Introduction

À ses débuts, la géométrie algorithmique s'est intéressée à la résolution de divers problèmes géométriques pour des objets que nous appellerons "linéaires" : des ensembles de points, de segments, de droites, et des polygones, dans le plan, et respectivement leurs généralisations en dimensions supérieures, pour les études effectuées dans $\mathbb{R}^d$. Les opérations de base nécessaires sur ces objets (comme le calcul de l'intersection de deux droites données, ou le calcul de la droite passant par deux points donnés) se traduisaient par des équations ou des systèmes d'équations de degré un, qui pour des données rationnelles ont des solutions rationnelles et pouvaient raisonnablement être considérés comme solubles de manière exacte et en temps $\mathcal{O}(1)$.

Depuis, deux tendances assez contradictoires se sont manifestées. D'un côté, un besoin croissant de précision s'est fait sentir, mettant (à juste titre) en doute l'exactitude des solutions fournies par l'ordinateur même pour les systèmes d'équation de degré un. Ceci a mené à un traitement plus fin de la manipulation des nombres réels, par l'introduction des arithmétiques paresseuses et d'intervalle ou par d'autres approches "arithmétiques". D'un autre côté, dans la modélisation des objets du monde réel est requis un plus grand réalisme que celui assuré par les approximations polygonales. Ceci implique le travail sur des "courbes" (notion à préciser en fonction du contexte), entités autrement plus complexes que les objets linéaires, et a pour conséquence la génération de systèmes algébriques de degré élevé qui devraient être résolus de manière exacte. Le parti généralement pris par les auteurs s'étant penchés sur des concrétisations de cette dernière tendance est d'utiliser un certain nombre de primitives, ou "oracles", fournissant les solutions des problèmes élémentaires posés pour des courbes (comme le calcul des droites d'appui communes de deux courbes) et de considérer que la résolution de ces primitives se fait de façon exacte et en temps constant (dépendant éventuellement des courbes traitées). Nous aurons ainsi d'une part une primitive calculant de manière exacte l'intersection de deux courbes, et d'une autre part le doute sur l'exactitude de la réponse de l'ordinateur à la question de l'appartenance d'un point à une droite.

Le principe sur lequel les travaux présentés dans cette thèse sont basés se situe quelque part entre les deux tendances mentionnées. L'utilité de l'emploi des courbes dans la modélisation géométrique est indéniable, nous traitons donc des courbes planes satisfaisant certaines propriétés. Mais le fait que, généralement, les opérations de base sur ces courbes ne peuvent pas être effectuées de manière exacte par l'ordinateur est aussi indéniable, et de point de vue géométrique et topologique cela peut générer des incohérences dans la scène étudiée. Une solution serait d'utiliser des arithmétiques exactes, mais cette approche nous éloignerait des aspects géométriques du problème à résoudre. Nous avons donc choisi de nous concentrer sur la topologie des objets donnés et de la solution cherchée, la notion de "topologie" étant à définir dans le contexte du problème à traiter.

Notre méthode consiste à utiliser, pour un objet courbe donné, une suite d'approximations polygonales convergeant, dans un sens précisé, vers l'objet. Plus précisément, si $\mathcal{O}$ est l'objet courbe et $\{\mathcal{P}_i\}$ est la suite de polygones l'approchant, nous avons prouvé que la solution $\mathcal{S}(\mathcal{O})$ du problème étudié pour l'objet $\mathcal{O}$ est à son tour approchée par la suite $\{\mathcal{S}(\mathcal{P}_i)\}$ des solutions respectives du problème pour les polygones $\mathcal{P}_i$. Plus, nous avons montré qu'il existe un indice i_0 (dépendant de $\mathcal{O}$) tel que la topologie de $\mathcal{S}(\mathcal{P}_i)$ soit "la même" que la topologie de $\mathcal{S}(\mathcal{O})$ pour tout $i \geq i_0$. Bien sûr, nous avons défini d'abord de manière rigoureuse ce que nous comprenons par "topologie de la solution" et par "même topologie".

