

Projet Maths

O. BRIODEAU, C. GOUMENT, A.S. MILEA, B. PALACHEV, D. TOOMEY

Décembre 2016

L'INTERPOLATION PAR SPLINE

A l'attention de M.Gleyse
2016-2017

Sommaire

1	Introduction	3
1.1	Formulation du problème	5
1.2	Point Historique	5
1.2.1	Les origines de l'interpolation	5
1.2.2	L'interpolation par splines	8
2	Courbes de Bézier	12
2.1	Contexte	12
2.2	Formulation du problème	15
2.3	Partie Mathématique	16
3	Splines cubiques	19
3.1	Contexte	19
3.2	Résolution mathématique	21
3.3	Méthode de calcul	26
3.4	Exemple de calcul d'une fonction spline cubique	30
3.5	Présentation de l'outil informatique	38
3.5.1	Analyse descendante	38
3.5.2	Présentation du contenu du programme	40
4	B-splines	43
4.1	Contexte	43
4.2	Les bases des B-splines	44
4.3	Courbe de degré 3 ($k=3$)	45

4.4	Contrôle local	47
4.5	Autres propriétés	47
4.6	Présentation de l'outil informatique	48
5	Avantages et inconvénients	49
5.1	Courbes de Bézier : avantages et inconvénients	49
5.2	Splines cubiques : avantages et inconvénients	50
5.3	B-splines : avantages et inconvénients	50
6	Exemples de fonctions	52
6.1	Fonction de Runge	52
6.1.1	Introduction	52
6.1.2	Exemple	52
6.2	Fonction Cosinus	55
6.3	Fonction Exponentielle	56
7	Applications des splines cubiques	58
7.1	Les splines dans la typographie et les logiciels de dessin	59
7.2	Les splines dans les transports : l'aéronautique et l'automobile .	61
7.3	Les splines en chimie	62
7.4	Les splines et l'imagerie médicale	63
7.5	Application de splines : le Taj Mahal	64
8	Conclusion	65
9	Crédits d'illustration	66
10	Bibliographie	68

1 Introduction

Depuis toujours, les savants cherchent de meilleures méthodes de résolution à l'aide des mathématiques pour résoudre des problèmes de calcul et de représentation graphique les plus courants dans les domaines de l'industrie et des sciences. Un des aspects essentiels de ces domaines est la précision des mesures et notamment la fidélité des approximations.

Du fait que les études de la nature ne sont en général pas modélisées par des formules mathématiques standards, nous nous trouvons alors obligés de passer par des estimations à l'aide d'opérations avec plusieurs fonctions de base. Ces méthodes entraînent dans la majorité des cas des erreurs et des incertitudes plus ou moins importantes. Ainsi, nous nous demandons : comment pourrait-on diminuer ces inexactitudes afin d'obtenir des résultats plus cohérents avec la réalité ?

De nos jours, chaque scientifique est confronté à tracer et à étudier une courbe expérimentale de la façon la plus pertinente possible en partant de l'unique connaissance de quelques points. Il est alors essentiel de donner avec le plus de précision possible les courbes de tendance et courbes lisses qui relient ces points que ce soit en chimie (courbes ph-métriques, graphiques potentiométriques), en thermodynamique (courbes d'enthalpies), en économie ou en mécanique (courbes de résistance des matériaux, etc)

La plupart des domaines scientifiques ou industriels requièrent la meilleure précision d'approximation de courbe afin de trouver des résultats expérimentaux qui sont bien en accord avec la réalité. Aujourd'hui, un des algorithmes les plus efficaces utilisés pour l'étude et la représentation des courbes est l'interpolation numérique.

D'après le dictionnaire Larousse, l'interpolation est «l'opération consistant à déterminer, à partir d'une série statistique succincte aux valeurs trop espacées, de nouvelles valeurs correspondant à un caractère intermédiaire pour lequel aucune mesure n'a été effectuée». Ce terme est employé dans plusieurs domaines avec des significations variées :

- en philologie (étude critique de textes), une interpolation est un extrait de texte introduit dans une œuvre à laquelle il n'appartient pas ;
- en imagerie numérique l'interpolation consiste à redimensionner l'image en diminuant ou en augmentant la matrice de l'image initiale, et donc le nombre de pixels ;
- en musique, une interpolation permet d'obtenir une valeur de note ou de rythme située entre deux valeurs connues ;
- dans l'industrie des films et d'animation, le principe d'interpolation est connu sous le nom de «Tweening», un procédé qui permet de créer des images intermédiaires successives de telle sorte qu'une image s'enchaîne agréablement et de façon fluide avec la suivante ;
- en méthodologie, l'interpolation est le rapprochement d'une situation avec des situations-type ou modèles ;
- en informatique, une interpolation permet l'évaluation de variables ou d'expressions

à l'intérieur d'une chaîne de caractères littérale (par exemple Perl 6 est particulièrement riche en convention d'interpolation) ;

- en mathématiques : l'interpolation est une opération permettant de construire une courbe à partir de fonctions, elles-mêmes trouvées à partir d'un nombre fini de points donnés ;

Dans ce rapport, nous nous concentrerons uniquement sur le domaine des mathématiques. Plus précisément, nous nous intéresserons à la technique de construction de courbes grâce aux fonctions qu'on établira à partir d'un nombre donné de points et des conditions initiales du problème.

En mathématique, on peut distinguer plusieurs méthodes d'interpolation :

- l'interpolation linéaire : on constitue une courbe d'interpolation qui est une succession de segments, à savoir relier des points entre eux par des portions de droites ;

- l'interpolation polynomiale, consiste à utiliser un polynôme unique et non des tronçons, de degrés aussi grands que nécessaire, pour estimer localement l'équation représentant la courbe, afin de déterminer la valeur entre les échantillons à l'aide de la méthode de détermination lagrangienne, la méthode plus développée d'interpolation d'Hermite ou grâce aux différences finies de Newton ;

- l'interpolation par cosinus : on applique la fonction cosinus en deux points fixés pour modéliser localement la courbe. La tangente à chaque pic est horizontale, ce qui signifie que chaque pic de la fonction correspond réellement à un point connu de la courbe discrète ;

- l'interpolation cubique : on utilise une équation cubique pour modéliser localement la courbe et on évalue la fonction en quatre points qui remplace la courbe discrète. Concernant les conditions de continuité, la forme de la cubique peut varier et donner une interpolation différente (interpolation cubique de Keys ou interpolation par spline cubique). La tangente en chaque point d'indice i possède la même pente que le segment reliant les points d'indice -1 et $i + 1$, ce qui signifie que chaque pic de la courbe peut être dépassé par la courbe interpolée ;

Nous pouvons définir le terme spline par une fonction polynomiale définie par morceaux avec différents polynômes sur chaque intervalle. Ainsi, une interpolation par spline est obtenue par l'assemblage de plusieurs polynômes de faibles degrés en des points donnés. Notre but principal est de présenter la spécificité et la méthodologie de l'interpolation par spline cubique. De ce fait, nous considérons les conditions de continuité spécifiques à cette technique.

Il existe également d'autres procédés pour étudier et tracer des courbes. Les courbes de Bézier et les B-splines sont des méthodes parmi les plus efficaces. Elles sont habituellement utilisées pour la modélisation de surfaces complexes sur des logiciels informatiques de modélisation de type CAO (Conception Assistée par Ordinateur). Dans ce rapport, nous allons aussi présenter dans leur aspect général ces deux techniques avec leurs points forts et leurs points faibles.

1.1 Formulation du problème

Soient des points A_i quelconques de coordonnées (x_i, y_i) , $i = 0, , n$. La méthode d'interpolation par spline cubique consiste à trouver une fonction $s(x)$ définie sur un intervalle $I = [a; b]$ définie sur \mathbb{R} tel que:

- cette fonction relie chaque points A_i et A_{i+1} par une spline cubique (polynôme de degrés 3 maximum) avec n indices naturels ;
- chaque spline est de classe C^2 (deux fois dérivables avec les dérivées d'ordre 1 et 2 continues) en tout point A_i . La classe C^2 est une condition nécessaire pour l'interpolation par spline cubique.

On se demande alors comment utiliser l'interpolation par spline cubique pour tracer une courbe.

Afin de répondre à cette question, nous aborderons le problème d'un point de vue mathématique puis informatique. Nous verrons ainsi par des démonstrations, des exemples, et par l'intermédiaire de deux programmes (un pour les splines cubiques, l'autre pour les B-splines), comment obtenir une fonction spline cubique. Enfin, nous tenterons de cerner les différents enjeux scientifiques et industriels de cette méthode.

1.2 Point Historique

Découvrir et innover nécessitent une certaine distanciation par rapport à notre passé et présent. Il reste cependant nécessaire de s'inspirer des grandes découvertes qui sont venues auparavant. Comprendre le contexte de l'utilisation de cet outil mathématique constitue un intérêt important pour réussir à nous placer dans le cœur du sujet et comprendre toutes les influences fondatrices et, l'évolution de la façon de résoudre notre problème. En effet, comme le disait Leibniz, «C'est une chose extrêmement utile que d'avoir connaissance de l'origine véritable de découvertes mémorables, et en particulier de celles qui ont été faites sciemment et par réflexion, et non par chance ». C'est pour cela qu'il est intéressant de connaître l'histoire et le développement d'un champ auquel on s'intéresse.

Dans le cadre de ce projet il est essentiel de présenter à la fois les origines de l'interpolation en générale et notamment la naissance et l'utilisation des splines au cours du temps.

1.2.1 Les origines de l'interpolation

L'interpolation comme outil mathématique n'est pas un concept nouveau. En effet, on retrouve des exemples d'utilisation très tôt dans l'histoire de l'Homme. Ses premières applications ont principalement eu lieu dans le domaine de l'astronomie.

En effet, les astronomes Babyloniens sont réputés pour avoir utilisé l'interpolation linéaire et polynomiale de degrés élevés aux alentours de 300 av. JC afin de prévoir les positions des astres dans les éphémérides (ouvrage qui contient les positions des astres à des moments donnés). Par ailleurs, quelques tables de pierre montrent aussi des techniques d'interpolation complexe. Malheureusement, la plupart de ces recherches se sont trop dégradées au cours de temps.



Figure 1: Ancienne table de pierre

Puis en 150 av. JC, Hipparque, un astronome grec, utilise à son tour l'interpolation linéaire afin de rassembler ses tables de cordes, une sorte de table trigonométrique primitive afin de prévoir les trajets de la lune et du soleil et les positions des corps célestes.

En 160 après JC, Claudio Ptolémy, un astronome mathématicien grec d'origine égyptienne, utilise des tables contenant une grande diversité de fonctions trigonométriques pour décrire l'univers d'un point de vue géocentrique. Pour éviter les calculs incommodes, il utilise une interpolation linéaire adaptée avec deux bornes pour trouver les valeurs intermédiaires de tableaux. Il est à l'origine de Almagest («La compilation mathématique»). C'est un ouvrage contenant les connaissances les plus avancées de l'Antiquité dans les domaines des mathématiques et de l'astronomie.

Plus tard, entre 600 et 800 après JC, les chinois et les indiens développent chacun de leur côté l'interpolation du deuxième ordre. Ces découvertes correspondent à des versions équivalentes aux formules de Gregory-Newton et Newton-Gauss pour des intervalles inégaux trouvées aux XIIIème siècle.

En l'an 1000, le scientifique arabe al-Biruni écrit une œuvre sur l'interpolation d'ordre 2 en reprenant quelques données des recherches indiennes et presque 300 ans plus tard, les chinois mettent en place les bases de l'interpolation d'ordre 3. On remarque en particulier le mathématicien Zhu Shijie qui écrit en 1303 son œuvre majeure. Il effectue des recherches très innovatrices concernant l'interpolation et trouve une solution proche de la formule de Gregory-Newton. Malgré tout, toutes ces méthodes donnent des résultats avec une précision trop faible. Il faudra attendre 1611 pour la découverte d'Harriot, un mathématicien anglais. Celle-ci consiste en une méthode d'interpolation précise qui va jusqu'aux différences d'ordre cinq. Sa méthode est considérée comme le plus ancien exemple du monde occidental qui utilise les techniques d'interpolation.

Quelques années plus tard, Briggs poursuit le travail de Napier et décrit quelques cas particuliers en considérant les différences d'ordres supérieures ou égales à trois étant égales à zéro.

Le mot «interpolation» apparaît à cette période dans le domaine des mathématiques. Le mathématicien John Wallis en est à l'origine dans son ouvrage en latin, *Arithmetica Infinitorum*, publié en 1655.

Dans la lettre adressée à Collins (secrétaire de la Royal Society), datée du 23 novembre 1670, Gregory écrit explicitement la fameuse formule d'Interpolation Gregory-Newton. En effet, un des mathématiciens les plus importants dans le domaine de l'interpolation est Isaac Newton. Son enthousiasme pour ce sujet, quelques formules et concepts théoriques peuvent être observés dans les lettres pour Smith et Oldenburg.

A partir de 1711, Isaac Newton commence à publier son travail sur l'interpolation en plusieurs livres et manuscrits. Stirling est le premier à s'intéresser aux techniques newtoniennes. Ainsi, en 1730, il publie un livre où il détaille la méthode Stirling-Newton.

Le comte de Lagrange s'est penché sur le problème de l'interpolation au XVI-IIème siècle mais il faut attendre l'arrivée des ordinateurs pour que le domaine se développe réellement. Un aspect très intéressant de sa théorie de 1795 était déjà trouvé par Waring et Euler seize années plus tôt, mais la formule est tout de même connue sous le nom de Lagrange.

Au XVIIIème siècle, l'interpolation et les approximations servent surtout à relier ou à approcher des mesures dans un but scientifique. Avec la révolution industrielle, les machines font leur apparition et il faut faire la conception des pièces pour pouvoir les fabriquer. Pour tracer des courbes, les dessinateurs utilisent des méthodes manuelles qui reposent sur la déformation de lames de métal, de ressorts, l'utilisation de pistolets à colle et des feuilles de graphiques à l'échelle réelle. Pour les surfaces, des gabarits sont construits manuellement en trois dimensions.

Le XIXème siècle est plus marqué par des évolutions et juste quelques innovations de la méthode d'interpolation. Tout d'abord Gauss, Bessel et Cauchy font des recherches sur le sujet, mais uniquement le mathématicien français trouve une nouvelle formule appelé le reste de Cauchy. Vers la fin de ce siècle, Tchebychef et Hermite innoveront la recherche des polynômes d'interpolation et ensuite Weierstrass démontre le théorème d'approximation qui dit que chaque intervalle peut être approximé par un polynôme, ce qui justifie leur utilité pour l'interpolation.

L'époque moderne est fortement marquée par le développement de la technologie et de la communication. Ainsi, de nombreux livres et ouvrages concernant l'interpolation par spline ont été publiés. De plus, du fait que les domaines scientifiques soient en plein essor, l'interpolation commence à être utilisée dans des études et expériences de plus en plus variées, ce qui entraîne l'évolution des

techniques, notamment dans le domaine de l'informatique vers la fin du XXème siècle.

En 1915, E.T.Whittaker développe la formule de Newton-Gauss pour un nombre infini d'abscisses équidistantes des deux côtés d'un point donné et montre que dans certaines conditions, la fonction de la courbe résultante converge vers la fonction qu'il appelle « la fonction cardinal» qui est une combinaison linéaire des fonctions décalées sous la forme $\sin(x)/x$. Après cette découverte, l'accent est mis sur le développement de l'interpolation oscillatoire.

A la fin des années 1940 Shanon et Schoenberg forment une théorie novatrice en considérant les fonctions d'interpolation comme des combinaisons linéaires de fonctions de base ou des noyaux. Au cours des années qui suivent, cette nouvelle approche est de plus en plus utilisée puis améliorée et de nouvelles techniques d'interpolation sont exploitées et développées.

Toutefois, après l'apparition des splines, les scientifiques se rendent compte que ces fonctions ont de meilleures propriétés pour l'interpolation, ce qui permet aux scientifiques d'évoluer et de perfectionner ces nouvelles méthodes de résolution.

En effet, en 1978 l'utilisation des splines dans les images digitales est révolutionnée par Hou & Andrews qui démontre la supériorité absolue de l'interpolation par B-spline. En outre, des études récentes mettent en exergue l'avantage des splines dans le domaine de l'économie. On remarque que la meilleure précision d'interpolation est obtenue en utilisant les grains dits O-MOMS récemment introduits par Blu et ses collaborateurs en 2001.

Par la suite, nous allons présenter la genèse des splines dans le but de comprendre leur évolution et pourquoi l'on considère que c'est le meilleur outil pour l'interpolation.

1.2.2 L'interpolation par splines

Avant l'invention des ordinateurs, tous les calculs étaient effectués à la main. Les fonctions et les fonctions par morceaux sont utilisées mais habituellement les polynômes sont adoptés du fait de leur facilité de manipulation. À l'origine, le concept de spline est développé pour l'élaboration et la fabrication de différentes structures pour les bateaux avant que la modélisation digitale soit implémentée à large échelle.

Dès l'apparition des ordinateurs, les splines remplacent les polynômes dans les problèmes d'interpolation et par la suite, elles sont exploitées dans le cadre de la construction des formes lisses et flexibles aussi au niveau des représentations graphiques numériques. Initialement, le mot « spline » est utilisé pour une lamelle mince en métal ou en bois dans le dialecte des anglais de l'est de l'Angleterre.

