

Projet PRISME

Géométrie, Algorithmes et Robotique

Sophia Antipolis

THÈME 2B



*R*apport
d'Activité

1999

Table des matières

| | | |
|----------|--|-----------|
| 1 | Composition de l'équipe | 4 |
| 2 | Présentation et objectifs généraux | 5 |
| 3 | Fondements scientifiques | 5 |
| 3.1 | Géométrie Algorithmique | 5 |
| 3.1.1 | Géométrie combinatoire | 5 |
| 3.1.2 | Techniques algorithmiques | 6 |
| 3.1.3 | Triangulations et diagrammes de Voronoï | 7 |
| 3.1.4 | Planification de trajectoires | 8 |
| 3.2 | Calcul géométrique | 8 |
| 3.2.1 | Robustesse et dégénérescences | 9 |
| 3.2.2 | Arithmétique et filtres | 9 |
| 3.2.3 | Régularisation | 11 |
| 3.2.4 | La bibliothèque CGAL | 11 |
| 4 | Domaines d'applications | 12 |
| 4.1 | Panorama | 12 |
| 4.2 | Reconstruction de formes | 12 |
| 4.2.1 | Imagerie médicale | 12 |
| 4.2.2 | Géologie | 13 |
| 4.3 | Algorithmique moléculaire | 13 |
| 4.4 | Placement de formes | 14 |
| 4.4.1 | Découpe | 15 |
| 4.4.2 | Aménagement de satellites | 15 |
| 5 | Logiciels | 16 |
| 5.1 | Bibliothèque d'algorithmes géométriques CGAL | 16 |
| 5.2 | Évaluation exacte d'expressions numériques | 17 |
| 5.3 | Hierarchie de Delaunay | 17 |
| 5.4 | Reconstruction à partir de coupes. | 17 |
| 5.5 | Modeleur géométrique pour la géologie | 18 |
| 5.6 | Aménagement de satellites | 18 |
| 5.7 | Placement et déplacement | 18 |
| 6 | Résultats nouveaux | 18 |
| 6.1 | CGAL | 18 |
| 6.1.1 | Noyau | 19 |
| 6.1.2 | Modules d'arithmétique | 19 |
| 6.1.3 | Triangulations bidimensionnelles | 19 |
| 6.1.4 | Triangulations tridimensionnelles | 20 |
| 6.1.5 | Visualisation 3D | 21 |
| 6.2 | Algorithmes géométriques | 21 |

| | | |
|----------|--|-----------|
| 6.2.1 | Rôle des prédicats dans les algorithmes géométriques | 22 |
| 6.2.2 | Compression de structures géométriques | 22 |
| 6.2.3 | Union de boules | 23 |
| 6.3 | Calcul géométrique | 23 |
| 6.3.1 | Degré algébrique des algorithmes | 23 |
| 6.3.2 | Signe de déterminants | 24 |
| 6.4 | Applications de la géométrie algorithmique | 24 |
| 6.4.1 | Reconstruction de formes | 24 |
| 6.4.2 | Métrologie | 25 |
| 6.4.3 | Dessin de graphes et molécules | 25 |
| 7 | Contrats industriels (nationaux, européens et internationaux) | 26 |
| 7.1 | IFP | 26 |
| 7.2 | REALVIZ | 27 |
| 7.3 | EDF | 27 |
| 7.4 | Kréon | 27 |
| 8 | Actions régionales, nationales et internationales | 27 |
| 8.1 | Actions nationales | 27 |
| 8.1.1 | Action de recherche coopérative INRIA FIABLE | 27 |
| 8.1.2 | Action de recherche coopérative INRIA GEOMETRICA | 28 |
| 8.1.3 | Accueil de chercheurs | 28 |
| 8.2 | Actions européennes : GALIA | 29 |
| 8.3 | Relations bilatérales | 29 |
| 8.3.1 | France-Espagne | 29 |
| 8.3.2 | France-Israël | 29 |
| 8.4 | Accueils de chercheurs étrangers | 29 |
| 9 | Diffusion de résultats | 30 |
| 9.1 | Animation de la communauté scientifique | 30 |
| 9.1.1 | Organisation de conférences | 30 |
| 9.1.2 | Comités de rédaction de revues scientifiques | 30 |
| 9.1.3 | Comités de sélection | 30 |
| 9.1.4 | Comités de programme de conférences | 30 |
| 9.1.5 | Serveur WWW | 30 |
| 9.2 | Formation | 31 |
| 9.2.1 | Enseignement universitaire | 31 |
| 9.2.2 | Autres enseignements | 31 |
| 9.2.3 | Stages | 31 |
| 9.2.4 | Thèses en cours | 32 |
| 9.2.5 | Thèses soutenues | 32 |
| 9.3 | Participation à des colloques, séminaires, invitations | 32 |
| 9.3.1 | Conférence invitée | 32 |
| 9.3.2 | Séjours | 32 |

| | | |
|-----------|---|-----------|
| 9.3.3 | Exposés à des colloques et séminaires | 32 |
| 9.3.4 | Participation à des colloques | 33 |
| 10 | Bibliographie | 33 |

1 Composition de l'équipe

Responsable scientifique

Jean-Daniel Boissonnat [DR]

Responsable permanent

Olivier Devillers [CR]

Assistante de projet

Agnès Clément-Bessière [TR, projets PRISME et ICARE]

Personnel Inria

Hervé Brönnimann [CR, en sabbatique à Princeton University]

Frédéric Cazals [CR]

Monique Teillaud [CR]

Mariette Yvinec [CR]

Chercheurs doctorants

Sophie Balaven [en collaboration avec l'IFP]

Frank Da [allocataire MENRT]

Julia Flötotto [allocataire MENRT depuis le 1^{er} octobre]

Pierre-Marie Gandoin [allocataire MENRT]

Sylvain Pion [AMN jusqu'au 31 août, boursier INRIA du 1 sept. au 30 nov.]

Chercheurs post-doctorants

François Rebufat

Alexandra Fronville [jusqu'au le 30 septembre, financée par l'action coopérative INRIA FIABLE jusqu'au 31 mars]

Stagiaires

Louai Adhami [DEA ARAVIS, UNSA, mars-septembre, avec CHIR]

Carine Bonetto [INSA-Rouen, juillet-septembre]

Julia Flöttoto [DEA ARAVIS, UNSA, mars-septembre]

Alexandre Guyennon [ESSI, janvier-juin]

Johann Liberman [ESSI, janvier-juin]

Radu Lupşa [DEA algorithmique, Paris, avril-juin]

Antoine Vigneron [DEA algorithmique, Paris, avril-juin]

Chercheurs invités

Bruce Simpson [University of Waterloo, Canada, du 1er janvier au 30 avril]

Jeff Vitter [Duke University, Caroline du nord, USA jusqu'au 30 juin]

2 Présentation et objectifs généraux

De nombreux domaines d'application demandent de savoir construire et traiter de manière efficace des objets géométriques. Citons, parmi beaucoup d'autres, la robotique, l'informatique graphique et la conception assistée par ordinateur.

L'objectif général du projet PRISME consiste à développer, dans le domaine de la géométrie, une algorithmique effective. Nos recherches portent en premier lieu sur les *aspects combinatoires et algorithmiques*, et se fondent sur des analyses réalistes. Ceci nous a conduit à mener des analyses en moyenne plutôt que dans le pire des cas (randomisation), à prendre en compte des paramètres significatifs autres que la taille des entrées, notamment la taille du résultat (algorithmes adaptatifs) et à définir de nouveaux modèles de calcul fidèles à l'arithmétique des ordinateurs.

Nous abordons, depuis quelques années, de manière systématique les questions liées au *calcul géométrique* : robustesse, programmation et analyse expérimentale des algorithmes, conception et développement d'une bibliothèque d'algorithmes (CGAL). Ces recherches devraient largement faciliter, à moyen terme, la diffusion et l'impact des techniques de la géométrie algorithmique et de nos résultats.

Le placement et le déplacement de robots nous ont fourni un sujet d'études et d'applications depuis de nombreuses années. Nos domaines d'application et nos collaborations industrielles se sont également développés au delà de la robotique. Les succès les plus marquants concernent le domaine médical (commercialisation du logiciel NUAGES), la modélisation géométrique en géologie, l'aménagement spatial et la modélisation moléculaire.

3 Fondements scientifiques

3.1 Géométrie Algorithmique

Mots clés : algorithme randomisé, algorithme adaptatif, triangulation, diagramme de Voronoï, triangulation de Delaunay, planification de trajectoires.

Résumé : *La géométrie algorithmique [4] a pour objet l'étude systématique des algorithmes géométriques, de leur conception et de leur analyse. C'est un domaine relativement récent né autour des années 1975. Le projet PRISME s'intéresse aux principales structures géométriques : triangulations, arrangements et diagrammes de Voronoï, étudie des techniques algorithmiques générales et efficaces, et cherche à résoudre quelques problèmes clé comme la planification de trajectoires de robots.*

3.1.1 Géométrie combinatoire

Élucider la combinatoire des objets géométriques est l'un des pré-requis indispensables à la conception d'algorithmes efficaces pour résoudre les problèmes de nature géométrique. L'un des apports majeurs de la géométrie algorithmique est d'avoir mis en évidence le rôle central joué par un petit nombre de structures fondamentales : polytopes, triangulations, arrangements et

diagrammes de Voronoï. Dans ce domaine, les membres du projet PRISME ont notamment étudié la complexité des arrangements dans le plan, l'enveloppe convexe d'un ensemble de sphères en toutes dimensions ou encore différents diagrammes de Voronoï. Plus récemment, nous nous sommes intéressés à des problèmes combinatoires liés à certains problèmes d'optimisation géométrique (percement et couverture).

3.1.2 Techniques algorithmiques

Glossaire :

Algorithmes randomisés Les algorithmes randomisés sont des algorithmes qui effectuent des choix aléatoires au cours de leur déroulement^[MR95].

Contrairement aux algorithmes probabilistes (dits de Monte Carlo) les algorithmes randomisés (encore appelés algorithmes de Las Vegas) fournissent la solution exacte d'un problème déterministe et ne font aucune hypothèse statistique sur les données traitées. Seule leur complexité dépend des choix aléatoires effectués et s'analyse en moyenne. En conférant aux situations défavorables un poids statistique négligeable, la randomisation conduit à des algorithmes simples et efficaces, notamment en géométrie où elle a fait son apparition récemment^[CS89]. La randomisation est aussi devenue une méthode de preuve permettant d'établir certains résultats combinatoires. Enfin, et ce n'est pas le moindre des paradoxes, la dérandomisation d'algorithmes randomisés permet, dans certains cas, d'établir des algorithmes déterministes optimaux.

Le projet PRISME s'est plus particulièrement spécialisé dans les algorithmes randomisés incrémentaux et a développé la méthode du graphe d'influence [4, 3, 10, 7]. Le graphe d'influence est une structure de données qui retrace l'histoire de la construction d'un objet. L'utilisation d'une telle structure conduit à des algorithmes semi-dynamiques capables d'insérer en ligne des données qui ne sont pas connues au départ. Pour la plupart des problèmes fondamentaux, on peut même obtenir un algorithme dynamique traitant non seulement les insertions mais aussi les suppressions de données.

Une autre utilisation de techniques randomisées a récemment été développée pour la recherche de voisins dans des espaces de très grandes dimensions en algorithmique moléculaire.

Glossaire :

Algorithmes adaptatifs Algorithmes dont la complexité est fonction de la valeur effective de certains paramètres significatifs (la taille de l'objet calculé, par exemple) et non de la valeur de ces paramètres dans le pire des cas.

