes, les stents servent à maintenir la patence des vaisseaux sanguins rétrécis ou bloqués. Ils sont généralement introduits dans l'artère dans un état compressé à l'aide d'un cathéter. Une fois positionnés précisément à l'emplacement du rétrécissement, ils sont déployés pour soutenir la paroi du vaisseau et permettre un écoulement sanguin normal. Actuellement, les méthodes de déploiement des stents vasculaires implantés dans les vaisseaux sanguins humains comprennent l'expansion par ballonnet (Figure 6-a) et l'auto-expansion (Figure 6-b) [11].

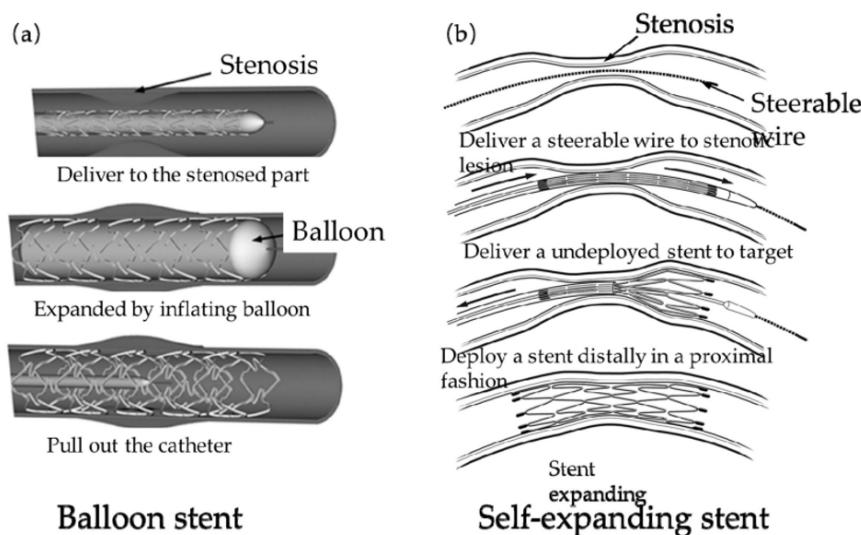


FIGURE 6 – Méthodes d'expansion des stents

Les deux méthodes d'expansion garantissent que le stent reste ouvert et en place après le déploiement, mais le choix de la méthode dépend de la situation clinique spécifique et du type de stent choisi.

2.3 Composants des stents

Dans cette partie, nous décrivons les différents composants des stents qui sont présentés dans le schéma (Figure 7) [12] :

- **Connecteurs** : Les connecteurs sont les éléments qui relient les différentes parties de la structure du stent. Ils jouent un rôle essentiel dans la flexibilité du stent, permettant à celui-ci de se conformer aux courbures naturelles du vaisseau sanguin tout en maintenant une structure de soutien intégrale. Ils peuvent également absorber et répartir les contraintes mécaniques que le

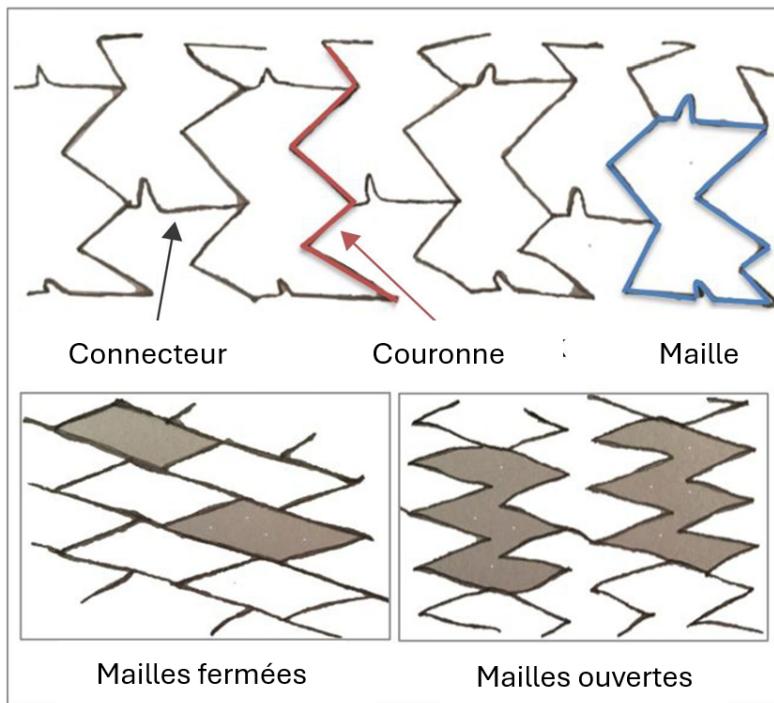


FIGURE 7 – Composants du stent

stent subit dans le corps, réduisant ainsi le risque de fatigue du matériau et de rupture du stent.

- **Couronne ou Sommet :** La couronne, ou sommet, se réfère aux points les plus élevés ou aux extrémités du motif en treillis du stent. Ce sont des points critiques pour l'ancrage du stent dans le vaisseau sanguin. Ils servent de points de contact principaux avec la paroi du vaisseau, aidant à fixer le stent en place et à empêcher son mouvement une fois déployé. Les couronnes doivent être conçues pour minimiser les traumatismes à la paroi du vaisseau tout en fournissant une fixation sûre.
- **Maille (Cellule) :** La maille est l'espace délimité par la structure en treillis du stent. La conception des cellules est cruciale pour plusieurs aspects de la performance du stent, y compris sa flexibilité, sa capacité à s'élargir correctement, et la surface de contact qu'il a avec la paroi du vaisseau. La taille et la forme (ouverte ou fermée) des cellules peuvent influencer la facilité avec laquelle un médecin peut insérer d'autres instruments à travers le stent après son placement, comme lors de procédures de cathétérisme ultérieures ou pour l'administration de traitements localisés.

2.4 Types des stents coronaires

Les stents peuvent être classés en 3 grandes familles : stents *métalliques nus* (BMS : Bare-metal stents), stents *couverts/actifs* (Stents élutants de drogues DES : Drug-Eluting Stents) et endoprothèses *bioresorbables* (BRS) (Figure 8).

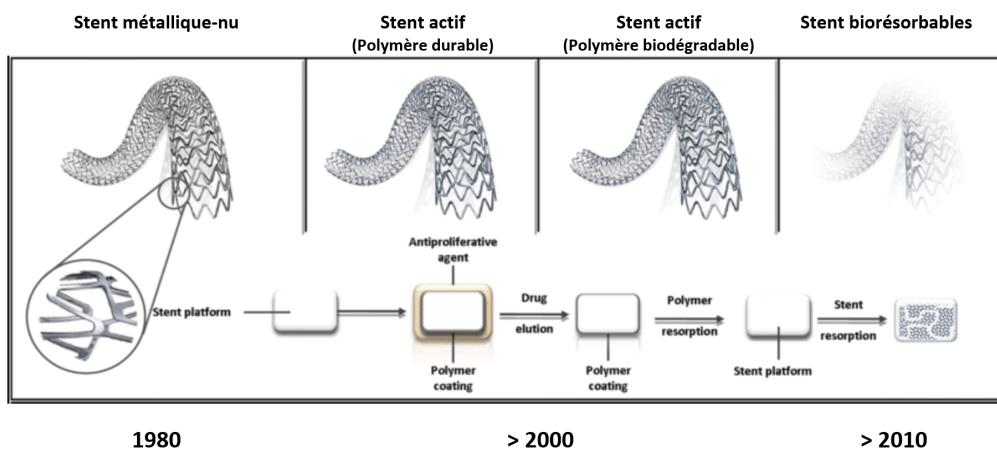


FIGURE 8 – Catégories des stents

1. **Stents métalliques nus (BMS)** : Les stents nus ont été développés en tout premier lieu. Ces stents sont des plateformes métalliques en chrome-cobalt ou chrome-platine. Ils sont à ce jour de moins en moins utilisés, car bien qu'ils soient efficaces sur le recoil et le remodelage vasculaire négatif, ils favorisent la prolifération intime. Si le stent diminue le phénomène de resténose au ballon, il persiste cependant avec les stents nus des taux de resténose importants, de l'ordre de 30 %. Dans le rapport de la HAS 2018 [13], l'avis de la Commission nationale d'évaluation des dispositifs médicaux et des technologies de santé (CNEDiMTs) recommande la prise en charge des stents nus dans une niche thérapeutique (lésion(s) à bas risque de resténose dans un contexte de risque hémorragique élevé ou de difficultés prévisionnelles concernant la prise de bithérapie antiagrégante plaquettaire).
2. **Stents à libération contrôlée de principe actif** : Les stents à libération contrôlée de principe actif ou « stents actifs », à la différence des stents nus (plateformes inertes), sont des plateformes libérant des molécules pharmacologiquement actives pour lutter plus efficacement contre la resténose en inhibant la prolifération des cellules musculaires lisses in situ. Le principe actif utilisé, qui se trouve en haute concentration au niveau de la lésion et en concentration infrathérapeutique au niveau systémique, possède des propriétés antiprolifératives / immunsuppressives qui permettent d'inhiber la prolifération des cellules musculaires lisses dans la lumière du stent abouïssant ainsi à une diminution du phénomène de resténose intrastent. Les

dernières recommandations de l'ESC 2018 préconisent l'utilisation de stent actif en 1ère intention [14]. Les différents stents actifs sont classés selon trois grandes caractéristiques :

- **Le matériau** : polymère non résorbable, polymère biorésorbable, sans polymère.
 - **L'alliage métallique** : chrome-platine, chrome-cobalt, acier.
 - **Le principe actif** : évérolimus, zotarolimus, biolimus, sirolimus (le paclitaxel n'ayant plus été utilisé à partir de la 2ème génération de stents actifs).
3. **Stents biorésorbables "BioResorbable Scaffolds" (BRS)** : L'idée est que les stents biorésorbables permettraient l'utilisation des propriétés mécaniques initiales du stent actif avec un effet médicamenteux contre la resténose et, par une résorption complète, permettraient de laisser le vaisseau libre de tout corps étranger, pouvant diminuer ainsi le taux de thromboses tardives de stent. On distingue 2 types de BRS selon leur nature : polymériques ou à base d'alliages métalliques [15]. Parmi les BRS métalliques, ceux constitués d'alliages de fer et de magnésium ont initialement été étudiés, mais actuellement seuls les BRS en magnésium sont encore en cours de développement [15]. Parmi les BRS polymériques, les plus étudiés sont ceux en acide polylactique.

2.5 Propriétés des stents coronaires

Le stent vasculaire ou coronaire, en tant que dispositif médical pour le traitement clinique, doit avoir des fonctions et des propriétés mécaniques idéales :

1. Assurer une **flexibilité** importante pour permettre la flexion et l'expansion ultérieure du stent dans le vaisseau sanguin.
2. Offrir une **résistance** élevée et une endurance face à la fatigue afin de soutenir les charges physiologiques récurrentes exercées sur les artères.
3. Favoriser une excellente **compatibilité** biologique pour réduire les risques de thrombose, de resténose vasculaire et pour atténuer les réactions de rejet de l'implant par l'organisme.

Un stent ne peut pas couvrir toutes les propriétés idéales et offre généralement plusieurs bonnes propriétés. La structure de conception du stent est liée à la resténose. Il existe encore de nombreuses complications dans les traitements actuels. Parmi celles-ci, la resténose intra-stent (RIS) est la complication la plus courante (comme illustré dans la Figure 9-b), ce qui constitue un défi important pour le génie biomécanique et a un impact sur la conception du stent. Des études ont montré qu'un an après l'implantation de stents métalliques ordinaires, la probabilité de resténose au niveau du vaisseau sanguin malade atteint jusqu'à 20-40%, et environ 10% des patients ont besoin d'une réimplantation du stent [16]. Même pour le stent à élution de médicament, largement utilisé ces dernières années, le taux de resténose dans le stent après l'implantation atteint jusqu'à 8-15% [17].