Ensuite, en 1895, la signification change et le mot est employé en particulier dans les industries de construction de bateaux et d'avions pour désigner la règle flexible utilisée pour tracer des courbes sur papier.

Pendant de nombreuses années, les concepteurs de bateaux ont utilisé des modèles réduits pour la conception de coques. Le meilleure modèle était ensuite dessiné sur du papier graphique et les points clés de la courbe étaient placés sur un papier graphique à échelle réelle. Une bande en bois était ensuite utilisée pour interpoler les points clés par des courbes lisses.

L'opération est simple : des bandes en bois (ou composées d'autres matériaux) sont fixées au niveau des points clés par des poids en plomb (nommés « ducks », « dogs » ou « rats » en langage technique) et entre ces points, les énergies de tension sont calculées de sorte qu'elles soient minimales selon le matériel utilisé pour la fabrication.

En 1987, Bartels et Robin Forrest décrivent le terme « lofting » comme étant une technique utilisée pendant la deuxième guerre mondiale par l'industrie britannique de construction des avions militaires pour fabriquer des modèles pour le fuselage en reliant des bandes minces (nommées splines) à travers des points clés choisis préalablement et disposées sur un plancher d'un grand loft de design. Un des avions les plus légendaires utilisés pendant cette période qui a révolutionné la stratégie de construction de ces machines est le fameux Spitfire, créé en utilisant la technique de « lofting ». L'entreprise Bristol Aeroplane Company a notamment utilisé cette technique avec des splines pour produire leurs premiers modèles.

Les splines sont apparus nettement après le théorème d'échantillonnage de Shannon qui s'intéresse à la quantité d'information contenue ou délivrée par une source. Les premières références mathématiques au concept de spline sont associées au mathématicien Isaac Schoenberg qui pose les bases de ce terme lorsqu'il est employé par l'armée américaine durant la deuxième guerre mondiale.

En 1946, Isaac Schoenberg est le premier à utiliser ce terme dans ses recherches concernant une approximation polynomiale lisse par morceaux. Il a montré comment utiliser les splines pour interpoler des échantillons équidistants pour une fonction. Cependant, Robin Forrest explique qu'une grande difficulté peut apparaître lorsque l'avion est frappé par une bombe lancé par l'opposition puisque la structure est susceptible de perdre de nombreux composants vitaux à cause de la faible rigidité. De ce fait, ils développent une autre méthode de fabrication nommé « conic lofting » qui utilise des sections coniques pour modéliser les positions des courbes qui se trouvent entre les points clés fixés par les dispositifs « ducks ».

Isaac Schoenberg a aussi introduit les B-splines, éléments nécessaires à la construction de splines polynomiaux, mais ces techniques ne sont utilisées qu'à partir des années 60. Les B-splines remplacent le « conic lofting » quand J. C. Ferguson, employé de Boeing, et plus tard, M. A. Sabin, employé de British Aircraft Corporation, réalisent que ces fonctions peuvent modéliser le processus physique de lissage de courbes.

En ce qui concerne l'utilisation des splines pour la conception et la modélisation des automobiles, on se rend compte que plusieurs projets indépendants

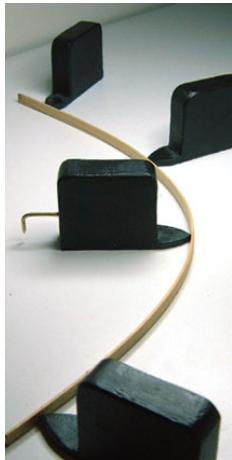


Figure 2: Ducks

sont majoritairement développés par des chercheurs pour des entreprises de ce milieu pendant la fin des années 50 et début des années 60. Ce fait s'explique par la forte concurrence des grands constructeurs automobiles.

Ces chercheurs ont posé les bases des algorithmes et des techniques pour évaluer les calculs des courbes. Paul de Casteljau, physicien et mathématicien de Citroën, a développé sa propre analyse numérique. Pierre Bézier, ingénieur français en mécanique et en électricité de Renault, a amélioré le travail de Casteljau et a créé les fameuses Courbes de Bézier, Birkhoff (qui a également étudié les problèmes d'interpolation en général) et finalement Garabedian et Boor de General Motors qui ont enrichi le sujet avec des théorèmes et des programmes numériques pour la production de voitures.

Une seule recherche de Paul de Casteljau n'a été publié en 1959. Cependant, les travaux de Boor de General Motors ont été publié au début des années 1960, incluant quelques théories fondamentales sur les B-splines. Les résultats du travail fait chez General Motors ont été détaillé dans les ouvrages de Birkhoff en 1990 et ensuite dans l'ouvrage de Young en 1997. De plus, en 1997, Davis résume toutes ces recherches théoriques.

Par ailleurs, chez Pratt & Whitney Aircraft, deux auteurs du premier livre de traitement des splines (Albert et ses collaborateurs en 1967) sont employés pour travailler sur ce sujet. Parallèlement, Feodor Theilheimer travaille chez David Taylor Model Basin pour augmenter l'utilité des splines.

Pour améliorer la manière de travailler avec les splines, depuis le début des splines, différents scientifiques, mathématiciens et ingénieurs ont tenté de trouver d'autres conditions et particularités afin d'obtenir une meilleure précision. Par exemple, Whittaker a dérivé une spline pour obtenir des données sur le lissage. Il est le premier à proposer d'introduire une mesure d'ajustement avec un terme de pénalité construit à partir des sommes des carrés de la troisième

différence de la fonction pour améliorer l'uniformité des solutions. En fait, il traite le terme de pénalité comme une distribution préalable puis il estime les allures et la construction des splines grâce au théorème de Bayes.

Le développement des splines reste tout de même majoritairement alloué à Bézier. En effet, il a été le premier à publier ses résultats lorsque Renault lui en a autorisé. La recherche sur les splines ne s'en est cependant pas arrêtée là et des équipes telles que le Biomedical Imaging Group de l'EPFL continuent à faire progresser le domaine.

De nos jours, les splines sont un outil indispensable pour des nombreux domaines allant de la conception des automobiles jusqu'au traitement des sons et le domaine médical.

Finalement, de nombreuses études conduisent à la conclusion que les splines offrent généralement le meilleur compromis coûts-performances. Grâce à l'évolution de la technologie, l'interpolation par spline continue à être améliorée et les méthodes de manipulation évoluent dans le but d'atteindre la meilleure approximation lorsque l'on étudie des problèmes de la vie réelle.

2 Courbes de Bézier

2.1 Contexte

Dans le cadre de la fabrication des automobiles la modélisation des courbes est une étape fondamentale qui s'est développé notamment pendant l'évolution de la technologie numérique, les ordinateurs en particulier. Pierre Bézier est un des ingénieurs français les plus importants qui a dédié sa carrière à la création d'un moyen simple et puissant pour modéliser des formes et faciliter la programmation des machines à commande numérique afin de rendre la production plus efficace.

Diplômé de l'école Nationale Supérieure d'Arts et Métiers et de l'école supérieure d'électricité, élève exceptionnel né à Paris en septembre 1910, Pierre Bézier entre chez Renault en 1933 où il y fera toute sa carrière jusqu'en 1975 au poste de directeur des méthodes mécaniques. Par la suite, dû à des divergences avec ses supérieurs, il est mis à l'écart. Cela lui donne l'occasion de s'intéresser à un domaine encore presque vierge : la modélisation des surfaces.



Figure 3: Pierre Bézier, 1958

Mathématicien, physicien, philosophe, ingénieur en mécanique et en électricité, il est renommé pour ses inventions révolutionnaires pour lesquelles il reçoit le Prix Nessim-Habif en 1972. à l'époque, un élément nouveau vient tout remettre en cause : le développement des machines à commande numérique. Apparues chez Renault au début des années soixante, elles n'étaient programmées que de façon relativement simple. Pierre Bézier chercha donc comment traduire mathématiquement une courbe, puis une surface dessinée à main levée, ce qui rend plus facile pour la commande numérique de prendre en compte instinctivement les modifications apportées à un modèle. Il s'attaque alors au problème de la modélisation des surfaces en trois dimensions, les commandes numériques se contentant jusqu'alors de courbes en deux dimensions. Comme Bézier voulait une interface accessible à toute le monde, il décide alors de considérer classiquement les surfaces comme une transformation de courbes. Son exigence de s'adapter

au dessinateur l'amène à une invention géniale, à savoir déduire le calcul à partir du dessin grâce à sa propre technique innovatrice nommée « poignée de contrôle ».



Figure 4: Renault 4CV

Au lancement de la Renault 4CV, en 1946, son objectif, très ambitieux pour l'époque, est d'assurer une cadence de 20 voitures par jour. Pour cela, il imagine une machine transfert à têtes indépendantes. En un an, son équipe réalise 750 unités d'usinage normalisées et 60 tables rotatives. Ainsi, la cadence atteint progressivement 300 voitures par jour. Ces machines spéciales sont vendues à travers le monde. Il devient alors chef du bureau d'études outillages mécaniques. En 1955, il s'intéresse à la commande numérique et trois ans plus tard, il met au point des perceuses avec des équipements aux commandes entièrement transistorisées. Il avance successivement sur l'échelle hiérarchique jusqu'en 1960 où il est nommé directeur générale, déchargé de toute responsabilité opérationnelle.

Bézier était un ingénieur très rigoureux qui avait une affinité formidable pour la précision des résultats en visualisant la modélisation comme une œuvre d'art. Par conséquent, il choisit la voie la plus difficile à explorer, à savoir transformer les courbes de forme des stylistes de carrosserie en expressions mathématiques qui devaient être utilisables par les ingénieurs grâce aux ordinateurs et ses liaisons avec les machines-outils. Ses recherches aboutirent à un logiciel appelé Unisurf.

Unisurf

Unisurf (« uni » pour unification et « surf » pour surface) est un pionnier des logiciels de DAO (Dessin Assisté par Ordinateur) et de CFAO (Conception et Fabrication Assistées par Ordinateur). Unisurf a été développé en 1971 par Pierre Bézier ayant pour but de passer directement du papier à la machine. Cette application, destinée à la conception des automobiles en association directe avec les machines à commandes numériques est la base des logiciels de CFAO (par exemple CATIA) développés ultérieurement. Unisurf provoque au début un mini tsunami culturel. En fait, les premières pièces qui pouvaient être réalisées grâce à la CAO étaient des pièces petites peu compliquées et les dessinateurs regardaient ce logiciel avec scepticisme car dessiner avec le logiciel

prenait plus de temps que de le réaliser sur papier. De plus, les logiciels et les machines provoquaient souvent des erreurs à cause de mauvaises manipulations et des petits moteurs des tables à dessin qui se bloquaient. Toutefois, l'évolution a été extraordinaire et peu à peu Renault passe de 10% des pièces fabriquées par CAO à 30% puis à 60%.

En 1968, Unisurf est présent partout, en particulier à Détroit, le temple de l'industrie automobile à l'époque. Une machine à dessiner, une fraiseuse, un ordinateur d'occasion avec une mémoire de 8 ko et un logiciel rudimentaire. Le système est opérationnel en 1972 et avec Unisurf 3 en 1975, la CFAO se généralise dans toute l'industrie, facilitant la définition graphique d'objets aux formes complexes (nez des motrices TGV, sièges de voiture etc.).

Cette modélisation mathématique s'appuie sur les fameuses « courbes de Bézier », représentatives de polynômes paramétriques en langage mathématique. Elles constituent le sujet de sa thèse de doctorat d'état en mathématiques défendue le 23 février 1977 sous le titre: « Essai de définition numérique des courbes et surfaces ».

Unisurf est adopté en 1967 par Peugeot qui le perfectionne et l'utilise pour concevoir la maquette complète de la Peugeot 104. Peugeot et Renault collaborent ainsi pendant quinze ans aux versions successives du système Unisurf. à partir de 1980, une vingtaine de personnes dont quatre à cinq informaticiens travaillent chez Renault avec le logiciel Unisurf sur des machines qui occupent l'espace d'une chambre.

2.2 Formulation du problème

La problématique

On considère $n+1$ points du plan ($n \geq 1$, le plus souvent $n = 3$), A_0, A_1, \dots, A_n et on définit une courbe paramétrée $M(t)$, $t \in [0, 1]$, associée à ces points, où $M(t)$ est un barycentre des A_i :

$$M(t) = B_0(t)A_0 + B_1(t)A_1 + \dots + B_n(t)A_n$$

où $M(t) \geq 0 \quad \forall i = 0, \dots, n$

$$\sum B_i(t) = 1 \quad \forall i = 0, \dots, n$$

Les coefficients $B_i(t)$ dépendent de t de manière la plus régulière possible. La solution adoptée par P. Bézier consiste à prendre les polynômes de Bernstein.

Définition

Les polynômes de Bernstein d'ordre n sont les polynômes

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

Lorsque n est fixé, on note simplement B_i au lieu de $B_{i,n}$.

On vérifie aussitôt que les B_i sont supérieurs à 0 sur $[0, 1]$ et que leur somme est bien égale à 1 (c'est la formule du binôme de Newton appliquée à $(1-t+t)^n$).

Remarques

1) On a $B_{0,n}(t) = (1-t)^n$ et $B_{n,n}(t) = t^n$

2) On notera que :

- les polynômes $B_i(t)$ forment une base de l'espace vectoriel des polynômes de degré inférieur à n (tous les polynômes sont éléments de R^n)
- l'espace est non nul et stable par combinaison linéaire, donc sous-espace vectoriel de R^n

3) Les polynômes B_i forment ce qu'on appelle une partition de l'unité. Ils sont bien connus en analyse. En particulier, si f est une fonction continue sur $[0, 1]$, on peut poser :

$$B_n(f) = \sum_{i=0}^n \left(\frac{i}{n}\right) B_i(f)$$

On montre que la suite de polynômes $B_n(f)$ converge uniformément vers f sur $[0, 1]$ (cela est le théorème de Weierstrass).

2.3 Partie Mathématique

Théorie générale

Quatre points P_0, P_1, P_2 et P_3 définissent une courbe de Bézier cubique. La courbe se trace en partant du point P_0 , en se dirigeant vers P_1 et en arrivant au point P_3 selon la direction $P_2 - P_3$, mais elle ne passe ni par P_1 ni par P_2 (ces points donnent une information de direction). La distance entre P_0 et P_1 détermine la "longueur" du déplacement dans la direction de P_1 avant de tourner vers P_3 .

Pour la forme paramétrique de la courbe, nous remarquons que la formule des coefficients du polynôme qui caractérise la courbe est inspirée d'une loi binomiale et montre que la courbe est toujours totalement contenue dans l'enveloppe convexe des quatre points donnés (ce qui aide également à vérifier les conditions de polynômes de Bernstein).

Les courbes de Bézier sont intéressantes pour le traitement des images car les points peuvent être rapidement calculés en utilisant une procédure récursive qui utilise la division et les opérations de base en évitant toutes les opérations de l'arithmétique des nombres réels flottants.

$$\begin{pmatrix} A' \\ B' \\ C' \\ D' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{2}{4} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{pmatrix} \begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix}$$

ou

$$D \leftarrow \frac{C + D}{2}$$

$$C \leftarrow \frac{B + C}{2}, \quad D \leftarrow \frac{C + D}{2}$$

$$B \leftarrow \frac{A + B}{2}, \quad C \leftarrow \frac{B + C}{2}, \quad D \leftarrow \frac{C + D}{2}$$

Plus précisément, on peut décomposer la courbe $P(t)$ en deux courbes P_L et P_R dont les points de contrôles sont respectivement (L_1, L_2, L_3, L_4) et (R_1, R_2, R_3, R_4) :

$$\begin{pmatrix} L'_1 \\ L'_2 \\ L'_3 \\ L'_4 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{2}{4} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{pmatrix} \begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} R'_1 \\ R'_2 \\ R'_3 \\ R'_4 \end{pmatrix} = \begin{pmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{2}{4} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix}$$

Courbes de Bézier d'ordre 2 (quadratique)

Une courbe de Bézier quadratique est la courbe $B(t)$ définie par les points de contrôle P_0, P_1 et P_2 .

$$B(t) = (1-t)^2 P_0 + 2t(1-t)P_1 + t^2 P_2, \quad t \in [0, 1]$$

On considère trois points distincts A, B, C du plan.

On pose $A = (a_1, a_2), B = (b_1, b_2), C = (c_1, c_2)$.

On considère la courbe de Bézier Γ définie par

$$M(t) = (1-t)^2 A + 2t(1-t)B + t^2 C$$

Elle passe par A et C et elle est tangente respectivement à (AB) et (BC) en ces points. Si A, B, C sont alignés, la courbe Γ est égale au segment $[AB]$. Sinon, la courbe Γ est une parabole.

Démonstration pour A, B, C non colinéaires: On a les formules:

$$x(t) = at^2 + bt + c \quad \text{et} \quad y(t) = a't^2 + b't + c'$$

avec $a = a_1 - 2b_1 + c_1, \quad b = 2(b_1 - a_1), \quad a' = a_2 - 2b_2 + c_2, \quad b' = 2(b_2 - a_2)$

On note que $a \neq 0$ et $a' \neq 0$ car sinon B est le milieu de $[AC]$. On élimine les termes en t^2 entre ces équations:

$$ax - ay = (a'b - ab')t + a'c - ac'$$

Si le coefficient de t est nul on obtient une droite, ce qui est absurde puisque les points A, B, C ne sont pas alignés (en outre les droites (AB) et (BC) sont tangentées à la courbe). Précisément, ce coefficient vaut

$$ab - ab' = 2(-a_1b_2 + b_1a_2 + c_1b_2 - b_1c_2 + a_1c_2 - c_1a_2)$$

et c'est le déterminant obtenu en bordant d'une ligne de 1 la matrice des coordonnées de A, B, C .