Théoriquement, ce procédé pourrait être appliqué à des courbes (de Jordan) satisfaisant un minimum de conditions assurant l'existence des suites de polygones les approchant. Dans la pratique, si nous voulons pouvoir implémenter les algorithmes de résolution des problèmes donnés qui découlent de notre méthode, il s'impose de restreindre la famille de courbes impliquées. Nous avons choisi de travailler avec des courbes paramétriques contrôlées par des points ; concrètement, les algorithmes ont été implémentés pour des courbes de Bézier, et pour la facilité de l'expression ce sont ces courbes dont nous parlerons tout au long de ce document. Remarquons toutefois, comme il sera précisé dans la deuxième partie du premier chapitre, que d'autres types de courbes paramétriques répondent aux propriétés demandées pour l'application de notre méthode : les courbes de Bézier rationnelles, les B-splines, les NURBS.

L'avantage majeur (de notre point de vue) de ces familles de courbes est d'avoir associée, par définition, la suite convergente de polygones dont nous parlions ; en plus, chacun de ces polygones satisfait, toujours par définition, toutes les bonnes propriétés assurant l'aboutissement de notre discours. La suite de polygones est construite de manière récursive par l'application d'un algorithme de subdivision qui, précision importante pour l'esprit de l'approche que nous présentons, obtient le polygone de l'itération $i+1$ en utilisant uniquement le polygone de l'itération i, sans avoir besoin d'informations contenues dans la courbe elle-même. Ainsi, une fois que les polygones initiaux sont connus, les courbes peuvent être complètement ignorées et nous sommes libérés de l'emploi des primitives dont nous avons parlé au début de l'introduction. Dans le cas des courbes de Bézier, l'algorithme de subdivision est évidemment celui de de Casteljau.

Nous nous devons maintenant de préciser que notre méthode, telle qu'elle est appliquée dans les travaux de cette thèse, ne fournit pas encore la solution topologique parfaitement rigoureuse aux problèmes abordésés. Ceci est dû au fait que, même en ne manipulant jamais directement les courbes concernées, des imprécisions de calcul vont quand même apparaître dans le traitement des données, car nous avons utilisé l'arithmétique flottante pour effectuer toutes les opérations élémentaires sur les points, segments et droites générés dans les calculs sur les polygones approchants. Pour que notre démarche atteigne pleinement son but, l'utilisation d'une arithmétique exacte, paresseuse ou d'intervalle, s'impose comme une continuation naturelle des travaux.

Le premier chapitre de la thèse est consacré à la définition rigoureuse des objets sur lesquels nous avons travaillé, les *polycourbes*. Le point de départ est constitué par le *splinegone*, introduit par Souvaine [Sou], mais certaines imprécisions (que nous avons commentées) dans la définition de celui-ci nous empêchent de nous en servir dans notre présentation. Néanmoins, quelques résultats intéressants concernant le splinegone (ou un objet qui, si il n'est pas explicitement appelé ainsi, en satisfait les propriétés) sont déjà parus dans la littérature, et par conséquent nous avons jugé important de prouver que le nouvel objet est un splinegone.

La deuxième partie du chapitre est dédiée au "mariage" de la polycourbe et des courbes de Bézier, qui a eu comme résultat les *polycourbes de Bézier*. Pour des raisons dûment expliquées, les courbes de Bézier générales ne conviennent pas à l'usage que nous voulons en faire, nous avons donc utilisé uniquement les courbes de Bézier de B-polygone (ou polygone de contrôle) simple et convexe, que nous avons nommées courbes de Bézier *complètement convexes*. Nous remarquons ici que nous avons préféré la dénomination "B-polygone" à celle, généralement employée, de "polygone de contrôle" pour le polygone associé à une courbe de Bézier, pour éviter les confusions entre le polygone de contrôle d'une courbe de Bézier et le *polygone de contrôle d'une polycourbe de Bézier*, notion introduite et largement utilisée dans cette thèse. Nous avons aussi prouvé, pour une polycourbe de Bézier quelconque (qui est par définition une courbe simple), l'existence, sous certaines conditions, d'un polygone de contrôle simple. Ainsi, une caractéristique topologique de base de la polycourbe est partagée par son polygone de contrôle.