Pour la plupart des problèmes géométriques, la taille du résultat est très variable non seulement en fonction du nombre de données traitées mais aussi en fonction du jeu de données lui-même. Par exemple, l'enveloppe convexe d'un ensemble de n points en dimension d est un polytope dont le nombre de faces peut varier de $(d+1)!$ (pour un $(d+1)$ -simplexe) à $\Omega(n^{\lfloor \frac{d+1}{2} \rfloor})$ (pour un polytope maximal à n sommets); le nombre d'intersections d'un ensemble de n

[MR95] R. MOTWANI, P. RAGHAVAN, *Randomized Algorithms*, Cambridge University Press, New York, NY, 1995.

[CS89] K. L. CLARKSON, P. W. SHOR, « Applications of random sampling in computational geometry, II », *Discrete Comput. Geom.* 4, 1989, p. 387–421.

segments du plan peut varier de 0 à $\frac{n(n-1)}{2}$. Pour ces exemples, il est donc particulièrement intéressant de disposer d'algorithmes adaptatifs dont la complexité dépend de la taille du résultat. Dans d'autres problèmes, ce sont d'autres paramètres qui peuvent jouer un rôle crucial. La contribution du projet PRISME dans ce domaine concerne essentiellement les problèmes d'enveloppes convexes d'objets plans et de percement (ou, par dualité, de couverture).

3.1.3 Triangulations et diagrammes de Voronoï

Glossaire :

Diagramme de Voronoï Soit \mathcal{S} un ensemble de n sites de l'espace euclidien \mathbb{E}^d . Pour chaque site p de \mathcal{S} , la cellule de Voronoï $V(p)$ de p est l'ensemble des points de \mathbb{E}^d qui sont plus proches de p que de tous les autres sites de \mathcal{S} . Le diagramme de Voronoï de \mathcal{S} est la décomposition de \mathbb{E}^d formée par les cellules de Voronoï des sites. Voir la figure 1.

Triangulation de Delaunay Soit \mathcal{S} un ensemble de n points de \mathbb{E}^d . La triangulation de Delaunay de \mathcal{S} est le dual géométrique du diagramme de Voronoï de \mathcal{S} : deux points de \mathcal{S} sont reliés par une arête dans la triangulation de Delaunay si et seulement si leurs cellules sont adjacentes dans le diagramme de Voronoï de \mathcal{S} .

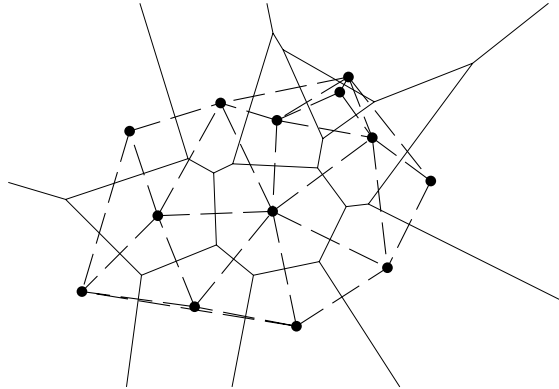


FIG. 1 – Diagramme de Voronoï (en traits pleins) et triangulation de Delaunay (en pointillés) d'un ensemble de points de \mathbb{E}^2 .

Les diagrammes de Voronoï sont des structures très utiles, rencontrées fréquemment car elles permettent de représenter des relations de distance entre objets et des phénomènes de croissance : il n'est pas étonnant de les voir utilisés pour modéliser des cristaux ou les grandes structures de l'univers, et de les trouver souvent dans la nature, par exemple sur la carapace d'une tortue ou sur le cou d'une girafe réticulée. Les diagrammes de Voronoï sont aussi des structures de données permettant de résoudre de nombreux problèmes : recherche de plus proches voisins et planification de mouvements notamment.

L'étude des diagrammes de Voronoï, de leurs propriétés mathématiques, de leur calcul et de leurs nombreuses variantes a été et reste un sujet d'importance majeure de la géométrie algorithmique ^[OBS92]. Les contributions du projet PRISME portent sur les aspects combinatoires

[OBS92] A. OKABE, B. BOOTS, K. SUGIHARA, *Spatial Tessellations: Concepts and Applications of Voronoi*

et algorithmiques, l'extension à différentes métriques non euclidiennes et l'application aux problèmes de reconstruction de formes [5] (cf. section 4.2) et d'approximation de surfaces.

3.1.4 Planification de trajectoires

Une tâche fondamentale en robotique consiste à planifier des trajectoires sans collision dans un environnement encombré d'obstacles. Ce problème est très difficile puisque, même dans sa version la plus simple où les contraintes sur les déplacements du robot sont purement géométriques, sa complexité croît rapidement (de façon probablement exponentielle) avec le nombre de degrés de liberté du robot.

Dans les vingt dernières années, ce problème a suscité des recherches allant des mathématiques (géométrie algébrique, combinatoire, théorie de la complexité), à la programmation et l'implantation effective sur des robots en passant par la conception et l'analyse d'algorithmes performants. L'idée centrale de tous les algorithmes est néanmoins toujours la même : construire une représentation de l'ensemble des configurations du robot pour lesquelles il n'y a pas de collision, ce qu'on appelle l'espace libre du robot. On ramène ainsi le problème du déplacement du robot dans l'espace où celui-ci évolue au problème du déplacement d'un point dans l'espace des configurations dont la dimension est en général élevée (six pour un manipulateur à six degrés de liberté). Moyennant une structuration appropriée de cet espace, on peut alors construire un chemin qui correspond à un déplacement sans collision du robot.

Dans un premier temps, nous avons proposé des méthodes approchées générales qui ont conduit au logiciel industriel ACT. Nous avons ensuite étudié les aspects combinatoires et algorithmiques de cas particuliers importants en pratique : robot mobile évoluant sur un sol plan [1] ou sur un sol accidenté, robot à pattes. On peut obtenir dans ces cas particuliers des solutions analytiques exactes et des algorithmes de calcul efficaces.

Plus récemment, nous avons abordé la planification de trajectoires de robots mobiles soumis à des contraintes cinématiques : on parle dans ce cas de systèmes non holonomes [2]. La grande différence de ces systèmes par rapport aux systèmes holonomes réside dans le fait que tout chemin de l'espace libre n'est pas exécutable par le robot : c'est le problème du créneau qui nécessite de faire des manœuvres si la voiture ne peut pas se déplacer en crabe. Nous nous sommes plus particulièrement intéressés au calcul de trajectoires de courbure bornée correspondant à une voiture dont le rayon de braquage est minoré. Le contrôle optimal permet de caractériser les plus courts chemins de courbure bornée en l'absence d'obstacle. Il est alors possible d'obtenir des algorithmes efficaces en présence d'obstacles.

Ces travaux ont conduit à plusieurs applications industrielles.

3.2 Calcul géométrique

Mots clés : robustesse, dégénérescence, arithmétique, filtre, régularisation, programmation des algorithmes géométriques, recherche expérimentale.

Résumé : *Le calcul géométrique demande, au-delà de la conception et de l'analyse des algorithmes, de résoudre les questions délicates posées par leur pro-*

grammation. Il s'agit tout d'abord de savoir représenter et organiser les structures géométriques de façon cohérente et efficace. Il s'agit également de garantir un calcul fiable, ce qui est particulièrement délicat avec les objets géométriques dont la représentation est à la fois symbolique et numérique. ^[Hof89] *Ces problèmes sont un des principaux obstacles à l'utilisation des résultats de la géométrie algorithmique.* ^[C+96] *Nous détaillons dans la première section les problèmes posés par le calcul géométrique, puis nous décrivons un paradigme pour les résoudre. Finalement, nous présentons le projet CGAL, qui pose les fondements d'une bibliothèque d'algorithmes géométriques.*

3.2.1 Robustesse et dégénérescences

Les algorithmes sont généralement exprimés dans un modèle où chaque nombre réel est représentable de façon exacte. En pratique, la représentation des nombres réels en machine pose de gros problèmes qui sont très vite observables. Le résultat des opérations arithmétiques pouvant être tronqué, certaines propriétés géométriques du plan euclidien ne sont pas vérifiées dans l'implantation. Par ailleurs, le traitement des cas dégénérés, souvent ignorés dans les articles, est fastidieux, allonge considérablement les programmes et constitue une source d'erreurs.

Plusieurs approches ont été suivies pour tenter de résoudre ces problèmes. Une méthode de perturbation symbolique, qui permet de résoudre les cas dégénérés, a été développée et raffinée. Pour les algorithmes incrémentaux randomisés, nous avons montré comment perturber la définition du problème pour prendre en compte les cas dégénérés.

Pour ce qui concerne les problèmes de stabilité numérique, une approche prometteuse consiste à utiliser une arithmétique des prédicats exacte (voir la prochaine section). De façon complémentaire, on peut chercher à développer des algorithmes nécessitant une précision minimale. Le formalisme utilisé, basé sur la notion de *degré* d'un algorithme, permet d'analyser les algorithmes et les problèmes en fonction de leur degré et de chercher des compromis entre degré et complexité [23].

3.2.2 Arithmétique et filtres

Les prédicats constituent l'interface entre les parties numérique et combinatoire des algorithmes géométriques. Le calcul exact de prédicats^[YD95] garantit donc la stabilité numérique des algorithmes géométriques et permet à l'implantation de vérifier et d'utiliser les résultats du plan euclidien, plutôt que du plan des nombres flottants (qui en est une discrétisation). Une arithmétique exacte des prédicats ne nécessite pas nécessairement le recours à des types

[Hof89] C. M. HOFFMANN, « The Problems of Accuracy and Robustness in Geometric Computation », *IEEE Computer* 22, 3, mars 1989, p. 31–41.

[C+96] B. CHAZELLE *et al.*, « Application Challenges to Computational Geometry: CG Impact Task Force Report », *Technical Report n° TR-521-96*, Princeton Univ., avril 1996, <http://www.cs.princeton.edu/~chazelle/taskforce/CGreport.ps>.

[YD95] C. K. YAP, T. DUBÉ, « The exact computation paradigm », *in: Computing in Euclidean Geometry*, D.-Z. Du et F. K. Hwang (éditeurs), 2nd édition, *Lecture Notes Series on Computing, 1*, World Scientific Press, Singapore, 1995, p. 452–492, <http://cs.nyu.edu/cs/faculty/yap/papers/paradigm.ps>.

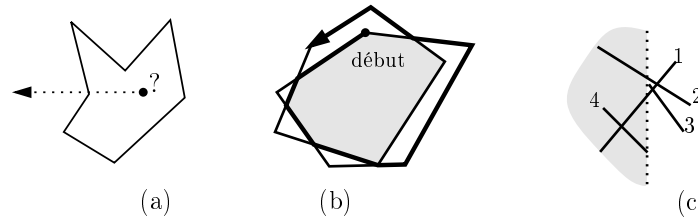


FIG. 2 – *Différents problèmes de robustesse :*

(a) Une méthode couramment utilisée pour localiser un point dans un polygone consiste à compter la parité du nombre d'intersections d'un rayon allant à l'infini. Ici, ce rayon passe par un sommet. Si les segments sont fermés, on compte deux intersections, aucune si les segments sont ouverts ; dans les deux cas, le point est placé incorrectement à l'extérieur du polygone.

(b) Une méthode couramment employée pour construire l'intersection de deux polygones consiste à parcourir, à partir d'un sommet, la frontière de chacun des polygones alternativement jusqu'au point de départ. Ici, on peut arranger les segments de façon à ce que la deuxième intersection ne soit pas détectée par un calcul en virgule flottante ; l'algorithme fournit alors une solution erronée.

(c) L'algorithme de Bentley-Ottmann, l'un des plus utilisés en géométrie algorithmique, calcule les intersections d'un ensemble de segments en balayant le plan et en s'arrêtant quand la droite passe par un sommet ou une intersection. Ici, on peut disposer les segments de façon à ce qu'un calcul flottant traite les événements dans le mauvais ordre. Si l'intersection entre 1 et 2 est considérée à gauche de l'extrémité de 3, alors l'intersection entre 1 et 3 n'est pas détectée.

arithmétiques exacts. On peut, par exemple, montrer que certains prédicats comme le signe des déterminants peuvent s'évaluer en simple précision de manière fiable et efficace [6].