On peut écrire une expression pour t comme on est sûr de ne pas diviser par 0:

$$t = \frac{a'x - ay + ac' - a'c}{a'b - ab'}$$

Si on désigne par X cette quantité (ce qui revient à faire un changement de coordonnées cartésiennes) l'équation de Γ s'écrit $y = a'X^2 + b'X + c'$, ce qui correspond à l'équation d'une parabole.

Ces courbes quadratiques sont encore très utilisées aujourd'hui (par exemple dans les définitions de glyphes des polices de caractères des logiciels de traitement de texte au format TrueType et les polices OpenType dans leur variété compatible TrueType).

Ce type de courbes permet d'assurer la continuité en tangence de deux courbes raccordées mais en général, elles ne permettent pas de conserver la continuité de la courbure aux points d'interconnexion. Pour diminuer cet inconvénient, il est alors nécessaire d'augmenter le nombre d'arcs interconnectés afin de réduire les ruptures de courbure entre chacun d'eux, ce qui en limite l'intérêt et peut conduire à une complexification de la conception des courbes (avec davantage de sommets et de points de contrôles à positionner).

Courbes de Bézier d'ordre 3 (cubiques)

Une courbe de Bézier cubique est la courbe $B(t)$ définie par les points de contrôle P_0, P_1, P_2 et P_3 . Sa forme paramétrique est:

$$B(t) = P_0(1-t)^3 + 3P_1t(1-t)^2 + 3P_2t^2(1-t) + P_3t^3, \quad t \in [0, 1]$$

Ce sont les courbes de Bézier qui sont les plus utilisées en conception graphique car elles permettent d'assurer non seulement la continuité en tangence de deux courbes raccordées mais aussi celle de leur courbure en évitant de positionner de nombreux sommets et points de contrôle comme en ordre 2. Elles sont utilisées pour les polices des logiciels de traitement de texte par exemple dans le langage PostScript et la définition des glyphes des polices de caractères de « type 1 », ainsi que dans les polices OpenType dans leur variété CFF (Compact Font Format) qui reprennent les mêmes définitions de sommets et de points de contrôle.

Il en résulte un théorème général prouvé ci-dessous que la courbe est une portion cubique, c'est-à-dire qu'elle est définie par une équation $F(x, y) = 0$ où F est un polynôme de degré inférieur ou égal à 3. Les polynômes de Bernstein $(1-t)^3, 3(1-t)^2t, 3(1-t)t^2$ et t^3 constituent une base de l'espace vectoriel des polynômes de degré inférieur ou égal à 3.

Démonstration : on a quatre points A, B, C, D et on pose :

$$M(t) = (1-t)^3A + 3(1-t)^2tB + 3(1-t)t^2C + t^3D$$

Il suffit de noter que leurs valeurs sont respectivement 0, 1, 2, 3. Par conséquent, les courbes de Bézier d'ordre 3 sont exactement toutes les courbes paramétrées par des polynômes de degré inférieur ou égal à 3. En effet, si on pose $A = (a_1, a_2)$, $B = (b_1, b_2)$ et de même pour tous les points, on obtient tous les couples de polynômes de degré inférieur ou égal à 3 possibles. Les courbes de Bézier sont des courbes de degré 3. Elles sont donc déterminées par quatre points de contrôle. Si les coordonnées des points de contrôle sont $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ et (x_4, y_4) , l'équation de la courbe est :

$$x(t) = (1-t)^3x_1 + 3t(1-t)^2x_2 + 3t^2(1-t)x_3 + t^3x_4 \text{ pour } 0 \leq t \leq 1$$

$$y(t) = (1-t)^3y_1 + 3t(1-t)^2y_2 + 3t^2(1-t)y_3 + t^3y_4 \text{ pour } 0 \leq t \leq 1$$

3 Splines cubiques

3.1 Contexte

Quand on parle de l'interpolation mathématiques, on s'intéresse aux méthodes de modélisation et paramétrisation des courbes entre plusieurs points de contrôle déjà fixés à l'aide des fonctions fondamentales. Les polynômes sont très bien adaptés pour ce genre de problèmes grâce à leurs particularités qui favorisent une résolution simple et rapide.

Cependant, comme leur précision n'est pas très élevée, dans les problèmes d'interpolation on préfère utiliser les splines dans la majorité des cas puisque on trouve des résultats similaires avec les interpolations polynomiales de degré très grand tout en évitant les instabilités dus au Phénomène du Runge que peuvent entraîner des résultats avec une précision moyenne. En effet, au niveau des graphiques numériques, les splines cubiques sont les fondements des nombreuses opérations grâce à la simplicité de construction, à la précision assez importante des solutions et également à leur capacité de décrire des formes complexes avec des ajustements et design interactif des courbes. L'évolution des splines est notamment observée après l'apparition des ordinateurs. En s'appuyant sur l'interpolation polynomiale déjà présente à l'époque, les scientifiques et les mathématiciens ont développé une méthode d'interpolation par morceaux de polynômes afin d'augmenter l'efficacité des résultats tout en gardant la même simplicité de manipulation.

Un des plus importants fondateurs des méthodes de l'interpolation en utilisant des polynômes définis par morceaux est Isaac Schoenberg. Mathématicien roumain né au début du XXème siècle, réputé pour ses travaux sur les splines, fait ses études notamment en Roumanie, Allemagne et également aux États-Unis. Vers 1940, Schoenberg a réalisé des travaux extrêmement importants sur le sujet de l'analyse mathématiques générale et sur le sujet d'approximation en utilisant l'interpolation par des fonctions splines développées par lui-même. Pendant 15 années, il est le seul qui s'intéresse à ce type d'interpolation en réussissant à écrire une quarantaine de documents qui seront indispensables pour les ouvrages développés plus tard par d'autres scientifiques. De ce fait, en 1960, quand les ordinateurs sont de plus en plus répandus dans le monde scientifique, on commence à utiliser les splines notamment pour les ajustements des données numériques et pour le design géométrique assisté par ordinateur (ce qui est à la base de la CAO moderne qui actuellement utilise les B-splines, plus évolués). Malgré sa retraite prise en 1973, il n'arrête pas ses recherches sur les splines, ce qui lui permet d'introduire un autre concept d'importance majeure, à savoir les splines cardinales.

Une courbe spline est une fonction polynomiale par morceaux définie sur un intervalle $[a;b]$ divisé en sous-intervalles. Le degré de la spline est défini comme celui du polynôme de plus haut degré utilisé dans sa construction. On appelle spline uniforme, une spline construite sur l'intervalle $[a;b]$ seulement avec des

polynômes qui ont le même degré. Dans le cas contraire, elle est non uniforme.

En ce qui concerne la dérivabilité d'une spline, sachant que la dérivabilité d'un polynôme est infinie, elle dépend de la continuité au niveau de la jointure des courbes polynomiales définies sur chaque sous-intervalle de $[a;b]$.

La plus simple fonction spline est celle uniforme de degré 1 appelé aussi ligne polygonale ou ligne brisée. Graphiquement, la méthode consiste juste à relier chaque point avec des segments de droite, ce qui donnera une continuité C^0 sauf si les points sont alignés.

Le cas le plus courant et le plus important des splines est la spline cubique uniforme. Ces fonctions ont une utilité fondamentale dans une large gamme de domaines scientifiques, notamment dans la CAO. En fonction des contraintes imposées en général par la spécificité de chaque problème, on a multiples manières d'interpoler:

- Interpolation avec les tangentes
- Interpolation avec méthode de Catmull-Rom
 - Spline circulaire
 - Spline serrée
 - Spline naturelle
- Interpolation en courbure

3.2 Résolution mathématique

Définition

Considérons $n + 1$ points x_i distincts. On définit un polynôme s_i sur $I = [x_i, x_{i+1}]$ tel que

$$s_i(x) = a_i x + b_i x + c_i x + d_i \quad \text{avec } i = 0, 1, 2, \dots, n$$

Posons

$$s_i(x_i) = f(x_i) = f_i$$

On doit avoir $s \in C^2$. Nous devons donc satisfaire le système suivant:

$$\begin{cases} s'_{i-1}(x_i) = s'_i(x_i) & i = 1, 2, \dots, n-1 \\ s''_{i-1}(x_i) = s''_i(x_i) \end{cases}$$

On obtient

$$s''_i(x) = f''_i \frac{(x_{i+1}-x)}{h_i} + f''_{i+1} \frac{(x-x_i)}{h_i} \quad \text{où } h_i = x_{i+1} - x_i$$

Démonstration

$$s_i(x_i) = a_i x_i^3 + b_i x_i^2 + c_i x_i + d_i$$

En dérivant:

$$s'_i(x_i) = 3a_i x_i^2 + 2b_i x_i + c_i$$

$$s''_i(x_i) = 6a_i x_i + 2b_i$$

$$\Rightarrow b_i = \frac{s''_i(x_i) - 6a_i x_i}{2} = \frac{f''_i - 6a_i x_i}{2}$$

De plus,

$$s_i''(x_{i+1}) = 6a_i(x_{i+1}) + 2b_i$$

$$b_i = 6a_i x_{i+1} + 2 \frac{(s_i''(x_i) - 6a_i x_i)}{2}$$

$$s_i''(x_{i+1}) = 6a_i x_{i+1} + f_i'' - 6a_i x_i = f_{i+1}''$$

$$a_i[6(x_{i+1}) - 6x_i] = f_{i+1}'' - f_i''$$

$$a_i = \frac{(f_{i+1}'' - f_i'')}{6(x_{i+1} - x_i)}$$

On a

$$s_i''(x) = 6a_i x + 2b_i$$

En remplaçant a_i et b_i par leurs expressions, on obtient :

$$s_i''(x) = 6 \frac{(f_{i+1}'' - f_i'')}{6(x_{i+1} - x_i)} x + 2 \frac{(f_i'' - 6a_i x_i)}{2}$$

$$s_i''(x) = \frac{(f_{i+1}'' - f_i'')}{x_{i+1} - x_i} x + f_i'' - 6a_i x_i$$

$$s_i''(x) = \frac{(f_{i+1}'' - f_i'')}{x_{i+1} - x_i} x + f_i'' - 6 \frac{(f_{i+1}'' - f_i'')}{6(x_{i+1} - x_i)} x_i$$

$$s_i''(x) = \frac{x(f_{i+1}'' - f_i'')}{(x_{i+1} - x_i)} + f_i'' - \frac{x_i(f_{i+1}'' - f_i'')}{(x_{i+1} - x_i)}$$

Or $h_i = x_{i+1} - x_i$,

$$s_i''(x) = \frac{x(f_{i+1}'' - f_i'')}{(x_{i+1} - x_i)} + f_i'' - \frac{x_i(f_{i+1}'' - f_i'')}{(x_{i+1} - x_i)}$$

$$s_i''(x) = \frac{xf_{i+1}'' - xf_i'' + f_i''(x_{i+1} - x_i) - x_i f_{i+1}'' + x_i f_i''}{h_i}$$

$$s_i''(x) = \frac{xf_{i+1}'' - xf_i'' + x_{i+1} f_i'' - x_i f_i'' - x_i f_{i+1}'' + x_i f_i''}{h_i}$$

$$s_i''(x) = \frac{f_{i+1}''(x - x_i) + f_i''(x_{i+1} - x)}{h_i}$$

$$s_i''(x) = f_i'' \frac{(x_{i+1} - x)}{h_i} + f_{i+1}'' \frac{(x - x_i)}{h_i}$$

On intègre deux fois et en considérant les constantes d'intégration a_i et b_i : On intègre une première fois :

$$s'_i(x) = \int \left(f''_i \frac{(x_{i+1} - x)}{h_i} + f''_{i+1} \frac{(x - x_i)}{h_i} \right) dx$$

$$s'_i(x) = -f''_i \frac{(x_{i+1} - x)^2}{2h_i} + a_i + f''_{i+1} \frac{(x - x_i)^2}{2h_i} + b_i$$

On intègre une deuxième fois :

$$s_i(x) = \int \left(-f''_i \frac{(x_{i+1} - x)^2}{2h_i} + a_i + f''_{i+1} \frac{(x - x_i)^2}{2h_i} + b_i \right) dx$$

$$s_i(x) = f''_i \frac{(x_{i+1} - x)^3}{6h_i} + f''_{i+1} \frac{(x - x_i)^3}{6h_i} + a_i(x_{i+1} - x) + b_i(x - x_i) \quad i = 0, 1, \dots, n - 1$$

Nous allons maintenant déterminer les expressions de a_i et b_i :

Ces constantes peuvent être déterminées en écrivant les relations de continuité aux noeuds.

$$s_i(x_i) = f(x_i) \text{ et } s_i(x_{i+1}) = f(x_{i+1})$$

Pour a_i :

$$s_i(x_i) = f''_i \frac{(x_{i+1} - x_i)^3}{6h_i} + f''_{i+1} \frac{(x_i - x_i)^3}{6h_i} + a_i(x_{i+1} - x_i) + b_i(x_i - x_i) \text{ avec } h_i = x_{i+1} - x_i$$

$$f_i = f''_i \frac{h_i^2}{6} + a_i h_i$$

$$a_i = \frac{(f_i - f''_i) \frac{h_i^2}{6}}{h_i}$$

$$a_i = \frac{f_i}{h_i} - f''_i \frac{h_i}{6}$$

Pour b_i :

$$s_i(x_{i+1}) = f_i'' \frac{(x_{i+1} - x_i)^3}{6h_i} + f_{i+1}'' \frac{(x_{i+1} - x_i)^3}{6h_i} + a_i(x_{i+1} - x_i) + b_i(x_{i+1} - x_i) \text{ avec } h_i = x_{i+1} - x_i$$

$$f_{i+1} = f_{i+1}'' \frac{h_i^2}{6} + b_i h_i$$

$$b_i = \frac{(f_{i+1} - f_{i+1}'') \frac{h_i^2}{6}}{h_i}$$

$$b_i = \frac{f_{i+1}}{h_i} - f_{i+1}'' \frac{h_i}{6}$$

$$\text{ainsi } a_i = \frac{f_i}{h_i} - f_i'' \frac{h_i}{6} \text{ et } b_i = \frac{f_{i+1}}{h_i} - f_{i+1}'' \frac{h_i}{6}$$

On peut alors remplacer les expressions de a_i et b_i dans $s_i(x)$

$$s_i(x) = f_i'' \frac{(x_{i+1} - x)^3}{6h_i} + f_{i+1}'' \frac{(x - x_i)^3}{6h_i} + a_i(x_{i+1} - x) + b_i(x - x_i) \quad i = 0, 1, \dots, n-1$$

$$s_i(x) = f_i'' \frac{(x_{i+1} - x)^3}{6h_i} + f_{i+1}'' \frac{(x - x_i)^3}{6h_i} + \left(\frac{f_i}{h_i} - f_i'' \frac{h_i}{6} \right) (x_{i+1} - x) + \left(\frac{f_{i+1}}{h_i} - f_{i+1}'' \frac{h_i}{6} \right) (x - x_i)$$

$$s_i(x) = f_i'' \left(\frac{(x_{i+1} - x)^3}{6h_i} - \frac{h_i}{6} (x_{i+1} - x) \right) + f_{i+1}'' \left(\frac{(x - x_i)^3}{6h_i} - \frac{h_i}{6} (x - x_i) \right) + \frac{f_i}{h_i} (x_{i+1} - x) + \frac{f_{i+1}}{h_i} (x - x_i)$$

Il faut ensuite respecter les conditions de continuité pour les dérivées premières :

$$s'_i(x) = f_i'' \left(\frac{-(x_{i+1} - x)^2}{2h_i} + \frac{h_i}{6} \right) + f_{i+1}'' \left(\frac{(x - x_i)^2}{2h_i} - \frac{h_i}{6} \right) - \frac{f_i}{h_i} + \frac{f_{i+1}}{h_i}$$