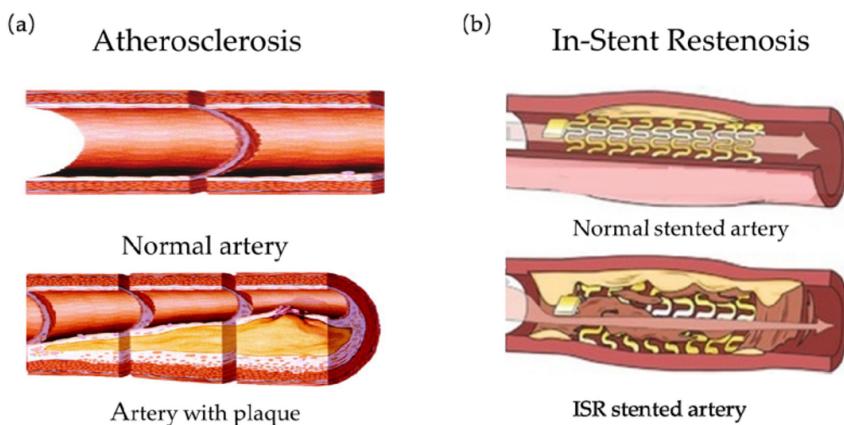


FIGURE 9 – Athérosclérose des vaisseaux sanguins et resténose intra-stent : (a) Athérosclérose ; (b) RIS

2.6 Objectif de notre travail

L'objectif de ce projet est de fournir une représentation géométrique détaillée des composants fondamentaux des stents, à savoir la maille et les connecteurs, et d'offrir la capacité de rééchantillonner ces éléments en segments linéaires.

Dans le troisième chapitre, nous explorerons les méthodes et outils qui seront utilisés pour développer notre programme. Nous utiliserons notamment les courbes de Bézier avec le logiciel de graphisme vectoriel Inkscape, qui nous permettra de tracer avec précision les points de contrôle et les tangentes sur les schémas techniques fournis par les fabricants, pour ensuite les convertir en tracés SVG. Ces tracés pourront par la suite être importés et manipulés en Python en utilisant la bibliothèque SvgPathTools, spécialisée dans le traitement des courbes de Bézier.

Le chapitre quatre détaillera les fonctions implémentées et le processus de développement étape par étape, en utilisant le stent modèle "Synergy" comme étude de cas pratique pour illustrer notre approche. Nous détaillerons aussi les limitations et les challenges que nous avons eu.

Le résultat final de notre projet est de fournir les schémas échantillonnés sous format du fichier CSV qui servira pour une prochaine étape de combinaison des mailles résultantes en couronnes et les couronnes en stent.

3 Méthodes et Outils

3.1 Méthodes

3.1.1 Courbes paramétrées

1. Utilités des courbes paramétrées

Les courbes paramétrées sont des outils essentiels en modélisation géométrique, utilisées dans des domaines variés allant de la conception industrielle à l'animation graphique. Elles se définissent par une représentation mathématique où chaque point de la courbe est associé à un ou plusieurs paramètres. Cette approche permet de décrire des formes complexes et fluides avec un haut degré de précision et de contrôle. Ces courbes peuvent être simples, comme les lignes droites, ou plus complexes, comme les courbes de Bézier ou les splines. Leur principal avantage réside dans leur capacité à représenter des formes continues et lisses à partir de points de contrôle, rendant la manipulation et la modification des formes beaucoup plus intuitive et efficace.

Dans la pratique, les courbes paramétrées offrent une flexibilité et un contrôle précis, permettant de créer des formes avec un petit nombre de points de contrôle. Cela rend leur stockage et traitement plus efficace, notamment dans les logiciels de CAO. Elles sont idéales pour l'interpolation et l'approximation de formes, utiles dans la reconstruction de surfaces à partir d'un ensemble limité de données. Dans l'animation, elles permettent de générer des mouvements fluides et naturels en ajustant les paramètres au fil du temps. Les courbes paramétrées sont également incontournables pour la modélisation de surfaces complexes en 3D, comme les surfaces NURBS. Leur intégration harmonieuse avec d'autres outils de CAO facilite la création de formes tridimensionnelles à partir de profils bidimensionnels. Enfin, leur adaptabilité en fait un choix privilégié pour répondre à des besoins spécifiques, que ce soit pour la simplicité et la facilité de manipulation ou pour une précision mathématique élevée.

2. Formulation mathématique

Une courbe paramétrée plane est une application d'un sous-ensemble D de R dans R^2 . Il s'agit d'une application qui, à un réel t (le paramètre), associe un point du plan. On parle aussi d'**arc paramétré**. On peut la noter :

$$\begin{aligned} f : \quad D \subset R &\rightarrow R^2 \\ t &\mapsto f(t) \end{aligned} \tag{1}$$

ou écrire en abrégé :

$$t \mapsto M(t) \text{ ou } t \mapsto \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} \tag{2}$$

Enfin en identifiant C avec R^2 , on note aussi :

$$t \mapsto z(t) = x(t) + iy(t) \quad (3)$$

avec l'identification usuelle entre le point $M(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$ et son affixe $z(t) = x(t) + iy(t)$.

Nous notons pour que cette définition ait un sens, il faut que $x(t)$ et $y(t)$ existent simultanément. C'est pourquoi le domaine de définition D de la courbe (C) est l'intersection des domaines de définition D_x et D_y des fonctions $x(t)$ et $y(t)$. On a donc $D = D_x \cap D_y$.

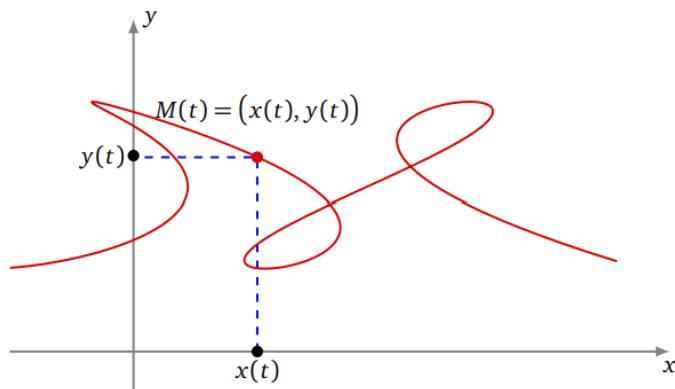


FIGURE 10 – Exemple de représentation d'une courbe paramétrée

3.1.2 Courbes de Bézier

1. Choix des courbes de Bézier pour la modélisation

Dans le cadre de la modélisation géométrique des mailles d'un stent, l'adoption des courbes de Bézier cubiques se présente comme une solution particulièrement adaptée. Les courbes de Bézier cubiques, une forme spécifique de courbes paramétrées, sont définies par quatre points : deux points de contrôle et deux points d'ancre. Cette configuration offre un équilibre optimal entre simplicité et flexibilité, permettant de créer des courbes lisses et précises, indispensables pour la conception précise des mailles complexes d'un stent. L'utilisation de ces courbes facilite la création de modèles qui doivent non seulement respecter des critères stricts de précision dimensionnelle, mais aussi imiter les propriétés mécaniques et flexibles du stent réel. En contrôlant la courbure et la forme des mailles à travers les points de contrôle des courbes de Bézier cubiques, il est possible de simuler avec précision la structure du stent, garantissant ainsi une conception à la fois efficace et fidèle à la réalité clinique.

2. Formulation mathématique

Les courbes de Bézier sont définies au moyen des polynômes de Bernstein, présentant un unique segment de courbe avec une continuité C. Pour $n \in N$ et $0 \leq k \leq n$, on appelle polynôme de Bernstein B_k^n le polynôme :

$$B_k^n(t) = \binom{n}{k} t^k (1-t)^{n-k} \quad (4)$$

Soient P_0, \dots, P_n des points d'un plan affine P . On appelle **courbe de Bézier** de points de contrôle P_0, \dots, P_n , la courbe paramétrée définie, pour $t \in [0, 1]$, par :

$$M(t) = \text{Bar}((P_0, B_0^n(t)), (P_1, B_1^n(t)), \dots, (P_n, B_n^n(t))) \quad (5)$$

Autrement dit, pour tout $t \in [0, 1]$, $M(t)$ est l'unique point de P tel que :

$$\forall A \in \mathcal{P}, \overrightarrow{AM}(t) = \sum_{k=0}^n B_k^n(t) \overrightarrow{AP}_k \quad (6)$$

La courbe de Bézier a les propriétés suivantes :

- elle est contenue dans l'enveloppe convexe des points P_0, \dots, P_n .
- elle a pour extrémité les points P_0 et P_n .
- le vecteur tangent à la courbe en P_0 est $\overrightarrow{P_0P_1}$, le vecteur tangent en P_n est $\overrightarrow{P_{n-1}P_n}$.

Lorsqu'un point de contrôle individuel est déplacé, les formes sont modifiées de manière intuitive, influençant de manière marginale les parties éloignées de la courbe (contrôle pseudo-local). Lorsque le nombre de points de contrôle, et par conséquent le degré de la courbe, est relativement élevé, l'influence d'un point de contrôle individuel devient proportionnellement plus faible, et par conséquent, des déplacements plus importants, souvent de points de contrôle éloignés, peuvent être nécessaires pour définir la forme souhaitée.

3.1.3 Courbes de Bézier cubique

Une courbe de Bézier cubique est un type particulier de courbe mathématique définie par un polynôme de troisième degré, caractérisée par quatre points de contrôle (P_0, P_1, P_2, P_3) . Ces points de contrôle influent sur la forme et la trajectoire de la courbe, avec les extrémités définissant les points de départ et d'arrivée, tandis que les deux points intermédiaires, également appelés "handles" ou "points de contrôle", jouent un rôle crucial dans la détermination de la courbure et de l'allure de la courbe.

L'intervalle de définition est $[a, b] = [0, 1]$. La courbe commence au point P_0 (pour $t = 0$) et finit au point P_3 (pour $t=1$). La dérivée au point P_0 est contrôlée par le point P_1 . Le vecteur dérivé en $t=0$ est :

$$D_0 = 3(P_1 - P_0) \quad (7)$$

La droite (P_0, P_1) est donc tangente à la courbe au point P_0 . De manière similaire, la dérivée au point P_3 est :

$$D_0 = 3(P_3 - P_2) \quad (8)$$

La figure suivante montre deux exemples de courbe de Bézier cubique :

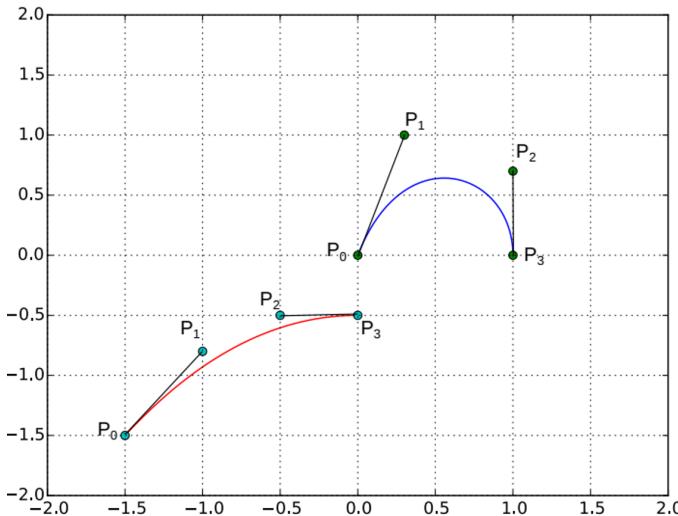


FIGURE 11 – Exemple de courbes de Bézier cubiques

Il est prouvé que la seule courbe polynomiale de troisième degré qui satisfait ces deux propriétés est :

$$P(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t) P_2 + t^3 P_3 \quad (9)$$

La formule ci-dessus permet de calculer les coordonnées du point P pour un paramètre t donné sur la courbe.