Une autre méthode consiste à utiliser des types arithmétiques exacts disponibles dans des bibliothèques telles que GMP ou Leda. Pour rester efficace, il faut avoir recours à des *filtres* qui permettent d'améliorer progressivement le résultat flottant jusqu'à atteindre la fiabilité requise.^[FvW96] En moyenne, dans la grande majorité des cas, le calcul flottant suffit pour conclure et l'arithmétique exacte n'est que rarement invoquée. Globalement, la perte de performance est minime [19].

Nous prévoyons donc de produire et automatiser ces outils de filtrage et de les intégrer dans la bibliothèque CGAL. Le but recherché est de libérer le programmeur de ces considérations de robustesse en lui procurant des prédicats exacts à coût très abordable. Des études préliminaires sur des algorithmes classiques ont montré que ce but est réaliste [12].

3.2.3 Régularisation

Si le calcul des prédicats est exact, l'algorithme retourne un résultat combinatoire correct (carte planaire, diagramme de Voronoï, etc.) Reste le problème d'arrondir les coordonnées des objets géométriques sous une forme représentable en machine. Le calcul flottant ne garantit pas que la structure arrondie vérifie des propriétés comme la planarité ou la convexité. Bien que cela n'affecte pas le déroulement de l'algorithme lui-même, cela peut avoir des incidences sur les algorithmes qui utilisent ensuite ce résultat. Des méthodes ont été étudiées pour *régulariser* certaines constructions. Elles supposent toutes de savoir arrondir les constructions géométriques sur une grille.

Nous avons développé une méthode permettant d'arrondir correctement des constructions géométriques à un point entier le plus proche. Des outils d'automatisation pour la bibliothèque CGAL sont en cours d'étude. Il faudra aussi évaluer le coût de cette approche. Il reste sur ce sujet beaucoup de recherches à effectuer, en particulier pour des structures plus complexes comme les diagrammes de Voronoï de segments ou les arrangements tridimensionnels [25].

3.2.4 La bibliothèque CGAL

Comme on l'a vu, la programmation des algorithmes géométriques pose des problèmes difficiles qui ne relèvent pas seulement de l'ingénierie logicielle. Ces problèmes se posent à tous ceux qui utilisent des algorithmes géométriques et il n'existe pas actuellement de solution simple. C'est pourquoi une bibliothèque d'algorithmes géométriques, baptisée CGAL est développée en collaboration avec d'autres équipes européennes dans le cadre de projets européens Esprit : Freie Universität Berlin (Allemagne), Universität Halle (Allemagne), RISC Linz (Autriche), Max-Planck-Institut Saarbrücken (Allemagne), Tel Aviv University (Israël), Universiteit Utrecht (Pays-Bas), et ETH Zürich (Suisse).

Cette bibliothèque est écrite en C++ et veut permettre une programmation facile, fiable et efficace d'applications nécessitant la mise en œuvre d'algorithmes géométriques (voir aussi 5.1). La généricité est une des caractéristiques de cette bibliothèque. Les algorithmes peuvent

[FvW96] S. FORTUNE, C. J. VAN WYK, « Static Analysis Yields Efficient Exact Integer Arithmetic for Computational Geometry », *ACM Trans. Graph.* 15, 3, juillet 1996, p. 223–248.

utiliser différents types de données dès lors que certaines méthodes sont définies ce qui rend le code largement réutilisable. Cette approche contraste avec celle des « perles » utilisées par exemple en infographie qui doivent être adaptées à chaque nouveau besoin. La généralité doit aussi permettre de comparer les caractéristiques de différents algorithmes dans un même contexte et de conduire des recherches expérimentales.

4 Domaines d'applications

4.1 Panorama

Mots clés : ingénierie inverse, CAO, médical, géologie, satellite, placement, découpe, chimie, molécule.

<http://www.ics.uci.edu/~eppstein/geom.html>

La géométrie algorithmique trouve des applications dans tous les domaines où l'on doit acquérir, modéliser, manipuler ou transformer des données géométriques en quantité importante.

Jusqu'à présent nous avons surtout abordé des applications en reconstruction de formes, en robotique et en modélisation moléculaire mais nos travaux peuvent trouver des applications dans d'autres domaines tels que la conception assistée par ordinateur, la synthèse d'image ou les systèmes d'information géographique.

4.2 Reconstruction de formes

Mots clés : modélisation géométrique, reconstruction de formes, imagerie médicale, géologie.

Résumé : *Reconstruire la forme d'un objet tridimensionnel [5] à partir d'un certain nombre de mesures effectuées sur cet objet est une opération de base dans de nombreux domaines applicatifs. Le projet PRISME s'est plus particulièrement intéressé aux problèmes de reconstruction tels qu'ils se posent dans les domaines de l'imagerie médicale et de la géologie. Dans ces deux domaines, des solutions originales, largement basées sur les concepts de diagrammes de Voronoï et triangulations de Delaunay, ont été proposées.*

4.2.1 Imagerie médicale

Dans le domaine médical, les données proviennent généralement d'un système d'imagerie comme la RMN qui permet d'acquérir plusieurs coupes parallèles de la région concernée. La première opération consiste alors à extraire de chaque coupe (de façon manuelle ou en utilisant un procédé automatique d'analyse d'image) un ensemble de contours polygonaux approximant les sections de l'organe à reconstruire. Le procédé de reconstruction considère alors successivement chaque paire de coupes consécutives et reconstruit, à l'aide de la triangulation de Delaunay, la tranche d'organe comprise entre ces deux coupes. On obtient ainsi une représentation volumique de l'objet, une triangulation tridimensionnelle, qui peut se prêter à divers traitements ultérieurs ; visualisation, simulation d'intervention ou planification de trajectoire pour guider

une endoscopie (figure 3). Ces travaux [9] ont conduit au logiciel NUAGES commercialisé par plusieurs sociétés (voir 5.4).

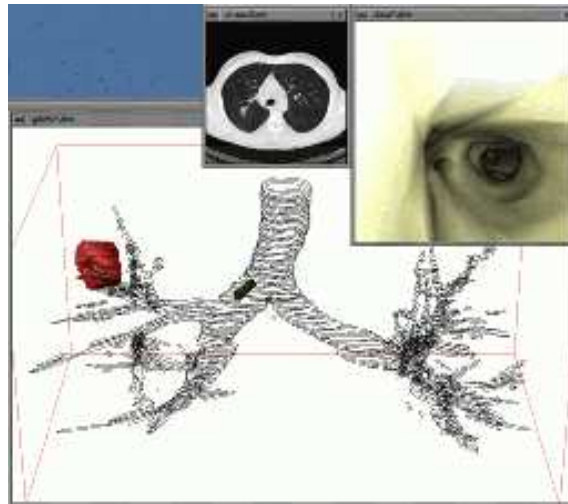


FIG. 3 – *Simulation d'endoscopie.*

4.2.2 Géologie

Parvenir à une bonne connaissance du sous-sol d'une région est un enjeu majeur dans l'exploitation des ressources minières, le génie civil ou la protection de l'environnement. Pour y parvenir le géologue dispose de plusieurs sources d'informations : sondages, études sismiques et relevés topographiques qui constituent généralement un ensemble de données hétérogènes, éparses et imprécises. Le projet PRISME développe actuellement une méthode de reconstruction automatique de formations géologiques. Cette méthode se distingue des approches plus traditionnelles d'interpolation de surfaces et construit directement une représentation volumique des formations. Entre autres avantages, cette approche permet de segmenter automatiquement les données et de s'affranchir dans une large mesure des problèmes d'instabilité numérique (figure 4, voir 5.5).

4.3 Algorithmique moléculaire

Mots clés : molécule, chimie, docking.

L'émergence de méthodes algorithmiques en biologie moléculaire remonte sans doute aux années 80, mais le processus s'est accéléré depuis environ cinq ans comme en témoignent l'apparition de journaux et conférences spécifiques, ainsi que la publication d'articles relatifs au sujet dans les forums traditionnels comme le symposium ACM de géométrie algorithmique ou *Algorithmica*.

L'objectif de ces recherches est de trouver des médicaments plus efficaces, moins nocifs, et plus faciles à extraire ou à synthétiser. L'enjeu est donc considérable et ce d'autant plus

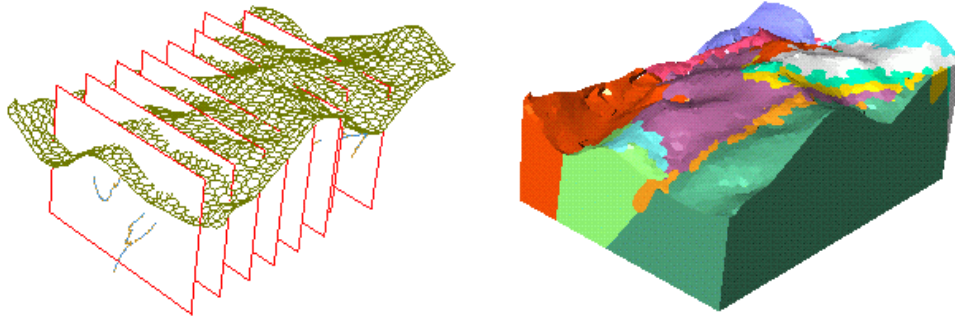


FIG. 4 – *Reconstruction du sous-sol de Morges (Alpes).*

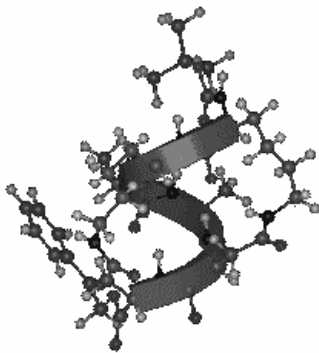
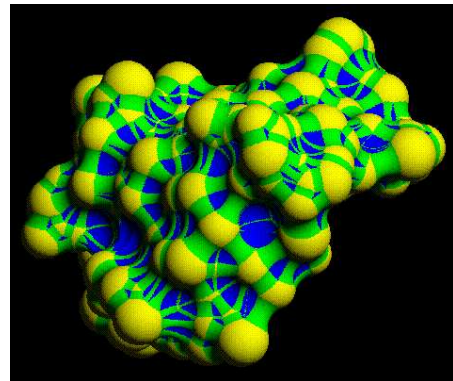


FIG. 5 – (a) *Modèle moléculaire*



(b) *Surface moléculaire*

que des méthodes telle que la synthèse combinatoire donnent accès à une diversité moléculaire gigantesque.

Parmi les problèmes algorithmiques bien identifiés, figurent l'étude des surfaces moléculaires (figure 5), l'investigation des conformations moléculaires stables, le docking ou étude de la complémentarité ligand-récepteur, ainsi que le clustering de grandes bases de données moléculaires. Dans le cadre d'une collaboration avec Sanofi Recherche, nous avons fait une contribution à ce dernier problème (voir 6.4.3).

4.4 Placement de formes

Mots clés : placement géométrique, découpe industrielle, aménagement de satellites.

Résumé : *Le placement de formes rigides ou articulées est un champ d'application privilégié de la géométrie algorithmique. Il s'agit de savoir si un objet donné peut être placé parmi une collection d'objets. Nous avons plus particulièrement étudié deux applications d'origine industrielle : la découpe et l'aménagement de satellites.*

4.4.1 Découpe

Il s'agit de découper sur un support plan (peau, tôle, tissu, etc) des formes géométriques données, souvent polygonales. Deux objectifs sont poursuivis : rapidité du placement des formes sur le support et minimisation des chutes.

Nous avons résolu les problèmes de placement et de déplacement dans le cas d'un polygone (resp. polyèdre) en translation dans un univers polygonal (resp. polyédrique) en proposant des algorithmes de complexité quasi-optimale. Ces algorithmes permettent de calculer un placement s'il en existe un, mais également l'ensemble des positions admissibles de l'objet à placer. Le placement simultané de deux polygones en translation dans un environnement polygonal, et de trois polygones dans un parallélogramme, a également été résolu. Le calcul de l'espace admissible pour un polygone en translation et rotation peut aussi être effectué avec une complexité presque optimale.