La condition de continuité $s'_i(x_i) = s'_{i-1}(x_i)$ en x_i s'écrit alors :

d'une part,

$$\begin{aligned} s'_i(x_i) &= f''_i\left(\frac{-(x_{i+1}-x_i)^2}{2h_i} + \frac{h_i}{6}\right) + f''_{i+1}\left(\frac{(x_i-x_{i-1})^2}{2h_i} - \frac{h_{i-1}}{6}\right) - \frac{f_i}{h_i} + \frac{f_{i+1}}{h_i} \\ s'_i(x_i) &= f''_i\left(-\frac{h_i}{2} + \frac{h_i}{6}\right) - f''_{i+1}\frac{h_i}{6} - \frac{f_i}{h_i} + \frac{f_{i+1}}{h_i} \\ s'_i(x_i) &= f''_i\left(-\frac{2h_i}{6}\right) - f''_{i+1}\frac{h_i}{6} - \frac{f_i}{h_i} + \frac{f_{i+1}}{h_i} \\ s'_i(x_i) &= -f''_i\frac{h_i}{3} - f''_{i+1}\frac{h_i}{6} - \frac{f_i}{h_i} + \frac{f_{i+1}}{h_i} \end{aligned}$$

d'autre part,

$$\begin{aligned} s'_{i-1}(x_i) &= f''_{i-1}\left(\frac{-(x_i-x_{i-1})^2}{2h_{i-1}} + \frac{h_{i-1}}{6}\right) + f''_i\left(\frac{(x_i-x_{i-1})^2}{2h_{i-1}} - \frac{h_{i-1}}{6}\right) - \frac{f_{i-1}}{h_{i-1}} + \frac{f_i}{h_{i-1}} \\ s'_{i-1}(x_i) &= f''_{i-1}\frac{h_{i-1}}{6} + f''_i\left(\frac{h_{i-1}^2}{2h_{i-1}} - \frac{h_{i-1}}{6}\right) - \frac{f_{i-1}}{h_{i-1}} + \frac{f_i}{h_{i-1}} \quad \text{avec } h_{i-1} = x_i - x_{i-1} \\ s'_{i-1}(x_i) &= f''_{i-1}\frac{h_{i-1}}{6} + f''_i\left(\frac{h_{i-1}}{2} - \frac{h_{i-1}}{6}\right) - \frac{f_{i-1}}{h_{i-1}} + \frac{f_i}{h_{i-1}} \\ s'_{i-1}(x_i) &= f''_{i-1}\frac{h_{i-1}}{6} + f''_i\frac{h_{i-1}}{3} - \frac{f_{i-1}}{h_{i-1}} + \frac{f_i}{h_{i-1}} \end{aligned}$$

ainsi, en utilisant le raccord des dérivées premières, on a :

$$\begin{aligned} s'_i(x_i) &= s'_{i-1}(x_i) \\ -f''_i\frac{h_i}{3} - f''_{i+1}\frac{h_i}{6} - \frac{f_i}{h_i} + \frac{f_{i+1}}{h_i} &= f''_{i-1}\frac{h_{i-1}}{6} + f''_i\frac{h_{i-1}}{3} - \frac{f_{i-1}}{h_{i-1}} + \frac{f_i}{h_{i-1}} \\ -\frac{f_i}{h_i} + \frac{f_{i+1}}{h_i} + \frac{f_{i-1}}{h_{i-1}} - \frac{f_i}{h_{i-1}} &= f''_i\frac{h_i}{3} + f''_{i+1}\frac{h_i}{6} + f''_{i-1}\frac{h_{i-1}}{6} + f''_i\frac{h_{i-1}}{3} \\ \frac{f_{i+1}-f_i}{h_i} - \frac{f_i-f_{i-1}}{h_{i-1}} &= \frac{2f''_ih_i + f''_{i+1}h_i + f''_{i-1}h_{i-1} + 2f''_ih_{i-1}}{6} \\ 6\left(\frac{f_{i+1}-f_i}{h_i} - \frac{f_i-f_{i-1}}{h_{i-1}}\right) &= 2f''_i(h_i + h_{i-1}) + f''_{i+1}h_i + f''_{i-1}h_{i-1} \\ h_if''_{i+1} + 2(h_i + h_{i-1})f''_i + h_{i-1}f''_{i-1} &= 6\left(\frac{f_{i+1}-f_i}{h_i} - \frac{f_i-f_{i-1}}{h_{i-1}}\right) \quad i = 1, 2, \dots, n-1 \end{aligned}$$

On obtient donc un système linéaire de $(n-1)$ équations à $(n+1)$ inconnus. Si on fixe les valeurs de f''_0 et f''_n (par exemple à 0) on peut résoudre ce système.

3.3 Méthode de calcul

Dans cette section, nous allons montrer comment résoudre ce système pour des points données. Considérons 5 points: x_0, x_1, x_2, x_3 , et x_4 , d'ordonnées respectives f_0, f_1, f_2, f_3 , et f_4 . En appliquant l'égalité (1), on obtient :

$$\begin{cases} h_0 f''_0 + 2(h_1 + h_0) f''_1 + h_1 f''_2 = 6\left(\frac{f_2 - f_1}{h_1} - \frac{f_1 - f_0}{h_0}\right) \\ h_1 f''_1 + 2(h_2 + h_1) f''_2 + h_2 f''_3 = 6\left(\frac{f_3 - f_2}{h_2} - \frac{f_2 - f_1}{h_1}\right) \\ h_2 f''_2 + 2(h_3 + h_2) f''_3 + h_3 f''_4 = 6\left(\frac{f_4 - f_3}{h_3} - \frac{f_3 - f_2}{h_2}\right) \end{cases}$$

Pour une résolution plus facile, on peut réécrire le système sous forme matricielle en factorisant par $f''_0, f''_1, f''_2, f''_3, f''_4$, on obtient :

$$\begin{pmatrix} h_0 & 2(h_1 + h_0) & h_1 & 0 & 0 \\ 0 & h_1 & 2(h_2 + h_1) & h_2 & 0 \\ 0 & 0 & h_2 & 2(h_3 + h_2) & h_3 \end{pmatrix} \begin{pmatrix} f''_0 \\ f''_1 \\ f''_2 \\ f''_3 \\ f''_4 \end{pmatrix} = \begin{pmatrix} 6\left(\frac{f_2 - f_1}{h_1} - \frac{f_1 - f_0}{h_0}\right) \\ 6\left(\frac{f_3 - f_2}{h_2} - \frac{f_2 - f_1}{h_1}\right) \\ 6\left(\frac{f_4 - f_3}{h_3} - \frac{f_3 - f_2}{h_2}\right) \end{pmatrix}$$

On obtient un système de 3 équations à 5 inconnues ($f''_0, f''_1, f''_2, f''_3$ et f''_4), il faut donc compléter ce système par 2 autres relations afin de pouvoir le résoudre. On va donc imposer les valeurs des dérivées secondes nulles aux deux extrémités, soit $f''(x_0) = 0$ et $f''(x_5) = 0$. On qualifie alors la courbe obtenue de spline naturelle. En rajoutant ces deux équations dans les matrices, on obtient :

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ h_0 & 2(h_1 + h_0) & h_1 & 0 & 0 \\ 0 & h_1 & 2(h_2 + h_1) & h_2 & 0 \\ 0 & 0 & h_2 & 2(h_3 + h_2) & h_3 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f''_0 \\ f''_1 \\ f''_2 \\ f''_3 \\ f''_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 6\left(\frac{f_2 - f_1}{h_1} - \frac{f_1 - f_0}{h_0}\right) \\ 6\left(\frac{f_3 - f_2}{h_2} - \frac{f_2 - f_1}{h_1}\right) \\ 6\left(\frac{f_4 - f_3}{h_3} - \frac{f_3 - f_2}{h_2}\right) \\ 0 \end{pmatrix}$$

On généralise alors à $n + 1$ points:

$$\underbrace{\begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 \\ h_0 & 2(h_1 + h_0) & h_1 & \ddots & \ddots & \vdots \\ 0 & h_1 & 2(h_2 + h_1) & h_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \vdots \\ 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix}}_A \begin{pmatrix} f''_0 \\ f''_1 \\ f''_2 \\ f''_3 \\ \vdots \\ f''_{n-1} \\ f''_n \end{pmatrix} = \underbrace{\begin{pmatrix} 0 \\ 6\left(\frac{f_2-f_1}{h_1} - \frac{f_1-f_0}{h_0}\right) \\ 6\left(\frac{f_3-f_2}{h_2} - \frac{f_2-f_1}{h_1}\right) \\ \vdots \\ \vdots \\ 6\left(\frac{f_n-f_{n-1}}{h_{n-1}} - \frac{f_{n-1}-f_{n-2}}{h_{n-2}}\right) \\ 0 \end{pmatrix}}_G$$

On peut remarquer que l'on obtient un système de la forme $AX=G$ de dimension $(n+1)$. Pour le résoudre, nous utilisons l'algorithme de Thomas.

Algorithme de Thomas: La résolution du système $AX=G$ est alors ramenée à la résolution successive des systèmes $UX=Y$ et $LY=G$, pour ensuite avoir $LUX=G$, ce qui équivaut à $AX=G$.

Calcul de L et U: Soit la matrice A, avec des coefficients simplifiés :

$$A = \begin{pmatrix} s_1 & w_1 & 0 & \dots & \dots & 0 \\ e_2 & s_2 & w_2 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & e_{n-1} & s_{n-1} & w_{n-1} \\ 0 & \dots & \dots & 0 & e_n & s_n \end{pmatrix}$$

On cherche des matrices L et U suivantes tel que $A=LU$:

$$L = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ \beta_2 & 1 & \ddots & \ddots & \vdots \\ 0 & \beta_3 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & 0 & \beta_n & 1 \end{pmatrix} \quad U = \begin{pmatrix} \alpha_1 & w_1 & 0 & \dots & 0 \\ 0 & \alpha_2 & w_2 & \ddots & \vdots \\ \vdots & \ddots & \alpha_3 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & w_{n-1} \\ 0 & \dots & \dots & 0 & \alpha_n \end{pmatrix}$$

On cherche ensuite les coefficients α_i et β_i .

Pour cela, écrivons la matrice :

$$LU = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ \beta_2 & 1 & \ddots & \ddots & \vdots \\ 0 & \beta_3 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & 0 & \beta_n & 1 \end{pmatrix} \begin{pmatrix} \alpha_1 & w_1 & 0 & \dots & 0 \\ 0 & \alpha_2 & w_2 & \ddots & \vdots \\ \vdots & \ddots & \alpha_3 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & w_{n-1} \\ 0 & \dots & \dots & 0 & \alpha_n \end{pmatrix} = \begin{pmatrix} \alpha_1 & w_1 & 0 & \dots & 0 \\ \alpha_1\beta_2 & w_1\beta_2 + \alpha_2 & \ddots & \ddots & \vdots \\ 0 & \alpha_2\beta_3 & w_2\beta_3 + \alpha_3 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \alpha_{n-1}\beta_n & w_{n-1}\beta_n + \alpha_n \end{pmatrix}$$

Par identification avec A, on obtient les relations suivantes:

$$A = \begin{pmatrix} s_1 & w_1 & 0 & \dots & \dots & 0 \\ e_2 & s_2 & w_2 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & e_{n-1} & s_{n-1} & w_{n-1} \\ 0 & \dots & \dots & 0 & e_n & s_n \end{pmatrix}$$

$$\alpha_1 = s_1; \alpha_{i-1}\beta_i = e_i \Leftrightarrow \beta_i = \frac{e_i}{\alpha_{i-1}}; w_{i-1}\beta_i + \alpha_i = s_i \Leftrightarrow \alpha_i = s_i - \beta_i w_{i-1} \quad i = 2, \dots, n$$

Nous pouvons alors trouver X en résolvant successivement les systèmes données par les équations de matrices $UX=Y$ et $LY=G$. En développant:

$$UX = \underbrace{\begin{pmatrix} \alpha_1 & w_1 & 0 & \dots & 0 \\ 0 & \alpha_2 & w_2 & \ddots & \vdots \\ \vdots & \ddots & \alpha_3 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & w_{n-1} \\ 0 & \dots & \dots & 0 & \alpha_n \end{pmatrix}}_U \underbrace{\begin{pmatrix} f''_1 \\ f''_2 \\ f''_3 \\ \vdots \\ f''_{n-1} \\ f''_n \end{pmatrix}}_X = \underbrace{\begin{pmatrix} \alpha_1 f''_1 + w_1 f''_2 \\ \alpha_2 f''_2 + w_2 f''_3 \\ \vdots \\ \vdots \\ \alpha_{n-1} f''_{n-1} + w_{n-1} f''_n \end{pmatrix}}_Y$$

On obtient ainsi la relation :

$$UX = Y \Rightarrow y_i = \alpha_i x_i + w_i x_{i+1} \Leftrightarrow x_i = \frac{y_i - w_i x_{i+1}}{\alpha_i} \quad i = 1, \dots, n$$

Maintenant, faisons de même pour $LY=G$:

$$LY = \underbrace{\begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ \beta_2 & 1 & \ddots & \ddots & \vdots \\ 0 & \beta_3 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & 0 & \beta_n & 1 \end{pmatrix}}_L \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ y_{n-1} \\ y_n \end{pmatrix}}_Y = \underbrace{\begin{pmatrix} y_1 \\ \beta_2 y_1 + y_2 \\ \beta_3 y_2 + y_3 \\ \vdots \\ \vdots \\ \beta_n y_{n-1} + y_n \end{pmatrix}}_G$$

On obtient alors une seconde relation :

$$LY = G \Rightarrow y_1 = g_1 ; g_i = \beta_i y_{i-1} + y_i \Leftrightarrow y_i = g_i - \beta_i y_{i-1}, \quad i = 2, \dots, n$$

Nous obtenons ainsi les $f''_i = x_i = \frac{y_i - w_i x_{i+1}}{\alpha_i}$ nécessaires pour calculer le polynôme $s_i(x)$ sur chaque intervalle $x_i; x_{i+1}$ pour $i = 0, 1, \dots, n-1$.

3.4 Exemple de calcul d'une fonction spline cubique

Nous allons prendre ici les 5 points (x_i, f_i) suivants : $(1;6), (2;3), (3;0), (4;2), (5;6)$.

1ère étape: on remplace les coefficients dans la matrice A

On sait que $h_i = x_{i+1} - x_i$. Avec les points choisis ci-dessus, on remarque que $h_i = 1$ pour chacun des 5 points, ce qui simplifie le système d'équations.

On remplace dans A:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ h_0 & 2(h_1 + h_0) & h_1 & 0 & 0 \\ 0 & h_1 & 2(h_2 + h_1) & h_2 & 0 \\ 0 & 0 & h_2 & 2(h_3 + h_2) & h_3 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 \\ 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

2ème étape: on écrit et on simplifie notre système matriciel

On a :

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 \\ 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}}_A \underbrace{\begin{pmatrix} f''_0 \\ f''_1 \\ f''_2 \\ f''_3 \\ f''_4 \end{pmatrix}}_X = \underbrace{\begin{pmatrix} 0 \\ 6(f_2 - 2f_1 + f_0) \\ 6(f_3 - 2f_2 + f_1) \\ 6(f_4 - 2f_3 + f_2) \\ 0 \end{pmatrix}}_G$$

On obtient $f''_0 = f''_4 = 0$. On réduit nos matrices afin de simplifier les calculs:

$$\underbrace{\begin{pmatrix} 4 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 4 \end{pmatrix}}_A \underbrace{\begin{pmatrix} f''_1 \\ f''_2 \\ f''_3 \end{pmatrix}}_X = \underbrace{\begin{pmatrix} 0 \\ 30 \\ 12 \end{pmatrix}}_G \quad (1)$$

3ème étape: on résout notre système d'équation directement ou en utilisant l'algorithme de Thomas

Méthode 1 : On résout directement notre système:

$$(1) \Leftrightarrow \begin{cases} 4f''_1 + f''_2 = 0 \\ f''_1 + 4f''_2 + f''_3 = 30 \\ f''_2 + 4f''_3 = 12 \end{cases} \Leftrightarrow \begin{cases} f''_1 = -\frac{f''_2}{4} \\ -\frac{f''_2}{4} + 4f''_2 + 3 - \frac{f''_2}{4} = 30 \\ f''_3 = 3 - \frac{f''_2}{4} \end{cases} \Leftrightarrow \begin{cases} f''_1 = -\frac{f''_2}{4} \\ -\frac{f''_2}{2} + \frac{8f''_2}{2} = 27 \\ f''_3 = 3 - \frac{f''_2}{4} \end{cases} \Leftrightarrow \begin{cases} f''_1 = -\frac{f''_2}{4} \\ \frac{7f''_2}{2} = 27 \\ f''_3 = 3 - \frac{f''_2}{4} \end{cases}$$

$$\Leftrightarrow \begin{cases} f''_1 = -\frac{27}{14} \\ f''_2 = \frac{54}{7} \\ f''_3 = \frac{15}{14} \end{cases}$$

À partir de ces résultats et des hypothèses, on obtient:

$$\begin{cases} f''_0 = 0 \\ f''_1 = -\frac{27}{14} \\ f''_2 = \frac{54}{7} \\ f''_3 = \frac{15}{14} \\ f''_4 = 0 \end{cases}$$

Méthode 2 : On utilise l'algorithme de Thomas :

On a les matrices suivantes :

$$A = \begin{pmatrix} 4 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 4 \end{pmatrix} = \begin{pmatrix} s_1 & w_1 & 0 \\ e_2 & s_2 & w_2 \\ 0 & e_3 & s_3 \end{pmatrix} \quad L = \begin{pmatrix} 1 & 0 & 0 \\ \beta_2 & 1 & 0 \\ 0 & \beta_3 & 1 \end{pmatrix} \quad U = \begin{pmatrix} \alpha_1 & c_1 & 0 \\ 0 & \alpha_2 & c_2 \\ 0 & 0 & \alpha_3 \end{pmatrix}$$

Appliquons les relations sur les coefficients α_i et β_i .