Deux problèmes fondamentaux de la géométrie algorithmique ont été abordés dans cette thèse : l'enveloppe convexe et les arrangements. À chacun des deux, nous avons dédié deux chapitres, l'un présentant des résultats de nature théorique et l'autre présentant en détail les algorithmes découlant de notre méthode, avec leurs justifications de correction et de convergence, et des résultats numériques.

Les chapitre deux et trois sont ainsi consacrés à l'étude de l'enveloppe convexe d'une polycourbe (de Bézier) fermée. Si pour les arrangements la notion de topologie (combinatoire) est bien connue, la topologie de l'enveloppe convexe est une notion assez floue. En conséquence, nous avons commencé, dans le chapitre deux, par définir de façon rigoureuse le résultat que nous souhaitons obtenir, en introduisant la notion de *pertinence topologique*. Nous avons ensuite prouvé l'existence d'une suite de polygones telle qu'à partir d'un certain indice tous les polygones satisfassent les conditions de pertinence topologique relatives à la polycourbe qui leur est associée.

Mais cette preuve fait intervenir des calculs sur les courbes de Bézier composant la polycourbe, elle ne peut donc pas fournir de critère implémentable pour la pertinence topologique. Une partie du chapitre trois est par conséquent allouée à la recherche d'un critère de pertinence topologique basé uniquement sur le polygone de contrôle. Une fois le critère trouvé, l'algorithme qui a été conçu en l'utilisant est présenté et commenté, avec une discussion complète sur sa complexité. Une attention spéciale est accordée aux cas dégénérés, la bête noire du programmeur. À la fin du troisième chapitre, les résultats numériques obtenus sont interprétés, non sans remarquer qu'ils relèvent une complexité pratique de l'algorithme largement inférieure à celle théoriquement calculée. Enfin, quelques exemples que nous avons considérés intéressants peuvent être trouvés dans le dernier paragraphe.

Les chapitres quatre et cinq abordent le problème des arrangements de polycourbes (de Bézier). La première partie du chapitre quatre fait un rappel des résultats sur les arrangements de courbes qui peuvent être trouvés dans la littérature. Remarquons que les courbes impliquées sont toujours des courbes non bornées ou fermées, donc des courbes qui partagent sans équivoque le plan en deux régions connexes disjointes, propriété que les polycourbes non fermées n'ont pas. Les arrangements de polycourbes seraient ainsi similaires aux arrangements de segments plutôt qu'aux arrangements de droites. Par conséquent, nous avons utilisé pour exprimer la notion d'équivalence (topologique, combinatoire) de deux arrangements de polycourbes les résultats de Vo Phi [VoP94]. Dans la deuxième partie du chapitre quatre nous avons prouvé l'existence d'une suite d'arrangements de lignes polygonales qui à partir d'un certain indice sont équivalents à l'arrangement de polycourbes.

Comme dans le cas de l'enveloppe convexe, la preuve mentionnée ci-dessus implique des calculs directement sur les courbes de Bézier intervenant dans l'arrangement et nous ne pouvons donc pas nous en inspirer pour l'élaboration d'un critère d'équivalence implémentable. En conséquence, la première partie du chapitre cinq présente des résultats nous permettant de déduire un test d'équivalence basé uniquement sur des calculs sur objets linéaires, faisant intervenir les polygones de contrôle des polycourbes, ainsi que leurs *polygones support*, ou *porteurs*. Ensuite, nous détaillons l'algorithme qui a été conçu pour calculer un arrangement de lignes polygonales équivalent à un arrangement de polycourbes de Bézier donné, en faisant une discussion sur son coût. La fin du chapitre est dédiée à la présentation et l'interprétation des résultats numériques, et quelques exemples jugés intéressants peuvent être trouvés dans le dernier paragraphe.