3.1.4 Différents types des images

1. **Image bitmap** : elles sont composées d'une grille de pixels individuels, chacun ayant sa propre couleur. La qualité d'une image bitmap dépend de sa résolution, qui est définie par le nombre total de pixels (par exemple, 1920x1080). Lorsqu'une image bitmap est agrandie, elle peut devenir floue ou pixelisée car les pixels individuels deviennent plus visibles. Ces images sont idéales pour capturer des détails complexes et une vaste gamme de couleurs, ce qui les rend parfaites pour les photographies et les images réalistes. Les formats courants incluent JPEG, PNG, GIF, BMP et TIFF. Pour la modification de ces images, des logiciels comme Adobe Photoshop ou GIMP sont généralement utilisés. Contrairement aux images vectorielles, les bitmaps ne sont pas bien adaptées aux redimensionnements importants, en particulier lorsqu'il s'agit d'agrandir une image.

2. **Image vectorielle** : elles sont définies par des chemins, qui sont des constructions mathématiques composées de points, de lignes et de courbes, associées à divers attributs de couleur. Leur principal avantage réside dans leur capacité à être redimensionnées sans aucune perte de qualité. Cela est dû au fait que les formes dans une image vectorielle sont calculées mathématiquement, permettant une mise à l'échelle propre et nette, quelle que soit la taille. Ces images sont parfaites pour des graphiques comme les logos, les schémas, les illustrations et les icônes. Les formats les plus courants comprennent SVG, EPS, AI et, dans certains cas, PDF. Les logiciels spécialisés tels qu'Adobe Illustrator ou Inkscape sont souvent utilisés pour la création et la modification d'images vectorielles. En outre, un fichier SVG (Scalable Vector Graphics) est un format de fichier basé sur XML utilisé pour décrire ces images vectorielles. Il est particulièrement apprécié pour sa capacité à définir des images qui peuvent être redimensionnées sans perte de qualité. Voici les éléments clés qu'on trouve généralement dans un fichier SVG :

- Balises XML : Comme SVG est basé sur XML, le fichier commence par une déclaration XML. Elle spécifie la version de XML et peut inclure d'autres attributs comme l'encodage.
- Élément racine `<svg>` : C'est l'élément racine qui définit le cadre de travail pour le reste du fichier SVG. Il inclut des attributs comme `width` (largeur), `height` (hauteur), et `viewBox` (qui détermine la zone de l'image à afficher).
- Formes de base : SVG utilise des balises pour différentes formes géométriques telles que `<circle>` (cercle), `<rect>` (rectangle), `<line>` (ligne), `<ellipse>` (ellipse), `<polygon>` (polygone), et `<polyline>` (ligne brisée).
- Chemins (`<path>`) : L'élément `<path>` est l'un des éléments les plus puissants en SVG. Il est utilisé pour définir des formes complexes à l'aide d'une série de commandes de mouvement et de dessin (comme "M" pour déplacer, "L" pour ligne, "C" pour courbe de Bézier cubique, etc.).
- Styles et Couleurs : Les couleurs et les styles (comme le remplissage, la bordure, etc.) sont définis à l'aide d'attributs comme `fill`, `stroke`, `stroke-width`. SVG permet également l'utilisation de styles CSS pour le stylisme.
- Texte : Le texte est ajouté avec l'élément `<text>`, qui peut être positionné et stylisé.
- Transformations : Les éléments SVG peuvent être transformés (tournés, déplacés, mis à l'échelle, etc.) en utilisant des attributs de transformation.
- Groupes (`<g>`) : Les éléments peuvent être regroupés ensemble à l'aide de l'élément `<g>`, permettant de leur appliquer des transformations ou des styles en tant que groupe.
- Images et Autres SVGs : Des images raster peuvent être intégrées dans un SVG avec `<image>`, et d'autres SVGs peuvent être imbriqués.
- Animations et Interactions : SVG supporte les animations (via SMIL ou

CSS) et peut interagir avec le JavaScript pour des comportements dynamiques.

- Métadonnées et Liens : Les éléments comme `<metadata>`, `<desc>`, et `<a>` permettent d'ajouter des métadonnées, des descriptions et des hyperliens.

Un fichier SVG peut être créé et édité avec des éditeurs graphiques vectoriels comme Adobe Illustrator ou Inkscape, et peut être visualisé dans la plupart des navigateurs web modernes. Sa nature basée sur XML le rend modifiable avec des éditeurs de texte classiques, offrant une grande flexibilité pour les développeurs et les designers.

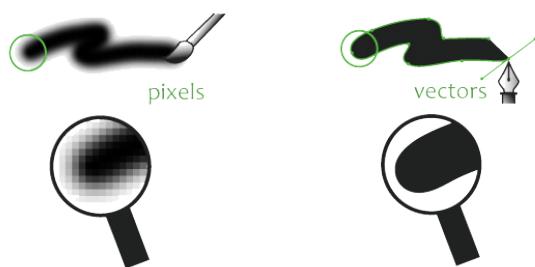


FIGURE 12 – Image Bitmap vs Image vectorielle

Nous notons comme exemple de ces fichiers le code suivant :

```

1 <svg width="100" height="100" xmlns="http://www.w3.org/2000/svg">
2   <circle cx="50" cy="50" r="40" stroke="black" stroke-width="4"
3     fill="gray" />
4   <text x="50" y="55" font-size="12" text-anchor="middle" fill=
5     white">Stent</text>
6   <path d="M20\u201440\u2014Q\u201450\u201420,\u201480\u201440" stroke="white" fill="gray"/>
7 </svg>

```

Listing 1 – Exemple du contenu d'un fichier SVG

Ce code résulte de l'image suivante :

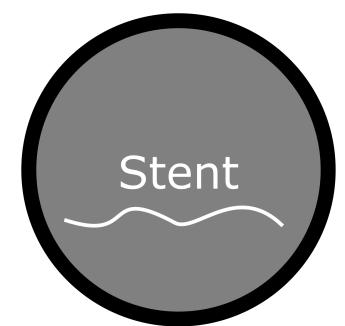


FIGURE 13 – Exemple d'une image矢量ielle résultante sur SVG

3.2 Outils

3.2.1 Inkscape

Inkscape, un logiciel de dessin vectoriel open-source, occupe une place prépondérante dans le monde de la conception graphique en offrant une palette d'outils avancés et polyvalents. Tout particulièrement, l'outil des Courbes de Bézier, représente l'un des atouts majeurs de ce logiciel. Cette fonctionnalité permet une modélisation précise et complexe des formes en utilisant des courbes mathématiquement définies. L'utilisation de l'outil des Courbes de Bézier dans Inkscape prend tout son sens dans la modélisation géométrique des stents. En effet, cette fonctionnalité permet de définir avec une grande précision les contours, les courbes et les dimensions des stents.

En pratique, les concepteurs de stents utilisent l'outil Courbe de Bézier pour définir les formes de mailles complexes. En ajustant les points de contrôle (nœuds) et en manipulant les courbes de Bézier.

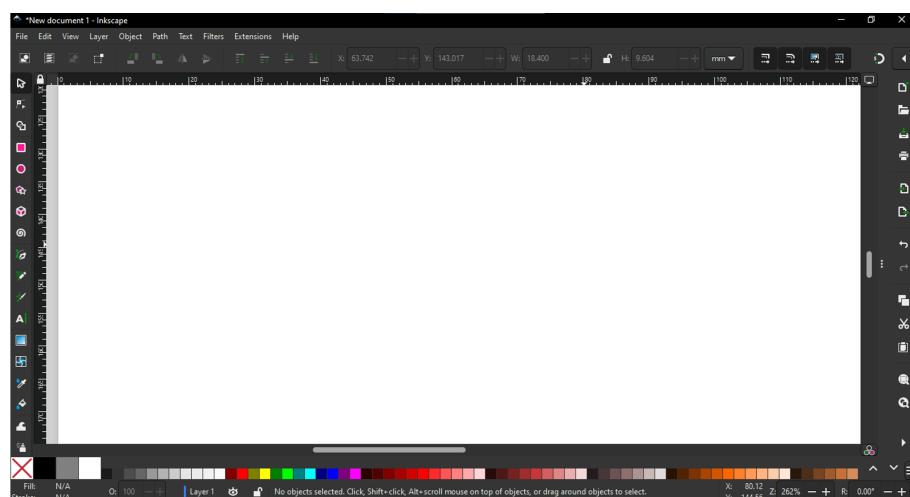


FIGURE 14 – Interface d'utilisateur d'InkScape

1. Ouverture de Inkscape et Sélection de l'Outil Courbe de Bézier :

Dans un premier temps, lançons l'application Inkscape et accédons à l'outil Courbe de Bézier situé sur le côté gauche de l'interface (également connu sous le nom de pen tool) (Figure 15).

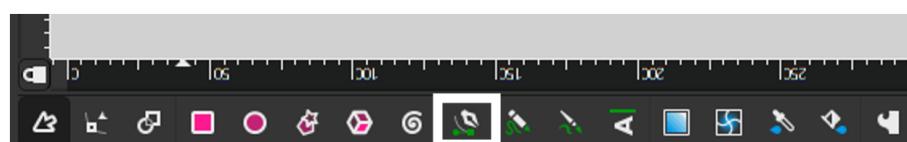


FIGURE 15 – Menu des options sur Inkscape

2. **Création de Formes en Définissant des Trajets** : Avec l'outil Courbe de Bézier, nous ne nous contentons pas de cliquer et dessiner, mais nous définissons des trajets. Pour illustrer, imaginons que nous souhaitons créer une forme géométrique spécifique. Nous procédons en déterminant les points de départ et d'arrivée des courbes de Bézier au-dessus de notre forme choisie (Figure 16).

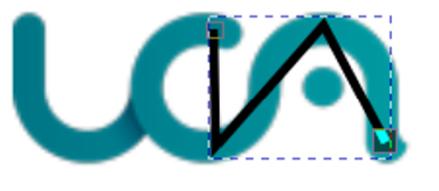


FIGURE 16 – Définition des trajets sur Inkscape

3. **Utilisation de l'Outil Node pour Affiner les Courbes** : Une fois les points définis, nous passons à l'outil *Node* pour ajuster avec précision les points de contrôle (tangentes) des courbes de Bézier (Figure 17). Cette étape vise à garantir que les courbes correspondent de manière optimale à la forme géométrique que nous avons déterminée.

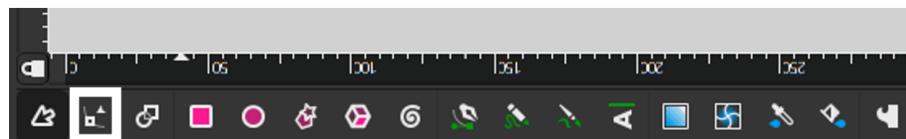


FIGURE 17 – Choix de l'outil "Node"

4. **Optimisation des points de contrôle (Tangentes)** : L'utilisation minutieuse de l'outil Node nous permet de travailler avec les contrôles points, ajustant les tangentes pour obtenir des courbes de Bézier aussi précises que possible, en parfaite adéquation avec la forme de maille que nous avons définie précédemment (Figure 18).



FIGURE 18 – Définition des courbes de bézier sur Inkscape

5. **Enregistrement du Résultat au Format SVG :** Une fois que les courbes ont été soigneusement ajustées, nous enregistrons le résultat sous forme de fichier SVG. Ce format, largement utilisé dans la création graphique vectorielle, permet de préserver la qualité et la flexibilité de notre création (Figure 19).