Les relations trouvées précédemment sont :

$$\alpha_1 = s_1, \beta_i = \frac{e_i}{\alpha_{i-1}}, \alpha_i = s_i - \beta_i w_{i-1} \quad i = 2, \dots, n$$

On trouve ici

$$\alpha_1 = s_1 = 4$$

$$\begin{aligned} \beta_2 &= \frac{e_2}{\alpha_1} = \frac{1}{4} & \alpha_2 &= s_2 - \beta_2 w_1 = 4 - \frac{1}{4} * 1 = \frac{15}{4} \\ \beta_3 &= \frac{e_3}{\alpha_2} = \frac{1}{\frac{15}{4}} = \frac{4}{15} & \alpha_3 &= s_3 - \beta_3 w_2 = 4 - \frac{4}{15} * 1 = \frac{56}{15} \end{aligned}$$

Finalement

$$L = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ 0 & \frac{4}{15} & 1 \end{pmatrix} \quad U = \begin{pmatrix} 4 & 1 & 0 \\ 0 & \frac{15}{4} & 1 \\ 0 & 0 & \frac{56}{15} \end{pmatrix}$$

On résout ensuite successivement les systèmes LY=G et UX=Y.
On pose

$$X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \quad Y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

$$LY = G \Leftrightarrow \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ 0 & \frac{4}{15} & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 30 \\ 12 \end{pmatrix}$$

$$\Leftrightarrow \begin{cases} y_1 = 0 \\ \frac{1}{4}y_1 + y_2 = 30 \\ \frac{4}{15}y_2 + y_3 = 12 \end{cases}$$

$$\Leftrightarrow \begin{cases} y_1 = 0 \\ y_2 = 30 \\ \frac{4}{15} * 30 + y_3 = 12 \end{cases}$$

$$\Leftrightarrow \begin{cases} y_1 = 0 \\ y_2 = 30 \\ y_3 = 12 - 8 \end{cases}$$

$$\Leftrightarrow Y = \begin{pmatrix} 0 \\ 30 \\ 4 \end{pmatrix}$$

$$\begin{aligned}
UX = Y &\Leftrightarrow \begin{pmatrix} 4 & 1 & 0 \\ 0 & \frac{15}{4} & 1 \\ 0 & 0 & \frac{56}{15} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 30 \\ 4 \end{pmatrix} \\
&\Leftrightarrow \begin{cases} 4x_1 + x_2 = 0 \\ \frac{15}{4}x_2 + x_3 = 30 \\ \frac{56}{15}x_3 = 4 \end{cases} \\
&\Leftrightarrow \begin{cases} x_2 = -4x_1 \\ \frac{15}{4} * (-4x_1) + x_3 = 30 \\ \frac{56}{15}x_3 = 4 \end{cases} \\
&\Leftrightarrow \begin{cases} x_2 = -4x_1 \\ x_3 = 30 + 15x_1 \\ \frac{56}{15} * (30 + 15x_1) = 4 \end{cases} \\
&\Leftrightarrow \begin{cases} x_2 = -4x_1 \\ x_3 = 30 + 15x_1 \\ 56x_1 = 4 - 112 \end{cases} \\
&\Leftrightarrow \begin{cases} x_2 = -4 * (-\frac{27}{14}) \\ x_3 = 30 + 15 * (-\frac{27}{14}) \\ x_1 = -\frac{27}{14} \end{cases} \\
&\Leftrightarrow \begin{cases} x_2 = \frac{54}{7} \\ x_3 = \frac{15}{14} \\ x_1 = -\frac{27}{14} \end{cases} \\
&\Leftrightarrow X = \begin{pmatrix} -\frac{27}{14} \\ \frac{54}{7} \\ \frac{15}{14} \end{pmatrix}
\end{aligned}$$

D'où :

$$\left\{ \begin{array}{l} f_0'' = 0 \\ f_1'' = -\frac{27}{14} \\ f_2'' = \frac{54}{7} \\ f_3'' = \frac{15}{14} \\ f_4'' = 0 \end{array} \right.$$

4ème étape: on calcul les splines

Nous pouvons ensuite construire les splines grâce à la formule :

$$s_i(x) = f_i'' \frac{(x_{i+1} - x)^3}{6h_i} + f_{i+1}'' \frac{(x - x_i)^3}{6h_i} + a_i(x_{i+1} - x) + b_i(x - x_i) \quad i = 0, 1, \dots, n-1 \text{ et } h_i = x_{i+1} - x_i$$

avec $a_i = \frac{f_i}{h_i} - f_i'' \frac{h_i}{6}$ et $b_i = \frac{f_{i+1}}{h_i} - f_{i+1}'' \frac{h_i}{6}$

Nous rappelons 5 points choisis pour cet exemple :

$$(1 ; 6), (2 ; 3), (3 ; 0), (4 ; 2), (5 ; 6)$$

$$(x_0 ; f_0), (x_1 ; f_1), (x_2 ; f_2), (x_3 ; f_3), (x_4 ; f_4)$$

De plus, nous avons dis que $h_i = 1$ pour chacun des 5 points. Ainsi, nous obtenons 4 splines :

Pour $s_0(x)$:

$$s_0(x) = f_0'' \frac{(x_1 - x)^3}{6h_0} + f_1'' \frac{(x - x_0)^3}{6h_0} + a_0(x_1 - x) + b_0(x - x_0)$$

On remarque $f_0'' = 0$. On remplace les expressions de a_i et b_i pour $i = 0$:

$$s_0(x) = 0 + f_1'' \frac{(x - x_0)^3}{6h_0} + \left(\frac{f_0}{h_0} - f_0'' \frac{h_0}{6} \right)(x_1 - x) + \left(\frac{f_1}{h_0} - f_1'' \frac{h_0}{6} \right)(x - x_0)$$

$$s_0(x) = -\frac{27}{14} \frac{(x - 1)^3}{6} + (6 - 0)(2 - x) + (3 - (-\frac{27}{14}) \frac{1}{6})(x - 1)$$

$$s_0(x) = -\frac{27}{14} \frac{(x - 1)^3}{6} + 6(2 - x) + 3(x - 1) + \frac{9}{28}(x - 1)$$

$$s_0(x) = -\frac{27}{14} \frac{(x - 1)^3}{6} + 6(2 - x) + 3(x - 1) + \frac{9}{28}(x - 1)$$

$$s_0(x) = -\frac{27}{14} \frac{(x - 1)^3}{6} + 6(2 - x) + \frac{93}{28}(x - 1)$$

Pour $s_1(x)$:

$$\begin{aligned}
s_1(x) &= f_1'' \frac{(x_2 - x)^3}{6h_1} + f_2'' \frac{(x - x_1)^3}{6h_1} + a_1(x_2 - x) + b_1(x - x_1) \\
s_1(x) &= -\frac{27}{14} \frac{(3 - x)^3}{6} + \frac{54}{7} \frac{(x - 2)^3}{6} + a_1(3 - x) + b_1(x - 2) \\
s_1(x) &= -\frac{27}{14} \frac{(3 - x)^3}{6} + \frac{54}{7} \frac{(x - 2)^3}{6} + (\frac{f_1}{h_1} - f_1'' \frac{h_1}{6})(3 - x) + (\frac{f_2}{h_1} - f_2'' \frac{h_1}{6})(x - 2) \\
s_1(x) &= -\frac{27}{14} \frac{(3 - x)^3}{6} + \frac{54}{7} \frac{(x - 2)^3}{6} + (3 - (-\frac{27}{14} \frac{1}{6}))(3 - x) + (\frac{0}{1} - \frac{54}{7} \frac{1}{6})(x - 2) \\
s_1(x) &= -\frac{27}{14} \frac{(3 - x)^3}{6} + \frac{54}{7} \frac{(x - 2)^3}{6} + (3 + \frac{9}{28})(3 - x) - \frac{9}{7}(x - 2) \\
s_1(x) &= -\frac{27}{14} \frac{(3 - x)^3}{6} + \frac{54}{7} \frac{(x - 2)^3}{6} + \frac{93}{28}(3 - x) - \frac{9}{7}(x - 2)
\end{aligned}$$

Pour $s_2(x)$:

$$\begin{aligned}
s_2(x) &= f_2'' \frac{(x_3 - x)^3}{6h_2} + f_3'' \frac{(x - x_2)^3}{6h_2} + a_2(x_3 - x) + b_2(x - x_2) \\
s_2(x) &= \frac{54}{7} \frac{(4 - x)^3}{6} + \frac{15}{14} \frac{(x - 3)^3}{6} + a_2(4 - x) + b_2(x - 3) \\
s_2(x) &= \frac{54}{7} \frac{(4 - x)^3}{6} + \frac{15}{14} \frac{(x - 3)^3}{6} + (\frac{f_2}{h_2} - f_2'' \frac{h_2}{6})(4 - x) + (\frac{f_3}{h_2} - f_3'' \frac{h_2}{6})(x - 3) \\
s_2(x) &= \frac{54}{7} \frac{(4 - x)^3}{6} + \frac{15}{14} \frac{(x - 3)^3}{6} + (\frac{0}{1} - \frac{54}{7} \frac{1}{6})(4 - x) + (2 - \frac{15}{14} \frac{1}{6})(x - 3) \\
s_2(x) &= \frac{54}{7} \frac{(4 - x)^3}{6} + \frac{15}{14} \frac{(x - 3)^3}{6} + (-\frac{54}{7} \frac{1}{6})(4 - x) + (2 - \frac{15}{14} \frac{1}{6})(x - 3) \\
s_2(x) &= \frac{54}{7} \frac{(4 - x)^3}{6} + \frac{15}{14} \frac{(x - 3)^3}{6} - \frac{9}{7}(4 - x) + (\frac{51}{28})(x - 3)
\end{aligned}$$

Pour $s_3(x)$:

$$s_3(x) = f_3'' \frac{(x_4 - x)^3}{6h_3} + f_4'' \frac{(x - x_3)^3}{6h_3} + a_3(x_4 - x) + b_3(x - x_3)$$

$$s_3(x) = \frac{15}{14} \frac{(x_4 - x)^3}{6h_3} + 0 + a_3(5 - x) + b_3(x - 4)$$

$$s_3(x) = \frac{15}{14} \frac{(x_4 - x)^3}{6h_3} + \left(\frac{f_3}{h_3} - f_3'' \frac{h_3}{6}\right)(5 - x) + \left(\frac{f_4}{h_3} - f_4'' \frac{h_3}{6}\right)(x - 4)$$

$$s_3(x) = \frac{15}{14} \frac{(5 - x)^3}{6} + \left(2 - \frac{15}{14} \frac{1}{6}\right)(5 - x) + (6 - 0)(x - 4)$$

$$s_3(x) = \frac{15}{14} \frac{(5 - x)^3}{6} + \frac{51}{28}(5 - x) + (6 - 0)(x - 4)$$

$$s_3(x) = \frac{15}{14} \frac{(5 - x)^3}{6} + \frac{51}{28}(5 - x) + 6(x - 4)$$

Lorsque l'on compare les valeurs théoriques au valeur trouvées par le programme, il y a évidemment une erreur d'arrondi, mais les valeurs correspondent. Nous avons représenté le résultat sur Géogébra, où l'on peut voir que l'on a relié de façon lisse et continue nos points.

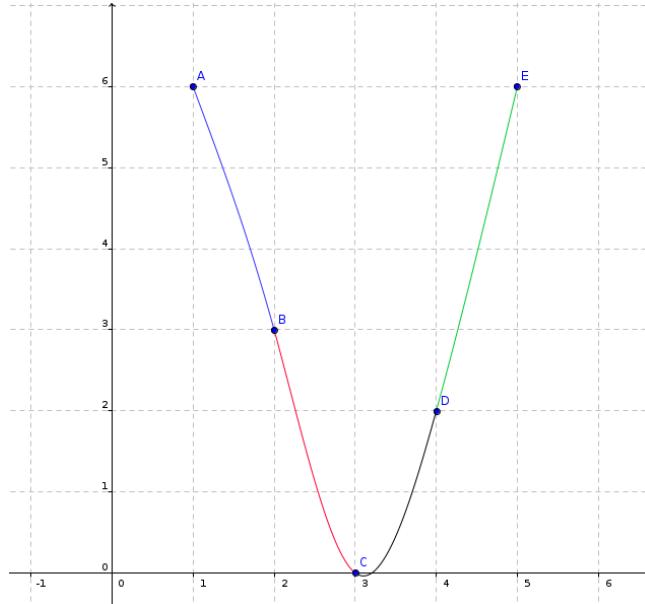


Figure 5: Exemple d'interpolation par spline cubique sur Geogebra

3.5 Présentation de l'outil informatique

Nous avons eu accès à un programme pascal permettant de calculer des splines en entrant le nombre de points que nous voulions raccorder ainsi que leurs coordonnées respectives. Afin d'avoir une meilleur ergonomie et une meilleure lisibilité, nous avons remanié le programme.

Voici donc le programme que nous avons utilisé.

3.5.1 Analyse descendante

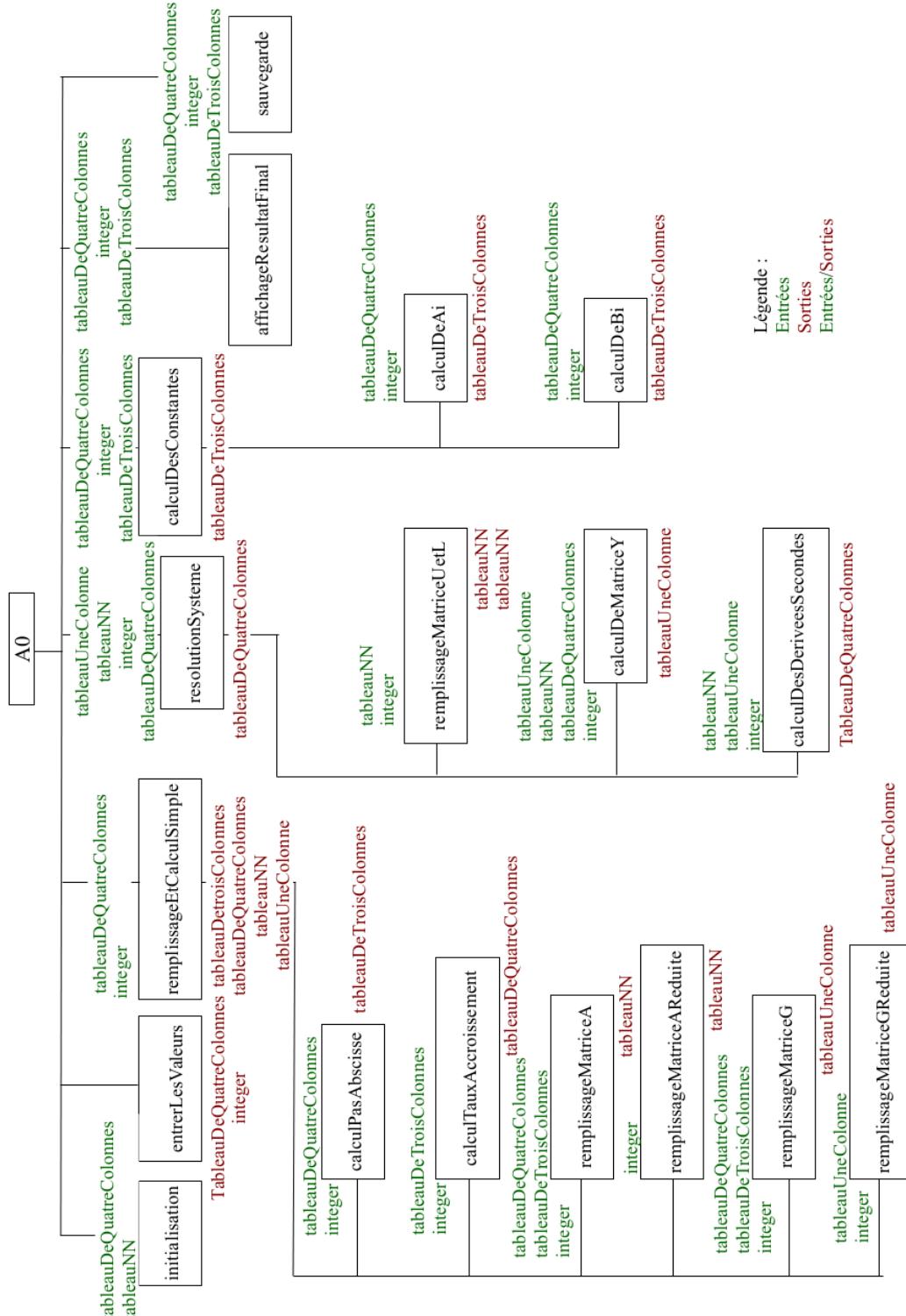


Figure 6: Analyse descendante du programme

3.5.2 Présentation du contenu du programme

- o *Function fileExists (nomDuFichier : String) : Boolean;*
Renvoie un booléen vrai si le fichier existe.
- o *Procedure sauvegarde (donneesAuPoint : tableauDeQuatreColonnes; nombrePoint : integer; donneesEntreDeuxPoints : tableauDeTroisColonnes);*
Permet de sauvegarder les résultats du programme dans un fichier .txt dans le répertoire du fichier exécutable.
- o *Procedure initialisation (Var donneesAuPoint : tableauDeQuatreColonnes; Var matriceA : tableauNN);*
Initialise les tableaux *donneesAuPoint* et *matriceA* à zéro.
- o *Procedure entrerLesValeurs (Var donneesAuPoint : tableauDeQuatreColonnes; Var nombrePoint : integer);*
Permet d'entrer le nombre de points et leurs coordonnées respectives.
- o *Procedure affichageResultatFinal (donneesAuPoint : tableauDeQuatreColonnes; nombrePoint : Integer; donneesEntreDeuxPoints : tableauDeTroisColonnes);*
Affiche les résultats à l'écran.
- o *Procedure calculPasAbscisse (Var donneesEntreDeuxPoints : tableauDeTroisColonnes; donneesAuPoint : tableauDeQuatreColonnes ; nombrePoint : Integer);*
La procédure calcule le pas, noté *h* dans la résolution mathématique entre les abscisses des points $[x_i, x_{i+1}]$.
- o *Procedure calculTauxAccroissement (Var donneesAuPoint : tableauDeQuatreColonnes; donneesEntreDeuxPoints : tableauDeTroisColonnes; nombrePoint : Integer);*
La procédure permet de calculer le taux d'accroissement à partir des valeurs entrées.
- o *Procedure remplissageMatriceA (Var matriceA : tableauNN; donneesAuPoint : tableauDeQuatreColonnes; donneesEntreDeuxPoints : tableauDeTroisColonnes; nombrePoint : integer);*
Cette procédure simplifie le système en le transposant dans une matrice.