Ces algorithmes permettent d'exhiber rapidement une solution au problème de découpe, en plaçant l'une après l'autre les différentes formes sur le support. Ils sont à la base du logiciel PIAF (voir 5.7).

4.4.2 Aménagement de satellites

L'aménagement d'un satellite consiste à positionner les équipements spécifiés par un schéma logique sur une plate-forme, en prenant en compte des contraintes physiques imposées par l'environnement (par exemple la protection contre les rayons solaires) et par les équipements entre eux (par exemple l'interférence électromagnétique et le respect du champ de vue d'un instrument d'observation). L'aménagement spatial d'un satellite s'inscrit dans un processus de conception complexe qui dure en général plusieurs mois et nécessite de nombreux allers retours entre bureau d'études et équipes spécialisées en électro-magnétisme, thermique et mécanique. L'objectif de notre travail est de démontrer, en développant un prototype, qu'il est possible d'accélérer considérablement la phase de conception en assistant, voire en automatisant, l'aménagement avec des méthodes issues de la géométrie algorithmique. Un résultat est présenté sur la figure 6.

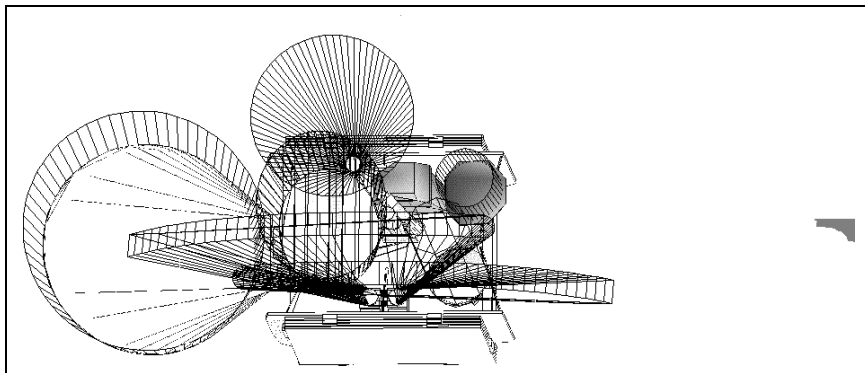


FIG. 6 – à droite : espace admissible pour l'instrument grisé.

5 Logiciels

5.1 Bibliothèque d'algorithmes géométriques CGAL

Participants : Jean-Daniel Boissonnat, Carine Bonetto, Hervé Brönnimann, Frédéric Cazals, Frank Da, Olivier Devillers, Alexandre Guyennon, Johann Liberman, Sylvain Pion, François Rebufat, Monique Teillaud, Mariette Yvinec [correspondante].

Voir aussi les paragraphes 3.2.4, 6.1 et 8.2.

Le site du projet : <http://cs.uu.nl/CGAL/>.

CGAL est une bibliothèque de programmes géométriques développée dans le cadre de deux projets Européen successifs (projet ESPRIT IV LTR CGAL décembre 97 - juin 98, projet ESPRIT IV LTR GALIA novembre 99- mai 99) par un consortium formé de huit équipes de recherches des organismes suivants : Universiteit Utrecht, Max-Planck-Institut, INRIA Sophia Antipolis, ETH Zürich, Tel Aviv University, Freie Universität Berlin, Universität Halle, RISC Linz. CGAL a pour but de rendre accessibles au monde industriel et aux domaines applicatifs les solutions avancées que propose la géométrie algorithmique.

La bibliothèque CGAL comprend un noyau, une bibliothèque de base et un certain nombre de modules support. Le noyau est formé de classes C++ qui représentent les objets géométriques élémentaires (points, vecteurs, directions, droites, rayons, plans, simplexes, boîtes isothétiques) ainsi que des transformations affines et un certain nombre de prédicats géométriques sur ces objets. Actuellement ces classes existent en dimension 2 et 3 ; à terme elles seront disponibles en toutes dimensions. Par le mécanisme des classes génériques, chaque classe peut être instanciée selon plusieurs modes de représentation : on peut choisir entre coordonnées cartésiennes ou homogènes, ainsi qu'entre différentes arithmétiques.

La bibliothèque de base propose un certain nombre de structures de données géométriques ainsi que des algorithmes. Les structures de données comprennent polygones, polyèdres, triangulations, cartes planaires, arrangements et diverses structures de recherche (arbre de segments, arbres k -dimensionnels, etc.). Des algorithmes sont disponibles pour calculer des enveloppes convexes et résoudre des problèmes d'optimisation (linéaires, quadratiques, de type linéaires généralisés). Par le biais des classes et des fonctions génériques, ces algorithmes peuvent s'utiliser soit avec les objets du noyau, soit avec des classes géométriques propres à l'utilisateur pourvu que celles-ci satisfassent un ensemble de requêtes spécifiées par une *classe d'implantation*.

Enfin, une bibliothèque support permet de générer des objets aléatoires, et d'interfacer CGAL avec plusieurs outils de visualisation (fichier ascii, Leda windows, OpenGL, Open Inventor, Tcl/Tk, PostScript).

CGAL compte actuellement environ 160 000 lignes de codes et supporte les plateformes suivantes : Mips Pro (CC) , SUNOS(egcs, g++), Linux(egcs, g++), WindowsNT(Visual C++). La version 2.0 de CGAL est disponible sur le web depuis juin 99. Elle a été téléchargée plus de 500 fois en deux mois et demi. La version 2.1 assurant le support des plateformes WindowsNT(Visual C++) est prévue pour novembre 99.

5.2 Évaluation exacte d'expressions numériques

Participants : Hervé Brönnimann, Olivier Devillers, Sylvain Pion [correspondant], Mariette Yvinec.

<http://www.inria.fr/prisme/logiciel/>.

L'un des points cruciaux en géométrie, concernant la vitesse autant que la robustesse, est le calcul des prédicats, tels les signes de déterminants.

Pour ce faire, une bibliothèque d'arithmétique d'intervalles permet d'accélérer les calculs, tout en vérifiant leur exactitude : c'est un filtre arithmétique. La bibliothèque proposée ici est plus adaptée aux besoins des prédicats que d'autres bibliothèques disponibles, car elle permet de s'affranchir des changements répétitifs de modes d'arrondis, et est donc plus rapide.

En complément, pour effectuer les calculs exacts en cas d'échec d'un filtre, on peut utiliser la bibliothèque d'arithmétique modulaire, qui permet de calculer en particulier les signes de déterminants et d'expressions polynômiales plus générales, qui interviennent dans la quasi totalité des algorithmes géométriques.

Nous proposons également des méthodes plus spécifiques pour le calcul de déterminants de petites dimensions sur des entiers.

5.3 Hiérarchie de Delaunay

Participant : Olivier Devillers [correspondant].

Versions de démonstrations disponibles,

<http://www.inria.fr/prisme/logiciel/del-hierarchy/fr.html>

Nous proposons une nouvelle structure de données pour le calcul de la triangulation de Delaunay de points du plan permettant de combiner simultanément une bonne complexité théorique dans le cas le pire, un très bon comportement pratique et une occupation mémoire réduite. Cette structure permet également une mise à jour dynamique (insertions et suppressions).

Ce programme peut calculer la triangulation de 500 000 points aléatoires dans le plan en 22 secondes et 150 000 points sur la surface d'un objet tridimensionnel en 40 secondes (sur SUN Ultra10) [8].

5.4 Reconstruction à partir de coupes.

Participants : Jean-Daniel Boissonnat [correspondant], Bernhard Geiger.

<http://www.inria.fr/prisme/personnel/geiger/nuages.html>

NUAGES, logiciel de reconstruction 3D à partir de sections 2D (voir 4.2.1) développé par Bernhard Geiger, est utilisé notamment pour l'imagerie médicale, la microscopie électronique et la reconstruction de surface. Il est commercialisé par Noesis, Cril, Siemens, Leibinger. Des exécutable pour diverses plateformes sont disponibles par ftp anonyme. Plusieurs publications dans des domaines appliqués décrivent des résultats obtenus avec NUAGES [9].

5.5 Modeleur géométrique pour la géologie

Participant : Jean-Daniel Boissonnat [correspondant].

<http://www.inria.fr/prisme/fiches/Geologie/>

GEO3D (voir 4.2.2) permet de construire automatiquement un modèle volumique à partir de données hétérogènes et irrégulières. Ce logiciel est développé dans le cadre du programme national GÉOFrance3D de cartographie du sous-sol de la France. Une première version est utilisée au BRGM. Ce travail est le résultat de la thèse de Stéphane Nullans.

5.6 Aménagement de satellites

Participants : Jean-Daniel Boissonnat, Eelco de Lange, Monique Teillaud [correspondante].

<http://www.inria.fr/prisme/fiches/Satellite/>

Le logiciel GÉOTOOLS pour l'aménagement d'un satellite permet le placement itératif d'instruments (antennes et autres équipements) sur une plate-forme plane. Le logiciel a été conçu dans le cadre de notre collaboration avec MATRA MARCONI SPACE.

5.7 Placement et déplacement

Participant : Jean-Daniel Boissonnat [correspondant].

Bernard Faverjon et le projet PRISME sont à l'origine du logiciel ACT permettant la planification de trajectoires pour des robots manipulateurs. Ce logiciel est commercialisé par Aleph Technologies.

Francis Avnaim a développé un logiciel de placement de formes pour la découpe de pièces bidimensionnelle (tissus, métal, cuir...). Ce logiciel est commercialisé par la société Nakache.

6 Résultats nouveaux

6.1 CGAL

Mots clés : géométrie algorithmique, degré algébrique des algorithmes, triangulation, triangulation de Delaunay, reconstruction de formes, alpha-formes.

Voir aussi les paragraphes 3.2.4, 5.1 et 8.2.

Le site du projet : <http://cs.uu.nl/CGAL/>

Résumé : *Le développement de la bibliothèque de programmes géométriques CGAL (Computational Geometry Algorithms Library) s'est poursuivie très activement cette année. Trois versions publiques de CGAL ont vu le jour au cours de l'année 99 : à la version 1.2 [36] sortie en février 99 a succédé la version 2.0 [38] qui introduit dans CGAL le mécanisme des namespace indispensable au portage de CGAL sur WindowsNT. La sortie de la version 2.1 [37] qui réalise effectivement le portage de CGAL sur WindowsNT est prévue pour novembre 99. Au sein du*

consortium européen qui développe CGAL, le projet PRISME est plus particulièrement chargé de la partie cartésienne du noyau, de certains modules arithmétiques et des modules de triangulations.

6.1.1 Noyau

Participant : Hervé Brönnimann.

Hervé Brönnimann a étendu et maintenu le code du noyau en coordonnées cartésiennes (plus de 13000 lignes de code). Le noyau tourne maintenant sur Visual C++ de Microsoft Corp. La conception du noyau a été entièrement remanié de façon à permettre une modularité totale (chaque classe est interchangeable indépendamment et séparément, en réutilisant le code existant), et coexiste avec l'ancien design (qui reste pour certains compilateurs ne pouvant supporter le nouveau design).

6.1.2 Modules d'arithmétique

Participant : Sylvain Pion.

La gestion des problèmes de précision dans CGAL est effectuée par le choix d'un type numérique pour les données géométriques. On peut, par exemple, utiliser des types numériques exacts pour garantir le bon fonctionnement de l'algorithme. Dans ce cadre, nous développons plusieurs types de nombres adaptés à l'évaluation des prédicats géométriques.

Nous avons, l'année passée, développé des filtres arithmétiques basés sur l'arithmétique d'intervalle, effectuant du calcul d'erreur lors de l'exécution du programme. Cette année, nous avons introduit de nouveaux filtres qui opèrent dès la compilation. Cette approche fonctionne dans moins de cas, mais est plus avantageuse du point de vue de la performance.