FIGURE 19 – Exemple d'une image vectorielle résultante sur InkScape

3.2.2 SVGPathTools

'svgpathtools' est une bibliothèque spécialement conçue pour manipuler et analyser les objets "Path" (chemin) des fichiers SVG ainsi que les courbes de Bézier. Cette bibliothèque offre une variété de fonctionnalités axées sur la géométrie pour transformer et examiner les éléments de chemin. Elle inclut également un sous-module nommé `bezier.py`, qui contient des outils dédiés aux courbes de Bézier d'ordre n, représentées comme des n-uplets. Nous notons ses fonctionnalités clés suivantes :

- **Lecture, écriture et affichage** de fichiers SVG contenant des éléments Path et autres éléments SVG.
- Conversion de segments de chemin de Bézier en objets polynomiaux numpy. `poly1d`.
- Conversion de polynômes (sous forme standard) en leur forme de Bézier.
- Calcul de **vecteurs tangents** et de **vecteurs normaux** (selon la règle de la main droite).
- Calcul de la **courbure**.
- Séparation des chemins discontinus en leurs sous-chemins continus.
- Calcul efficace des **intersections** entre chemins et/ou segments.
- Détermination de la **boîte englobante** pour un chemin ou un segment.
- **Inversion** de l'orientation d'un segment ou d'un chemin.
- **Découpage** et **division** de chemins et de segments.
- **Lissage** des chemins pour éliminer les irrégularités et rendre les chemins différentiables.
- **Cartes de transition** du domaine du chemin au domaine du segment et inversement (`T2t` et `t2T`).
- Calcul de la **surface** délimitée par un chemin fermé.
- Calcul de la **longueur d'arc**.
- Calcul de l'**inverse de la longueur d'arc**.

- Conversion des tuples de couleurs RGB en chaînes de couleurs hexadécimales et inversement.

Ces outils font de `svgpathtools` une ressource précieuse pour les tâches de modélisation et d'analyse géométrique, en particulier lorsqu'il s'agit de travailler avec des graphiques vectoriels et des courbes de Bézier dans des fichiers SVG.

3.2.3 Organisation du programme en Orienté objet

Pour organiser notre programme, nous avons utilisé le paradigme de programmation "Programmation orientée objet (POO)" qui présente comme avantages :

- **Organisation du Code** : La POO aide à structurer et organiser le code de manière logique, ce qui le rend plus lisible, maintenable et évolutif.
- **Réutilisation du Code** : L'héritage et la composition favorisent la réutilisation du code, ce qui réduit la redondance et améliore l'efficacité du développement.
- **Modularité** : En encapsulant le code, les objets deviennent des modules indépendants, ce qui facilite le débogage et le test.
- **Flexibilité et Scalabilité** : Le polymorphisme et l'abstraction permettent de construire des systèmes flexibles et évolutifs, capables de s'adapter aux changements avec un impact minimal sur le code existant.
- **Conception Intuitive** : La POO permet de modéliser des concepts complexes du monde réel de manière intuitive, rendant le processus de développement plus naturel et compréhensible.

3.3 Bases de données

Nous avons des schémas d'une partie de la représentation longitudinale des stents de quelques constructeurs sur lesquels nous nous sommes basés pendant notre travail (Figures 20 et 21).

Φ4.0mm	Ultimaster Nagomi	Ultimaster Tansei	Xience Skypoint	Resolute Onyx	Synergy	Orsiro
Stent Design						

FIGURE 20 – Schémas longitudinales de quelques modèles de stents

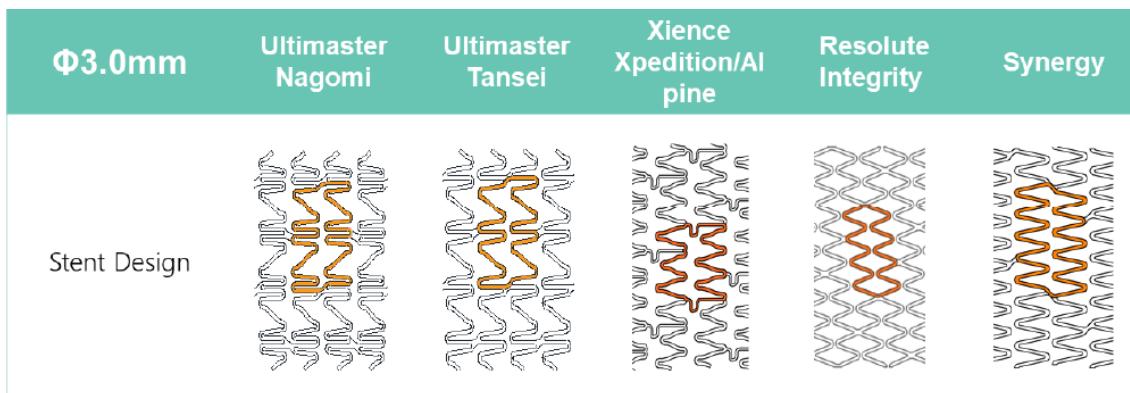


FIGURE 21 – Schémas longitudinales de quelques modèles de stents- partie 2

Dans ce qui suit, nous afficherons au fil des étapes les résultats obtenus sur le modèle "Synergy". Les résultats correspondant aux échantillonnages des autres modèles seront en annexes.

4 Implémentation de notre programme

4.1 Réalisation des schémas sur inkscape

La première phase de création de notre modèle géométrique des mailles et des connecteurs du stent consistait à élaborer une représentation vectorielle du dessin en question. À cette fin, nous avons employé le logiciel Inkscape, dont les fonctionnalités sont exhaustivement décrites dans le quatrième chapitre. Étant donné que cette tâche implique un dessin manuel sur le schéma (Figure 22), il est crucial d'atteindre un haut degré de précision.

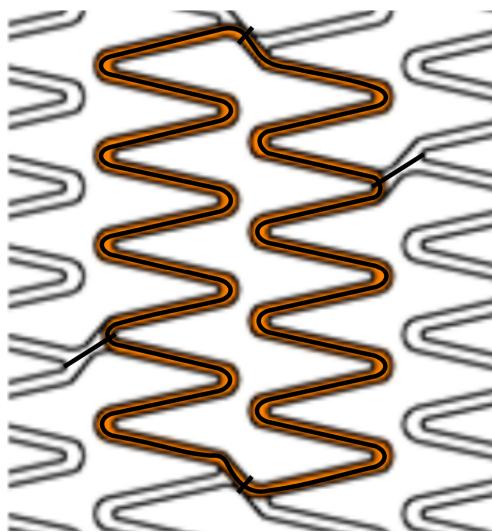


FIGURE 22 – Réalisation de schéma vectoriel du modèle "Synergy" sur Inkscape

Pour ce faire, nous avons privilégié l'utilisation des courbes de Bézier cubiques d'Inkscape, ce qui nous permet de déterminer avec exactitude les points de contrôle ainsi que les tangentes associées. Il est important de noter que même pour les segments qui semblent être des lignes droites, nous avons opté pour les représenter à l'aide de courbes de Bézier afin d'uniformiser leur manipulation ultérieure.

Ce processus aboutit à la création de chemins au format SVG (Figure 23), qui constituent les données d'entrée de notre script Python que nous analyserons plus avant dans les sections à venir. En règle générale, le résultat se compose de cinq chemins : un correspondant à la maille et quatre autres associés à chaque connecteur (gauche, droite, haut et bas). Ces derniers sont essentiels pour reconstituer l'intégralité de la forme du stent, car ils représentent les connexions entre les couronnes et les mailles.

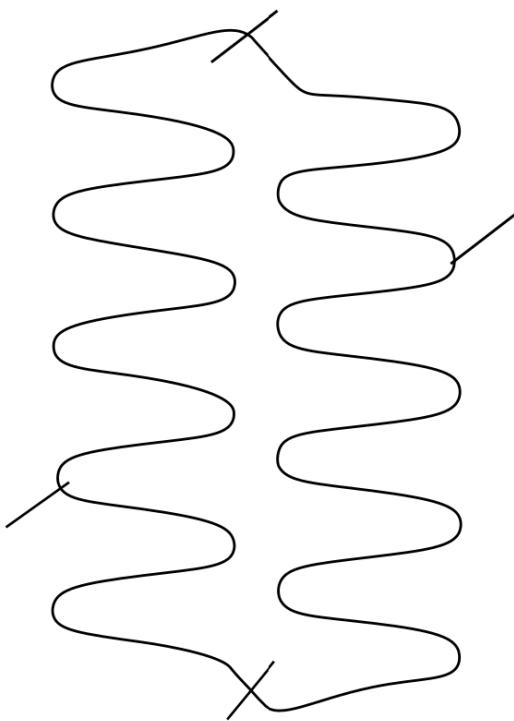


FIGURE 23 – Représentation sous forme d'image vectorielle du modèle "Synergy"

4.2 Lecture et manipulation des chemins SVG

4.2.1 Lecture des chemins SVG

Dans notre programme, nous avons conçu la classe *svgPathReader* pour lire et analyser les données vectorielles d'une image SVG représentant le schéma élémentaire d'un stent. Pour cela nous nous sommes servis de la librairie de python "svgPathtools". Nous présentons dans ce qui suit une reformulation des fonctions principales regroupées par leur finalité :

- **Initialisation et Préparation des Données :**

- La classe SvgPathReader est initialisée avec le chemin d'une image SVG, et les chemins ainsi que les attributs sont extraits en utilisant svg2paths.
- Plusieurs listes pour stocker différents aspects des chemins SVG (comme les segments de courbes de Bézier et leurs longueurs) sont préparées.
- Un dictionnaire est créé pour associer automatiquement à la sortie du programme des noms aux différentes parties du stent (telles que la maille et les connecteurs).

- **Analyse de la Forme Générale :**

La méthode '*set_shape_length*' calcule la longueur totale de la forme en sommant les longueurs des segments de courbe de Bézier. '*set_paths*' collecte tous les chemins non nuls, c'est-à-dire ceux qui contiennent des données. Cette méthode est une mesure préventive pour assurer que nous avons un input correcte.

— **Détail des Chemins et Segments :**

- '*set_segments_list*' rassemble les segments de courbes de Bézier de chaque chemin pour un traitement individuel.
- '*set_paths_list*' convertit chaque segment de courbe de Bézier en un format utilisable, comprenant les points de contrôle et les points terminaux de chacune des courbes.

— **Conversion et Manipulation Mathématique :**

- '*set_bezier_curves_to_poly_list*' convertit les courbes de Bézier en objets polynomiaux pour faciliter les calculs mathématiques ultérieurs.

— **Calcul des Longueurs et Détermination des Extrémités :**

- '*path_length*' est une fonction utilitaire pour mesurer la longueur d'un chemin individuel.
- '*set_paths_length_list*' génère une liste des longueurs de tous les chemins actifs.
- '*extremities_of_path*' calcule les valeurs extrêmes (xmin, xmax, ymin, ymax) pour chaque chemin, ce qui est utile pour comprendre la position et la dimension de chaque composant.

— **Séparation et Identification des Composants :**

- La méthode '*separate_components*' classe les chemins en identifiant le composant principal (probablement le maille) et les connecteurs basés sur leurs dimensions et leurs positions relatives. Elle utilise les longueurs et les extrêmes des chemins pour distinguer les différents composants et les associe aux noms pertinents dans le dictionnaire.

4.2.2 Manipulation des chemins SVG

Le code de la classe ‘**SvgPathOperator**’ travaille de pair avec la classe ‘**SvgPathReader**’ pour manipuler et analyser les chemins dans une image SVG représentant un stent. Voici un résumé des principales fonctions et opérations réalisées par cette classe :

— **Initialisation et Configuration :**

- L'initialisation de ‘**SvgPathOperator**’ reçoit une instance de ‘**SvgPathReader**’ et exécute plusieurs méthodes de ‘**SvgPathReader**’ pour préparer les

données. Cela inclut la définition des chemins, des segments, des courbes de Bézier, des longueurs des chemins, de la longueur totale de la forme et la séparation des composants du stent.