- o *Procedure remplissageMatriceAReduite (Var matriceA, matriceAR : tableauNN; nombrePoint : Integer);*

Dans la matrice A initiale, on a plusieurs lignes composées de 1 et de 0. On peut donc réduire la matrice aux lignes correspondantes aux valeurs des f_i'' non nulles. On réécrit ainsi les lignes du "milieu" de la matrice A dans la matrice AR (A Réduite).

- o *Procedure remplissageMatriceG (Var matriceG : tableauUneColonne; donneesAuPoint : tableauDeQuatreColonnes; donneesEntreLesPoints : tableauDeTroisColonnes; nombrePoint : Integer);*

On a la matrice G résultante de A multipliée par les dérivées seconde.

- o *Procedure remplissageMatriceGReduite (Var matriceG, matriceGR : tableauUneColonne; nombrePoint : Integer);*

On effectue le même principe qu'avec la matrice A pour la matrice G : On a des valeurs de f_i'' nulles. On réduit alors la matrice G.

- o *Procedure remplissageEtCalculSimple (Var donneesEntreDeuxPoints : tableauDeTroisColonnes; Var donneesAuPoint : tableauDeQuatreColonnes; nombrePoint : Integer; Var matriceAR : tableauNN; Var matriceGR : tableauUneColonne);*

Cette procédure appelle les procédures précédentes pour calculer le pas, le taux d'accroissement, les matrices A et G réduites.

- o *Procedure calculDeAi (donneesAuPoint : tableauDeQuatreColonnes; nombrePoint : Integer; Var donneesEntreDeuxPoints : tableauDeTroisColonnes);*

On calcule ici les coefficients a_i .

- o *Procedure calculDeBi (donneesAuPoint : tableauDeQuatreColonnes; nombrePoint : Integer; Var donneesEntreDeuxPoints : tableauDeTroisColonnes);*

On calcule ici les coefficients b_i .

- o *Procedure calculDesConstantes (donneesAuPoint : tableauDeQuatreColonnes; nombrePoint : Integer; Var donneesEntreDeuxPoints : tableauDeTroisColonnes);*

Appelle les deux procédures précédentes : *calculDeAi* et *calculDeBi*.

- o *Procedure remplissageMatriceUetL (Var matriceL : tableauNN; Var matriceU : tableauNN; matriceAR : tableauNN; nombrePoint : Integer);*

On cherche ici à décomposer A en deux matrices bidiagonales telles que $A=UL$.

- o *Procedure calculDeMatriceY (Var matriceY, matriceGR : tableauUneColonne; matriceL : tableauNN; donneesAuPoint : tableauDeQuatreColonnes; nombrePoint : Integer);*

Cette procédure permet de calculer la matrice Y qui va être multipliée par L.

- o *Procedure calculDesDeriveesSecondes (Var donneesAuPoint : tableauDeQuatreColonnes; matriceU : tableauNN; matriceY : tableauUneColonne; nombrePoint : Integer);*

Cette procédure permet de calculer les dérivées secondes de f .

- o *Procedure resolutionSystème (matriceG : tableauUneColonne; matriceA : tableauNN; nombrePoint : Integer; Var donneesAuPoint : tableauDeQuatreColonnes);*

Ici, on appelle les procédures de remplissage de U et L, de calcul de la matrice Y et de calcul des dérivées pour résoudre le système.

Le programme principal exécute finalement les fonctions et les procédures pour calculer les fonctions splines cubiques. A l'exécution du programme, un répertoire nommé « Résultats » est créé s'il n'existe pas encore, dans le répertoire où se trouve le programme.

4 B-splines

4.1 Contexte

Les courbes de Bézier sont totalement modifiées si un changement de point de contrôle est effectué. Pour cette raison la méthode de ces courbes peut être qualifiée de méthode globale.

Les courbes de B-splines ont été définies par Cox et De Boor dans les années 1970 et développées par Boeing dans les années 1980. Ils cherchaient un moyen pour utiliser les courbes de Bézier sans ces inconvénients. Boeing a donc remplacé les polynômes de Bernstein (B_i^k) par des fonctions de base des B-splines ($B_{i,k}$). Ainsi, les courbes de B-splines sont déterminées par ces fonctions de base, des points de contrôle et un vecteur noeud. Les courbes de B-splines sont stables numériquement, ayant des coefficients toujours positifs. De plus elles sont faciles à interpréter géométriquement. La variation d'un point de contrôle modifie seulement une partie limitée de la courbe, ce qui représente un grand avantage pour la Conception Assistée par Ordinateur. Alors, la méthode de B-splines est considérée comme une méthode locale.

De plus, les courbes de B-splines, étant équivalentes aux courbes de Bézier continues sur C^2 , sont les plus utilisées dans l'interpolation des splines unidimensionnelles.

NURBS

Les B-splines rationnelles non-uniformes, ou NURBS de leur acronyme anglais (*Non-Uniform Rational Basis Splines*), donnent la possibilité de représenter facilement des courbes qui ne peuvent pas être construites par les B-splines uniformes. Les points de contrôle étant des rationnels, l'utilisation de ces courbes permet de réduire le nombre de noeuds d'une approximation tout en conservant une bonne précision pour un arc ou une facette de courbe quelconque.

Après avoir effectué toutes les transformations affines (translation, rotation, homothéties) et aussi quelques transformations non isométriques comme les projections ou la perspective, les NURBS conservent toutes ces propriétés, et même certaines caractéristiques physiques essentielles comme la continuité, la préservation des tangentes ou angles aux sommets.

Grâce à sa grande précision, les NURBS sont utilisées dans la cartographie. Étant compatibles avec la plupart des projections dans ce domaine, les NURBS interpolent des distances et des angles à partir d'un volume réduit de mesures géodésiques.

En outre, les NURBS sont une partie indispensable dans les systèmes de CAO les plus récents : ceci est dû à leur efficacité et à leur précision dans la représentation des images en 2 et 3 dimensions. Et elles sont aussi exploitées dans la majorité des instruments de mesures de haute précision comme les miroirs et lentilles utilisés en astronomie.

En conclusion, les B-splines et les NURBS sont les méthodes d'interpolation

les plus précises, mais elles sont encore en développement.

4.2 Les bases des B-splines

On prend le vecteur noeud $T = (t_0, t_1, \dots, t_m)$ avec $t_i \leq t_{i+1}$, $i \in [0, m - 1]$ et $m \in \mathbb{N}$. On choisit aussi les points de contrôle P_0, P_1, \dots, P_m et on note $P_i = \overrightarrow{OP_i}$. On doit élaborer une courbe $X_0(t)$ avec $X_0(t_i) = P_i$ pour tout $i \in [0, m]$. La courbe est alors définie par :

$$X_0(t) = \sum_{i=0}^n B_{i,0}(t).P_i$$

Par contre, les points obtenus à l'aide de cette fonction ne seront pas suffisants. En effet, la courbe sera discontinue (avec des sauts). C'est pour cette raison que l'on cherche à approcher $X_0(t)$ par une courbe linéaire par morceaux $X_1(t)$, où t varie de t_i à t_{i+1} , qui relie les points P_{i-1} et P_i .

$$X_1(t) = \left(\frac{t_{i+1} - t}{t_{i+1} - t_i} \right) P_{i-1} + \left(\frac{t - t_i}{t_{i+1} - t_i} \right) P_i, \text{ pour } t \in [t_i, t_{i+1}] \quad (1)$$

Donc on a :

$$X_1(t_i) = P_{i-1}$$

$$X_1(t_{i+1}) = P_i$$

Si t_0, \dots, t_m sont différents, alors $X_1(t)$ est continue et peut s'écrire sous la forme:

$$X_1(t) = \sum_{i=0}^n B_{i,1}(t).P_i$$

Mais $B_{i,1}(t)$ n'est pas définie.

D'où, pour $t \in [0, m]$, l'équation (1) nous donne :

$$\begin{aligned} X_1(t) &= \sum_i \left(\frac{t_{i+1} - t}{t_{i+1} - t_i} \cdot 1_{[t_i, t_{i+1}]} \cdot P_{i-1} + \frac{t - t_i}{t_{i+1} - t_i} \cdot 1_{[t_i, t_{i+1}]} \cdot P_i \right) \\ &= \sum_i \frac{t_{i+1} - t}{t_{i+1} - t_i} \cdot 1_{[t_i, t_{i+1}]} \cdot P_{i-1} + \sum_i \frac{t - t_i}{t_{i+1} - t_i} \cdot 1_{[t_i, t_{i+1}]} \cdot P_i \end{aligned}$$

On pose $i' = i - 1$ pour la première somme:

$$\begin{aligned}
X_1(t) &= \sum_{i'} \left(\frac{t_{i'+2} - t}{t_{i'+2} - t_{i'+1}} \cdot 1_{[t_{i'+1}, t_{i'+2}[} \cdot P_{i'} \right) + \sum_i \left(\frac{t - t_i}{t_{i+1} - t_i} \cdot 1_{[t_i, t_{i+1}[} \cdot P_i \right) \\
&= \sum_i \left(\frac{t_{i+2} - t}{t_{i+2} - t_{i+1}} \cdot 1_{[t_{i+1}, t_{i+2}[} + \frac{t - t_i}{t_{i+1} - t_i} \cdot 1_{[t_i, t_{i+1}[} \right) \cdot P_i \\
&= \sum_i \left(\frac{t_{i+2} - t}{t_{i+2} - t_{i+1}} \cdot B_{i+1,0}(t) + \frac{t - t_i}{t_{i+1} - t_i} \cdot B_{i,0}(t) \right) \cdot P_i
\end{aligned}$$

Alors:

$$B_{i,1}(t) = \frac{t_{i+2} - t}{t_{i+2} - t_{i+1}} \cdot B_{i+1,0}(t) + \frac{t - t_i}{t_{i+1} - t_i} \cdot B_{i,0}(t)$$

Donc la fonction de degré 1, $X_1(t) = \sum_{i=0}^n B_{i,1}(t) \cdot P_i$ est définie en utilisant $X_0(t) = \sum_{i=0}^n B_{i,0}(t) \cdot P_i$, de degré 0. De la même façon, en généralisant, on trouve:

$$X_k(t) = \sum_i B_{i,k}(t) \cdot P_i$$

les P_i étant les points de contrôle de la courbe. Il y a évidemment le même nombre de $B_{i,k}$ que de P_i soit de P_0 à P_n . Le vecteur noeud (t_0, t_1, \dots, t_m) sera tel que $m = n + k + 1$ étant donné que les $B_{i,k}$ s'étalent sur $k + 1$ noeuds et

$$B_{i,k}(t) = \frac{t_{i+1+k} - t}{t_{i+1+k} - t_{i+1}} B_{i+1,k-1}(t) + \frac{t - t_i}{t_{i+k} - t_i} B_{i,k-1}(t)$$

$B_{i,k}(t)$ est une fonction de base d'une B-spline, qui est généralement appelée **base de B-splines**.

Dans la partie qui suit, on va s'intéresser aux B-splines de degré 3, bien que les démarches soient généralisables.

4.3 Courbe de degré 3 (k=3)

Si on veut placer n points de contrôle, on doit déterminer $n + 4$ noeuds. Il nous faut aussi un vecteur noeud. Par exemple, on prend le vecteur $(1, 2, 3, 4, 5, 6, 7)$ et

on définit les points de contrôle $(1, 1), (1, 3), (2, 4), (3, 1)$. Après des calculs et en variant t de 0 à 7, on obtient la courbe en rouge. On remarque que cette courbe a une forme particulière. Elle commence et finit à l'origine du repère et elle ne passe pas par le premier et le dernier point de contrôle. Cela peut s'expliquer par le fait que, en $t = 0$ et en $t = 7$, les $B_{i,3}$ sont nulles et par conséquent la courbe de B-splines est aussi nulle. De plus, pour éviter cet inconvénient, on fait varier t entre 3 et 4 tel que $B_{i,3}$ est égale à 1. On obtient la partie de la courbe colorée en vert, qui est relativement plus proche de la courbe modélisée en bleu par les points de contrôle P_0 à P_3 .

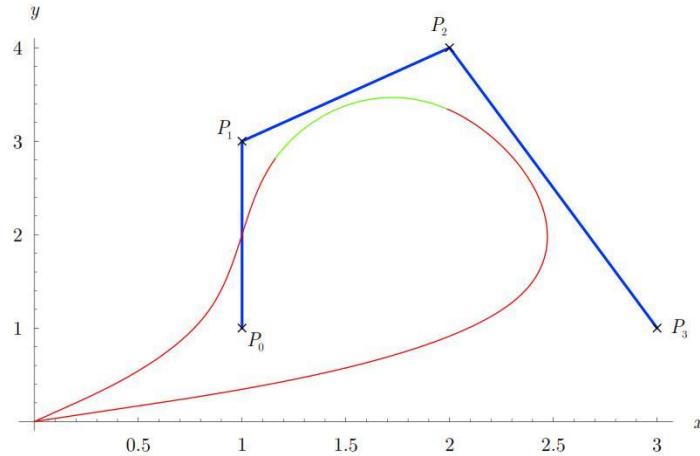


Figure 7: Courbe de B-spline de degré 3

Répétition des nœuds On peut, sans problème, répéter des nœuds si on veut que la courbe passe par un point précis. Pour les nœuds internes, il suffit par exemple de répéter le point trois fois pour le degré 3 (soit $k = 3$). Pour les points extrêmes, il faut encore une répétition et ce phénomène s'appelle B-splines à nœud ouvert. Par contre, avec cette action, la fonction de la courbe n'est plus dérivable, ce qui s'explique par le fait qu'une fonction de base $B_{i,k}$ est dérivable $k - r$ fois, où r est la multiplicité d'un point.

Répétition des points de contrôle La répétition des points 3 fois implique aussi que la courbe passe par un point précis. Par exemple, en prenant 8 nœuds distincts et en choisissant $P_1 = P_2 = P_3$, on calcul $X_3(t_4)$:

$$X_3(t_4) = B_{1,3} \cdot P_1 + B_{2,3} \cdot P_2 + B_{3,3} \cdot P_3$$

Les autres bases $B_{4,3}, B_{5,3}, \dots$ ne figurent pas dans la formule car en t_4 , elles sont égales à 0.

$$X_3(t_4) = (B_{1,3} + B_{2,3} + B_{3,3}).P_1$$

Où $B_{1,3} + B_{2,3} + B_{3,3} = 1$, donc $X_3(t_4) = P_1$

Par contre, la fonction de la courbe reste indérivable après la répétition des points.

4.4 Contrôle local

Modification d'un point de contrôle (exemple pour $k = 3$) Le fait de changer un point de contrôle P_i modifie la courbe seulement entre t_i et t_{i+4} .

Modification d'un nœud Le changement d'un nœud ne fait varier que les $B_{i,k}$ autour. Pour $k = 3$, les fonctions des courbes modifiées varient entre les bases $B_{i-4,k}$ et $B_{i,k}$ (ce qui n'est pas valable pour les extrémités) et modifie les t de t_{i-4} à t_{i+4} .

4.5 Autres propriétés

Les B-Splines ont beaucoup d'autres propriétés dont:

- la différentiabilité des B-Splines;
- l'ajout d'un nœud sans modifier la courbe;
- l'élévation du degré sans changer la courbe;
- la symétrie des points de contrôle impliquant une courbe symétrique.