Nous avons également développé un type de nombre exact *filtré*, c'est-à-dire qui n'effectue le calcul exact multiprécision qu'en cas de demande, lorsque la précision des intervalles est insuffisante. Ce type de nombre, bien que moins performant que l'approche précédente basée sur l'étude des prédicats, est d'une utilisation plus simple.

6.1.3 Triangulations bidimensionnelles

Participants : Jean-Daniel Boissonnat, Frank Da, Mariette Yvinec.

Le module triangulation de CGAL a encore été légèrement remanié cette année pour permettre une meilleure distinction entre les aspects combinatoires et géométriques des triangulations et assurer un traitement cohérent des situations dégénérées. La représentation des triangulations dans CGAL est maintenant pleinement cohérente : quelle que soit la dimension d de la triangulation, l'ensemble des faces et des sommets représentés est combinatoirement équivalent à la triangulation d'une d -sphère topologique. Le module triangulation s'est aussi enrichi. Les triangulations contraintes supportent maintenant l'insertion ou la suppression en ligne de nouvelles contraintes. Une classe gérant les triangulations de Delaunay contraintes a été ajoutée. Cette classe abrite aussi quelques fonctions permettant le raffinement d'un maillage. Cette fonctionnalité constitue l'embryon de ce qui, à terme, doit devenir un module séparé de

traitement des maillages. Enfin des classes permettant la construction, à partir d'un ensemble de points, d'alpha-formes et d'alpha-formes pondérées font dorénavant partie de la distribution de CGAL.

6.1.4 Triangulations tridimensionnelles

Participant : Monique Teillaud.

La version 2.1 de la bibliothèque CGAL contient un module de triangulation tridimensionnelle mis au point cette année [29, 41].



FIG. 7 – *Triangulation de Delaunay tridimensionnelle calculée par la bibliothèque CGAL*

Lors de la conception, l'accent a été mis sur une systématisation de la distinction entre combinatoire et géométrie, déjà présente dans les modules de CGAL existants. Deux classes principales structurent le module :

- la classe représentant la triangulation géométrique, avec deux paramètres : une classe où l'utilisateur peut spécifier son type de points ainsi que les opérations élémentaires sur ces points, la deuxième classe étant la triangulation combinatoire ; une triangulation géométrique d'un ensemble A de points de \mathbb{R}^3 est vue comme une partition de l'espace par des tétraèdres dont les sommets sont quatre points de A , et par des cellules non bornées dont trois sommets sont des points de A formant une face de l'enveloppe convexe de A , et le quatrième sommet est à l'infini,

- la classe représentant la triangulation combinatoire, dans laquelle aucun aspect géométrique n'intervient ; cette triangulation est un graphe avec des sommets et des cellules liés par des relations d'incidence et d'adjacence, les points n'apparaissent pas, le sommet à l'infini ne peut être distingué des autres sommets.

Une attention toute particulière a été portée au traitement des dimensions dégénérées (cas de points tous coplanaires ou de points tous alignés dans l'espace). Ainsi la triangulation combinatoire proposée par ce module de CGAL représente une triangulation d'une sphère topologique S^d de \mathbb{R}^{d+1} , pour $d \in \{0,1,2,3\}$. Selon la dimension, une cellule a donc 1, 2, 3 ou 4 sommets.

Le module propose des triangulations de Delaunay (Figure 7) et des triangulations régulières de points pondérés, représentées par des classes dérivées de la triangulation géométrique.

6.1.5 Visualisation 3D

Participants : Frédéric Cazals, François Rebufat.

Les outils de visualisation 3D simples, modulaires et performants tels qu'Open Inventor n'étant pas dans le domaine public, nous avons écrit en C++ `Viewer_3`, un visualisateur basé sur OpenGL. Il est spécialement conçu pour des objets géométriques 3D, e.g. issus de CGAL, et dispose de méthodes spécifiques permettant d'entrer, supprimer, modifier, et sélectionner les données manipulées. Des exemples de telles fonctionnalités sont l'insertion points dans les objets géométriques, la définition de plans de *clipping*, etc.

Son architecture logicielle s'appuie sur le *multi-threading*¹. Plus précisément, les parties calcul géométrique et visualisation sont exécutées dans deux threads différents. Cela permet entre autres choses, lorsque l'application est lancée sous le débogueur, l'examen interactif des objets géométriques manipulés.

Le visualiseur a d'autre part été conçu pour que l'utilisateur puisse redéfinir sa propre gestion d'événements et y intégrer ses propres panneaux de boutons et menus. Une interface simple pour les «objets graphiques» le rend extensible pour tout type d'objet, ce qui en fait un outil de visualisation pouvant être utilisé dans un cadre plus large que celui de la librairie CGAL.

Ce outil est utilisé par une démonstration 3D présentant les objets et fonctions tridimensionnelles de CGAL. L'idée est de présenter les objets géométriques dans une interface visuelle de qualité et d'offrir un ensemble de fonctionnalités permettant à l'utilisateur de «jouer» avec ces objets. Le graphisme est géré par OpenGL et l'interface par Xform.

Une démonstration est également disponible sur le *web* sous forme d'une *applet java* interfacée avec une librairie CGAL. Le code CGAL s'exécute sur le serveur alors que l'applet tourne côté client.

6.2 Algorithmes géométriques

Mots clés : géométrie algorithmique, triangulation de Delaunay, diagramme de Voronoï.

1. Un *thread* est un flot de contrôle indépendant à l'intérieur d'un process. Le multi-threading est ici utilisé pour la raison suivante : deux threads lancés par le même process ont accès aux mêmes variables globales : les structures de données géométriques ainsi que leur traduction en fonctions OpenGL.

6.2.1 Rôle des prédicats dans les algorithmes géométriques

Participants : Jean-Daniel Boissonnat, Antoine Vigneron.

Ce travail [45, 31] prolonge les résultats obtenus les années précédentes avec F. Preparata et J. Snoeyink. Soit un ensemble de n objets courbes du plan, on veut trouver les k paires qui s'intersectent en limitant la complexité des prédicats utilisés. En particulier, on s'interdit de compter les points d'intersection et de les ordonner. On veut ainsi améliorer la fiabilité et faciliter la programmation en limitant le nombre de situations dégénérées à envisager. Un premier algorithme pour des objets épais de même taille est proposé. Il s'exécute en temps $O(n \log n + \frac{D^2}{S}(n+k))$ si S est l'aire minimale de ces objets et D le diamètre maximum. Il se généralise en dimension quelconque, et sa restriction aux disques ou aux boules de même rayon en basse dimension est optimale en temps, en espace, en degré et en nombre d'entrées-sorties. On propose ensuite plusieurs algorithmes dans le cas des arcs monotones qui n'utilisent que le prédicat d'intersection, le prédicat dessus-dessous entre un point et un arc, et la comparaison des abscisses des extrémités des points. Ces algorithmes ne demandent donc pas de connaître le nombre de points d'intersection ni leur positions respectives. En particulier, le temps de calcul est fonction du nombre de paires de courbes qui se coupent et non du nombre de points d'intersection à la différence des autres algorithmes non triviaux connus. Le cas bleu-rouge est traité en temps $O((n+k) \log n)$. Dans le cas général pour des courbes ou des pseudo-segments, on présente des algorithmes qui se rapprochent des bornes inférieures connues.

6.2.2 Compression de structures géométriques

Participants : Olivier Devillers, Pierre-Marie Gandoin.

La compression de structures géométriques est un domaine de recherche relativement nouveau, situé entre la géométrie algorithmique et la compression de données standard. L'essor rapide des applications de la synthèse d'image rend nécessaires la manipulation et l'échange de données géométriques de manière rapide et économique. En particulier, les nombreuses possibilités offertes par le World Wide Web en matière de réalité virtuelle sont conditionnées par un accès rapide aux données. Cela implique — en particulier lorsque la bande passante est faible — d'organiser et de comprimer les données géométriques de manière optimale. Depuis 1995, plusieurs articles scientifiques ont traité le problème du codage de maillages (scènes géométriques en 2 ou 3 dimensions composées de polygones) de la façon suivante : les sommets du maillage sont transmis dans un ordre établi pour contenir partiellement la topologie du maillage (c'est-à-dire les arêtes des polygones). Parallèlement, un ensemble de règles simples permet de prédire la position d'un sommet à transmettre à partir des positions de ses voisins déjà transmis. Dans notre article, nous décrivons un algorithme de compression dont le principe général est complètement différent : l'ordre de transmission des sommets du maillage est exploité pour comprimer leurs coordonnées, et la topologie du maillage est ensuite reconstruite à partir des sommets. Cet algorithme atteint des facteurs de compression légèrement supérieurs à ceux obtenus par les algorithmes actuellement disponibles pour la compression géométrique, et il permet en outre une transmission progressive et interactive des données.

6.2.3 Union de boules

Participants : Hervé Brönnimann, Olivier Devillers.

La complexité (le nombre d'arêtes et de faces) de l'union, de l'enveloppe convexe ou de l'intersection d'un ensemble de boules est quadratique dans le cas le pire, mais si on se restreint à des boules de même rayon la complexité de l'intersection et de l'enveloppe convexe deviennent linéaire. Nous avons développé un contre-exemple démontrant qu'il n'en était pas de même pour l'union, et ce même si on imposait à ces boules d'avoir une intersection commune[32].

6.3 Calcul géométrique

Mots clés : robustesse, perturbation, dégénérescence, vérification, balayage, enveloppe convexe.

6.3.1 Degré algébrique des algorithmes

Participants : Olivier Devillers, Alexandra Fronville, Monique Teillaud.

Travail effectué avec le soutien de l'action coopérative FIABLE et en collaboration avec Bernard Mourrain (SAGA).

Un algorithme géométrique comprend en général une partie combinatoire, qui calcule une structure de données géométrique, et une partie numérique, qui calcule des points. La robustesse de l'algorithme dépend de choix découlant de prédicats géométriques, faire ces choix de manière exacte garantit que la structure de données sera cohérente, alors que les calculs numériques peuvent rester approchés sans que le bon déroulement de l'algorithme en soit affecté. Les prédicats sont des combinaisons de polynômes, et leur degré est une mesure de la précision nécessaire pour les calculs.

La comparaison des abscisses de deux points d'intersection est un prédicat de base pour un certain nombre d'algorithmes. Ce prédicat a été étudié en détail pour le cas d'intersections de segments de droites^[BP97].

Les problèmes analogues pour des objets courbes n'ont pas encore été abordés dans la littérature. Nous nous sommes concentrés sur le cas le plus simple, celui des arcs de cercles [33]. Nous proposons une méthode permettant de comparer de façon exacte deux points d'intersection entre arcs de cercles, sans calculer ces points d'intersection. Cette méthode fait intervenir le calcul de Bézoutiens multivariés. Nous montrons qu'il peut être nécessaire d'évaluer exactement le signe de polynômes de degré 12, mais que dans certaines configurations, évaluer les signes de polynômes de degré 6 est suffisant.

Nous avons implanté la méthode en C++ et évalué expérimentalement ses performances, ce qui nous a permis de démontrer son efficacité pratique. Plusieurs filtres arithmétiques ont été testés [12, 28].

[BP97] J.-D. BOISSONNAT, F. P. PREPARATA, « Robust Plane Sweep for Intersecting Segments », *Research Report n° 3270*, INRIA, 1997, <http://www.inria.fr/RRRT/RR-3270>.

6.3.2 Signe de déterminants

Participant : Alexandra Fronville.

Travail effectué avec le soutien de l'action coopérative FIABLE et en collaboration avec Bernard Mourrain (SAGA).