— **Détection des Intersections :**

- ‘*set_maille_connectors_intersections*’ identifie et stocke les points d’intersection entre la maille (le composant principal du stent) et chacun des connecteurs.
- ‘*set_maille_connectors_seg_intersections*’ enregistre les segments de la maille où se produisent des intersections avec les connecteurs.
- ‘*set_maille_connectors_t_intersections*’ compile les paramètres spécifiques (‘*t*’) des courbes de Bézier à ces points d’intersection.

— **Calcul de la Courbure :**

- ‘*calculate_curvature_at_one_point*’ calcule la courbure d’un point sur une courbe de Bézier à un paramètre ‘*t*’ donné, en utilisant la première et la seconde dérivée de la courbe.
- Pour une courbe paramétrée définie par $x = g(t)$ et $y = h(t)$, la formule pour calculer la courbure κ en un point est donnée par :

$$\kappa = \frac{|g'(t)h''(t) - h'(t)g''(t)|}{([g'(t)]^2 + [h'(t)]^2)^{\frac{3}{2}}} \quad (10)$$

‘*calculate_curvature_of_curve*’ détermine la courbure totale d’une courbe de Bézier en calculant la courbure en plusieurs points le long de la courbe.

- ‘*set_max_curvature*’ et ‘*set_min_curvature*’ trouvent respectivement la courbure maximale et minimale parmi une série de courbes de Bézier en utilisant un nombre défini de points pour le calcul.

— - **Compilation des Courbures :**

- ‘*set_curvatures_list*’ crée une liste des courbures pour chaque courbe de Bézier dans les données du stent, ce qui permet une analyse plus approfondie de la forme du stent.
- ‘*set_total_curvature*’ additionne toutes les courbures des courbes pour obtenir une mesure de la courbure totale de la structure du stent.

Le but de cette classe est d’analyser la géométrie du stent en termes d’intersections et de courbure, ce qui pourrait être utile pour comprendre le comportement mécanique du stent, optimiser sa conception ou préparer des simulations.

4.3 Echantillonnage

Dans cette section, nous explorons en détail le processus d’échantillonnage utilisé pour numériser notre maillage de stent et ses connecteurs. Nous partons d’images

SVG créées avec Inkscape, que nous pouvons ensuite importer et traiter en Python. L'objectif est de ré-échantillonner ces images, qui se composent initialement de courbes de Bézier, en les convertissant en segments de droite. Ces segments de droite faciliteront les manipulations ultérieures dans nos analyses.

Nous adoptons deux approches distinctes pour l'échantillonnage :

1. **Échantillonnage Équidistant** : Cette technique consiste à diviser chaque courbe en points espacés uniformément le long de son parcours. L'avantage de cette méthode est qu'elle garantit une distribution homogène des points, ce qui peut être particulièrement utile pour des analyses où la distance constante entre les points est essentielle. Cependant, elle peut ne pas capturer avec précision les subtilités dans les régions de haute courbure.
2. **Échantillonnage Basé sur la Courbure** : Contrairement à l'échantillonnage équidistant, cette méthode place plus de points dans les zones où la courbure de la courbe est plus grande. Cela permet une représentation plus fidèle des changements de direction et des détails fins du maillage, ce qui est crucial pour les analyses structurelles ou dynamiques où les détails de la géométrie ont un impact significatif sur les résultats.

4.3.1 Echantillonnage équidistant

La classe ‘EquidistantSampler‘ a pour but d’effectuer un échantillonnage équidistant des chemins dans une image SVG, typiquement ceux représentant un stent. Nous détaillons en ce qui suit les fonctions principales et le processus d’échantillonnage en question :

- **Initialisation :**

La classe ‘EquidistantSampler‘ est initialisée avec un objet ‘SvgPathOperator‘. Pendant l’initialisation, elle exécute des méthodes de ‘SvgPathOperator‘ pour déterminer les intersections entre la maille principale du stent et ses connecteurs.

- **Échantillonnage Équidistant :**

- La méthode ‘*my_equidistant_sampling*‘ effectue un échantillonnage équidistant sur les segments de la maille du stent.
- Pour déterminer les points échantillonnes, la méthode calcule d’abord la distance uniforme entre les points basée sur la longueur totale de la forme et le nombre souhaité d’échantillons.
- Ensuite, elle parcourt la liste des segments du stent, ajoutant des points équidistants à la liste ‘sampled_points‘.

- **Gestion des Intersections :**

- Des conditions spéciales sont vérifiées pour s’assurer que lorsqu’un segment

contient un point d'intersection avec un connecteur, ce point est correctement inclus dans l'échantillonnage.

- Des corrections sont appliquées pour ajuster les points d'intersection et s'assurer qu'ils sont positionnés de manière appropriée par rapport à la structure globale du stent. (Nous détaillerons dans la section suivante quelques problèmes que nous avons dû corriger pour avoir un échantillonnage correcte).

— **Résultat (Figure 24) :**

- La méthode ‘*my_equidistant_sampling*’ retourne une liste de points qui sont répartis de manière équidistante le long de la forme du stent, en tenant compte des points d'intersection avec les connecteurs.

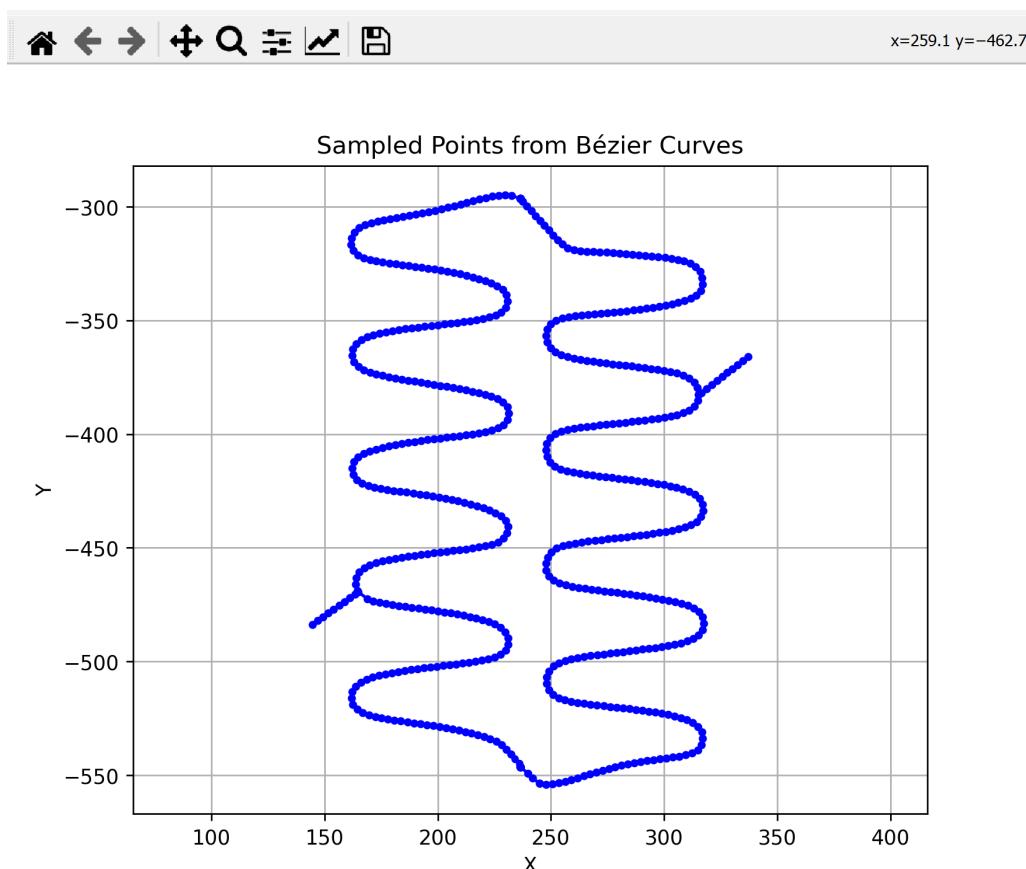


FIGURE 24 – Échantillonnage équidistant du modèle Synergy

Ce processus d'échantillonnage est important pour créer une représentation discrète de la forme du stent qui pourrait être utilisée pour des analyses numériques, telles que les simulations de mécanique des fluides ou les calculs de résistance des matériaux.

4.3.2 Echantillonnage basé sur la courbure

Dans cette partie, nous allons présenter une deuxième méthode d'échantillonnage basée sur la courbure des schémas. Cette méthode permet une représentation plus fidèle des caractéristiques géométriques complexes du stent, car les régions avec des courbures élevées (qui sont souvent des régions de contraintes mécaniques importantes) sont échantillonnées avec plus de points. Cela peut améliorer la précision des analyses mécaniques ou des simulations, car ces régions de haute courbure peuvent être critiques pour le comportement du stent une fois implanté. Dans notre programme, nous avons créé la classe '*CurvatureSampler*' pour implémenter ce nouvel échantillonnage.

— **Initialisation :**

- La classe 'CurvatureSampler' est initialisée avec un objet 'SvgPathOperator'. Elle invoque des méthodes de 'SvgPathOperator' pour préparer les données relatives à la courbure, ainsi que pour identifier les intersections entre la maille principale et les connecteurs.

— **Échantillonnage Basé sur la Courbure :**

- La méthode '*curvature_based_sampling*' échantillonne des points sur les courbes de Bézier basées sur leur courbure relative.
- Elle calcule le nombre de points d'échantillonnage pour chaque courbe proportionnellement à la part de la courbure totale de la courbe dans la courbure totale de tous les chemins.

— **Insertion des Points d'Intersection :**

- Lors de l'échantillonnage des segments de la maille qui contiennent des intersections avec les connecteurs, la méthode insère des points d'échantillonnage aux paramètres ' t ' correspondant à ces intersections, pour garantir que ces points critiques sont inclus.

— **Calcul et Distribution des Points :**

- Les valeurs de ' t ' sont distribuées de manière linéaire sur chaque segment, puis converties en points sur les courbes de Bézier.

— **Résultat (Figure 25) :**

- La méthode '*curvature_based_sampling*' retourne une liste de points échantillonnés qui reflètent la courbure des chemins du stent, fournissant ainsi une description détaillée pour des analyses ultérieures.

En somme, l'échantillonnage basé sur la courbure est un moyen raffiné de préparer des données pour des modèles qui nécessitent une résolution plus élevée dans les régions de changement géométrique significatif, ce qui est souvent lié à des caractéristiques physiques ou mécaniques importantes.