4.6 Présentation de l'outil informatique

Les commentaires de ce programme se trouvent dans le fichier bspline.pas. Programme B-splines
Signatures :

- o Function BI0 (I:Integer; X:Real):Real ;
- o Function BI1 (I:Integer; X:Real):Real ;
- o Function BI2 (I:Integer; X:Real):Real ;
- o Function BI3 (I:Integer; X:Real):Real ;
- o Function Bspline (N:Integer; A:Tabulation; X:Real): Real ;
- o Procedure TraceBspline (A:Tabulation; N:Integer; X0, X1: Real) ;
- o Procedure CalculCoeff (N:Integer; var A:Tabulation) ;
- o Procedure Init (NP, C:Integer; XD, XF:Real) ;

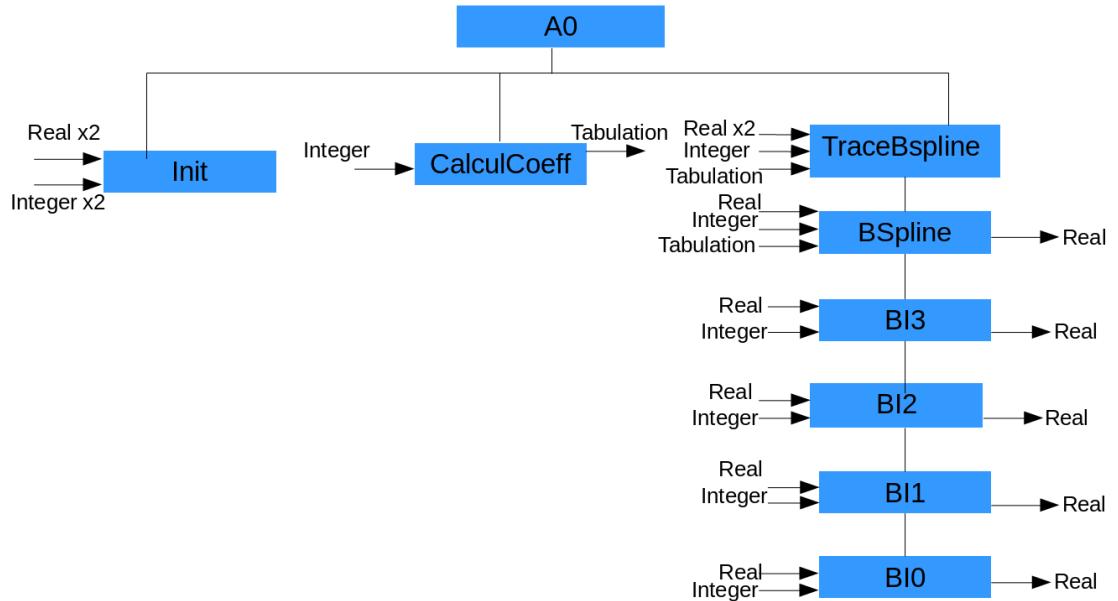


Figure 8: Analyse descendante du programme

5 Avantages et inconvénients

5.1 Courbes de Bézier : avantages et inconvénients

AVANTAGES :

Les courbes de Bézier présentent de nombreux avantages mais aussi des inconvénients que nous allons présenter afin de comprendre l'évolution vers les splines cubiques. Utilisant des polynômes de degrés assez petits, les courbes de Bézier sont faciles à calculer et à tracer sans que cela nuise pour autant à la complexité des formes géométriques représentées. Il est en effet possible de décrire des formes élémentaires variées comme des courbes concaves ou convexes, des inflexions, noeuds ou rebruissements, formes non représentables à l'aide d'un polygone. Une particularité importante des courbes de Bézier concerne la méthode de traçage. Cette technique utilise des points de contrôle et non des points d'interpolation ainsi la courbe ne passe pas par les points donnés- comme c'est le cas pour les interpolations- mais les approche. Cette approximation possède plusieurs avantages :

- les points de contrôle guident géométriquement la courbe, ce qui permet d'obtenir plus facilement une courbe d'aspect naturel.
- Il est possible de prévoir les déformations des courbes grâce à sa stabilité satisfaisante.
- La courbe est facile à modifier en déplaçant les points de contrôle qui sont peu nombreux. Les agrandissements, déformations et toutes les transformations que l'on peut réaliser sur un polynôme (rotation, translation, mise à l'échelle, inclinaison...) sont ainsi applicables.

INCONVENIENTS :

Malheureusement, les courbes de Bézier possèdent aussi des inconvénients non négligeables. Ces courbes ont un contrôle global et manquent de contrôle local. La modification d'un point fait bouger toute la courbe, ce qui est problématique dans des domaines tels que l'industrie automobile : il est gênant que toute la pièce change de forme lorsque nous voulons seulement faire varier une partie de la pièce. Au niveau des calculs, il faudra calculer à nouveau toute la pièce. La courbe ne passant pas par les points de contrôle, il est difficile de la contrôler précisément. Pour réaliser une forme complexe, les courbes de Bézier nécessitent beaucoup de points de contrôle et donc des degrés de courbe plus élevés ce qui les rend difficile à manipuler. La fonction de Runge mets en évidence ce dernier point. L'interpolation polynomiale n'est pas adaptée à l'approximation de fonctions pour grand n.

5.2 Splines cubiques : avantages et inconvénients

AVANTAGES

Les principaux avantages de l'interpolation par spline sont sa stabilité et simplicité de calcul. Une spline cubique étant une fonction obtenue à l'aide de divers polynômes de degrés inférieurs ou égaux à 3. Ceci permet de limiter l'augmentation du degré du polynôme et ainsi éviter le phénomène de Runge (problème d'oscillation de Lagrange non présent). En effet, lorsque l'on augmente n , cela augmente le nombre de morceaux et non le degré des polynômes. De plus, les faibles degrés d'interpolation facilitent l'implémentation, ce qui les rend très populaires. En utilisant des morceaux de polynômes pour relier les points, on améliore aussi la stabilité de la courbe : le changement d'un point n'influence pas toute la courbe. Cet avantage est très important pour les utilisations pratiques, il comble le défaut principal des courbes de Bézier (manque de contrôle local). Pour finir, les courbes sont harmonieuses, sans pointe ni détour... Les interpolations par splines cubiques sont donc très utilisées dans le domaine du design pour approcher des contours complexes sans avoir de cassure du rayon de courbure.

INCONVENIENTS :

Avec l'interpolation par spline cubique, nous devons toutefois faire face à deux inconvénients : la spline peut elle aussi devenir oscillante si les dérivées de la fonction à interpoler deviennent trop grandes (»1) et sa résolution est un peu plus complexe.

5.3 B-splines : avantages et inconvénients

AVANTAGES :

Comme les splines cubiques, les B-splines résolvent les problèmes que nous avons constaté pour les courbes de Bézier. Le contrôle local est possible pour les B-splines et l'ajout de points n'augmente pas le degré de la courbe. Nous pouvons ajouter que le degré du polynôme est indépendant du nombre de points de contrôle.

INCONVENIENTS :

Le principal inconvénient des B-splines est leur complexité. En effet, il n'est pas facile de calculer les fonctions de base. Les points de contrôle ne sont plus

les seuls paramètres des courbes, il y a aussi le vecteur nœud. Il est difficile de gérer les points et les nœuds en même temps. C'est pourquoi nous ne faisons que varier les points de contrôle, les nœuds sont généralement ouverts aux extrémités et uniformes au milieu.

6 Exemples de fonctions

6.1 Fonction de Runge

6.1.1 Introduction

La logique des interpolations consiste à ce que, plus on augmente le nombre de noeuds d'interpolation, plus la courbe tracée sera précise. Or, en 1901, Carle David Tolme Runge, mathématicien allemand, a étudié un résultat différent : il s'agit du phénomène de Runge, associé à sa fonction, qui se retrouve dans le cas de l'interpolation polynomiale de Lagrange. En effet, aux bords d'un intervalle d'interpolation, des oscillations, ne correspondant pas à la fonction recherchée, peuvent apparaître. Ces oscillations augmentent avec le degré du polynôme d'interpolation mais il y a tout de même convergence du polynôme vers la fonction sur un intervalle réduit par rapport au domaine défini initialement.

6.1.2 Exemple

Nous nous sommes intéressés à la fonction de Runge, c'est à dire à $f(x) = \frac{1}{1+25x^2}$. Nous voulons étudier l'allure de cette fonction sur l'intervalle $[-1;1]$. Pour cela, nous avons tracé la fonction sur Open Office Excel à l'aide des points suivants:

x	-1	-0.5	0	0.5	1
$1/(1+25x^2)$	0.0384615385	0.1379310345	1	0.1379310345	0.038461538

Figure 9: Tableau de résultats de la fonction de Runge en fonction des valeurs de x

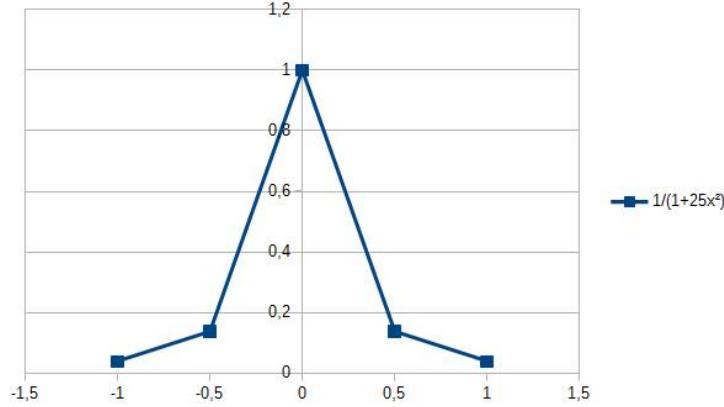


Figure 10: Représentation graphique de la fonction de Runge sur l'intervalle $[-1;1]$

La courbe obtenue à l'aide d'un tableur est une succession de segments de droites donc très linéaire et n'a pas l'allure lisse et courbée attendue de la fonction Runge.

Pour comparer, nous avons utilisé les mêmes coordonnées de points dans notre programme FonctionSplineCubique, que nous avons modifié afin d'y mettre des fractions ($\frac{1}{26}$ dans notre cas). Après avoir obtenu une série de fonctions splines, nous avons tracé la courbe correspondante sur le logiciel Geogebra.

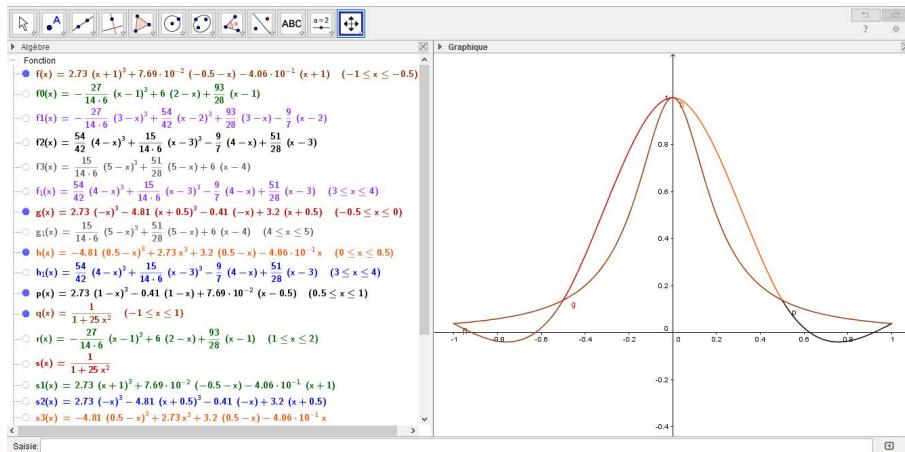


Figure 11: Représentation graphique de l'interpolation de la fonction de Runge par spline cubique

Nous obtenons alors une courbe bien plus lisse. On remarque que vers le milieu de l'intervalle de définition, la fonction d'interpolation se rapproche de la fonction recherchée mais que le phénomène de Runge est présent aux extrémités de l'intervalle, comme expliqué précédemment.

Voici une autre représentation de cette même fonction avec 10 puis 20 noeuds d'interpolation :

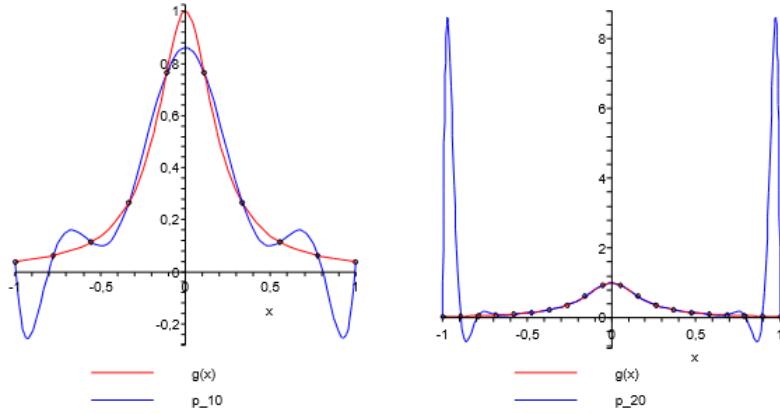


Figure 12: Représentation graphique du phénomène de Runge visible aux extrémités de l'intervalle de définition

Ce phénomène de Runge n'apparaît pas dans tous les cas (comme par exemple avec $f(x) = \frac{1}{1+x^2}$) et peut être atténué par l'utilisation de courbes B-splines.

6.2 Fonction Cosinus

A titre de second exemple, nous nous sommes intéressés à la fonction cosinus. Initialement, nous avions prévu de la tracer sur l'intervalle $[0; \pi]$, mais la fonction étant impaire sur le point $(\pi/2; 0)$, nous nous sommes limités à l'intervalle $[0; \pi/2]$.

x	0	0.3927	0.7854	1.1781	1.5708
cos(x)	1	0.9239	0.7071	0.3827	0

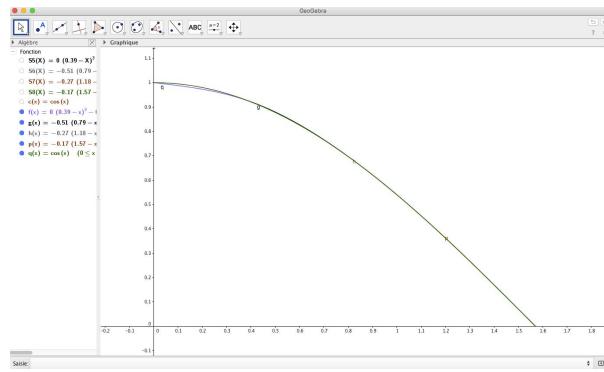


Figure 13: Modélisation de la fonction Cosinus par spline cubique

Là encore, la fonction n'a pas l'allure attendue, elle n'est pas lisse et a un aspect quelque peu linéaire. De même que pour la fonction de Runge, en utilisant notre programme FonctionSplineCubique, nous avons relevé les mêmes coordonnées puis utilisé les splines obtenues pour tracer la courbe sur GeoGebra.

6.3 Fonction Exponentielle

Nous nous sommes finalement intéressés à la fonction exponentielle modélisée par une interpolation polynomiale. Nous avons tout d'abord décidé d'étudier cette fonction avec Open Office Excel sur l'intervalle $[-1;3]$, sans utiliser d'interpolation.

x	-1	0	1	2	3
$\exp(x)$	0.3679	1	2.7183	7.3891	20.0855

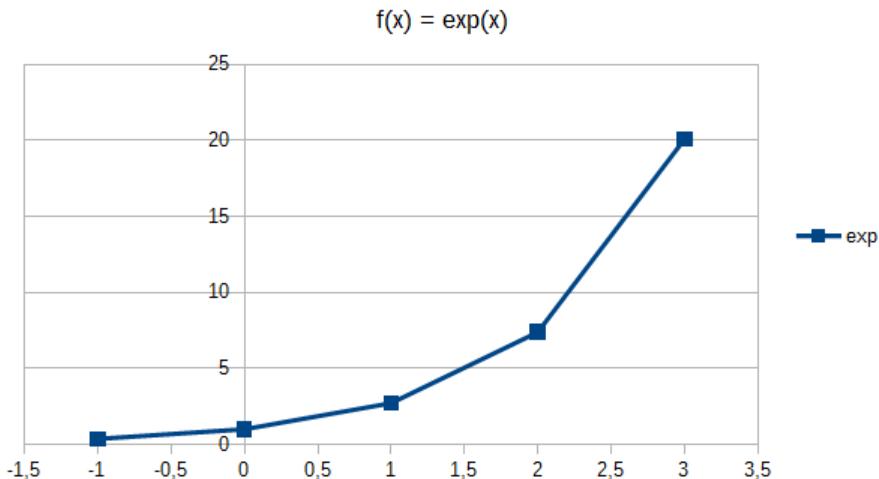


Figure 14: Représentation graphique de la fonction exponentielle sur l'intervalle $[-1;3]$

La courbe obtenue à l'aide du tableur est une succession de segments de droite, elle est donc linéaire. Mais elle a tout de même une allure assez proche de la fonction exponentielle recherchée.

Ensuite, pour comparer, nous avons appliqué la méthode en utilisant cette fois le logiciel Geogebra. On a ainsi modélisé la fonction exponentielle à l'aide de quatre nœuds, répartis sur l'intervalle $[0;2]$.

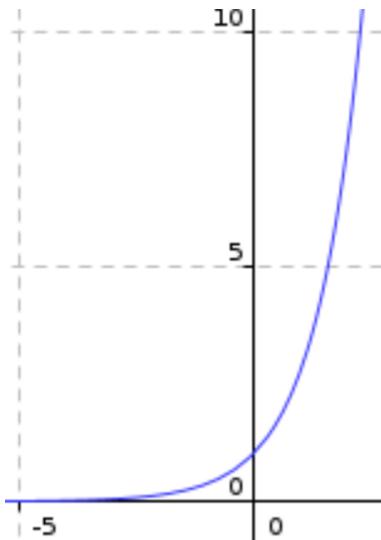


Figure 15: Nouvelle représentation graphique de la fonction exponentielle sur l'intervalle $[-1;3]$ avec Geogebra

On retrouve une courbe à l'allure lisse comme celle de la fonction exponentielle. Les noeuds dans l'intervalle $[0;2]$ sont confondus avec la courbe.

Par conséquent, nous nous sommes servis de trois exemples de fonctions usuelles afin de révéler la précision de la fonction d'interpolation par spline dans le traçage de différentes courbes de fonctions. En effet, à l'aide de notre programme FonctionSplineCubique, nous avons obtenu une série de fonctions splines puis, à l'aide du logiciel GeoGebra, nous avons utilisé ces fonctions pour tracer les courbes. Nous avons ainsi pu retrouver une courbure fidèle à l'allure attendue des fonctions de Runge, cosinus et exponentielle dont l'aspect lisse provient des propriétés de l'interpolation par splines cubiques.