Les prédicats géométriques pouvant être reliés, dans beaucoup de cas, au signe de certains déterminants, nous nous sommes penchés sur des algorithmes certifiés et rapides utilisant des arithmétiques en simple ou double précision pour le calcul de tels signes. Dans le cas où la matrice est loin d'être singulière, une décomposition LU nous permet de certifier le signe. Sinon, ce même algorithme nous donne des informations sur son noyau, ce qui nous permet de transformer le problème initial en une meilleure situation pour le calcul numérique. Un article est en cours de rédaction sur les différents résultats que nous avons obtenus dans cette direction.

6.4 Applications de la géométrie algorithmique

6.4.1 Reconstruction de formes

Participants : Jean-Daniel Boissonnat, Frédéric Cazals, Frank Da, Radu Lupşa.

Soit $\mathcal{A} = \{A_1, \dots, A_n\}$ un ensemble fini de points de \mathbb{R}^d et $Vor(\mathcal{A})$ le diagramme de Voronoï associé. On note V_i la cellule du point A_i . Soit X un point de \mathbb{R}^d et V_X la cellule de X dans le diagramme de Voronoï $Vor(\mathcal{A} \cup \{X\})$. On définit les voisins naturels d'un point comme les points A_i de \mathcal{A} pour lesquels $V_{X,i} = V_i \cap V_X \neq \emptyset$. On note $nat(X)$ l'ensemble des voisins naturels de X .

On définit les coordonnées naturelles de X $\lambda_i(X) = \frac{|V_{X,i}|}{|V_X|}$, comme les rapports des volumes de $V_{X,i}$ et V_X pour chacun des voisins naturels de X .

Sibson a montré que les $\lambda_i(X)$ sont C^0 partout et C^1 partout sauf aux points de \mathcal{A} . Par ailleurs X est le barycentre de ses voisins naturels affectés des coordonnées naturelles.

Nous avons montré comment utiliser ce schéma d'interpolation pour reconstruire des surfaces échantillonnées. \mathcal{A} est un ensemble de points pris sur une surface et, de plus, on suppose connu en chaque point de \mathcal{A} un approximant local de la surface $h_i(X)$, typiquement le plan tangent. L'interpolation naturelle fournit en tout point $X \in \mathbb{R}^d$ par $h(X) = \sum_{i \in nat(X)} \lambda_i(X) h_i(X)$, le lieu des points où $h(X)$ s'annule est l'interpolation recherchée. Un algorithme de maillage permet d'en calculer une approximation polyédrique.

Cette méthode fournit une approximation C^1 de la forme cherchée qui converge vers celle-ci quand la densité d'échantillonnage augmente.

La méthode a été validée sur des exemples variés issus de la synthèse d'images, de la CAO et de la modélisation moléculaire.

Un point intéressant de cette méthode (et qui la distingue des méthodes classiques) est qu'elle dissocie le calcul de la surface interpolante de la structure géométrique sous-jacente, ici le diagramme de Voronoï des points. Ceci permet de fournir une surface C^1 qui peut être exploitée directement. Des applications en cours portent sur la simplification de gros fichiers de points tels qu'on en obtient par exemple avec des caméras 3D. Un autre atout de cette

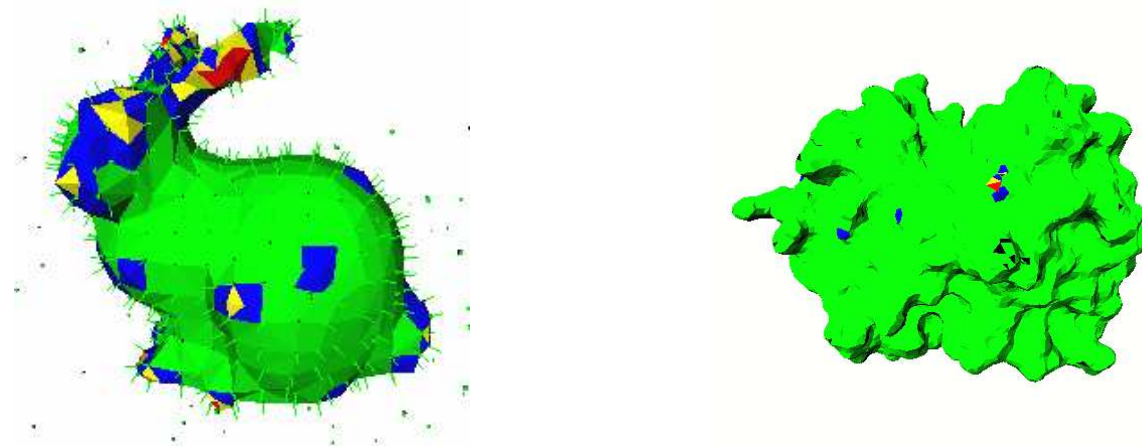


FIG. 8 – Exemples de surfaces reconstruites: (a)Modèle de lapin à partir de 500 points (b)Surface moléculaire à partir de 18,000 points

méthode est de fournir toutes les surfaces de niveaux : on peut ainsi sans calcul supplémentaire calculer des offsets, déformer localement la forme reconstruite ou calculer un maillage frontal.

Pour plus de détails, voir [22, 30].

6.4.2 Métrologie

Participant : Olivier Devillers.

Travail en collaboration avec Franco Preparata, Brown University, Providence, USA.

La zone cylindrique minimale d'un ensemble de points en trois dimensions est la couronne cylindrique définie par deux cylindres coaxiaux contenant tous les points et telle que la différence des rayons (épaisseur) soit minimale. Pour une application en métrologie, l'ensemble de points est nominalelement cylindrique, c'est-à-dire que les points sont proches d'un cylindre de référence connu. En utilisant des approximations valides au voisinage de ce cylindre de référence, on obtient une très bonne approximation de la zone cylindrique minimale. La méthode fournie des approximations successives ; chaque itération résout un problème de programmation linéaire en dimension six. L'erreur entre l'approximation et la solution optimale converge très rapidement (trois itérations en pratique) vers une erreur limite de $\frac{8\omega_0^2}{R}$ (ω_0 est l'épaisseur minimale et R le rayon du cylindre de référence) [35].

6.4.3 Dessin de graphes et molécules

Participants : Jean-Daniel Boissonnat, Frédéric Cazals, Julia Flötotto.

Les méthodes modernes de conception de médicaments font pour une large part appel à l'informatique : calcul et représentation des surfaces moléculaires, étude de la complémentarité ligand-récepteur, "screening" de grandes bases de données moléculaires, recherche de groupes d'atomes communs à plusieurs molécules, etc. En particulier, l'amélioration de l'effica-

citée de molécules traitant une pathologie donnée passe inévitablement par la comparaison des fonctions chimiques qu'elles présentent, d'où l'importance d'une représentation «normalisée» des formules développées des molécules permettant de mettre en évidence «différences» et «similarités».

La question du dessin de formules développées a fait l'objet du stage de DEA de Julia Flötotto [42] et peut s'énoncer comme suit :

Étant donné une famille de molécules partageant certains groupements chimiques, dessiner toutes ces molécules en représentant ces groupements de façon identique. Naturellement, les dessins proposés doivent respecter certaines contraintes comme le non croisement des liaisons chimiques ou encore les angles de liaison.

La solution apportée restreint le problème au cas où les graphes moléculaires ne contiennent pas de cycle autre que pentagonal ou hexagonal, et repose sur l'identification des parties isomorphes de la famille de molécules. On calcule pour ce faire une représentation de toute la famille appelée super-arbre. Celui-ci est ensuite utilisé pour coordonner les dessins. L'algorithme de dessin consiste en un parcours en largeur du graphe moléculaire. Pour chaque arête, un angle est choisi parmi les angles autorisés satisfaisant un certain critère d'optimisation —par exemple la longueur ou l'aire du dessin. Les molécules sont traitées de façon séquentielle, et les décisions sont propagées à l'aide du super-arbre (voir figure 9).

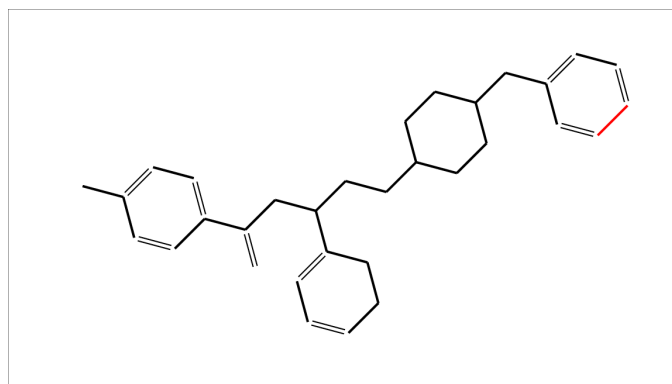


FIG. 9 – Dessin de molécule

7 Contrats industriels (nationaux, européens et internationaux)

7.1 IFP

Participants : Jean-Daniel Boissonnat, Sophie Balaven.

Il s'agit d'un contrat de conseil et d'encadrement d'une doctorante, Sophie Balaven, rémunérée par l'IFP.

Le travail est réalisé avec la bibliothèque CGAL et fait l'objet d'un dépôt de brevet.

7.2 REALVIZ

Participant : Jean-Daniel Boissonnat.

Techniques de triangulation pour la reconstruction 3D à partir d'images.

Il s'agit d'intégrer dans le produit ImageModeler développé par la société REALVIZ des outils géométriques de reconstruction de surfaces. Ce travail est réalisé par S. Nullans, ancien doctorant du projet et maintenant employé de Realviz.

7.3 EDF

Participant : Olivier Devillers.

Dans le cadre d'études menées par EDF en liaison avec le projet INRIA OMEGA, il est nécessaire de modéliser le déplacement de nombreuses particules dans un maillage. EDF était confronté dans ce cadre à des problèmes numériques que les approches développées dans le projet PRISME contribuent à résoudre (voir 6.3).

7.4 Kréon

Participant : Olivier Devillers.

http://www.kreon3d.com/ La société KRÉON, qui fabrique des capteurs laser permettant de numériser des points sur des objets, a acquis notre logiciel de triangulation de Delaunay 3D (voir 5.3) pour intégration dans son système d'ingénierie inverse.

8 Actions régionales, nationales et internationales

8.1 Actions nationales

8.1.1 Action de recherche coopérative INRIA FIABLE

Participants : Jean-Daniel Boissonnat, Hervé Brönnimann, Olivier Devillers, Alexandra Fronville, Pierre-Marie Gandoin, Sylvain Pion, Monique Teillaud, Mariette Yvinec.

Action Fiable : *http://www.inria.fr/prisme/fiable*

Cette action regroupe six équipes, cinq projets ou avant-projets INRIA : ALADIN (UR-Rennes), ARENAIRE (UR-Rhone-Alpes), POLKA (UR-Lorraine), PRISME (UR-Sophia) et SAGA (UR-Sophia) et une équipe extérieure : CHPV (Paris VI).

L'objectif est d'étudier les méthodes et techniques permettant la réalisation de logiciels garantissant la fiabilité des résultats. Comme indiqué plus haut (cf. 3.2), le calcul géométrique est très sensible aux erreurs de précision qui peuvent induire des incohérences dans la topologie des objets manipulés.

L'action FIABLE a débuté en septembre 1997. Son suivi a été assuré par plusieurs réunions communes et s'est concrétisé par les journées *Outils pour un calcul numérique fiable* organisées à Jussieu en septembre 1999. Ces journées ont réunies d'une part les équipes de FIABLE qui

ont présentés leurs travaux et d'autre part des participants extérieurs parmi lesquels quelques orateurs invités prestigieux.

En ce qui concerne Prisme, le travail effectué s'est concentré sur la conception de prédicats géométriques robustes (voir 6.3.1) et a largement bénéficié des contacts établis avec notamment SAGA et ALADIN.

8.1.2 Action de recherche coopérative INRIA GEOMETRICA

Participants : Jean-Daniel Boissonnat, Jeff Vitter, Mariette Yvinec.

Action Geometrica : <http://www.inria.fr/prisme/Geometrica/>

Géométrica a pour but de développer une approche axiomatique du calcul géométrique. Cette approche tend à définir les algorithmes à partir de primitives simples vérifiant un ensemble fini d'axiomes.