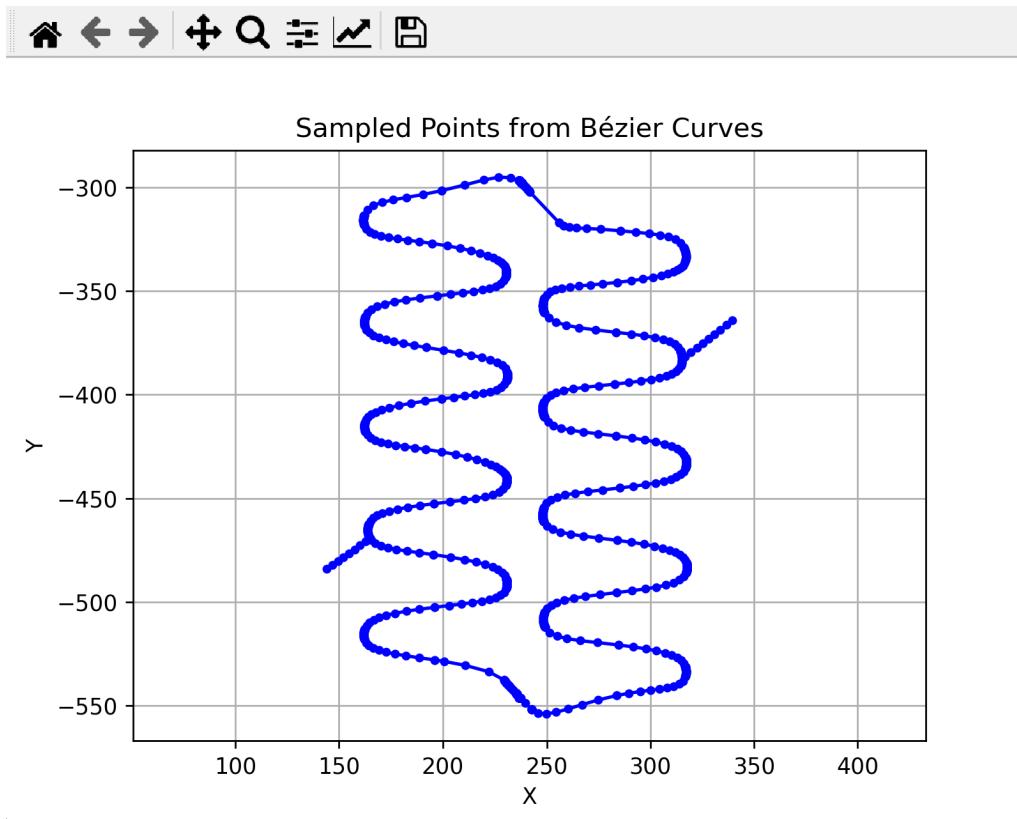


FIGURE 25 – Échantillonnage basé sur la courbure du modèle Synergy

4.3.3 Correction des connexions des connecteurs et la maille

Au cours du développement de notre programme, nous avons rencontré un problème concernant l'exactitude des points de jonction entre les connecteurs et la maille du stent. L'échantillonnage a produit deux types d'erreurs distinctes :

1. Dans le premier cas, illustré par la partie gauche de la figure 26, l'échantillonnage ne produisait aucun point d'intersection entre le connecteur et la maille, ce qui résultait en une absence totale de liaison entre ces deux éléments essentiels du stent.
2. Dans le second cas, bien qu'il y ait eu une intersection entre le connecteur et la maille, l'échantillonnage créait un chevauchement incorrect où le connecteur pénétrait dans la maille au lieu de s'y joindre de manière propre et précise ce qui est illustré dans la figure 26.

Ces imprécisions sont problématiques car les points de jonction sont cruciaux pour la reconstruction fidèle de la structure complète du stent. Afin de remédier à ces problèmes et pour garantir la capture des points d'intersection lors de l'échantillonnage, nous avons intentionnellement conçu le schéma initial de sorte qu'un connecteur traverse la maille. Cette approche nous assure d'obtenir une intersection

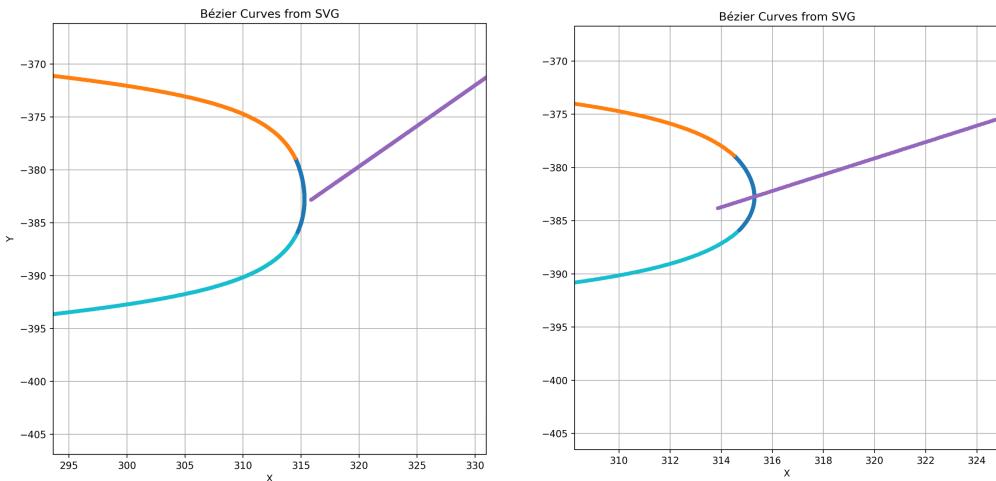


FIGURE 26 – Erreurs d’interesection du connecteur avec la maille

pendant le processus d’échantillonnage. Une fois cette intersection obtenue, nous procédons à la définition précise des coordonnées du point d’intersection. La dernière phase de notre processus consiste à éliminer la portion du connecteur qui s’étend indûment à l’intérieur de la maille, rectifiant ainsi toute superposition incorrecte et assurant que le modèle final du stent reflète une connexion exacte et propre entre la maille et ses connecteurs.

Suite à la mise en place des ajustements nécessaires, nous avons réussi à obtenir des points d’intersection précis entre le connecteur et la maille. Cela a permis de corriger les erreurs précédentes où le connecteur pénétrait incorrectement dans la maille ou manquait d’intersection.

En analysant la figure 27 qui montre l’échantillonnage après correction :

1. **Image avec les Points Échantillonnés (Non-Zoomée)** : On observe que les points bleus, représentant les échantillons prélevés des courbes de Bézier, suivent maintenant correctement le tracé des courbes sans débordements ni interruptions. Les points d’intersection semblent alignés de manière à refléter une jonction naturelle entre la maille et le connecteur.
2. **Image avec les Points Échantillonnés (Zoomée)** : En se concentrant davantage sur la zone d’intersection, il est clair que les corrections ont amélioré la précision de l’échantillonnage. Les points sont disposés de façon à former une transition fluide, ce qui est essentiel pour une modélisation précise du stent.

Ces ajustements sont cruciaux car un alignement précis des points d’intersection est nécessaire pour la reconstruction fidèle du stent dans les simulations de dynamique des fluides ou les analyses structurelles. La correction empêche les anomalies dans la modélisation qui pourraient entraîner des erreurs dans l’évaluation de la performance ou de la sécurité du stent.

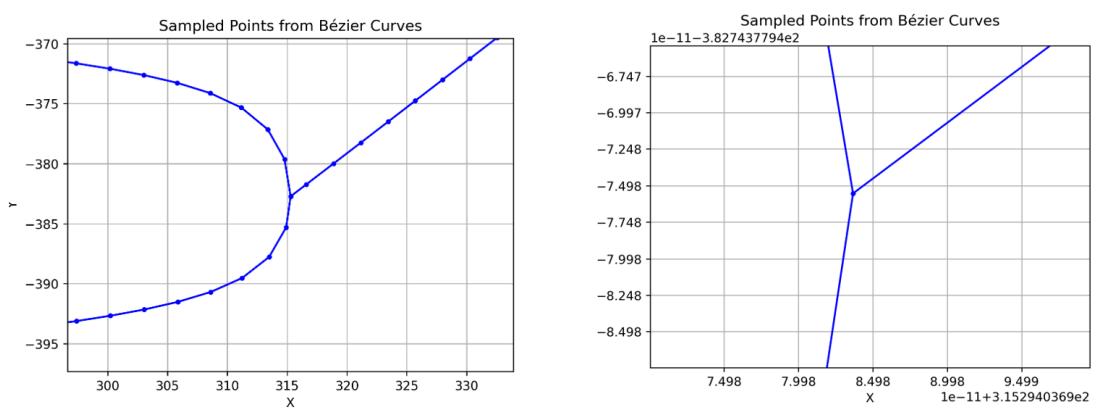


FIGURE 27 – Correction d’intersection de la maille avec un connecteur

4.4 Intérface d’utilisateur

Afin de rendre l’utilisation de notre programme utilisable par une grande majorité des utilisateurs, nous avons implémenté une interface d’utilisateur simple (Figure 28) qui permet les actions suivantes :

- **Bouton de Sélection de Fichier :**
- "Select a SVG file" : Un bouton permettant à l’utilisateur de sélectionner un fichier SVG à partir de son système de fichiers.
- **Menu Déroulant pour Sélectionner le Type d’Échantillonnage :**
- "Select Sampler" : Un menu déroulant où l’utilisateur peut choisir le type d’échantillonneur à utiliser. Les options visibles sont "Equidistant Sampler" et "Curvature Sampler", ce qui suggère que l’application peut échantillonner les points du SVG de façon équidistante ou basée sur la courbure du stent.
- **"Champ de Saisie Numérique" :**
- "Choose the total number of points" : Une zone de texte où l’utilisateur peut entrer le nombre total de points d’échantillonnage souhaités.
- **Bouton de Lancement :** - "Launch Program" : Un bouton que l’utilisateur peut cliquer pour démarrer le processus d’échantillonnage après avoir sélectionné le fichier SVG et le type d’échantillonneur, et saisi le nombre de points.

L’interface est simple et conçue pour une tâche spécifique avec une interaction utilisateur minimale, ce qui la rend facile à utiliser. L’application sert d’outil pour les utilisateurs qui ont besoin de créer des échantillons de points à partir de la géométrie d’un fichier SVG pour une analyse ultérieure.

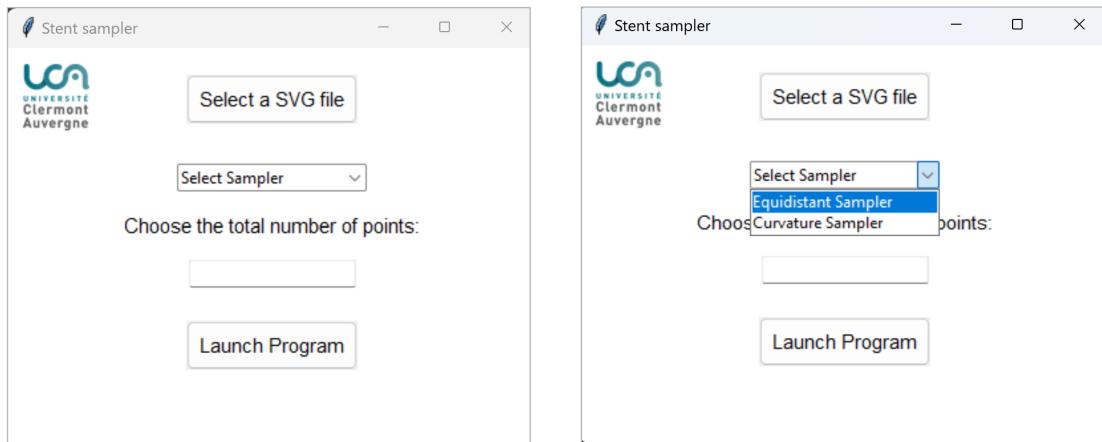


FIGURE 28 – Interface d’utilisateur

4.5 Visualisation des schémas et Fichiers CSV comme livrables

La classe ‘Visualiser’ a pour but de visualiser et enregistrer les points échantillonnés à partir des courbes de Bézier. La classe peut travailler avec des échantillons basés sur l’équidistance ou sur la courbure, en utilisant respectivement ‘EquidistantSampler’ ou ‘CurvatureSampler’. Voici les principales fonctions du code :

- **Initialisation :**

- Lorsqu’une instance de ‘Visualiser’ est créée, elle est initialisée avec un objet sampler, qui peut être soit un ‘EquidistantSampler’ soit un ‘CurvatureSampler’.

- **Correction de la Forme :**

- La méthode ‘correct_shape’ ajuste les coordonnées des points échantillonnés pour s’assurer qu’ils respectent la géométrie du stent, notamment en s’alignant correctement avec les points d’intersection.

- **Enregistrement des Points dans un Fichier CSV :**

- ‘points_in_csv_file’ enregistre les coordonnées des points échantillonnés dans un fichier CSV (Figure 30) qui a comme nom le format présenté dans la figure 29. Cela permet une analyse ultérieure ou une utilisation dans d’autres applications.