7 Applications des splines cubiques

Comme la partie historique de ce dossier le souligne, l'interpolation est un vieux outil mathématique. Son ancienneté nous montre qu'elle est utile à toute les époques. En effet, l'interpolation permet de déduire une infinité de points à partir de seulement quelques mesures, ce qui est infiniment utile dans tous les métiers scientifiques. Si l'interpolation est d'une aussi grande utilité, l'interpolation par spline l'est a fortiori puisqu'il s'agit d'une des méthodes d'interpolations présentant la plus petite erreur et le moins de déviation. Comme le dit Thomas Grandine de la RD de Boeing : «les splines sont très utilisées chez Boeing ainsi que dans la plus grande part du monde industriel. Il y a très peu de domaines où ces fonctions n'ont pas encore eu de rôle, et leur utilisation continue à s'intensifier.»

Depuis le début de son développement dans les années 50, celle-ci a connu de nombreuses applications dans des domaines variés. Etant limités par le temps (et la considération écologique du nombre de pages à imprimer bien sûr !) nous présenterons dans cette section une liste non exhaustive de ces domaines. Nous passerons bien évidemment par l'automobile mais aussi la typographie, la chimie, l'aéronautique et le domaine de l'imagerie médicale.

7.1 Les splines dans la typographie et les logiciels de dessin

Une nouvelle application des splines cubiques est apparue avec l'arrivée des ordinateurs. On les retrouve en typographie et plus précisément, ils sont à la base des différentes polices de caractère que l'on trouve dans les traitements de texte. Par exemple, le Postscript est un langage informatique et de programmation complexe, créé dans les années 1980, qui permet de communiquer avec les imprimantes. Les polices de caractère en Postscript sont définies par des courbes de Bézier mais il n'y a pas d'interaction homme-machine comme pour les traitements de texte. On ne se rend donc pas compte que ce langage est utilisé.

La photo ci-dessous montre à gauche, la lettre « e » grossie par un ordinateur des années 80, et à droite, la même lettre avec un ordinateur d'aujourd'hui. La méthode des splines cubiques est utilisée pour cette dernière.

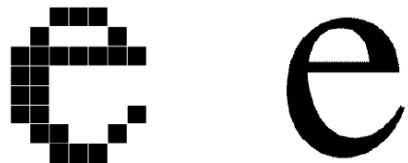


Figure 16: La lettre e avec un ordinateur des années 80 (gauche) et un ordinateur d'aujourd'hui (droite)

La méthode des splines cubiques permet, lorsque l'on agrandit ou diminue la taille de la police, de supprimer le phénomène de « pixellisation ». On remarque alors que les lettres d'aujourd'hui sont plus lisibles, plus agréables à lire que celles d'il y a 30 ans. Il est également possible d'écrire la même lettre de différentes façons en raison des très nombreuses polices existantes.

Avec le développement de l'informatique, cette application s'est aussi étendue aux logiciels de dessin (modélisation), tels que Paint ou Geogebra. Les courbes de Bézier et les B-Splines sont utilisées afin de modéliser toutes les formes, principalement géométriques, souhaitées.

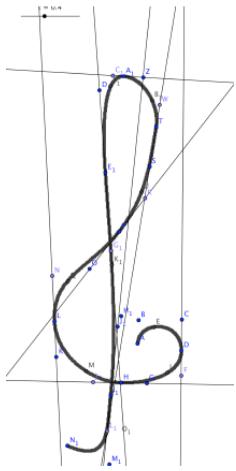


Figure 17: Dessin (à l'aide de B-splines) réalisé par ordinateur avec le logiciel Geogebra.

Les points bleus visibles sur l'image représentent les noeuds d'interpolation.

Cette méthode s'est ensuite appliquée dans les logiciels de C.A.O. (Conception Assistée par Ordinateur) ou dans le domaine du design. Nous expliquerons cela dans la partie qui suit.

7.2 Les splines dans les transports : l'aéronautique et l'automobile

Avant l'avènement de l'ordinateur et des splines, il fallait utiliser d'autres moyens pour trouver une courbe lisse passant par des points données. La photo ci-contre montre un ingénieur de chez Boeing, dans les années avant l'essor de l'ordinateur, en plein processus de création. Il utilise ce que les anglosaxons appellent un « draftsman's spline » (une corde reliée à des poids qui permettent de lui donner une forme). D'autres méthodes consistaient par exemple en l'utilisation de règles flexibles. La photo ci-contre illustre bien la pénibilité de ce travail et nous donne une idée du temps de travail nécessaire.



Figure 18: Ingénieur chez Boeing

La deuxième moitié du 20ème siècle a amené avec elle l'ordinateur dans le monde du travail. Ce fut une véritable révolution pour l'industrie. Les employés de bureau pouvaient abandonner leurs règles pliantes au profit de logiciels de C.A.O. (Conception Assistée par Ordinateur). On pouvait maintenant faire faire les calculs rapidement par l'ordinateur avec des algorithmes et modéliser les structures que l'on souhaitait construire. Ces logiciels sont utilisés dans tous les domaines de l'industrie mais plus particulièrement dans l'aéronautique et l'automobile où l'aérodynamisme est un critère important et l'obtention de courbes lisses est ainsi nécessaire. L'utilisation des splines ne s'arrête cependant pas là ! En effet, en industrie, une fois le produit conçu vient la phase de fabrication où l'on retrouve les splines : la Fabrication Assistée par Ordinateur (F.A.O.) simplifie la fabrication de pièces complexes aux contours ou à courbure précise.

7.3 Les splines en chimie

Comme le dit CJC Kruger de la société d'hydrauliques Korf, « L'interpolation par splines cubiques est une technique utile [en ingénierie chimique] pour interpoler des valeurs entre des points de données ». Elle a cependant la particularité de varier fortement entre ses noeuds. Même si cette variation reste moins importante qu'avec une interpolation polynomiale, ceci peut se révéler problématique. La réponse est donc de contraindre les variations de la spline entre ses noeuds. Dans ce but, Korf, Riscona et d'autres, tels que Huang, ont développé des méthodes d'interpolation par fonctions splines restreintes (restrained splines functions).

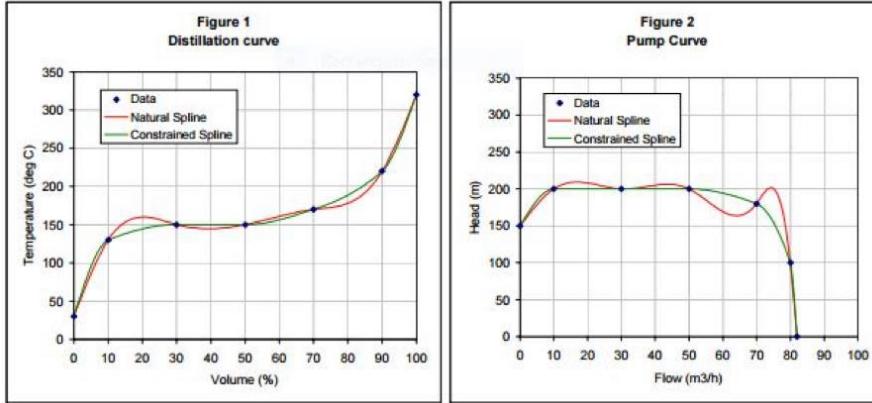


Figure 19: Spline Restreintes

Les deux graphiques ci-dessus montrent les courbes obtenues à travers l'interpolation par splines cubiques restreintes ou non entre 7 points. Pour la distillation ainsi que pour la pompe, on remarque une assez nette différence entre ces deux courbes : les splines restreintes varient bien moins ! On peut donc voir que même si elle peut s'avérer très pratique, la méthode d'interpolation par splines cubiques n'est pas optimale pour toutes les applications et peut être améliorée dans certains domaines.

7.4 Les splines et l'imagerie médicale

Nous avons vu que l'industrie regorge de domaines d'application pour les splines. Il ne faut cependant pas en conclure qu'il s'agit des seules applications possibles des splines ! Certains chercheurs essayent en effet de faire valoir les avantages des splines dans le domaine de l'imagerie médicale. L'Ecole Polytechnique Fédérale de Lausanne (Suisse) se trouve au premier plan de cette bataille. Elle comporte un groupe de recherche entier baptisé Biomedical Imaging Group (BIG) qui fait beaucoup de zèle dans cette direction.

Nous citerons particulièrement E. Meijering et M.Unser qui ont publié un grand nombre d'articles et de publications. Les chercheurs du BIG considèrent en effet que les splines présentent le meilleur rapport « temps de calcul/prix » de toutes les méthodes d'interpolation, ce qui les rend idéales dans un domaine où le traitement du signal est aussi important que l'imagerie médicale.

7.5 Application de splines : le Taj Mahal

Afin de voir par nous-même la fidélité de modélisation des splines, nous avons décidé de modéliser une courbe en 2D : une image du Taj Mahal. A l'aide du logiciel Gimp, nous avons pris les coordonnées de quelques points du contour de la structure sur l'image d'origine puis nous avons utilisé le logiciel Geogebra pour modéliser les splines obtenus entre ces points.



Figure 20: Photo du Taj Mahal

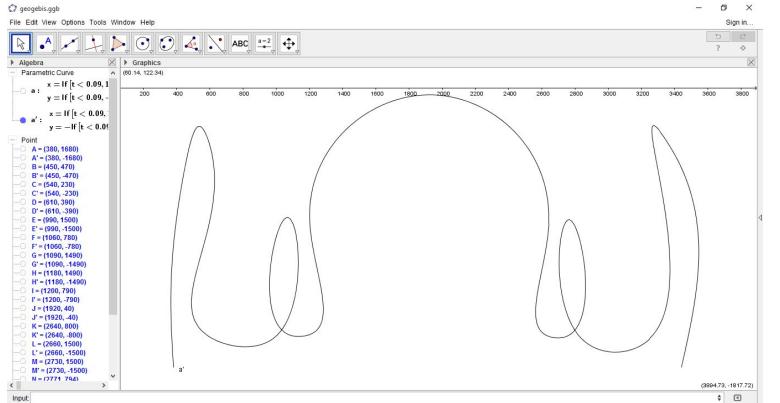


Figure 21: Modélisation du Taj Mahal par splines cubiques sur Geogebra

On remarque que même si les contours de l'image obtenue ont tendance à être plus courbés que sur l'image originale, la structure du Taj Mahal reste reconnaissable. Le manque de précision vient aussi du nombre de points à interpoler: nous en avons pris 20, mais en rajoutant des points, les détails de la structure commenceront à apparaître.

8 Conclusion

Après avoir retracé l'histoire de l'interpolation, nous avons approfondi, dans l'ordre chronologique, les différentes méthodes : tout d'abord les courbes de Bézier puis les splines cubiques pour ensuite finir avec les B-Splines. Pour chaque technique d'interpolation, nous avons présenté le contexte, développé la résolution mathématiques et informatique et expliqué les avantages et inconvénients. S'ensuit des exemples de fonctions construites par splines cubiques. Nous avons achevé notre rapport par la présentation d'applications possibles de cette méthode. Notons que les exemples et applications exposés dans notre travail ne représentent qu'un petit échantillon de la réalité.

Nous avons aimé travailler sur ce sujet. En effet, ce projet nous a permis d'explorer la méthode d'interpolation par splines cubiques qui nous était auparavant méconnue. La compréhension mathématiques de l'interpolation par splines cubiques est complexe mais après de nombreuses recherches, manipulations du programme et calculs de splines, nous avons réussi à aboutir à un travail intéressant et enrichissant.

Nous avons ainsi pu nous rendre compte de l'utilité et de l'importance de cette méthode dans de nombreux domaines d'applications non seulement en mathématiques mais aussi en aéronautique, chimie, typographie, etc. Comme nous l'avons vu, les courbes de Bézier sont la base de l'interpolation par splines cubiques qui a ensuite évolué vers les B-splines : ce procédé est en constante évolution. Bref, l'actualité de ce domaine est très grande : les courbes et les surfaces d'interpolation et d'approximation sont utilisées partout. Il suffit de choisir un objet au hasard, il est probable qu'une des notions étudiées dans notre travail ait été utilisées.

Nous avons de plus découvert le langage LaTex qui est très utilisé pour l'écriture de documents scientifiques ainsi que réutilisé des logiciels de modélisation géométrique comme Geogebra. Un certain temps a été nécessaire afin de comprendre et s'approprier ce langage mais la connaissance du logiciel Texmaker nous sera très probablement utile pour l'élaboration d'autres documents scientifiques.

9 Crédits d'illustration

Figure 1 - Ancienne table de pierre.

Source : <https://imagescience.org/meijering/research/chronology/>

Figure 2 - Ducks.

Source : <http://www.duckworksmagazine.com/03/r/articles/splineducks/splineDucks.htm>

Figure 3 - Pierre Bézier, 1958.

Source : <http://www.fondam.org/Portraits/PierreBezier>

Figure 4 - Renault 4CV.

Source : <https://quatrecylindres.com/2010/12/08/dans-le-retro-de-pierre-hugonnaud-renault-4cv-la-puce-de-billancourt/>

Figure 5 - Exemple d'interpolation par spline cubique sur Geogebra.

Figure 6 - Analyse descendante du programme.

Figure 7 - Courbe de B-spline de degré 3.

Source : <http://www.sens-neuchatel.ch/bulletin/no34/art3-34.pdf>

Figure 8 - Analyse descendante du programme.

Figure 9 - Tableau de résultats de la fonction de Runge en fonction des valeurs de x.

Figure 10 - Représentation graphique de la fonction de Runge sur l'intervalle $[-1; 1]$.

Figure 11 - Représentation graphique de l'interpolation de la fonction de Runge par spline cubique.

Figure 12 - Représentation graphique du phénomène de Runge visible aux extrémités de l'intervalle de définition.

Source : http://johan.mathieu.free.fr/math/soc_maths/oral_1_capes/phenomene_Runge_projet_licence.pdf

Figure 13 - Modélisation de la fonction Cosinus par spline cubique.

Figure 14 - Représentation graphique de la fonction exponentielle sur l'intervalle $[-1; 3]$.

Figure 15 - Nouvelle représentation graphique de la fonction exponentielle

sur l'intervalle $[-1; 3]$ avec Geogebra.

Figure 16 - La lettre e avec un ordinateur des années 80 (gauche) et un ordinateur d'aujourd'hui (droite).

Source : BezierDP.pdf

Figure 17 - Dessin (à l'aide de B-splines) réalisé par ordinateur avec le logiciel Geogebra.

Source : BezierDP.pdf

Figure 18 - Ingénieur chez Boeing.

Source : <http://www.boeingimages.com/archive/Engineer-Drafting-with-a-Physical-Spline-2F3XC5LBLBL.html>

Figure 19 - Splines restreintes.

Source : <http://www.korf.co.uk/spline.pdf>

Figure 20 - Photo du Taj Mahal.

Source : <http://img0.svstatic.com/wallpapers/7516c8b89bf97f0219f074a24515b7c9-large.jpeg>

Figure 21 - Modélisation du Taj Mahal par splines cubiques sur Geogebra.

10 Bibliographie

Sources que nous avons utilisé pour développer le rapport :

<https://patrimoine.gadz.org/gadz/bezier.htm>

<http://www-groups.dcs.st-and.ac.uk/history/Biographies/Schoenberg.html>

<http://www.sens-neuchatel.ch/bulletin/no34/art3-34.pdf>

<http://www.math.u-psud.fr/~perrin/CAPES/geometrie/BezierDP.pdf>

<https://en.wikipedia.org/wiki/B-spline>

https://fr.wikipedia.org/wiki/Spline#Spline_serr.C3.A9e

[https://en.wikipedia.org/wiki/Spline_\(mathematics\)#History](https://en.wikipedia.org/wiki/Spline_(mathematics)#History)

<http://bigwww.epfl.ch/publications/meijering0201.pdf>

https://fr.wikipedia.org/wiki/Interpolation_num%C3%A9rique

<http://bigwww.epfl.ch/publications/meijering0201.pdf>

<http://www.math.univ-toulouse.fr/~cnegules/Article/beziers.pdf>

https://fr.wikipedia.org/wiki/Courbe_de_B%C3%A9zier

<https://imagescience.org/meijering/research/chronology/>

https://www.math.u-bordeaux.fr/~pfischer/Teaching_files/cours.pdf (page 8 et 9)

<https://www.youtube.com/watch?v=kF-75RRvCbs>

<http://www.labri.fr/perso/schlick/simg/cours10.pdf>

<http://pulsar.webshaker.net/2012/08/29/les-courbes-de-bezier-1/>

http://perso.univ-lyon1.fr/jean-claude.iehl/Public/educ/ENS/chap6_Courbes.pdf

<http://ccmuzzo.free.fr/perso/projets/Splines.pdf>

<http://www.giref.ulaval.ca/~afortin/mat17442/documents/splines.pdf>

http://homepages.ulb.ac.be/~majansen/teaching/INFO-F-205/diapositives05interpolation_4.pdf

<http://www.math.univ-metz.fr/~croisil/M1-0809/2>

<https://fr.wikipedia.org/wiki/NURBS>

<https://fr.wikipedia.org/wiki/Spline>

http://johan.mathieu.free.fr/math/doc_maths/oral_1_capes/phenomene_Runge_projet_licence.pdf

<http://w3.bretagne.ens-cachan.fr/math/people/gregory.vial/files/cs/interp.pdf>

Pour l'écriture du rapport avec Texmaker :

<http://tex.stackexchange.com/>