En dehors du projet PRISME cette action regroupe :

– Équipe GÉCOAL, ENS Paris (Michel Pocchiola, Maître de conférences; Pierre Angelier, doctorant). – ISA, INRIA Lorraine (Sylvain Petitjean, Chargé de recherches). – Collaborateurs extérieurs: Franco Preparata, Professeur, Université de Brown; Gert Vegter, Associate Professeur, Université de Groningen; Stéphane Rivière, Maître de conférences, Université de Mulhouse.

8.1.3 Accueil de chercheurs

Séminaire PRISME : <http://www.inria.fr/prisme/seminaire/>

Le séminaire PRISME, organisé par Monique Teillaud, a donné la parole à plusieurs orateurs extérieurs ou d'autres projets INRIA :

Lutz Kettner, (*Département Informatik, ETH Zürich*),

George Drettakis (*INRIA-iMAGIS*),

Mathieu Desbrun (*Dept of Computer Science, California Institute of Technology*),

Bernard Mourrain (*INRIA-SAGA*),

Steven M. LaValle (*Department of Computer Science, Iowa State University*),

Marco Manzini (*CNR-IAN, Pavie*),

Fredo Durand (*INRIA-iMAGIS*),

Anne Verroust (*INRIA-SYNTIM*),

Etienne Bertin (*LABSAD UPMF, Grenoble*),

Philippe Pebay (*INRIA-Gamma et INSA-Lyon*),

Michel Barlaud (*I3S*) et Eric Debreuve, (*Laboratoire de Biophysique et de Traitement d'image de l'UFR médecine*),

Bruno Levy (*GOCAD et CNRS/Inria Lorraine*) et Stéphane Conreux (*GOCAD*),

Sylvain Lazard (*INRIA-ISA*),

Sue Whitesides (*Université Mc Gill*).

8.2 Actions européennes : GALIA

Voir aussi les paragraphes 3.2.4, 5.1 et 6.1. Le site du projet : <http://cs.uu.nl/CGAL/>

Le projet a participé au projet de recherche communautaire suivant :

- Acronyme : GALIA, numéro 28155
- Titre : Geometric Algorithms for Industrial Application
- Programme spécifique du projet : ESPRIT
- Modalité du projet : Long Term Research.
- Date de début : 15-11-98 - Durée : 18 mois
- Mode de participation du projet Inria: partenaire
- Coordinateur: MPI Saarbrücken (Allemagne)
- Liste des partenaires :
ETH Zürich (Suisse), Freie Universität Berlin (Allemagne), Halle Universität (Allemagne),
Utrecht University (Pays-Bas)
Tel Aviv University (Israël)
- Résumé du projet : Développer une bibliothèque de programmes géométriques pour favoriser la diffusion dans le monde industriel et dans les milieux applicatifs des solutions et méthodes proposées par la géométrie algorithmique.

8.3 Relations bilatérales

8.3.1 France-Espagne

Nous avons organisé avec l'université polytechnique de Catalogne un cours sur CGAL avec le soutien du ministère des affaires étrangères et du ministère de la recherche espagnol [41]. Ces contacts ont débouché également sur une proposition Picasso.

8.3.2 France-Israël

Une collaboration avec l'université de Tel-Aviv (Danny Halperin et Micha Sharir) sur des algorithmes robustes et efficaces de calcul d'opérations de type Minkowski a débuté en novembre. Cette action est co-financée par les ministères de la recherche et de l'industrie français et israélien dans le cadre du programme « Factory for the future » (AFIRST).

8.4 Accueils de chercheurs étrangers

Amérique

Jeff Vitter (*Duke University, Durham*) a passé une année sabbatique dans le projet. Bruce Simpson (*University of Waterloo*) a partagé son année sabbatique entre le projet PRISME et le projet GAMMA de Rocquencourt.

9 Diffusion de résultats

9.1 Animation de la communauté scientifique

9.1.1 Organisation de conférences

<http://www.inria.fr/prisme/manifestations/journees98/>

Hervé Brönnimann a organisé la 15ème édition du *Colloque Européen de Géométrie Algorithmique* (European Workshop on Computational Geometry CG'99). Ce colloque annuel permet à la communauté européenne du domaine de se rencontrer. Il s'est déroulé du 15 au 17 mars au Palais des Congrès de Juan les Pins, et a regroupé plus de soixante participants. Trois conférenciers invités prestigieux (Jeff Vitter, Duke University et INRIA Sophia-Antipolis; Bernard Chazelle, Princeton University et École Polytechnique; Jean-Marie Laborde, IMAG Grenoble) ont présenté un état de l'art de leur domaine: celui des algorithmes dits en mémoire externe, pour des données en nombre tellement grand qu'elles ne tiennent pas en mémoire (Vitter), celui de la dérandomisation (Chazelle), et celui de la géométrie dynamique avec Cabri-géomètre (Laborde). Les 42 exposés donnés par les participants ont couvert une multitude de sujets intéressant la communauté (théorie des graphes, visibilité, arithmétique pour la géométrie, algorithmique, systèmes d'information géographiques, reconstruction, etc.).

9.1.2 Comités de rédaction de revues scientifiques

Theoretical Computer Science, Algorithmica, International Journal of Computational Geometry and Applications, Computational Geometry: Theory and Applications (J-D. Boissonnat) et *Journal of Discrete Algorithms* (M. Yvinec).

9.1.3 Comités de sélection

Jean-Daniel Boissonnat a organisé et présidé la section d'audition du concours de recrutement des chargés de recherche INRIA à l'unité de recherche de Sophia et il a également été membre de la section d'audition du concours de l'unité de recherche de Lorraine. Il est également membre du CNU et de la commission de spécialistes de l'université de Nice Sophia-Antipolis et de celle de l'ENS (27ième section).

9.1.4 Comités de programme de conférences

Hervé Brönnimann a participé au comité de programme du *ACM Symposium on Computational Geometry* et Jean-Daniel Boissonnat à celui de COCOON'99 (*5th Annual International Computing and Combinatorics Conference*). Jean-Daniel Boissonnat a été élu membre du *steering committee of Computational Geometry*. Il est également du conseil d'administration de l'AFIT (Association Française d'Informatique Théorique).

9.1.5 Serveur WWW

<http://www.inria.fr/prisme/>

Le projet PRISME maintient sur son site [www](http://www.inria.fr/prisme/) une collection de fiches explicatives sur les sujets présentés dans ce rapport, ainsi que les logiciels téléchargeables.

9.2 Formation

J.-D. Boissonnat est chargé de mission auprès de la direction générale pour la formation par la recherche.

9.2.1 Enseignement universitaire

- DEA d’algorithmique de Paris, Géométrie Algorithmique, 20 h (J-D. Boissonnat, M. Yvinec)
- Cours de 3ème année à l’ESSI : Géométrie Algorithmique, 24 h (J-D. Boissonnat, M. Yvinec)
- Cours de 3ème année à l’ISIA : Géométrie Algorithmique, 20 h (J-D. Boissonnat, M. Yvinec)
- ENSTA 3ème année, cours de Géométrie algorithmique du module « Algorithmes de vision » 3h (Monique Teillaud)
- DEA Aravis (Nice), de la géométrie algorithmique au calcul géométrique, 15 h (O. Devillers)
- Cours de Géométrie Algorithmique, 40 h (H. Brönnimann).
- École Polytechnique (Paris), Informatique, (O. Devillers, chargé de travaux pratiques à temps partiel, 96h)
- Travaux pratiques de programmation, (S. Pion 96h)
- Travaux dirigés Photoshop, IUT Sophia-Antipolis. (F. Rebufat 50h)
- Analyse, DEUG Mass. (A.Fronville 40h)

9.2.2 Autres enseignements

- Cours CGAL, Université Polytechnique de Catalogne (O. Devillers, F. Rebufat et M. Teillaud) [41].

9.2.3 Stages

Les sujets de stage proposés pour 1999 peuvent être consultés <http://www.inria.fr/prisme/Stages/>

- Louai Adhami, *Planification d’interventions chirurgicales robotisées : chirurgie cardiaque*, DEA Aravis, université de Nice Sophia-Antipolis.
- Carine Bonetto, *Sortie PostScript d’objets CGAL*, stage technicien de l’INSA-Rouen, juillet à septembre.
- Julia Flötotto, *Dessin de graphes et similarités moléculaires*, DEA Aravis, université de Nice Sophia-Antipolis.
- Alexandre Guyennon et Johann Liberman, *Un utilitaire pour la librairie CGAL*, projet de fin d’études, ESSI-DESS.
- Radu Lupşa, *Reconstruction de formes et métamorphoses*, DEA Algorithmique de Paris.
- Antoine Vigneron, *Algorithmes élémentaires pour reporter les intersections d’objets courbes*, DEA Algorithmique de Paris.

9.2.4 Thèses en cours

- Sophie Balaven, *Génération de maillages complexes 3D pour les simulateurs de réservoirs de nouvelle génération*, université de Nice-Sophia Antipolis, en collaboration avec l'IFP.
- Frank Da, *Interpolation de formes*, université de Nice-Sophia Antipolis.
- Julia Flötotto, *Dessin de graphes et similarités moléculaires*, université de Nice-Sophia Antipolis.
- Pierre-Marie Gandoin, *Représentation discrète des structures géométriques*, université de Nice-Sophia Antipolis.

9.2.5 Thèses soutenues

- Sylvain Pion, *De la géométrie algorithmique au calcul géométrique* [12], Thèse de doctorat de l'université de Nice, soutenue le 19 novembre.

9.3 Participation à des colloques, séminaires, invitations

9.3.1 Conférence invitée

- International Conference on Curves and Surfaces, « On the role of predicates in geometric computing » St Malo, juillet (J.-D. Boissonnat).

9.3.2 Séjours

Olivier Devillers a séjourné une semaine à Brown University.

9.3.3 Exposés à des colloques et séminaires

Les membres du projet ont présenté des articles lors de plusieurs conférences, on se reportera à la bibliographie pour en avoir la liste.

De plus, ils ont fait des présentations lors des manifestations suivantes :

- Workshop CGAL, Sophia-Antipolis, février. « New results on the analysis of arithmetic filters » (O. Devillers), « Triangulations in CGAL » (M. Yvinec)
- Séminaire IGN, « Triangulation de Delaunay » Paris, mars, (O. Devillers).
- Action coopérative FIABLE « Prédicats exacts pour arrangement de cercles » Paris, mars, (A. Fronville).
- « Géométrie algorithmique et diagrammes de Voronoï », INRIA Sophia, avril, (J.-D. Boissonnat).
- Mini Workshop on Geometric Software, ETH Zürich, avril, « Exact predicates for arrangements of arcs of circles » (M. Teillaud); « Generic and efficient implementation of filtered predicates in CGAL » (S. Pion).
- « Reconstruction de formes », Université de Strasbourg, avril, (J.-D. Boissonnat).
- Israeli Spring Workshop on Computational Geometry, Nachsholim, mai, « On the role of predicates in geometric computing » (J.-D. Boissonnat).
- « Reconstruction de formes », LORIA, mai, (J.-D. Boissonnat).

- Séminaire Brown University, « Robustness issues in computational geometry; approaches used in CGAL » juin, (O. Devillers).
- Séminaire SAGA, Université J. Fourier, Grenoble, « Structures et algorithmes géométriques pour les applications » (M. Yvinec)
- Review meeting galia : « Triangulations », Bruxelles, septembre (M. Yvinec)
- Journées *Outils pour un calcul numérique fiable*, Paris, septembre. « Précision numérique et incohérence géométrique » (O. Devillers), « Résolution de systèmes polynômes : application pour la fiabilité en géométrie algorithmique » (A. Fronville).
- Colloque du *Center for Geometric Computing*, Baltimore, octobre (H. Brönnimann).
- Séminaire Dassault Systèmes, « Calcul géométrique et prédicats », Paris, octobre et Aix en Provence, novembre (O. Devillers).
- Journées du thème INRIA 2B, Ventron, novembre (F. Cazals, M. Yvinec).
- « Interpolation naturelle sur une surface », Université de Montpellier, novembre, (J.-D. Boissonnat).
- Séminaire annuel du CIEEIST (centre interdisciplinaire d'étude de l'évolution des idées scientifiques et techniques) « Interpolation naturelle sur une surface », Université d'Orsay, décembre, (J.-D. Boissonnat).
- « Calcul géométrique », Université de Versailles, décembre, (J.-D. Boissonnat).