- **Visualisation :**

- La méthode ‘plot_shape’ trace les points échantillonnés sur un graphique. Elle distingue entre les échantillonnages équidistants et basés sur la courbure et applique la méthode de correction de forme appropriée.

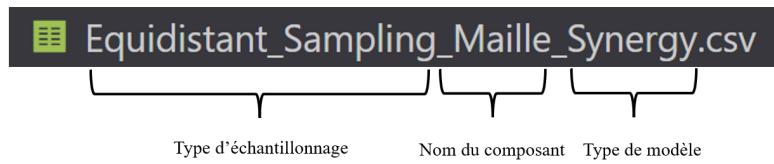


FIGURE 29 – Format du nom du fichier CSV résultant

	X,Y
1	245.12675402190197, -553.5060802091676
2	247.9375780463697, -554.0551554549506
3	250.80665886366702, -553.9344388563326
4	253.63931960319726, -553.447962719718
5	256.43245562904764, -552.7674121192575
6	259.19891693168734, -551.9846660681599
7	261.95094487155694, -551.1523727957186
8	264.69781166445347, -550.3031514453812
9	267.4461173486197, -549.4586063039351
10	270.2004123398807, -548.6338306364438
11	272.963722388141, -547.8398143895947
12	275.7379367936624, -547.0848113406155
13	278.5240886455115, -546.3751566386327
14	281.322558814108, -545.7157780508828
15	284.1330124209611, -545.1094934868484
16	286.9544552799113, -544.5566824950996
17	289.7866629274372, -544.061916344295
18	292.62745193988417, -543.6189970498385

FIGURE 30 – Représentation des points échantillonnés dans un fichier CSV

- Après la correction, elle enregistre les points dans un fichier CSV et les trace sur un graphique en utilisant Matplotlib.
- Les points échantillonnés sont affichés avec des marqueurs sur le graphique, et des informations telles que les axes et le titre sont ajoutées pour contextualiser le graphique.

— Affichage :

- La méthode ‘plt.show()’ est appelée à la fin pour afficher le graphique des points échantillonnés. Le graphique inclut une grille et des étiquettes pour une meilleure lisibilité.

En résumé, ‘Visualiser’ est une classe utilitaire pour l’analyse et la visualisation des modèles de stents. Elle fournit un moyen de vérifier visuellement l’exactitude des échantillons de points et de sauvegarder les résultats pour des usages futurs.

4.6 Limitations

Dans le contexte de notre projet visant à créer des modèles géométriques pour les mailles et les connecteurs des stents, nous sommes actuellement confrontés à la contrainte suivante : l'incapacité de définir précisément les connecteurs supérieurs et inférieurs. Cette limitation découle de notre approche qui repose sur l'utilisation de points d'intersection plutôt que sur des courbes, réduisant ainsi la précision des modèles de stents que nous pouvons générer. Jusqu'à présent, nous avons réussi à identifier uniquement des points d'intersection (Figure 31), laissant en suspens la tâche de déterminer les sections exactes où les mailles se superposent en haut et en bas.

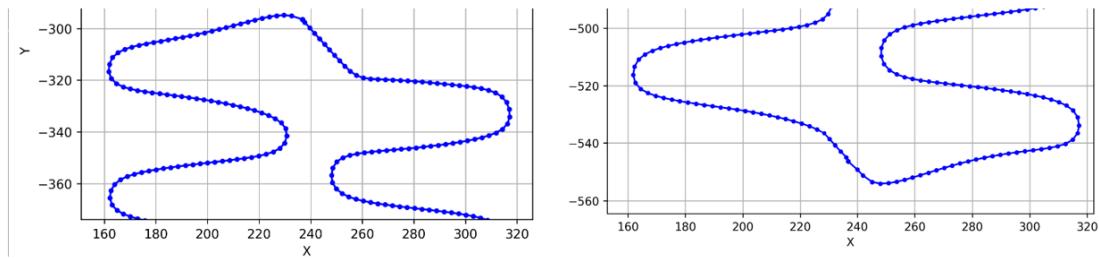


FIGURE 31 – Limitations des connecteurs haut et bas

Par ailleurs, une perspective d'amélioration réside dans la généralisation de la définition des symétries possibles pour chaque maille, ce qui contribuerait à affiner l'exactitude de nos modélisations géométriques.

5 Conclusion et perspectives

En conclusion, notre travail au sein du projet VSTENT a abouti au développement de deux méthodes d'échantillonnage pour la représentation géométrique des stents métalliques. La méthode 'my_equidistant_sampling' offre une approche systématique, générant une liste de points équidistants le long de la forme du stent, tout en prenant en compte les points d'intersection avec les connecteurs. D'un autre côté, la méthode 'curvature_based_sampling' propose une approche plus sophistiquée, fournissant une liste de points échantillonnés qui reflètent la courbure des chemins du stent.

Cette dernière méthode, axée sur la courbure, offre une description détaillée permettant des analyses approfondies. Elle se révèle particulièrement précieuse pour les modèles nécessitant une résolution plus élevée dans les régions de changement géométrique significatif, souvent lié à des caractéristiques physiques ou mécaniques importantes.

Ainsi, notre travail ouvre de nouvelles perspectives pour la modélisation précise des stents, contribuant à l'amélioration continue des pratiques médicales dans le domaine des interventions cardiovasculaires. Les méthodes d'échantillonnage que nous avons développées se positionnent comme un pont essentiel entre la complexité anatomique et la représentation numérique, promettant des applications pratiques et des avancées significatives dans la visualisation des dispositifs médicaux cardiovasculaires.