9.3.4 Participation à des colloques

Outre les colloques précédemment cités, des membres du projet ont assisté à diverses manifestations :

- Numérisation 3D, Paris, mai (O. Devillers).
- IMACS Applications of Computer Algebra Conference, Madrid, juin (A. Fronville).
- Workshop on Applied Geometric Computing, McGill University, La Barbade, novembre (M. Teillaud).

10 Bibliographie

Ouvrages et articles de référence de l'équipe

- [1] F. AVNAIM, J.-D. BOISSONNAT, « Polygon placement under translation and rotation », *Informatique Théorique et Applications/Theoretical Informatics and Applications* 23, 1989, p. 5–28.
- [2] J.-D. BOISSONNAT, A. CÉRÉZO, J. LEBLOND, « Shortest paths of bounded curvature in the plane », *Journal of Intelligent Systems* 10, 1994, p. 1–16.
- [3] J.-D. BOISSONNAT, O. DEVILLERS, R. SCHOTT, M. TEILLAUD, M. YVINEC, « Applications of random sampling to on-line algorithms in computational geometry », *Discrete Comput. Geom.* 8, 1992, p. 51–71, <http://www-sop.inria.fr/prisme/publis/bdsty-arsol-92.ps.gz>.
- [4] J.-D. BOISSONNAT, M. YVINEC, *Algorithmic geometry*, Cambridge University Press, UK, 1998, traduit de la version française (Ediscience international) par Hervé Brönnimann, <http://www-sop.inria.fr/prisme/personnel/yvinec/livre.html>.
- [5] J.-D. BOISSONNAT, « Shape reconstruction from planar cross-sections », *Comput. Vision Graph. Image Process.* 44, 1988, p. 1–29.
- [6] H. BRÖNNIMANN, M. YVINEC, « Efficient Exact Evaluation of Signs of Determinants », in : *Proc. 13th Annu. ACM Sympos. Comput. Geom.*, p. 166–173, 1997.

- [7] O. DEVILLERS, « An introduction to randomization in computational geometry », *Theoret. Comput. Sci.* 157, 1996, p. 35–52, <http://www-sop.inria.fr/prisme/publis/d-ircg-96.ps.gz>.
- [8] O. DEVILLERS, « Improved incremental randomized Delaunay triangulation », in: *Proc. 14th Annu. ACM Sympos. Comput. Geom.*, p. 106–115, 1998, <http://www-sop.inria.fr/prisme/publis/d-iirdt-98.ps.gz>.
- [9] B. GEIGER, *Three-dimensional modeling of human organs and its application to diagnosis and surgical planning*, Thèse de doctorat en sciences, École Nationale Supérieure des Mines de Paris, France, 1993, <http://www.inria.fr/RRRT/RR-2105>.
- [10] M. TEILLAUD, *Towards dynamic randomized algorithms in computational geometry*, *Lecture Notes in Computer Science*, 758, Springer-Verlag, 1993.

Livres et monographies

- [11] H. BRÖNNIMANN (éditeur), *15th European Workshop on Computational Geometry*. INRIA, 1999, <http://www-sop.inria.fr/prisme/manifestations/ccg99/index.html>.

Thèses et habilitations à diriger des recherches

- [12] S. PION, *De la géométrie algorithmique au calcul géométrique*, Thèse de doctorat en sciences, Université de Nice-Sophia Antipolis, France, 1999, <http://www.inria.fr/RRRT/TU-0???>

Articles et chapitres de livre

- [13] J.-D. BOISSONNAT, A. CÉRÉZO, E. DEGTIARIORA-KOSTOVA, V. KOSTOV, J. LEBLOND, « Shortest plane paths with bounded derivative of the curvature », *C. R. Acad. Sci. t. 329, Série I*, 1999, p. 613–618.
- [14] J.-D. BOISSONNAT, J. CZYZOWICZ, O. DEVILLERS, J.-M. ROBERT, M. YVINEC, « Convex Tours of Bounded Curvature », *Comput. Geom. Theory Appl.* 13, 1999, p. 149–160, <http://www-sop.inria.fr/prisme/publis/bcdry-ctbc.ps.gz>.
- [15] H. BRÖNNIMANN, B. CHAZELLE, J. MATOUŠEK, « Product Range Spaces, Sensitive Sampling, and Derandomization », *SIAM J. Comput.* 28, 1999, p. 1552–1575.
- [16] H. BRÖNNIMANN, I. EMIRIS, V. PAN, S. PION, « Sign Determination in Residue Number Systems », *Theoret. Comput. Sci.* 210, 1, 1999, p. 173–197, Special Issue on Real Numbers and Computers, <http://www-sop.inria.fr/prisme/personnel/pion/publis/TCS.ps.gz>.
- [17] O. DEVILLERS, M. KATZ, « Optimal Line Bipartitions of Point Sets », *Internat. J. Comput. Geom. Appl.* 9, 1999, p. 39–51, <http://www-sop.inria.fr/prisme/publis/dk-olbps-99.ps.gz>.
- [18] O. DEVILLERS, G. LIOTTA, F. PREPARATA, R. TAMASSIA, « Checking the convexity of polytopes and the planarity of subdivisions », *Comput. Geom. Theory Appl.* 11, 1998, p. 187–208, <http://www-sop.inria.fr/prisme/publis/dlpt-ccpps-98.ps.gz>.
- [19] O. DEVILLERS, F. P. PREPARATA, « Further Results on Arithmetic Filters for Geometric Predicates », *Comput. Geom. Theory Appl.* 13, 1999, p. 141–148, <http://www-sop.inria.fr/prisme/publis/dp-frafg-99.ps.gz>.
- [20] L. TANCREDI, M. TEILLAUD, « Application de la géométrie synthétique au problème de modélisation géométrique directe des robots parallèles », *Mechanism and Machine Theory* 34, 1999, p. 255–269.

Communications à des congrès, colloques, etc.

- [21] J.-D. BOISSONNAT, F. CAZALS, F. DA, O. DEVILLERS, S. PION, F. REBUFAT, M. TEILLAUD, M. YVINEC, « Programming with CGAL: The Example of Triangulations », *in: Proc the 15th Annu. ACM Sympos. Comput. Geom.*, p. 421–423, 1999, <http://www-sop.inria.fr/prisme/publis/bcddprty-pcet-99.ps.gz>.
- [22] J.-D. BOISSONNAT, F. CAZALS, S. NULLANS, « Smooth shape reconstruction », *in: 15th European Workshop Comput. Geom.*, p. 63–68, 1999.
- [23] J.-D. BOISSONNAT, J. SNOEYINK, « Line and curve segment intersection with restricted predicates », *in: Proc. 15th Annu. ACM Sympos. Comput. Geom.*, p. 370–379, 1999.
- [24] O. DEVILLERS, P.-M. GANDOIN, « Compression géométrique pour une transmission progressive », *in: 12ème journées de l'AFIG*, 1999.
- [25] O. DEVILLERS, P.-M. GANDOIN, « Rounding Voronoi Diagram », *in: Proc. 8th Discrete Geometry and Computational Imagery conference (DGCI99), Lecture Notes in Computer Science, 1568*, Springer-Verlag, p. 375–387, 1999.
- [26] O. DEVILLERS, F. P. PREPARATA, « Further Results on Arithmetic Filters for Geometric Predicates », *in: 15th European Workshop Comput. Geom.*, p. 113–116, 1999.
- [27] O. DEVILLERS, « On Deletion in Delaunay Triangulation », *in: Proc. 15th Annu. ACM Sympos. Comput. Geom.*, p. 181–188, 1999, <http://www-sop.inria.fr/prisme/publis/d-ddt-99.ps.gz>.
- [28] S. PION, « Interval Arithmetic: an efficient implementation and an application to computational geometry », *in: Workshop on Applications of Interval Analysis to systems and Control*, p. 99–110, 1999, http://www-sop.inria.fr/prisme/personnel/pion/publis/Misc_99.ps.gz.
- [29] M. TEILLAUD, « 3D triangulations in CGAL », *in: 15th European Workshop Comput. Geom.*, p. 175–178, 1999, <http://www-sop.inria.fr/prisme/publis/t-3tc-99.ps.gz>.

Rapports de recherche et publications internes

- [30] J.-D. BOISSONNAT, F. CAZALS, « Smooth shape reconstruction », *Rapport de recherche*, INRIA, 1999.
- [31] J.-D. BOISSONNAT, A. VIGNERON, « Elementary Algorithms for Reporting Intersections of Curve Segments », *Rapport de recherche n° 3825*, INRIA, 1999, <http://www.inria.fr/RRRT/RR-3825>.
- [32] H. BRÖNNIMANN, O. DEVILLERS, « The union of unit balls has quadratic complexity, even if they all contain the origin », *Rapport de recherche n° 3758*, INRIA, 1999, <http://www.inria.fr/RRRT/RR-3758>.
- [33] O. DEVILLERS, A. FRONVILLE, B. MOURRAIN, M. TEILLAUD, « Exact predicates for circle arcs arrangements », *Rapport de recherche n° 3826*, INRIA, 1999, <http://www.inria.fr/RRRT/RR-3826>.
- [34] O. DEVILLERS, P.-M. GANDOIN, « Compression géométrique pour une transmission progressive », *Rapport de recherche n° 3766*, INRIA, 1999, <http://www.inria.fr/RRRT/RR-3766>.
- [35] O. DEVILLERS, F. P. PREPARATA, « Evaluating the cylindricity of a nominally cylindrical point set », *Rapport de recherche n° 3747*, INRIA, 1999, <http://www.inria.fr/RRRT/RR-3747>.

Divers

- [36] « CGAL Manuals (5 volumes) », Février 1999, Release 1.2, supported by *Esprit LTR project 28155 GALIA*, <http://www.cs.uu.nl/CGAL/>.

-
- [37] « The CGAL Reference Manual », Décembre 1999, Release 2.1, supported by *Esprit LTR project 28155 GALIA*, <http://www.cs.uu.nl/CGAL/>.
 - [38] « The CGAL Reference Manual », Juin 1999, Release 2.0, supported by *Esprit LTR project 28155 GALIA*, <http://www.cs.uu.nl/CGAL/>.
 - [39] L. ADHAMI, *Planification d'interventions chirurgicales robotis/ées : chirurgie cardiaque*, Rapport de DEA Aravis, Université de Nice, France, 1999.
 - [40] C. BONETTO, *Sortie Postscript d'objets CGAL*, Rapport de stage technicien, INSA de Rouen, France, 1999.
 - [41] O. DEVILLERS, F. REBUFAT, M. TEILLAUD, « Cours CGAL », 1999, Barcelone, <http://www-sop.inria.fr/prisme/mediatheque/exposes/cgal-barcelona>.
 - [42] J. FLÖTOTTO, *Dessin de graphes et similarités moléculaires*, Rapport de DEA Aravis, Université de Nice, France, 1999.
 - [43] J. LIBERMAN, A. GUYENNON, *Un utilitaire pour la librairie CGAL*, Rapport de projet en entreprise, ESSI, 3ème année, France, 1999.
 - [44] R. LUPŞA, *Reconstruction de formes et métamorphoses*, Rapport de DEA algorithmique, Paris, France, 1999.
 - [45] A. VIGNERON, *Algorithmes élémentaires pour reporter les intersections d'objets courbes*, Rapport de DEA algorithmique, Paris, France, 1999.