Annexes

Modélisation géométrique des autres modèles de notre base de donnée

A - Ultimaster Nagomi

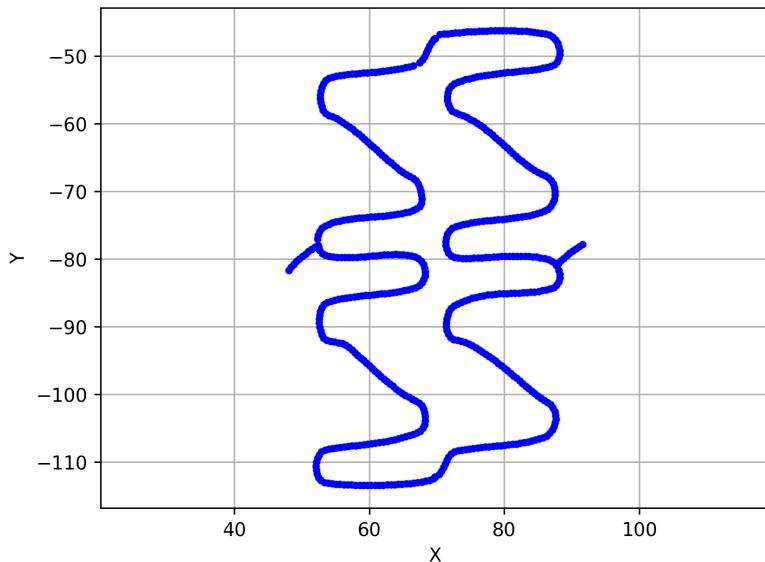


FIGURE 32 – Echantillonnage équidistant du modèle Ultimaster Nagomi

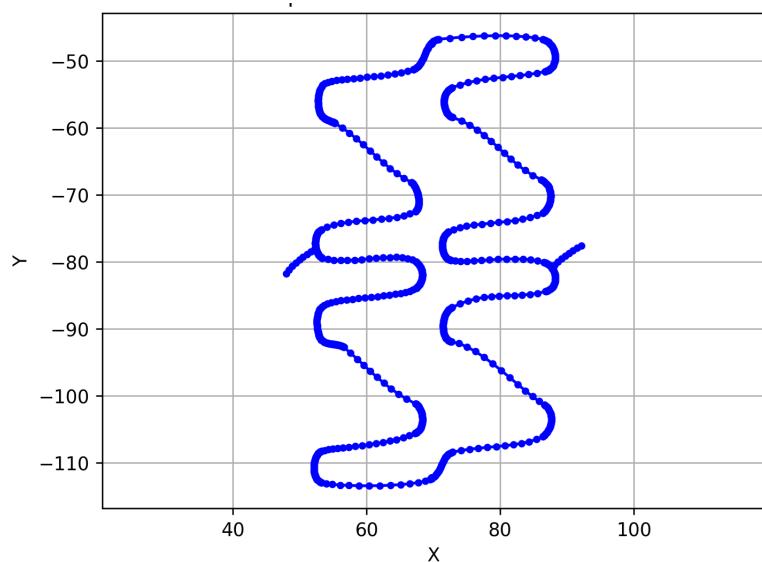


FIGURE 33 – Echantillonnage basé sur la courbure du modèle Ultimaster Nagomi

B - Ultimaster Tansei

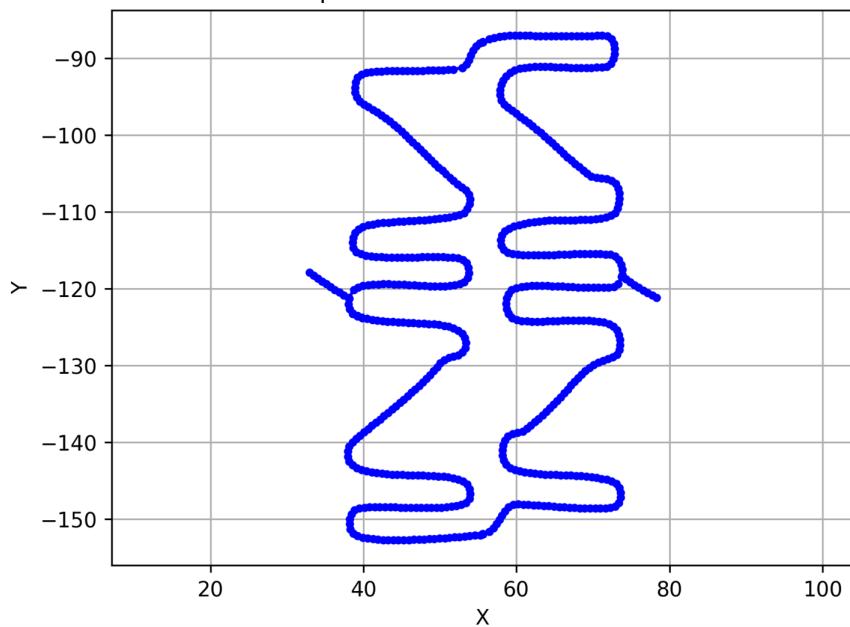


FIGURE 34 – Echantillonnage équidistant du modèle Utlimaster Tansei

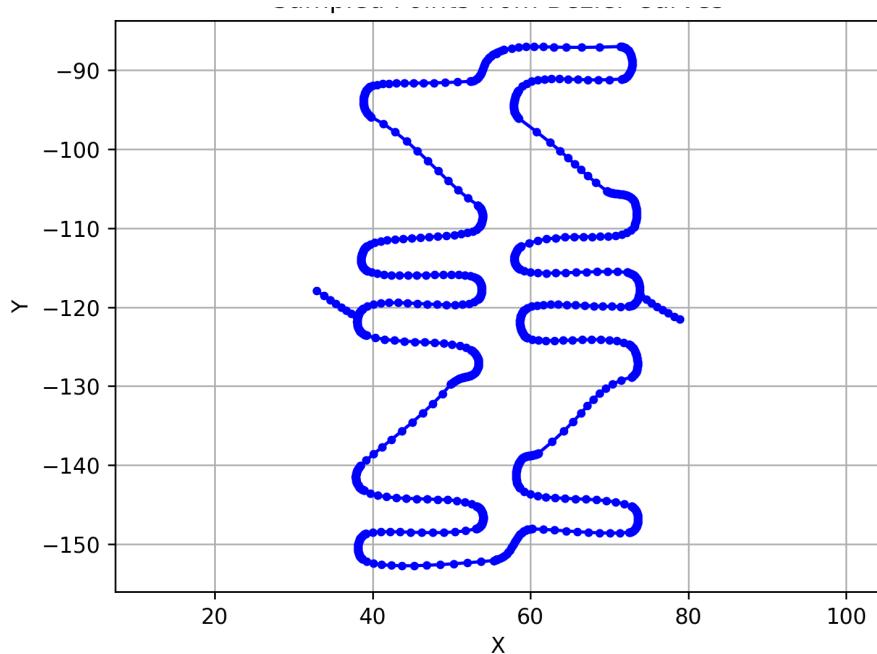


FIGURE 35 – Echantillonnage basé sur la courbure du modèle Utlimaster Nagomi

C - Xience Skypoint

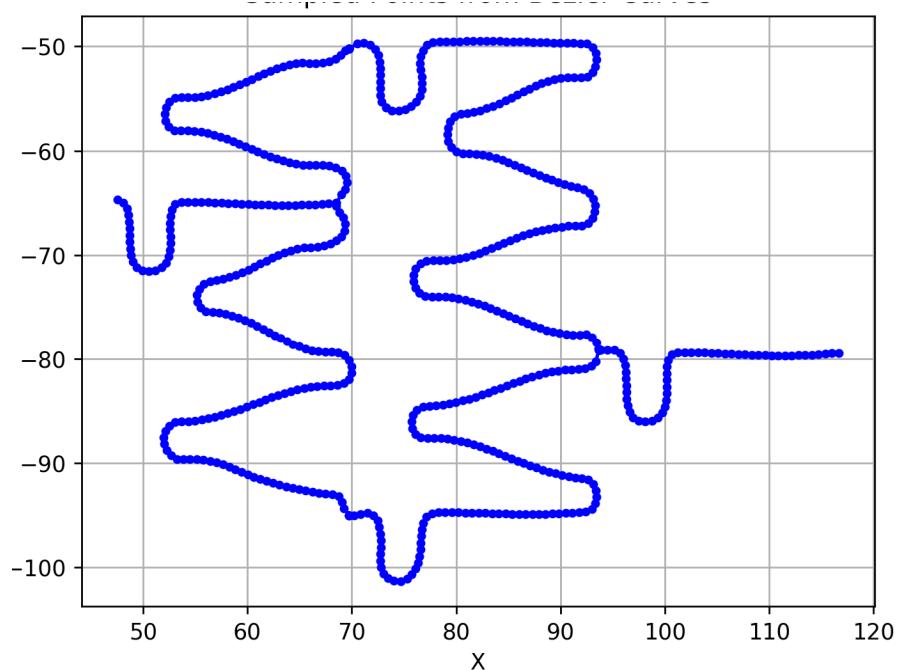


FIGURE 36 – Echantillonnage équidistant du modèle Xience Skypoint

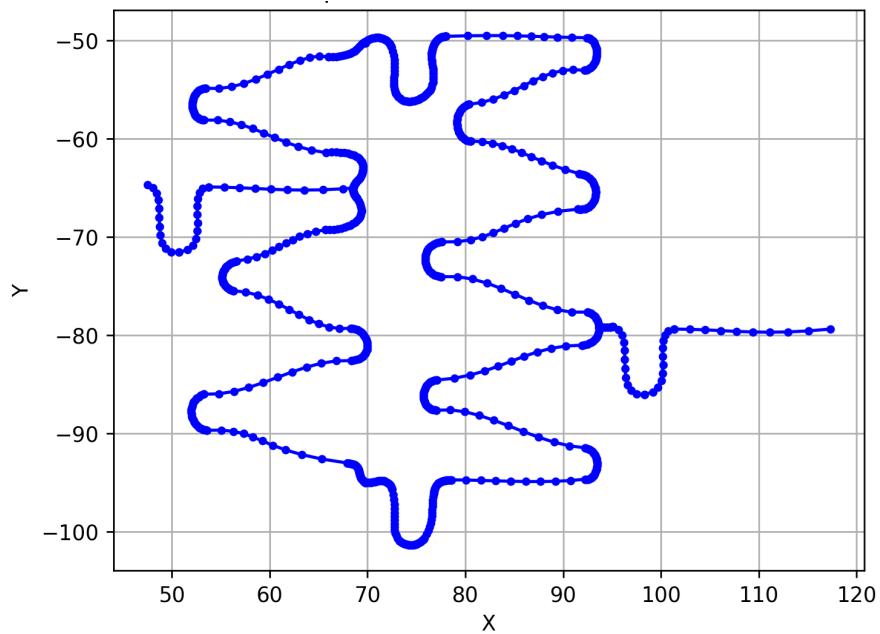


FIGURE 37 – Echantillonnage basé sur la courbure du modèle Xience Skypoint

C - Xience Xpedition/Alpine

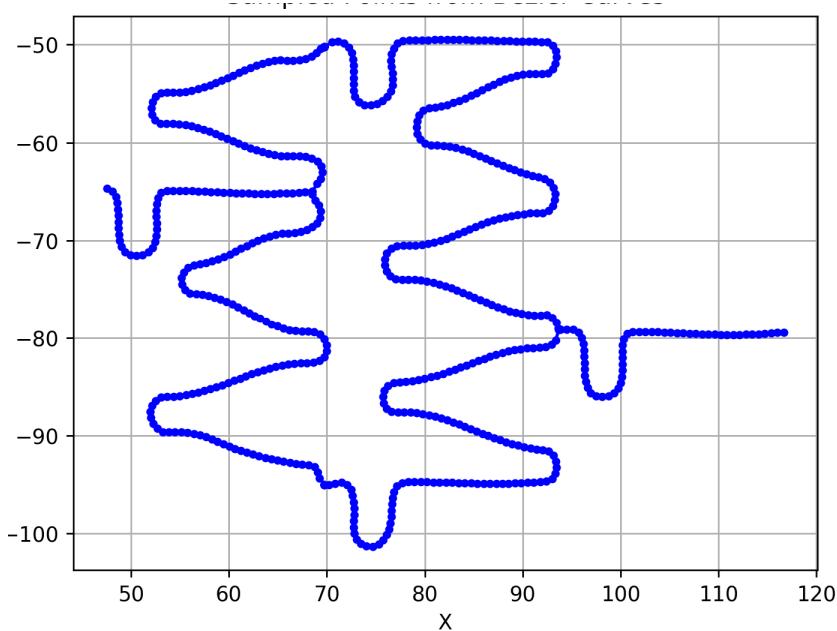


FIGURE 38 – Echantillonnage équidistant du modèle Xience Xpedition/Alpine

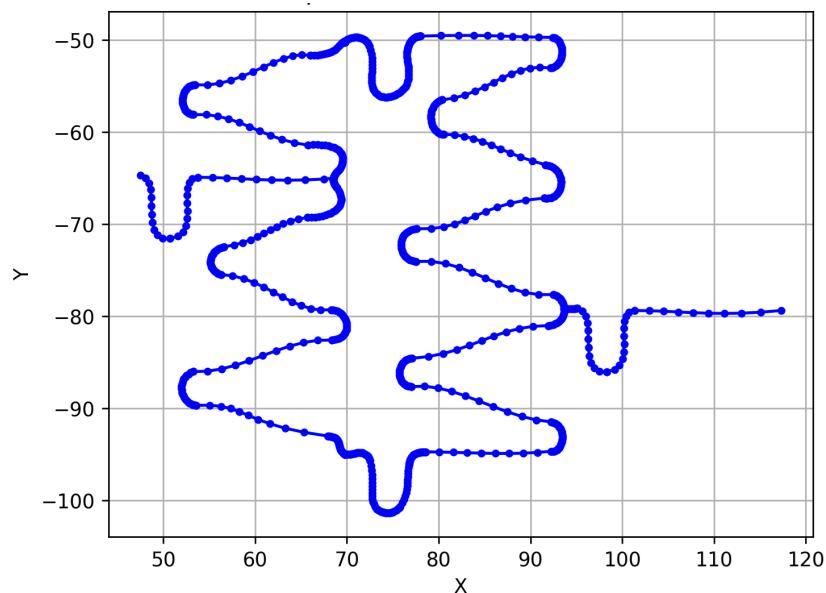


FIGURE 39 – Echantillonnage basé sur la courbure du modèle Xience Xpedition/Alpine

Références

- [1] Abdul Waheed Manati. Mort subite d'origine cardiaque à la phase aigüe de l'infarctus du myocarde : physiopathologie des troubles du rythmes ventriculaire, 2018.
- [2] Thomas N. James. *Anatomy of the coronary arteries in health and disease*. Circulation, 1965.
- [3] Birju Shah Daniel Beckmann Annie Chen Luan Ngo Ferdinand R. Flores Yanlin Gao et Matthew J. Budoff Song Shou Mao, Nasir Ahmadi. Normal thoracic aorta diameter on cardiac computed tomography in healthy asymptomatic adults : Impact of age and gender. *Academic Radiology*, 2008.
- [4] Bruce Blaus. The structure of an artery wall, 2024.
- [5] Bernhard Meier Morton J. Kern. Evaluation of the culprit plaque and the physiological significance of coronary atherosclerotic narrowings. *Circulation*, 2001.
- [6] Grant N. Pierce Andrew A. Francis. An integrated approach for the mechanisms responsible for atherosclerotic plaque regression. *Experimental and clinical cardiology*, 2011.
- [7] Rene G. Favaloro. Saphenous vein autograft replacement of severe segmental coronary artery occlusion : Operative technique. *The Annals of Thoracic Surgery*, 1968.
- [8] Samuel M. Ohayon Avi Shimony et Mark J. Eisenberg Jonathan Afilalo, Mandana Rasti. Off-pump vs. on-pump coronary artery bypass surgery : an updated meta-analysis and metaregression of randomized trials. *European Heart Journal*, 2011.
- [9] A.F Rickards J.R Hampton S.B King C.W Hamm J Puel W Hueb J-J Goy et A Rodriguez S.J Pocock, R.A Henderson. Meta-analysis of randomised trials comparing coronary angioplasty with bypass surgery. *The Lancet*, 1995.
- [10] Kathryn McDonald Martial G. Bourassa et Mark A. Hlatky Ida Sim, Munish Gupta. A meta-analysis of randomized trials comparing coronary artery bypass grafting with percutaneous transluminal coronary angioplasty in multi-vessel coronary artery disease. *The American Journal of Cardiology*, 1995.
- [11] Yafeng Hamm Jiping Lu Chen Pan. Structural design of vascular stents : A review. *MDPI*, 2021.

- [12] Giorgio Iacopo Baldereschi Carlo Di Mario Benedetta Tomberli, Alessio Mattesini. A brief history of coronary artery stents. *Revista Espanola de Cardiologia*, 2017.
- [13] Rapport HAS 2018. Endoprothèses (stents) coronaires, 2024.
- [14] Ahlsson A Alfonso F Banning AP Benedetto U Byrne RA Collet JP-Falk V Head SJ Jüni P Kastrati A Koller A Kristensen SD Niebauer J Richter DJ Seferovic PM Sibbing D Stefanini GG Windecker S Yadav R Zembala MO Neumann FJ, Sousa-Uva M. Guidelines on myocardial revascularization. *ESC Scientific Document Group.*, 2019.
- [15] É. Durand. *Stents coronaires biodégradables : un tournant thérapeutique*. 2015.
- [16] M. ; Gershlick A.H. ; Wijns W. ; Garcia E. ; Schampaert E. ; Breithardt G. Schofer, J. ; Schlüter. Sirolimus-eluting stents for treatment of patients with long atherosclerotic lesions in small coronary arteries : Double-blind, randomised controlled trial (e-sirius). *Lancet*, 2003.
- [17] M.B. ; Popma J.J. ; Fitzgerald P.G. ; Kuntz R.E. Moses, J.W. ; Leon. Sirolimus-eluting stents versus standard stents in patients with stenosis in a native coronary artery. *N. Engl. J. Med.*, 2003.